Classificação semissupervisionada de séries temporais extraídas de imagens de satélite

Bruno Ferraz do Amaral

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: _____

Bruno Ferraz do Amaral

Classificação semissupervisionada de séries temporais extraídas de imagens de satélite

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

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientadora: Profa. Dra. Elaine Parros Machado de Sousa

USP – São Carlos
Junho de 2016

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

A485c Amaral, Bruno Ferraz do
Classificação semissupervisionada de séries
temporais extraídas de imagens de satélite / Bruno
Ferraz do Amaral; orientadora Elaine Parros Machado
de Sousa. - São Carlos - SP, 2016.
95 p.

Dissertação (Mestrado - Programa de Pós-Graduação
em Ciências de Computação e Matemática Computacional)
- Instituto de Ciências Matemáticas e de Computação,
Universidade de São Paulo, 2016.

1. Classificação semissupervisionada. 2. Grafos.
3. Séries temporais. 4. Sensoriamento Remoto. I.
Sousa, Elaine Parros Machado de, orient. II. Título.

Bruno Ferraz do Amaral

**Semi-supervised classification of time series extracted from
satellite images**

Master dissertation submitted to the Instituto de
Ciências Matemáticas e de Computação – ICMC-
USP, in partial fulfillment of the requirements for the
degree of the Master Program in Computer Science
and Computational Mathematics. *FINAL VERSION*

Concentration Area: Computer Science and
Computational Mathematics

Advisor: Profa. Dra. Elaine Parros Machado de Sousa

USP – São Carlos
June 2016

AGRADECIMENTOS

Aos meus pais, por todo o apoio, paciência e confiança, principalmente nos momentos mais difíceis.

À minha namorada Luciana, por todo o apoio e carinho que sempre me impulsionam a seguir em frente.

À minha orientadora Profa. Dra. Elaine Parros Machado de Sousa, por todo o conhecimento transmitido, lições ensinadas, pelos conselhos dados e pela confiança em mim e no meu trabalho.

Ao meu amigo Samuka, por todo o estímulo e palavras de incentivo, que sempre foram de essencial importância.

À Dra. Luciana Alvim Santos Romani, por toda a ajuda, paciência e confiança no meu trabalho.

À Dra. Renata Ribeiro Gonçalves, por toda a ajuda, conversas e momentos compartilhados.

À Rachel Scrivani, por toda a ajuda, apoio e sugestões.

Aos amigos do GBDI, por todo o apoio, conversas e sugestões.

Ao Instituto de Ciências Matemáticas e de Computação (ICMC-USP).

À FAPESP (Fundo de Amparo à Pesquisa do Estado de São Paulo) e à Fundação CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) pelo apoio financeiro à realização deste trabalho.

*“Aqueles que acharam que se deram bem,
dêem um passo à frente. Não tão rápido, George Banks.”
(George Banks)*

RESUMO

AMARAL, B. F.. **Classificação semissupervisionada de séries temporais extraídas de imagens de satélite**. 2016. 95 f. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação (ICMC/USP), São Carlos – SP.

Nas últimas décadas, com o crescimento acelerado na geração e armazenamento de dados, houve um aumento na necessidade de criação e gerenciamento de grandes bases de dados. Logo, a utilização de técnicas de mineração de dados adequadas para descoberta de padrões e informações úteis em bases de dados é uma tarefa de interesse. Em especial, bases de séries temporais têm sido alvo de pesquisas em áreas como medicina, economia e agrometeorologia. Em mineração de dados, uma das tarefas mais exploradas é a classificação. Entretanto, é comum em bases de séries temporais, a quantidade e complexidade de dados extrapolarem a capacidade humana de análise manual dos dados, o que torna o processo de supervisão dos dados custoso. Como consequência disso, são produzidos poucos dados rotulados, em comparação a um grande volume de dados não rotulados disponíveis. Nesse cenário, uma abordagem adequada para análise desses dados é a classificação semissupervisionada, que considera dados rotulados e não rotulados para o treinamento do classificador. Nesse contexto, este trabalho de mestrado propõe 1) uma metodologia de análise de dados obtidos a partir de séries temporais de imagens de satélite (SITS) usando tarefas de mineração de dados e 2) uma técnica baseada em grafos para classificação semissupervisionada de séries temporais extraídas de imagens de satélite. A metodologia e a técnica de classificação desenvolvidas são aplicadas na análise de séries temporais de índices de vegetação obtidas a partir de SITS, visando a identificação de áreas de plantio de cana-de-açúcar. Os resultados obtidos em análise experimental, realizada com apoio de especialistas no domínio de aplicação, indicam que a metodologia proposta é adequada para auxiliar pesquisas em agricultura. Além disso, os resultados do estudo comparativo mostram que a técnica de classificação semissupervisionada desenvolvida supera métodos de classificação supervisionada consolidados na literatura e métodos correlatos de classificação semissupervisionada.

Palavras-chave: Classificação semissupervisionada, Grafos, Séries temporais, Sensoriamento Remoto.

ABSTRACT

AMARAL, B. F.. **Classificação semissupervisionada de séries temporais extraídas de imagens de satélite**. 2016. 95 f. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação (ICMC/USP), São Carlos – SP.

The amount of digital data generated and stored as well as the need of creation and management of large databases has increased significantly, in the last decades. The possibility of finding valid and potentially useful patterns and information in large databases has attracted the attention of many scientific areas. Time series databases have been explored using data mining methods in several domains of application, such as economics, medicine and agrometeorology. Due to the large volume and complexity of some time series databases, the process of labeling data for supervised tasks, such as classification, can be very expensive. To overcome the problem of scarcity of labeled data, semi-supervised classification, which benefits from both labeled and unlabeled data available, can be applied to classify data from large time series databases. In this Master dissertation, we propose 1) a framework for the analysis of data extracted from satellite image time series (SITS) using data mining tasks and 2) a graph-based semi-supervised classification method, developed to classify temporal data obtained from satellite images. According to experts in agrometeorology, the use of the proposed method and framework provides an automatic way of analyzing data extracted from SITS, which is very useful for supporting research in this domain of application. We apply the framework and the proposed semi-supervised classification method in the analysis of vegetation index time series, aiming at identifying sugarcane crop fields, in Brazil. Experimental results indicate that our proposed framework is useful for supporting researches in agriculture, according to experts in the domain of application. We also show that our method is more accurate than traditional supervised methods and related semi-supervised methods.

Key-words: Semi-supervised classification, Graph-based, Time Series, Remote Sensing.

LISTA DE ILUSTRAÇÕES

Figura 1 – Com poucas instâncias rotuladas, não é possível estimar a distribuição das classes positiva e negativa acuradamente. Com uso de algumas instâncias não rotuladas é possível obter mais informações a respeito das duas distribuições.	3
Figura 2 – Processo de Descoberta de Conhecimento em Bases de Dados.	8
Figura 3 – Exemplo de conjunto hipotético de instâncias rotuladas com dois atributos (a_1, a_2). Usando um algoritmo de classificação supervisionada, um dos possíveis resultados de treinamento do classificador é a definição da fronteira de separação das classes ilustrada.	11
Figura 4 – Exemplo da comparação entre classificação supervisionada e semissupervisionada, para um conjunto X com poucas instâncias rotuladas e grande número de instâncias não rotuladas.	15
Figura 5 – Exemplo da comparação entre classificação supervisionada e semissupervisionada usando modelos generativos. A fronteira de separação de classes é ilustrada em linha tracejada.	17
Figura 6 – Exemplo da comparação entre o uso do método KNN e da abordagem de <i>cluster-then-label</i> .	19
Figura 7 – Exemplo da comparação entre o uso da SVM (supervisionada) e da TSVM (semissupervisionada) para encontrar o hiperplano de máxima margem.	20
Figura 8 – A distância Euclidiana calcula a diferença entre pares de observações no mesmo instante de tempo, para as duas séries temporais A e B .	31
Figura 9 – A similaridade entre as séries temporais A e B é reconhecida pela DTW.	32
Figura 10 – A presença de ruído interfere no cálculo de dissimilaridade usando a DTW.	33
Figura 11 – Reconhecimento de similaridade entre séries temporais com ruído usando LCSS.	34
Figura 12 – <i>Framework</i> SITSMining e seus componentes.	36
Figura 13 – Processo de extração de valores das SITS para geração da série temporal multidimensional. Dado um par de coordenadas, são encontrados os pixels correspondes a essas coordenadas em cada imagem de cada SITS. Em seguida, os valores são extraídos, organizados por dimensão e ordenados pelo tempo, na série temporal multidimensional.	39
Figura 14 – Módulos e conjunto de instância da Camada de Extração e Pré-processamento.	40
Figura 15 – Camada de Mineração de Dados.	41
Figura 16 – Camada de Saída.	42

Figura 17 – Gráficos ilustrados usando os resultados produzidos pela Camada de Saída do <i>framework</i>	43
Figura 18 – Interface da ferramenta SatImagExplorer.	45
Figura 19 – Exemplo funcionamento da técnica CSPG, desde a criação do grafo até a propagação de rótulos.	52
Figura 20 – Região de interesse para geração do conjunto de dados.	59
Figura 21 – Máscara da safra 2004/2005 da cana-de-açúcar (a) e aplicação da máscara para rotulação dos pixels (b) da região de interesse.	62
Figura 22 – Processo de geração do conjunto de instâncias rotuladas e não rotuladas, e definição dos conjuntos de treinamento e teste para os casos supervisionado e semissupervisionado.	62
Figura 23 – Resultados MCC da comparação entre CSPG e as variações $CSPG_{st}$ e $CSPG_{prox}$	71
Figura 24 – Número de intersecções dos tipos de aresta em função de k	73
Figura 25 – Resultados de classificação (MCC) para a safra 2004/2005 e satélites TERRA e NOAA.	75
Figura 26 – Teste-t pareado dos resultados de classificação da safra 2004/2005 dos satélites TERRA e NOAA.	77
Figura 27 – Resultados de classificação (MCC) para as safras 2004/2005 e 2011/2012 do satélite TERRA.	78
Figura 28 – Teste-t pareado dos resultados de classificação das safras 2004/2005 e 2011/2012 do satélite TERRA.	79
Figura 29 – Visualização geoespacial dos resultados de classificação da CSPG e <i>framework</i> SITSMining (a), e a ilustração das áreas de cana-de-açúcar obtidas a partir de máscara de cana, para safra de 2004/2005 (b).	80

LISTA DE TABELAS

Tabela 1 – Informações das imagens dos satélites TERRA e NOAA.	59
Tabela 2 – Definição do conjunto de dados.	61
Tabela 3 – Melhor e pior caso na avaliação de um resultado de classificação usando MCC.	64
Tabela 4 – Parametrização do experimento para análise do parâmetro r	65
Tabela 5 – Resultados de MCC para o experimento de análise do parâmetro r	66
Tabela 6 – Valores dos parâmetros para cada algoritmo de classificação.	67
Tabela 7 – Configuração do experimento de função de distância para KNN.	67
Tabela 8 – Valores dos parâmetros das funções de distância DTW e LCSS.	67
Tabela 9 – Valores de MCC obtidos no experimento de função de distância para KNN.	68
Tabela 10 – Configuração do experimento para análise do parâmetro α	69
Tabela 11 – Resultados de MCC para o experimento de análise do parâmetro α	69
Tabela 12 – Configuração de experimento de comparação entre CSPG, CSPG _{st} e CSPG _{prox}	70
Tabela 13 – Resultados de MCC para o experimento de análise dos parâmetros w e k	72
Tabela 14 – Configuração do experimento de análise comparativa para TERRA e NOAA, safra 2004/2005.	74
Tabela 15 – Configuração do experimento de função de distância para CSPG.	77
Tabela 16 – Configuração do experimento usando <i>framework</i> SITSMining e CSPG.	79
Tabela 17 – Tabela verdade para experimento usando <i>framework</i> SITSMining e CSPG.	80

SUMÁRIO

1	INTRODUÇÃO	1
1.1	Contextualização e motivação	3
1.2	Principais contribuições	5
1.3	Objetivos	5
1.4	Organização do trabalho	6
2	MINERAÇÃO DE DADOS E CLASSIFICAÇÃO SEMISSUPERVISEDADA	7
2.1	Aprendizado não supervisionado	9
2.2	Aprendizado supervisionado	10
2.3	Aprendizado semissupervisionado	12
2.3.1	<i>Modelos generativos</i>	16
2.3.2	<i>Cluster-then-label</i>	17
2.3.3	<i>Separação por baixa densidade</i>	18
2.3.4	<i>Selftraining</i>	19
2.3.5	<i>Co-training</i>	20
2.3.6	<i>Baseada em grafos</i>	21
2.4	Considerações finais	26
3	SÉRIES TEMPORAIS	27
3.1	Pré-processamento	28
3.2	Transformação e representação de séries temporais	28
3.3	Tarefas de mineração de séries temporais	29
3.3.1	<i>Medidas de dissimilaridade</i>	30
3.4	Considerações finais	34
4	FRAMEWORK SITSMINING	35
4.1	Dados de entrada	37
4.2	Camada de Extração e Pré-processamento	38
4.3	Camada de Mineração de Dados	40
4.4	Camada de Saída	41
4.5	Ferramenta SatImagExplorer	42
4.6	Considerações Finais	44

5	TÉCNICA CSPG	47
5.1	Dados de entrada	48
5.2	Criação do grafo	49
5.3	Rotulação do grafo	51
5.4	Considerações finais	54
6	EXPERIMENTOS	57
6.1	Séries Temporais de Imagens de Satélite	58
6.2	Metodologia do Estudo Experimental	60
6.2.1	<i>Conjunto de dados</i>	60
6.2.2	<i>Algoritmos de classificação</i>	63
6.2.3	<i>Avaliação dos resultados</i>	63
6.3	Análise de parâmetros	64
6.3.1	<i>Parametrização do conjunto de dados</i>	65
6.3.2	<i>Parametrização dos algoritmos de classificação</i>	65
6.3.3	<i>Parametrização da técnica CSPG</i>	67
6.4	Análise comparativa	73
6.5	CSPG e <i>framework</i> SITSMining	78
6.6	Considerações finais	81
7	CONCLUSÃO	83
7.1	Principais contribuições	83
7.2	Trabalhos futuros	85
	REFERÊNCIAS	87

INTRODUÇÃO

Com o aumento do volume de dados gerados e o crescente avanço nas tecnologias de geração e coleta de dados, surgiu a necessidade de criação de bases de dados capazes de armazenar grandes volumes de dados, de modo persistente. Com o elevado número de bases de dados utilizadas para armazenamento, gerenciamento e proteção de grandes volumes de dados em diversos domínios, incluindo os de pesquisa científica, surgiu o interesse no estudo desses dados, visando a obtenção de informações valiosas para os usuários de tais bases.

A descoberta de conhecimento em bases de dados (do inglês *KDD - Knowledge Discovery in Databases*) consiste na análise de grandes volumes de dados a fim de obter padrões úteis e informações relevantes contidas nos dados. Em KDD, a aplicação de técnicas de mineração de dados para análise de grandes bases de dados permite a extração de padrões e obtenção de informações que dificilmente são obtidos ao observar manualmente os dados. Por exemplo, no cenário financeiro, é de grande valia a obtenção de informações relacionadas à detecção de fraudes no uso de cartões de crédito, à predição de valores futuros de ações na bolsa de valores e às características de um grupo de clientes com mesmo padrão de compra ou que apresentam interesse em uma mesma aplicação comercial. Além do cenário financeiro, o uso de técnicas de mineração de dados é útil em áreas como medicina, agrometeorologia, biologia, estatística, entre outras.

Atualmente, o fato das bases de dados apresentarem grande quantidade de dados inviabiliza sua análise manual, o que torna imprescindível o uso de suporte computacional para mineração de dados. Além disso, outro fator que torna o processo de análise dos dados mais custoso, é a presença de dados com alta complexidade. Dados como imagens de satélite, séries temporais, documentos e sequências de DNA são considerados complexos, pois não é possível, ou não é significativo, realizar operações básicas como multiplicação, adição, subtração, igualdade e desigualdade. Por exemplo, é possível comparar se duas imagens de satélite são idênticas, ou seja, se os pixels correspondentes nas duas imagens possuem mesmo valor, porém o resultado

quase sempre será negativo, o que não tem valor semântico para o domínio de aplicação.

Em KDD, a etapa de análise dos dados, em geral, consiste na aplicação de uma técnica de mineração de dados. Existem diversas tarefas de mineração de dados, como classificação, agrupamento, regressão, identificação de regras de associação, detecção de anomalias e predição. As tarefas de mineração de dados possuem objetivos distintos e usam abordagens diferentes para análise dos dados. Uma das tarefas mais amplamente utilizadas é a classificação, cujo objetivo é associar cada instância de um conjunto de dados a uma ou mais classes pré-definidas no problema (HAN; KAMBER; PEI, 2011). Nessa tarefa, usualmente um classificador é treinado com base em um conjunto de instâncias previamente rotuladas, denominado conjunto de treinamento. Na maioria dos casos, o conjunto de treinamento é gerado por meio de um processo supervisionado de rotulação, em que o especialista analisa e rotula cada instância de um conjunto de dados com base em seu conhecimento específico do domínio do problema. Após a etapa de treinamento, o classificador é usado para rotular instâncias de classe, a princípio, desconhecida.

Um problema para aplicação da tarefa de classificação ocorre quando o conjunto de treinamento é muito pequeno. Nesses casos, o classificador é treinado usando poucas instâncias rotuladas, e a qualidade de classificação obtida pode ser insatisfatória. Por exemplo, na Figura 1, pode-se notar que, usando apenas 2 instâncias rotuladas para cada classe, não é possível representar a real distribuição das classes do problema. Em geral, a obtenção de dados rotulados pode ser dificultada devido ao elevado esforço humano necessário para rotulação manual de dados, à necessidade de dispositivos especiais/caros e à dependência de experimentos que apresentam alto custo, seja financeiro ou de tempo, entre outros (ZHU *et al.*, 2009). Uma abordagem de análise viável, nesses casos, é a aplicação de técnicas de classificação semissupervisionada. A diferença da classificação semissupervisionada para a tradicional (supervisionada) é que são usados dados rotulados e não rotulados para o treinamento do classificador. A classificação semissupervisionada pode obter uma qualidade de classificação superior à do caso supervisionado se houver poucos dados rotulados e alta disponibilidade de dados não rotulados. Isso torna a opção pela abordagem semissupervisionada muito atraente, pois é comum, em problemas com escassez de dados rotulados, haver abundância de dados não rotulados. Alguns exemplos desses problemas são classificação de documentos e páginas *web*, predição de estrutura 3D de proteínas, análise de uso de terra a partir de imagens de satélite, técnicas de filtro de *spam* em *e-mails* e vigilância por câmeras de segurança. Nesses casos, apesar da dificuldade/custo de supervisão dos dados, dados não rotulados são obtidos com facilidade e em grande quantidade, o que viabiliza a aplicação de classificação semissupervisionada. Na Figura 1b, o uso de várias instâncias não rotuladas permite identificar a estrutura de cada classe do problema.

Neste trabalho de mestrado, o foco é na análise de dados obtidos a partir de séries de imagens de satélite com aplicação na área de agrometeorologia. Nesse estudo, devido à alta complexidade e volume dos dados, há escassez de dados rotulados e, como há abundância de dados não rotulados, a análise de classificação semissupervisionada é favorecida.

Figura 1 – Com poucas instâncias rotuladas, não é possível estimar a distribuição das classes positiva e negativa acuradamente. Com uso de algumas instâncias não rotuladas é possível obter mais informações a respeito das duas distribuições.

Fonte: Elaborada pelo autor.

1.1 Contextualização e motivação

No Brasil, a agricultura apresenta importância significativa para o setor econômico do país, sendo responsável por grande parte dos recursos nacionais. A agricultura brasileira contribui com, aproximadamente, um terço do PIB nacional (Produto Interno Bruto) e 40% da mão-de-obra. Devido à grande extensão territorial brasileira, é relevante para o governo e empresas ligadas à agricultura a realização do monitoramento das áreas agrícolas ao longo dos anos. Como resultado desse acompanhamento, é possível realizar estudos como o cálculo de estimativa de valores de produção, identificação de expansão, redução e migração de áreas agrícolas, além de oferecer informações relevantes como suporte à tomada de decisão para produtores agrícolas.

Nos últimos anos, tem crescido o interesse no acompanhamento de áreas agrícolas, devido ao aumento na relevância dos impactos causados pelo aquecimento global e as mudanças climáticas em todo o mundo. Por isso, estudos focados na compreensão das mudanças no clima brasileiro e seu impacto nas áreas de plantio agrícola são imprescindíveis para o planejamento envolvido na produção e comercialização dos produtos agrícolas, no Brasil. Nesse cenário, os biocombustíveis têm apresentado papel fundamental na redução da emissão de gases efeito estufa, assim como no oferecimento de uma solução de energia renovável para substituição de combustíveis fósseis. Nesse contexto, a cana-de-açúcar tem recebido grande destaque, nos últimos anos, devido à produção do etanol, desempenhando importante papel na economia brasileira.

O Brasil é o principal produtor mundial de cana-de-açúcar, cuja produção concentra-se no estado de São Paulo, com 57% do total nacional. Por isso, o monitoramento das áreas de plantio de cana-de-açúcar ao longo dos anos é uma tarefa importante, principalmente para o planejamento

necessário para a expansão e migração das áreas produtoras, levando em consideração as recentes mudanças climáticas e o aquecimento global. Um dos modos de realizar o acompanhamento das áreas produtoras de cana-de-açúcar ao longo de safras é por meio da análise de séries temporais de imagens de satélite (SITS - do inglês *Satellite Image Time Series*). Nessa análise, o objetivo é considerar, para cada área real da região de interesse, uma série de valores, que indica, por exemplo, o comportamento da vegetação presente naquela área, ao longo de um período de tempo. Dessa maneira, é possível diferenciar uma área de cana-de-açúcar de uma área de floresta, pois os comportamentos da vegetação nessas áreas são distintos, ao decorrer de uma safra de cana-de-açúcar. As informações relativas à vegetação de uma área podem ser obtidas usando índices de vegetação, como o NDVI (*Normalized Difference Vegetation Index*) (ROUSE *et al.*, 1973). O NDVI está relacionado à quantidade e concentração de biomassa da vegetação e é um índice amplamente usado em pesquisas de agricultura e vegetação (WANG; RICH; PRICE, 2003; LUCAS; SCHULER, 2007).

Para este trabalho de mestrado, foram utilizadas imagens capturadas pelo sensor AVHRR (*Advanced Very High Resolution Radiometer*), a bordo do satélite NOAA (*National Oceanic and Atmospheric Administration*), e pelo sensor MODIS (*Moderate Resolution Imaging Spectroradiometer*), a bordo do satélite TERRA. As imagens de satélite usadas neste trabalho possuem baixa resolução espacial, de aproximadamente 1km^2 . Consequentemente, em comparação a imagens de média e alta resolução espacial, o número de pixels nas imagens usadas é menor, porém a probabilidade de mistura espectral e perda de informação é maior. No entanto, como a cana-de-açúcar é cultivada em áreas vastas e próximas, a perda de informação apresentada nas imagens de baixa resolução pode ser considerada menos impactante para esse tipo de aplicação do que para tentar identificar alvos como estradas ou riachos, por exemplo. Por isso, o uso dessas imagens é viável e, ao mesmo tempo desafiador, para o contexto deste trabalho. Além disso há disponibilidade gratuita de longas séries de imagens de baixa resolução, e pode-se analisar uma região de interesse maior, como um estado inteiro, devido à redução no número de pixels presentes nas imagens, aumentando o alcance do monitoramento das áreas de cana-de-açúcar.

Para os especialistas em agrometeorologia, uma técnica computacional para identificação automática de áreas de cana-de-açúcar é de grande valia. No entanto, o processo de supervisão para rotulação manual de séries temporais de NDVI demanda muito tempo do especialista. Como consequência disso, a porção de dados rotulados resultante da supervisão é muito pequena em relação ao volume de dados não rotulados a serem classificados pela técnica computacional. Como as imagens de satélite de baixa resolução estão disponíveis em grande quantidade, a utilização de classificação semissupervisionada é favorecida, nesse cenário. Assim, este trabalho de mestrado teve como objetivo principal o desenvolvimento de uma técnica de classificação semissupervisionada para aplicação na análise de dados obtidos a partir de imagens de satélite, visando o monitoramento de áreas de plantio de cana-de-açúcar.

O desenvolvimento deste trabalho de mestrado foi realizado em parceria com pesqui-

sadores da Embrapa Informática Agropecuária¹ e do Cepagri/Unicamp (Centro de Pesquisas Meteorológicas e Climáticas Aplicadas à Agricultura)². As tarefas de desenvolvimento, planejamento de experimentos e análise dos resultados foram realizadas com apoio dos pesquisadores parceiros.

1.2 Principais contribuições

Este trabalho de mestrado teve como principais contribuições:

- o desenvolvimento de uma metodologia de análise de dados extraídos a partir de séries temporais de imagens de satélite usando métodos de mineração de dados, denominada *framework* SITSMining. O *framework* proposto é de grande utilidade para pesquisas em agrometeorologia, de acordo com os especialistas, pois consiste em um processo automatizado de análise de dados de imagens de satélite, que compreende tarefas de extração e pré-processamento de séries temporais, análise de mineração de dados, e produção de resultados em formato adequado para interpretação do especialista em agrometeorologia;
- desenvolvimento de uma técnica de classificação semissupervisionada adequada para análise de dados geoespaciais extraídos a partir de séries temporais de imagens de satélite. A técnica proposta, denominada Classificação Semissupervisionada usando Proximidade Geoespacial (CSPG), utiliza a abordagem semissupervisionada baseada em grafos para classificação de dados obtidos a partir de imagens de satélite por meio da análise de informações geográficas e temporais das áreas reais referenciadas pela imagem de satélite.

1.3 Objetivos

Este trabalho de mestrado está inserido no contexto de análise de dados extraídos de imagens de satélite, usando técnicas de mineração de dados, e foi realizado com os seguintes objetivos:

- propor uma metodologia de análise de dados obtidos a partir de imagens de satélite, que apresente alto valor prático para especialistas em agrometeorologia e envolva as tarefas de: 1) extração de valores das imagens e pré-processamento dos dados, 2) aplicação de métodos de mineração de dados e 3) visualização adequada para o especialista em agrometeorologia dos resultados produzidos na tarefa 2);
- desenvolver uma técnica para classificação semissupervisionada de dados extraídos de séries temporais de imagens de satélite, validar a metodologia e as técnicas propostas por

¹ <https://www.embrapa.br/informatica-agropecuaria>

² <http://www.cepagri.unicamp.br/>

meio de experimentos de análise comparativa com técnicas de classificação supervisionada e semissupervisionada propostas na literatura, para classificação de dados reais referentes ao estado de São Paulo, Brasil, visando a identificação correta de áreas de plantio de cana-de-açúcar.

1.4 Organização do trabalho

Esta dissertação de mestrado está organizada da seguinte maneira:

Capítulo 1 – Introdução. São apresentados o contexto, motivação e objetivos deste trabalho e as principais contribuições do trabalho desenvolvido.

Capítulo 2 – Mineração de dados e classificação semissupervisionada. São apresentados conceitos básicos e principais trabalhos nas literaturas de mineração de dados e classificação semissupervisionada.

Capítulo 3 – Séries temporais. São apresentados conceitos básicos e trabalhos relacionados a funções de distância e tarefas de mineração de séries temporais.

Capítulo 4 – *Framework* SITSMining. É apresentada a metodologia de análise de séries temporais de imagens de satélite desenvolvida neste trabalho.

Capítulo 5 – Técnica CSPG. É detalhada a técnica de classificação semissupervisionada baseada em grafos desenvolvida neste trabalho.

Capítulo 6 – Experimentos. São apresentados os experimentos e resultados para validação da técnica de classificação semissupervisionada proposta e para a aplicação da técnica proposta usando a metodologia de análise de SITS desenvolvida.

Capítulo 7 – Conclusões. São apresentadas as conclusões, incluindo as principais contribuições, publicações, e propostas para trabalhos futuros.

MINERAÇÃO DE DADOS E CLASSIFICAÇÃO SEMISSUPERVISIONADA

Nas últimas décadas, a quantidade de informação gerada e armazenada de maneira digital tem crescido aceleradamente. Como resultado disso, bases de dados têm sido usadas para armazenar e gerenciar grandes volumes de dados, em diversos domínios de aplicação. Além de volumosas, em muitos casos, as bases armazenam dados complexos como imagens e séries temporais. Devido ao alto volume e complexidade dos dados, a análise manual dessas bases tem sido impraticável, o que fez surgir o interesse no desenvolvimento de técnicas computacionais capazes de realizar tal análise em tempo e com custo viáveis, impulsionando o crescimento da área de Descoberta de Conhecimento em Bases de Dados (KDD – do inglês *Knowledge Discovery in Databases*).

O KDD pode ser descrito como o processo iterativo e iterativo de identificar padrões válidos, novos, úteis e compreensíveis em bases de dados volumosas e complexas (MAIMON; ROKACH, 2010). A área de KDD possui natureza interdisciplinar e consiste na intersecção de áreas de pesquisa como aprendizado de máquina, reconhecimento de padrões, banco de dados, estatística, inteligência artificial, visualização de dados, entre outras (FAYYAD; PIATETSKY-SHAPIRO; SMYTH, 1996). O objetivo comum dessas áreas é a extração de conhecimento em alto nível de abstração, a partir da análise de grandes volumes de dados em baixo nível de abstração (FAYYAD; PIATETSKY-SHAPIRO; SMYTH, 1996).

Muitos autores tratam KDD como sinônimo de mineração de dados, enquanto outros definem a mineração de dados como uma das etapas do processo de KDD (HAN; KAMBER; PEI, 2011) e de modo geral, as atividades realizadas no processo de KDD podem ser agrupadas nas seguintes etapas, ilustradas na Figura 2 (FAYYAD; PIATETSKY-SHAPIRO; SMYTH, 1996):

1. **Seleção:** a porção de dados relevantes para a análise de mineração de dados é selecionada a partir da base de dados.

2. **Pre-processamento:** dados com ruídos ou inconsistentes são removidos. Também pode ser escolhida e aplicada uma abordagem para substituição de valores faltantes.
3. **Transformação:** é escolhida a estrutura e/ou representação dos dados adequada para a análise de mineração de dados, com possível uso de métodos de seleção de atributos e redução de dimensionalidade.
4. **Mineração de dados:** é escolhida a tarefa de mineração de dados apropriada e um método inteligente para extração de padrões a partir da análise dos dados transformados.
5. **Interpretação e avaliação:** os padrões e os resultados obtidos são avaliados. De acordo com os objetivos definidos, são identificados os padrões válidos, úteis e de interesse.

Figura 2 – Processo de Descoberta de Conhecimento em Bases de Dados.

Fonte: Adaptada de [Fayyad, Piatetsky-Shapiro e Smyth \(1996\)](#).

A etapa de mineração de dados é essencial para o processo de KDD, pois é nela que os dados pré-processados nas etapas anteriores são analisados por um método que retorna ao usuário padrões possivelmente úteis e valiosos.

Em geral, as principais tarefas de mineração de dados costumam ser organizadas em dois principais paradigmas: aprendizado supervisionado e não supervisionado. Outro paradigma que vem crescendo desde a década de 1990 e consiste em uma estratégia intermediária entre os aprendizados supervisionado e não supervisionado é o aprendizado semissupervisionado. Nas próximas seções, os três tipos de aprendizado são descritos em mais detalhes.

2.1 Aprendizado não supervisionado

Com o objetivo de uniformizar a apresentação de conceitos e algoritmos, são introduzidas a seguir notações e definições relevantes no contexto deste trabalho.

Seja um conjunto de instâncias definido como uma coleção $X = \{x_1, x_2, \dots, x_n\}$ de n instâncias. O conjunto X é descrito por um conjunto de atributos $A = \{a_1, a_2, \dots, a_D\}$, em que um dado atributo a_i pode ser categórico ou numérico. No caso de atributo categórico, pode-se definir o domínio de valores para esse atributo como $dom(a_i) = \{v_{i1}, v_{i2}, \dots, v_{i|dom(a_i)|}\}$, em que $|dom(a_i)|$ corresponde à cardinalidade desse domínio¹. No caso de atributo numérico, $dom(a_i)$ corresponde a $\mathbb{R}$ ou a um subconjunto de $\mathbb{R}$.

O espaço dos atributos de A é denotado $\mathcal{X}$ e definido como o produto cartesiano entre os domínios dos atributos: $\mathcal{X} = dom(a_1) \times dom(a_2) \times \dots \times dom(a_D)$. Assim, para o conjunto de n instâncias X , tem-se que $x_i \in \mathcal{X}$, para todo $i \in \{1, 2, \dots, n\}$. Por fim, uma instância pode ser definida como um vetor D -dimensional $x_i = \{x_{i1}, x_{i2}, \dots, x_{iD}\}$, em que cada dimensão x_{ij} corresponde ao valor do atributo $a_j \in A$ para a instância x_i . Em geral, assume-se que as instâncias de um conjunto X são amostradas aleatoriamente e independentemente a partir de uma distribuição de probabilidade desconhecida $P(x)$ em $\mathcal{X}$ (ZHU *et al.*, 2009).

Considerando essas definições, dado um conjunto de instâncias X , pode-se dizer que o objetivo geral em aprendizado não supervisionado é encontrar estruturas e padrões de interesse em X , a partir da análise das próprias instâncias, ou seja, sem conhecimento adicional além dos próprios atributos em A (CHAPELLE; SCHÖLKOPF; ZIEN, 2006; ZHU *et al.*, 2009). Pela ausência de supervisão em relação aos dados, os métodos de aprendizado não supervisionado visam encontrar estruturas de interesse, porém não são capazes de oferecer informações semânticas sobre o padrão encontrado (HAN; KAMBER; PEI, 2011).

As principais tarefas de aprendizado não supervisionado são:

- **Agrupamento:** o objetivo é organizar as instâncias de um conjunto X em subconjuntos denominados agrupamentos, de modo que instâncias de um mesmo agrupamento apresentem alta similaridade entre si e apresentem baixa similaridade em relação a instâncias de outros agrupamentos. A escolha de uma função de distância $Dist(x_i, x_j)$ é importante nessa análise, pois os agrupamentos são calculados com base na similaridade/dissimilaridade entre as instâncias de X . Diversos métodos de agrupamento têm sido propostos na literatura, como K-Means (HAN; KAMBER; PEI, 2011), K-Medoids (KAUFMAN; ROUSSEEUV, 1990), BIRCH (ZHANG; RAMAKRISHNAN; LIVNY, 1996), Chameleon (KARYPIS; HAN; KUMAR, 1999), DBSCAN (BIRANT; KUT, 2007) e STING (WANG *et al.*, 1997).
- **Detecção de anomalia (ou novidade):** o objetivo é identificar instâncias que apresentem alta dissimilaridade em relação à maioria das instâncias do conjunto X . As instâncias

¹ Baseado na definição em (MAIMON; ROKACH, 2010)

que apresentam comportamento diferente da expectativa são consideradas novidades ou anomalias. Essa tarefa é usada em muitas aplicações como detecção de fraudes com cartão de crédito, detecção de intrusos em redes, processamento de imagens, entre outras (HAN; KAMBER; PEI, 2011). Abordagens e métodos para detecção de anomalias são descritos em (HODGE; AUSTIN, 2004).

2.2 Aprendizado supervisionado

Em aprendizado supervisionado, os atributos do conjunto de instâncias X correspondem ao conjunto $(A \cup y)$, em que y é denominado atributo alvo. O atributo y pode corresponder a um ou mais valores categóricos ou numéricos e seu domínio é denotado por $\mathcal{Y}$ (MAIMON; ROKACH, 2010). Desse modo, as instâncias do conjunto X correspondem a pares $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, em que, geralmente, o valor y_i de cada instância é gerado como resultado de um processo de supervisão das instâncias de X . Em geral, assume-se que cada par (x_i, y_i) é amostrado independentemente a partir uma distribuição não conhecida $P(x, y)$ em $\mathcal{X} \times \mathcal{Y}$ (CHAPELLE; SCHÖLKOPF; ZIEN, 2006). O principal objetivo dos métodos supervisionados é construir um modelo f capaz de aprender um mapeamento $f: \mathcal{X} \mapsto \mathcal{Y}$ de maneira automática, a partir das instâncias de X . Esse processo de aprendizado do modelo é denominado treinamento, e o conjunto X é comumente chamado de conjunto de treinamento.

As duas principais tarefas de aprendizado supervisionado são:

- **Classificação:** é um problema supervisionado no qual o atributo alvo y corresponde a um ou mais valores de um conjunto de valores pré-definidos, sejam eles categóricos ou numéricos. Em outros termos, o domínio $\mathcal{Y} = \{c_1, c_2, \dots, c_m\}$, em que m é a quantidade de classes do problema. Em classificação, o objetivo é construir modelo, denominado classificador, treiná-lo usando o conjunto de treinamento X , e usá-lo para inferir o valor do atributo y de instâncias futuras x , ainda não rotuladas. Em problemas de classificação com apenas duas classes ($C = 2$), é comum denominar as classes como positiva e negativa, sendo a classe positiva a de interesse, na maioria dos casos. Um exemplo de problema de classificação é a identificação automática da presença de tumores em imagens médicas.
- **Regressão:** é similar ao problema de classificação, com a diferença que o domínio do atributo alvo é contínuo, ou seja, $\mathcal{Y} = \mathbb{R}$ ou $\mathcal{Y} = \mathbb{R}^d$. Nesse caso, o modelo a ser construído, denominado modelo de regressão, visa estimar um valor contínuo do atributo y para instâncias futuras. Um exemplo de problema de regressão é o cálculo da estimativa da demanda de um determinado produto, dadas suas características principais (MAIMON; ROKACH, 2010).

O principal objetivo neste trabalho é o desenvolvimento de um algoritmo de classificação

semisupervisionada, para aplicação na análise de dados obtidos a partir de imagens de satélite de baixa resolução. Por isso, o foco nesta seção é na tarefa de classificação.

Um exemplo hipotético da aplicação de classificação é apresentado na Figura 3. Nesse exemplo, é considerado um conjunto de treinamento X , em que $\mathcal{X} = \mathbb{R}^2$ e o número de classes $C = 2$, cujas instâncias de cada classe são exibidas na Figura 3a. A partir de X , um classificador é treinado e, como resultado do treinamento, identifica uma possível fronteira de separação de classes, ilustrada em tracejado na Figura 3b. Após o treinamento, ao submeter uma instância x não rotulada ao classificador do exemplo, ela será classificada como positiva se for localizada na região azul clara (abaixo da reta correspondente à fronteira), ou como negativa, caso contrário.

Figura 3 – Exemplo de conjunto hipotético de instâncias rotuladas com dois atributos (a_1, a_2). Usando um algoritmo de classificação supervisionada, um dos possíveis resultados de treinamento do classificador é a definição da fronteira de separação das classes ilustrada.

Fonte: Elaborada pelo autor.

Alguns dos métodos de classificação supervisionados bastante consolidados na literatura e frequentemente utilizados como *baseline*, inclusive neste trabalho, são o K-Nearest-Neighbors (KNN) (HAN; KAMBER; PEI, 2011), o Naive Bayes (MITCHELL, 1997) e o *Support Vector Machines* (SVM) (VAPNIK; VAPNIK, 1998). O KNN consiste em associar uma instância não rotulada à classe majoritária dentre as K instâncias rotuladas mais próximas a ela. O Naive Bayes é um classificador estatístico baseado no teorema de Bayes. A característica principal do Naive Bayes é que o método assume que os atributos do conjunto A são condicionalmente independentes, dado o atributo de classe y (MITCHELL, 1997). Sob essa suposição, um número muito menor de cálculos é necessário para resolver o problema de classificação usando o teorema de Bayes. O método SVM visa encontrar o hiperplano de máxima margem, ou seja, que melhor divida o espaço de instâncias em duas regiões, sendo que cada uma contém instâncias de apenas uma das classes (classificação binária) (MAIMON; ROKACH, 2010). No caso em que as classes não são linearmente separáveis, a SVM utiliza uma transformação e funções Kernel de modo que as classes sejam linearmente separáveis no espaço transformado. O hiperplano encontrado

no espaço transformado corresponde a uma superfície no espaço original, capaz de separar as instâncias das duas classes.

2.3 Aprendizado semissupervisionado

O aprendizado semissupervisionado pode ser visto como um paradigma intermediário entre o supervisionado e o não supervisionado, em que são considerados dados rotulados e não rotulados para o treinamento de um modelo (ZHU *et al.*, 2009). No caso semissupervisionado, o conjunto de treinamento $X = \{x_1, x_2, \dots, x_r, x_{r+1}, \dots, x_n\}$ pode ser dividido nos subconjuntos (ZHU *et al.*, 2009):

1. **Conjunto de instâncias rotuladas:** $X_R = \{(x_1, y_1), (x_2, y_2), \dots, (x_r, y_r)\}$, cujas instâncias (x_i, y_i) carregam valores para os atributos de $A = \{a_1, a_2, \dots, a_D\}$ e para o atributo de classe ou rótulo y .
2. **Conjunto de instâncias não rotuladas:** $X_{NR} = \{x_{r+1}, x_{r+2}, \dots, x_n\}$, que contém instâncias x_j , cujo rótulo é desconhecido.

Essa é a configuração padrão e mais utilizada para aprendizado semissupervisionado, porém há outros tipos de supervisão, como do tipo “as instâncias x_i e x_j possuem a mesma classe” ou “as instâncias x_i e x_j possuem classes distintas” (CHAPELLE; SCHÖLKOPF; ZIEN, 2006). De modo geral, a maioria das estratégias em aprendizado semissupervisionado consiste em estender uma abordagem supervisionada, por meio do uso de dados não rotulados, ou estender uma abordagem não supervisionada, por meio da utilização de dados, conhecimento ou qualquer recurso que forneça informação adicional supervisionada ao problema tratado (ZHU *et al.*, 2009).

As principais tarefas de aprendizado semissupervisionado são descritas a seguir:

- **Classificação semissupervisionada:** assim como no caso supervisionado, o atributo y corresponde a um ou mais valores categóricos ou numéricos, e representa a classe ou rótulo das instâncias. O objetivo da classificação semissupervisionada é treinar um classificador baseado em um conjunto de treinamento composto por instâncias rotuladas (de classe conhecida) e não rotuladas (de classe desconhecida). Em geral, é desejado que a qualidade de classificação obtida seja superior ao obtido por um classificador supervisionado treinado apenas usando as instâncias rotuladas do mesmo problema (ZHU *et al.*, 2009).
- **Agrupamento semissupervisionado:** como uma extensão do agrupamento não supervisionado, é considerado um conjunto de instâncias não rotuladas em adição a algum tipo de informação supervisionada, útil ao processo de agrupamento. Um exemplo de agrupamento semissupervisionado considera restrições do tipo “duas instâncias devem/não podem pertencer ao mesmo agrupamento”, ou restrições no tamanho de agrupamentos. O

principal objetivo no agrupamento semissupervisionado é obter uma qualidade de agrupamento superior ao do caso não supervisionado (CHAPELLE; SCHÖLKOPF; ZIEN, 2006).

Neste trabalho, a tarefa de interesse é a classificação semissupervisionada, que é especialmente útil em cenários com baixa quantidade de dados rotulados e alta disponibilidade de dados não rotulados, nos quais o uso de classificação supervisionada pode produzir resultados insatisfatórios devido ao pequeno número de instâncias rotuladas (CHAPELLE; SCHÖLKOPF; ZIEN, 2006).

Existem diversos cenários reais que apresentam baixa disponibilidade de dados rotulados. A dificuldade na obtenção do atributo de classe y pode ocorrer por diversos fatores, como dependência de supervisão manual de dados por especialistas, necessidade do uso de dispositivos especiais de coleta, elevado custo financeiro ou tempo necessário para realização de experimentos, entre outros (ZHU *et al.*, 2009). Por outro lado, em muitos desses cenários reais, há disponibilidade de grandes volumes de dados não rotulados, situação que favorece a aplicação de classificação semissupervisionada. Alguns exemplos desses cenários são:

- Em detecção de *spam* em *e-mails*, existe o esforço humano necessário para rotulação da mensagem, além da dependência da interpretação do indivíduo. Por outro lado, há bilhões de *e-mails* sendo enviados todos os dias.
- Em reconhecimento de palavras escritas à mão, o acesso a documentos é extremamente fácil. Entretanto, como uma mesma letra pode ser escrita diferentemente por pessoas distintas, são necessárias várias instâncias de treinamento para uma mesma palavra. Portanto, o tempo e esforço humano necessário para obter um conjunto de palavras escritas por um número significativo de pessoas são elevados.
- Na área de medicina, há disponibilidade de bases de dados volumosas contendo dados no formato de séries temporais, como EEG e ECG (eletroencefalograma e eletrocardiograma, respectivamente). No entanto, pela complexidade apresentada pela série temporal, o processo de supervisão manual pelo médico ou especialista tem custo significativo.

Sabe-se, portanto, que a possibilidade de aplicação prática de métodos de classificação semissupervisionada é muito grande. Porém, ela só apresenta garantias de melhoria na classificação em relação ao caso supervisionado se existe uma relação entre a distribuição dos dados não rotulados $p(x)$ e o atributo de classe y (ZHU *et al.*, 2009). Portanto, para aplicação de um método de classificação semissupervisionada, é necessário adotar certas suposições a respeito dos dados e das classes do problema, em que algumas das principais suposições são (CHAPELLE; SCHÖLKOPF; ZIEN, 2006; ZHU *et al.*, 2009):

- **Suposição semissupervisionada de suavidade:** se duas instâncias x_i e x_j estão próximas, em uma região de alta densidade de dados, então seus rótulos y_i e y_j devem ser iguais. Isso significa que duas instâncias devem pertencer à mesma classe se residem em uma região densa e a distância entre elas é pequena, segundo uma função de distância. Se duas instâncias estiverem próximas mas em uma região de baixa densidade de dados, seus rótulos não devem necessariamente ser iguais.
- **Suposição de agrupamento:** se duas instâncias x_i e x_j pertencem a um mesmo agrupamento, elas provavelmente pertencem à mesma classe. Essa suposição não assume que cada classe se organiza em um único agrupamento, e sim que é improvável que instâncias em um mesmo agrupamento apresentem rótulos diferentes.
- **Suposição de separação por baixa densidade:** a fronteira de separação de classes reside em uma região de baixa densidade de dados. Essa suposição apresenta grande similaridade com a suposição de agrupamento, já que, se instâncias de um mesmo agrupamento possuem a mesma classe, a fronteira de separação de classes não deve ter intersecção com nenhum agrupamento, portanto residindo em uma região de baixa densidade de dados.
- **Suposição de *manifold* (variedade):** as instâncias de um conjunto com alta dimensionalidade residem em um *manifold* de baixa dimensionalidade. Adotando essa suposição, pode-se trabalhar no espaço de menor dimensionalidade e evitar a maldição da dimensionalidade e problemas como a necessidade de um número exponencialmente maior de instâncias para tarefas estatísticas como estimativa de densidades e o fato de que distâncias entre pares de instâncias tendem a ser mais similares e menos expressivas no espaço com alta dimensionalidade.
- **Suposição baseada em grafos:** se duas instâncias x_i e x_j , modeladas como os vértices v_1 e v_2 de um grafo $G(V, E)$, são conectadas por uma aresta $(v_1, v_2) \in E$ e peso w_{ij} que represente alta similaridade ou baixa dissimilaridade entre os vértices, os rótulos y_i e y_j dessas instâncias devem ser iguais. Vale mencionar que a suposição vale para qualquer par de instâncias cujos vértices estão conectados, sejam elas rotuladas ou não rotuladas.

Na Figura 4, é ilustrado um exemplo hipotético da aplicação de classificação supervisionada e semissupervisionada, em que é dado um conjunto de treinamento X com dois atributos numéricos (a_1, a_2) e um atributo de classe que pode assumir dois valores (c_1, c_2) , em que c_1 é denominada classe positiva e c_2 classe negativa.

No exemplo, o conjunto X é composto por apenas 4 instâncias rotuladas (subconjunto X_R) e mais de 500 instâncias não rotuladas (subconjunto X_{NR}). Para o caso supervisionado, é considerado apenas o conjunto X_R , exibido na Figura 4a. O resultado produzido pelo treinamento de um classificador supervisionado usando X_R é apresentado na Figura 4b, em que a linha tracejada corresponde à fronteira de separação das classes. Para o treinamento do classificador

Figura 4 – Exemplo da comparação entre classificação supervisionada e semissupervisionada, para um conjunto X com poucas instâncias rotuladas e grande número de instâncias não rotuladas.

Fonte: Elaborada pelo autor.

semissupervisionado, são consideradas todas as instâncias do conjunto X , rotuladas e não rotuladas, mostradas na Figura 4c. Assumindo que as duas classes do problema se organizam naturalmente em agrupamentos (suposição de agrupamento), pode-se obter o resultado de treinamento ilustrado na Figura 4d.

Ao comparar os resultados de treinamento dos classificadores supervisionado e semissupervisionado, é possível concluir que a fronteira de separação de classes mais adequada para o problema de classificação dado foi obtida pelo classificador semissupervisionado. Isso se deve a dois fatores: 1) o pequeno número de instâncias rotuladas não é capaz de informar ao classificador as características relativas às estruturas das duas classes, nesse caso; 2) a suposição de que as duas classes se organizam no formato de agrupamento é, de fato, verdadeira, o que possibilita o uso das instâncias não rotuladas para melhorar o resultado da classificação, em relação ao caso supervisionado.

Diversos métodos propostos na literatura de aprendizado semissupervisionado podem

ser divididos em seis principais abordagens: 1) modelos generativos, 2) separação por baixa densidade, 3) *cluster-then-label*, 4) *self-training*, 5) *co-training* e 6) baseada em grafos. Cada abordagem de classificação semissupervisionada utiliza estratégias e assume suposições diferentes em relação aos dados analisados. As principais características de cada abordagem de classificação semissupervisionada são apresentadas nas próximas seções, com maior destaque para a abordagem baseada em grafos, adotada neste trabalho de mestrado.

2.3.1 Modelos generativos

Métodos generativos supervisionados assumem que existe uma mistura de distribuições de probabilidade, a partir das quais são geradas as instâncias pertencentes a cada classe de um problema (ZHU, 2005). Nesse caso, é assumido um modelo $p(x, y) = p(y)p(x|y)$, no qual $p(x|y)$ corresponde a uma mistura de distribuições identificáveis. O objetivo de métodos generativos é encontrar as distribuições que geram os dados (ZHU *et al.*, 2009). Por exemplo, em um problema de classificação binária, pode ser assumido que os dados são gerados por uma mistura de duas distribuições gaussianas, cujos parâmetros são desconhecidos. O objetivo é utilizar dados rotulados e não rotulados para estimativa dos parâmetros da mistura de distribuições. Essa análise pode ser feita em duas etapas: 1) com grande volume de dados não rotulados, é possível identificar as distribuições da mistura, porém sem informação de classe; 2) associar cada distribuição a uma das classes do problema usando os dados rotulados disponíveis. Para mistura de distribuições como a Gaussiana, Multinomial, entre outras, o algoritmo EM (*Expectation-Maximization*) (DEMPSTER; LAIRD; RUBIN, 1977) tem sido a principal escolha para a estimativa dos parâmetros, quando há dados não rotulados (NIGAM *et al.*, 2000; ZHU *et al.*, 2009).

Um exemplo hipotético é ilustrado na Figura 5, em que assume-se que existem duas classes no problema, e os dados são gerados por uma mistura de distribuições Gaussianas. Na Figura 5a são mostradas as instâncias rotuladas do conjunto de treinamento para o caso supervisionado, e na Figura 5b é ilustrado o modelo aprendido, em que a linha tracejada indica a fronteira de separação entre as duas classes. Um método baseado em modelos generativos semissupervisionado considera um conjunto de treinamento composto por instâncias rotuladas e não rotuladas, conforme exibido na Figura 5c. A presença de um grande número de instâncias não rotuladas permite obter uma estimativa mais fiel em relação às distribuições que geram os dados, como mostrado no modelo obtido usando a abordagem semissupervisionada na Figura 5d.

A principal suposição adotada pelos métodos generativos é definida a seguir (ZHU *et al.*, 2009):

Suposição de modelo de mistura: os dados do problema são gerados por uma mistura de distribuições, e para o modelo estimado, o número de componentes da mistura, a probabilidade $p(y)$ e a probabilidade condicional $p(x|y)$ estão corretas.

Figura 5 – Exemplo da comparação entre classificação supervisionada e semissupervisionada usando modelos generativos. A fronteira de separação de classes é ilustrada em linha tracejada.

Fonte: Adaptada de [Zhu \(2005\)](#).

Se essa suposição é verdadeira, há garantias de melhora na qualidade de classificação, ao utilizar dados não rotulados para estimativa do modelo ([RATSABY; VENKATESH, 1995](#); [CASTELLI; COVER, 1995](#); [NIGAM *et al.*, 2000](#)). No entanto, nos casos em que a suposição em relação ao modelo de mistura de distribuições não é correta, os resultados podem ser inferiores aos de classificação supervisionada ([COZMAN *et al.*, 2003](#); [ZHU *et al.*, 2009](#)). Em diversos trabalhos, tem sido mostrada a superioridade de métodos que utilizam a abordagem semissupervisionada de modelos generativos em relação a métodos de classificação supervisionada ([NIGAM *et al.*, 2000](#); [FUJINO; UEDA; SAITO, 2005](#)).

2.3.2 Cluster-then-label

Na abordagem de *cluster-then-label*, dado um conjunto de treinamento X , o objetivo é aplicar um método de agrupamento não supervisionado e combinar as informações resultantes da análise de agrupamento e as instâncias rotuladas para classificar cada instância não rotulada

de X . A abordagem de *cluster-then-label*, embora não probabilística, utiliza a mesma ideia básica da abordagem semissupervisionada de modelos generativos, que é a de analisar dados não rotulados para identificar estruturas correspondentes a classes do problema (DEMIRIZ; BENNETT; EMBRECHTS, 1999; DARA *et al.*, 2002; ZENG *et al.*, 2003; DAI *et al.*, 2007).

A suposição adotada pela abordagem de *cluster-then-label* é a suposição de agrupamento, de que instâncias em um mesmo agrupamento tendem a pertencer à mesma classe. Isso implica que todo agrupamento de instâncias corresponde a uma classe, embora as instâncias de uma classe possam estar distribuídas em mais de um agrupamento.

Um exemplo hipotético do uso da abordagem de *cluster-then-label* é ilustrado na Figura 6. Dado um conjunto de instâncias rotuladas e não rotuladas, exibidas na Figura 6a, em que cada agrupamento de instâncias corresponde a apenas uma classe, ao utilizar o algoritmo supervisionado KNN, são obtidos os resultados de classificação mostrados na Figura 6b. Como se sabe que cada agrupamento está associado a uma classe, pode-se concluir que o resultado obtido não é satisfatório. Por outro lado, ao utilizar a abordagem de *cluster-then-label*, pode-se aplicar um algoritmo de agrupamento para obter os agrupamentos de instâncias ilustrados na Figura 6c. Por fim, pode-se rotular cada agrupamento, de acordo com a classe majoritária, dentre as instâncias rotuladas de cada agrupamento, obtendo-se assim o resultado de classificação esperado, exibido na Figura 6d.

2.3.3 Separação por baixa densidade

Na abordagem de separação por baixa densidade, o objetivo é encontrar uma fronteira de separação de classes que não resida em uma região de alta densidade de dados rotulados e não rotulados (ZHU *et al.*, 2009). O método mais comum da abordagem de separação por baixa densidade é a *Transductive Support Vector Machines* (TSVM) ou *Semi-supervised Support Vector Machines* (S3VM). Na TSVM, de maneira similar à SVM supervisionada, busca-se encontrar um hiperplano capaz de separar completamente as classes do problema. No entanto, para o caso semissupervisionado, deseja-se que não haja instâncias rotuladas e não rotuladas dentro da margem do hiperplano de separação das classes. Na Figura 7, é possível notar a diferença do caso supervisionado e semissupervisionado, e a influência das instâncias não rotuladas na definição do hiperplano de separação de classes.

O problema de otimização da TSVM é de natureza não convexa e a solução apresenta múltiplos mínimos locais, de modo que algoritmos de aprendizado podem convergir para um mínimo local subótimo e não encontrar a solução correspondente ao ótimo global (ZHU *et al.*, 2009). Encontrar de maneira eficiente uma solução próxima à ótima tem sido o objetivo em pesquisas envolvendo o estudo de TSVMS (ZHU *et al.*, 2009). A primeira implementação amplamente utilizada foi a *SVMLight* (JOACHIMS, 1999). Desde então, diversas técnicas de otimização para TSVMS têm sido propostas (CHAPELLE; SINDHWANI; KEERTHI, 2008). A TSVM implementada na *SVMLight* é utilizada nos experimentos apresentados no Capítulo 6.

Figura 6 – Exemplo da comparação entre o uso do método KNN e da abordagem de *cluster-then-label*.

Fonte: Elaborada pelo autor.

2.3.4 Selftraining

Na abordagem de *self-training* ou auto-treinamento, um classificador é treinado com base em suas próprias previsões (CHAPELLE; SCHÖLKOPF; ZIEN, 2006). Seja X um conjunto de treinamento, composto por instâncias rotuladas (conjunto X_R) e instâncias não rotuladas (conjunto X_{NR}). Inicialmente, o classificador f , treinado a partir de X_R , é usado para classificar as instâncias de X_{NR} . O algoritmo então escolhe um subconjunto das instâncias de X_{NR} recém rotuladas e assume que essas previsões são corretas e, portanto, podem ser usadas para o treinamento do classificador. Então, esse subconjunto de instâncias é retirado de X_{NR} e incorporado a X_R . O classificador é retreinado com base no novo conjunto X_R e usado para classificar as instâncias restantes de X_{NR} . O processo se repete até que não haja mais instâncias em X_{NR} .

Na abordagem de *self-training* é assumida apenas uma suposição: de que suas próprias previsões anteriores, ou ao menos um subconjunto composto pelas previsões que apresentam maior confiabilidade, são corretas.

Figura 7 – Exemplo da comparação entre o uso da SVM (supervisionada) e da TSVM (semissupervisionada) para encontrar o hiperplano de máxima margem.

Fonte: Elaborada pelo autor.

É fácil notar que, quando a suposição não é verdadeira, ou seja, o classificador é repetidamente treinado com informações incorretas, o desempenho do algoritmo não é satisfatório (ZHU *et al.*, 2009). As principais vantagens da abordagem *self-training* são a simplicidade do algoritmo, e a possibilidade de usar um classificador f arbitrário, seja supervisionado ou semisupervisionado (WEI; KEOGH, 2006; RATANAMAHATANA; WANICHSAN, 2008; ZHU *et al.*, 2009; NGUYEN; LI; NG, 2011).

2.3.5 Co-training

A ideia na abordagem de *co-training* é utilizar duas visões diferentes dos dados a serem classificados (CHAPELLE; SCHÖLKOPF; ZIEN, 2006). Seja X o conjunto de treinamento e A o conjunto de atributos, tem-se duas visões A_1 e A_2 , em que cada visão corresponde a um subconjunto dos atributos de A ($A_1 \subset A$ e $A_2 \subset A$), e não deve haver intersecção entre as visões ($A_1 \cap A_2 = \emptyset$). No *co-training* são utilizados dois classificadores f_1 e f_2 , que são treinados independentemente, sendo cada um especializado em uma visão dos atributos (CHAPELLE; SCHÖLKOPF; ZIEN, 2006). De maneira similar ao *self-training*, as predições mais confiáveis de f_1 são usadas para treinamento de f_2 e vice-versa (ZHU *et al.*, 2009). Desse modo, cada classificador utiliza o resultado de predições com alta confiabilidade em dados não rotulados como informação para o treinamento do outro classificador. Nesse caso, mesmo em um cenário que apresente escassez de dados rotulados, uma grande quantidade de dados não rotulados permite o aprimoramento do treinamento de cada classificador.

É importante mencionar que os classificadores f_1 e f_2 podem ser quaisquer classificadores adequados ao problema. Isso é uma vantagem importante do *co-training*, pois com o conhecimento prévio do domínio de aplicação, pode-se escolher classificadores especializados

para análise dos atributos pertencentes a cada visão. Para o *co-training*, é necessário que cada classificador utilize uma medida de confiabilidade, para que possam determinar quais instâncias recém rotuladas devem ser usadas para treinamento do outro classificador.

Na abordagem de *co-training*, são assumidas duas principais suposições:

1. Cada visão deve ser suficiente para treinar um bom classificador, caso haja um número suficiente de instâncias rotuladas.
2. As duas visões devem ser condicionalmente independentes dado o valor do atributo de classe.

O algoritmo *co-training*, proposto em (BLUM; MITCHELL, 1998), e suas variações (COLLINS; SINGER, 1999; NIGAM; GHANI, 2000; GOLDMAN; ZHOU, 2000; LEVIN; VIOLA; FREUND, 2003; ZHOU; LI, 2005; WAN, 2009) têm sido aplicados em diversos domínios, como em processamento de texto, classificação de sentimento e sistemas de detecção visual.

2.3.6 Baseada em grafos

Seja X um conjunto de treinamento, composto por instâncias rotuladas $X_R = \{(x_1, y_1), (x_2, y_2), \dots, (x_r, y_r)\}$ e instâncias não rotuladas $X_{NR} = \{x_{r+1}, x_{r+2}, \dots, x_n\}$. A abordagem baseada em grafos modela o conjunto de treinamento como um grafo $G(V, E)$, em que $v_i \in V$ é o vértice correspondente à instância x_i e $(v_i, v_j) \in E$ é a aresta que conecta os vértices v_i e v_j (ZHU *et al.*, 2009).

Os vértices $\{v_1, v_2, \dots, v_r\}$ são associados ao valores de classe $\{y_1, y_2, \dots, y_r\}$, enquanto que os vértices $\{v_{r+1}, v_{r+2}, \dots, v_n\}$ são não rotulados. Após a construção do grafo, o objetivo é rotular o conjunto de vértices não rotulados por meio das arestas que os conectam aos vértices rotulados. Na maioria dos casos, uma aresta (v_i, v_j) indica que existe uma conexão entre esses vértices, e o peso da aresta w_{ij} indica a similaridade entre eles.

De acordo com a suposição baseada em grafos, se o valor w_{ij} é grande, então a similaridade entre os vértices v_i e v_j é grande, e espera-se que os rótulos y_i e y_j sejam os mesmos. Portanto, o modo como o grafo é construído, ou seja, a maneira como os vértices são conectados é parte importante de um método de classificação semissupervisionada baseado em grafos. As maneiras mais comumente utilizadas para se definir os pesos das arestas do grafo são (ZHU *et al.*, 2009):

- grafo totalmente conexo, em que o peso da aresta (v_i, v_j) é calculado como:

$$w_{ij} = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) \quad (2.1)$$

em que $\|x_i - x_j\|$ geralmente corresponde à distância Euclidiana entre o par de instâncias, e σ é um parâmetro da função Gaussiana. O peso w_{ij} é calculado usando a função Gaussiana, que nesse caso, indica similaridade entre as instâncias, pois w_{ij} cresce à medida que a distância $\|x_i - x_j\|$ decresce.

- grafo *KNN*, no qual existe uma aresta (v_i, v_j) se x_j pertence ao conjunto das K instâncias mais próximas a x_i , de acordo com uma função de distância $Dist(x_i, x_j)$. Se x_i não pertence ao conjunto de vizinhos de x_j , nem x_j pertence ao conjunto de vizinhos de x_i , não existe aresta que conecta v_i e v_j , o que é denotado pelo peso $w_{ij} = 0$. Se houver aresta entre um par de vértices, o valor do peso da aresta pode ser o mesmo para todas as arestas do grafo (arestas não ponderadas) ou pode corresponder a uma medida de similaridade envolvendo o valor da distância entre eles, como no caso da função Gaussiana (arestas ponderadas).
- grafo *eNN*, no qual v_i é conectado a v_j apenas se $\|x_i - x_j\| \leq e$. Usando a mesma abordagem que no grafo *KNN*, as arestas podem ser ponderadas ou não ponderadas.

Apesar dessas estratégias, a construção do grafo deve sempre estar associada ao domínio de aplicação. Quanto mais conhecimento a respeito do domínio e sobre como as instâncias se relacionam, melhor será a construção do grafo e, provavelmente, melhores serão os resultados de classificação obtidos.

Após a construção do grafo G , o objetivo é utilizar uma estratégia para classificação dos vértices não rotulados. A seguir são descritas as principais estratégias propostas na literatura de classificação semissupervisionada baseada em grafos.

- **Propagação de rótulos:** é um processo iterativo em que cada vértice rotulado pertencente à classe $c_i \in C = \{c_1, c_2, \dots, c_m\}$ propaga um sinal com valor de seu rótulo c_i para seus vizinhos a cada iteração. Os vértices não rotulados recebem e armazenam sinais correspondentes a cada um dos m rótulos e, na iteração seguinte repassam os rótulos armazenados para seus vizinhos, sejam eles rotulados ou não rotulados. Aqui, os vértices rotulados não armazenam os sinais de rótulo recebido, apenas enviam sinais com valor de rótulo correspondente à sua classe. Então, seja P uma matriz de probabilidades de transição, na qual cada elemento é calculado de acordo com a Equação 2.2:

$$P_{ij} = \frac{w_{ij}}{\sum_{k=1}^n w_{ik}} \quad (2.2)$$

em que n é o número de de instâncias no conjunto de treinamento.

Seja F uma matriz $n \times m$, em que F_{ij} corresponde à quantidade do rótulo c_j que está armazenado no vértice v_i , e Y a matriz correspondente aos rótulos iniciais dos vértices rotulados (se o vértice rotulado v_i pertence à classe c_j , então $Y_{ij} = 1$, caso contrário $Y_{ij} = 0$). O algoritmo de propagação de rótulos é definido pelo seguintes passos:

1. $F = PF$
2. $F_R = Y$
3. Repetir a partir do passo 1 até convergência.

em que F_R é a parte da matriz F correspondente aos vértices rotulados.

- **Mincut:** é usada uma estratégia similar à de fluxo máximo em redes, em que os vértices rotulados da classe positiva correspondem às fontes e os vértices rotulados da classe negativa correspondem ao poço ou sumidouro. Como numa rede de resíduos, o fluxo se inicia na fonte e termina no poço. O objetivo é encontrar o conjunto de arestas com soma mínima de pesos que, ao serem retirados, interrompem completamente o fluxo que vai dos vértices positivos até os vértices negativos. Esse conjunto de vértices é denominado o corte mínimo. Após o corte no grafo, todos os vértices conectados à fonte são rotulados como positivos, e todos os vértices conectados ao poço são rotulados como negativos.

Assim, o objetivo é minimizar o valor

$$\sum_{i,j:f(x_i) \neq f(x_j)} w_{ij} \quad (2.3)$$

em que $f(x)$ é o valor de rótulo inferido para a instância x . Esse valor corresponde ao corte mínimo, pois se a aresta w_{ij} é removida, obrigatoriamente $f(x_i) \neq f(x_j)$. Vale mencionar que essa estratégia somente é aplicável para o caso de classificação binária. Para resolução da Equação 2.3, é possível utilizar algoritmos com complexidade de tempo polinomial para encontrar o corte mínimo no grafo.

- **Função harmônica:** usando a suposição baseada em grafos, a abordagem de função harmônica tem por objetivo encontrar uma função $f : V \mapsto \mathbb{R}$ no grafo G , em que V é o conjunto de vértices, tal que ela seja suave em todo o grafo. Para isso, f é definida como a seguinte função harmônica (ZHU *et al.*, 2009):

$$f(x_i) = \begin{cases} y_i, & \text{se } x_i \text{ é rotulada} \\ \frac{\sum_{k=1}^n w_{jk} f(x_k)}{\sum_{k=1}^n w_{jk}}, & \text{se } x_i \text{ é não rotulada} \end{cases} \quad (2.4)$$

em que w_{ij} é o peso da aresta entre os vértices v_i e v_j e representa a similaridade entre eles.

Isso significa que a função $f(x)$ é igual ao valor da classe da instância, quando x é rotulada, e quando x é não rotulada, $f(x)$ corresponde à média dos valores de f nos vizinhos do vértice que representa x no grafo, o que torna a f uma função suave no grafo G . A solução harmônica $\Delta f = 0$ é escrita em termos da matriz L do Laplaciano do grafo G :

$$L = D - W \quad (2.5)$$

em que W é a matriz de adjacência de G e D a matriz diagonal, em que $D_{ii} = \sum_{j=1}^n w_{ij}$.

Pode-se decompor L em quatro submatrizes:

$$L = \begin{bmatrix} L_{rr} & L_{ru} \\ L_{ur} & L_{uu} \end{bmatrix} \quad (2.6)$$

em que r corresponde aos índices das instâncias rotuladas e u os índices das instâncias não rotuladas. Seja $\mathbf{f} = (f(x_1), f(x_2), \dots, f(x_{r+u}))^T$ o vetor da função f em todos os vértices, pode-se particionar $\mathbf{f} = (\mathbf{f}_r, \mathbf{f}_u)$. Considerando $\mathbf{y}_r = (y_1, y_2, \dots, y_r)^T$ o vetor de rótulos dos vértices rotulados, a solução harmônica de f pode ser calculada pela Equação 2.8.

$$\mathbf{f}_r = \mathbf{y}_r \quad (2.7)$$

$$\mathbf{f}_u = -L_{uu}^{-1} L_{ur} \mathbf{y}_r \quad (2.8)$$

Dessa maneira, são obtidos os valores $\mathbf{f}_u = (f(x_{r+1}), f(x_{r+2}), \dots, f(x_{r+u}))^T$ dos rótulos inferidos pelo classificador para cada instância não rotulada.

A abordagem baseada em grafos tem sido uma das mais exploradas em classificação semissupervisionada, principalmente a partir do início dos anos 2000. Em (BLUM; CHAWLA, 2001), foi proposta a estratégia semissupervisionada de rotulação do grafo usando a estratégia *mincut*. Um método de rotulação em grafo usando Markov *random walks* considerando dados rotulados e não rotulados é proposto em (JAAKKOLA; SZUMMER, 2002). Diversos trabalhos focam na construção de kernels para grafos, como kernels obtidos a partir da decomposição do Laplaciano do grafo e *diffusion* kernels (KONDOR; LAFFERTY, 2002; SMOLA; KONDOR, 2003; ZHU *et al.*, 2004). A estratégia de rotulação do grafo usando funções harmônicas e campos Gaussianos é proposta em (ZHU *et al.*, 2003). Em (JOACHIMS *et al.*, 2003), os autores propõem um método transdutivo de aprendizado semissupervisionado baseado no particionamento do grafo, cujo treinamento é feito usando métodos espectrais. Um método que usa uma estratégia similar à propagação de rótulos e agrupamento espectral é definido em (ZHOU *et al.*, 2004). Métodos semissupervisionados escaláveis para classificação de grandes grafos são propostos em (BELKIN; MATVEEVA; NIYOGI, 2004; LIU; HE; CHANG, 2010). Diversos outros métodos que priorizam à construção do grafo também foram propostos (HEIN; AUDIBERT; LUXBURG, 2007; WANG; ZHANG, 2008; SZLAM; MAGGIONI; COIFMAN, 2008). Os métodos de classificação semissupervisionada baseada em grafos também têm elevado valor prático, sendo aplicados em domínios como classificação de imagens (CAMPS-VALLS; MARSHEVA; ZHOU, 2007), classificação de emoções (GOLDBERG; ZHU, 2006), análise léxica (RAO; RAVICHANDRAN, 2009), entre outros.

Um método de classificação semissupervisionada que utiliza a abordagem baseada em grafos e é relacionado à técnica proposta neste trabalho de mestrado é o *Linear Neighborhood Propagation* (LNP) (WANG; ZHANG, 2008). O método LNP, assim como a técnica proposta,

utiliza uma estratégia baseada em propagação de rótulos no grafo para classificação das instâncias não rotuladas. A principal diferença é que cada método utiliza uma estratégia distinta para modelar o conjunto de treinamento como grafo. Devido à grande similaridade com a técnica proposta neste trabalho e por ser um dos principais trabalhos que aplicam propagação de rótulos propostos na literatura de classificação semissupervisionada baseada em grafo, o método LNP foi utilizado no experimento de análise comparativa, apresentado no Capítulo 6. As principais características do LNP são descritas a seguir.

Linear Neighborhood Propagation

O método LNP, diferentemente de muitos métodos de classificação semissupervisionada baseada em grafos, não utiliza a função Gaussiana para calcular os pesos das arestas do grafo. Para indicar a similaridade entre pares de vértices, o método LNP assume que uma dada instância x_i , representada pelo vértice v_i , pode ser reconstruída pela combinação linear de seus K vizinhos mais próximos. Para isso, é necessário que $x \in \mathbb{R}^D$ seja um vetor D -dimensional de valores reais.

Seja X um conjunto de treinamento, composto por instâncias rotuladas $X_R = \{(x_1, y_1), (x_2, y_2), \dots, (x_r, y_r)\}$ e instâncias não rotuladas $X_{NR} = \{x_{r+1}, x_{r+2}, \dots, x_n\}$. Inicialmente é calculada a vizinhança $\mathcal{N}(x_i)$ para cada instância $x_i \in X$, em que $x_j \in \mathcal{N}(x_i)$ se x_j estiver entre as K instâncias menos dissimilares de x_i , de acordo com uma função de distância $Dist(x_i, x_j)$. Para calcular o peso w_{ij} da aresta que conecta os vértices v_i e v_j , assume-se que cada instância x_i pode ser reconstruída pela combinação linear de seus vizinhos. Assim, se $x_j \in \mathcal{N}(x_i)$, o peso da aresta (v_i, v_j) equivale ao coeficiente de combinação linear de x_j , para formação de x_i . Portanto, o objetivo é minimizar o valor de erro:

$$\varepsilon = \sum_i \|x_i - \sum_{j: x_j \in \mathcal{N}(x_i)} w_{ij} x_j\|^2 \quad (2.9)$$

em que w_{ij} é a contribuição de x_j para combinação de x_i . Quanto mais similares são x_i e x_j , maior será w_{ij} , em que $w_{ij} = 1$ se $x_i = x_j$. A Equação 2.9 é restrita a $\sum_{j: x_j \in \mathcal{N}(x_i)} w_{ij} = 1$ e $w_{ij} \geq 0$.

Esse problema equivale a resolver n problemas de otimização quadrática:

$$\min_{w_{ij}} \sum_{j, k: x_j, x_k \in \mathcal{N}(x_i)} w_{ij} G_{jk}^i w_{ik} \quad (2.10)$$

sujeito a $\sum_j w_{ij} = 1$ e $w_{ij} \geq 0$, em que G_{jk}^i corresponde ao elemento (j, k) da matriz *Gram* local definida como (WANG; ZHANG, 2008):

$$G^i = \begin{pmatrix} (x_i - x_{i_1})^T (x_i - x_{i_1}) & \cdots & (x_i - x_{i_1})^T (x_i - x_{i_k}) \\ (x_i - x_{i_2})^T (x_i - x_{i_1}) & \cdots & (x_i - x_{i_2})^T (x_i - x_{i_k}) \\ \vdots & \ddots & \vdots \\ (x_i - x_{i_K})^T (x_i - x_{i_1}) & \cdots & (x_i - x_{i_K})^T (x_i - x_{i_k}) \end{pmatrix} \quad (2.11)$$

em que x_{ij} é o j -ésimo vizinho da vizinhança $\mathcal{N}(x_i)$.

Após a resolução dos problema de otimização quadrática, obtém-se a matriz W , em que $W(i, j) = w_{ij}$. Em seguida, é aplicada uma estratégia similar à propagação de rótulos, em que cada vértice envia o rótulo armazenado na iteração atual a seus vizinhos, por meio das arestas de vizinhança. Seja F a matriz correspondente à quantidade armazenada do rótulo de cada classe, em cada vértice e T a matriz de rótulos iniciais dos vértices rotulados. O processo iterativo de propagação de rótulo é então definido:

$$F^{t+1} = \alpha W F^t + (1 - \alpha) T \quad (2.12)$$

em que α indica a porcentagem dos rótulos que um vértice recebe de seus vizinhos, e $(1 - \alpha)$ é a porcentagem do rótulo inicial que é retido pelo vértice.

Após convergência, seja F^* a matriz de rótulos final, cada vértice não rotulado é associado à classe cujo rótulo tiver sido recebido pela propagação de seus vizinhos em maior quantidade. O rótulo final y_i de cada instância não rotulada x_i é dado pela Equação 2.13.

$$y_i = \operatorname{argmax}_{1 \leq j \leq m} F^*(i, j). \quad (2.13)$$

2.4 Considerações finais

Neste capítulo, foram apresentados resumidamente conceitos relacionados à descoberta de conhecimento em bases de dados e mineração de dados. As principais tarefas de mineração de dados podem ser divididas em aprendizado não supervisionado, supervisionado e semissupervisionado. Foram discutidas as principais características de classificação semissupervisionada, as abordagens utilizadas pelos trabalhos propostos na literatura e as suposições adotadas em cada abordagem.

Neste trabalho foi adotada a abordagem de classificação semissupervisionada baseada em grafos, com propagação de rótulos, para desenvolvimento da técnica “Classificação Semissupervisionada usando Proximidade Geoespacial” (CSPG), apresentada no Capítulo 5. Na CSPG, o objetivo é a classificação de séries temporais referentes a áreas reais, considerando informações geográficas da área referenciada e seu comportamento ao longo do tempo. A abordagem baseada em grafos foi escolhida, pois é possível modelar as séries temporais como vértices do grafo, e construir arestas que representem as relações de similaridade entre elas considerando informações geográficas e temporais. Dessa maneira, vértices que representam áreas muito próximas ou com comportamento temporal similar tendem a pertencer à mesma classe. Por meio da propagação de rótulos, cada vértice propaga seu valor de classe para seus vizinhos de acordo com a similaridade entre as áreas representadas por eles.

SÉRIES TEMPORAIS

Uma série temporal consiste em uma sequência de observações ordenadas pelo tempo e que apresenta intervalos de tempo iguais entre cada par de observações (MORETTIN; TOLOI, 2006). Devido ao aumento no número das bases de séries temporais em diversos domínios, como medicina, economia e agrometeorologia, surgiu o interesse na análise desses dados usando tarefas de mineração de dados. Além disso, em diversos domínios de pesquisa, dados tipicamente não temporais, como uma única imagem com palavras manuscritas, figuras geométricas, a face, mão ou postura corporal de um indivíduo, são representados e analisados como séries temporais (WEI; KEOGH, 2006; BATISTA; WANG; KEOGH, 2011; REN; YUAN; ZHANG, 2011; SHOKOOHI-YEKTA; WANG; KEOGH, 2015).

Uma série temporal pode ser definida como $S = \{s_1, s_2, \dots, s_m\}$ em que s_i , $i \in [1..m]$, corresponde à observação de S no tempo i e pode apresentar valor numérico (real) ou categórico. A série temporal pode ser unidimensional ou multidimensional. No caso unidimensional, s_i corresponde a um único valor numérico ou categórico. No caso multidimensional, a observação $s_i = \{s_{i1}, s_{i2}, \dots, s_{iD}\}$ é um vetor contendo D valores, cada um correspondente a uma dimensão da série temporal D -dimensional.

Embora séries temporais tenham sido estudadas por mais de 100 anos na estatística, muitos de seus métodos não são escaláveis para análise de grandes volumes de séries temporais, o que aumenta o interesse no uso de métodos de mineração de dados (MAIMON; ROKACH, 2010). A mineração de séries temporais corresponde a uma etapa do processo de descoberta de conhecimento em bases de dados, descrito no Capítulo 2. A seguir, são apresentados detalhes das etapas de Pré-processamento, Transformação e Mineração de Dados, do processo de KDD para análise de séries temporais.

3.1 Pré-processamento

Na etapa de pré-processamento, o objetivo é eliminar problemas presentes na base de séries temporais que inviabilizem a realização da análise de mineração de dados. Uma das tarefas de pré-processamento é o tratamento de valores faltantes em séries temporais. O problema consiste na ausência de uma ou mais observações da série temporal, ou presença de observações com valores inválidos, o que pode ser consequência de problemas no sensor ou na coleta de dados, erro humano, condições ambientais, entre outros (MITSA, 2010). O tratamento de valores faltantes depende da aplicação e da análise de mineração de dados. Uma alternativa é a substituição de valores utilizando, por exemplo, algum método de interpolação, como a interpolação linear (MITSA, 2010). Outra alternativa consiste em não substituir os valores faltantes e utilizar técnicas e funções de distâncias que considerem essa característica para análise das séries temporais.

Outro problema considerado na etapa de pré-processamento é o tratamento de ruídos, que são observações correspondentes a erros de medida e, por isso, apresentam valores incorretos na série temporal. Dois métodos bastante utilizados para tratamento de ruídos são *binning* e a suavização da série temporal (MITSA, 2010).

Uma tarefa comum no pré-processamento é a normalização de séries temporais cujo objetivo é representar todas as séries temporais de uma base de dados em uma mesma escala. Dois tipos de normalização amplamente utilizados são a 1) normalização *min-max*, em que cada observação de uma série temporal pertence ao intervalo definido como $[min, max]$, em que *min* e *max* são valores arbitrários; e 2) normalização *z-score*, em que as observações são transformadas de modo a apresentar uma média μ e desvio padrão σ arbitrários (MITSA, 2010).

3.2 Transformação e representação de séries temporais

Em alguns casos, a utilização da série temporal em seu formato original não é desejada para aplicação de métodos de mineração de dados. Assim, usando um modelo de representação, pode-se representar cada série temporal em um novo formato que ofereça vantagens em relação ao uso das séries no formato original. Por exemplo, o uso de modelos de representação que possibilitem uma redução significativa no tamanho das séries temporais é útil para melhorar o desempenho de tarefas de mineração de dados. Uma das características dos modelos de representação é que eles adotam uma representação que permita um baixo custo de reconstrução da série temporal para o formato original (KEOGH *et al.*, 2001). Outra propriedade importante, nesse caso, é que o valor retornado por uma dada função de distância para um par de séries temporais no espaço transformado deve ser menor ou igual ao valor dessa função de distância no espaço de representação original (SHIEH; KEOGH, 2008). Essa propriedade é denominada propriedade de *lower bounding* e é útil para as tarefas indexação e buscas por similaridade (FALOUTSOS; RANGANATHAN; MANOLOPOULOS, 1994; KEOGH *et al.*, 2001).

Diversos modelos de representação de séries temporais têm sido propostos, como: *Discrete Fourier Transform* (DFT) (HARRIS, 1978), *Discrete Wavelet Transform* (DWT) (SHENSA, 1992), *Piecewise Aggregate Approximation* (PAA) (KEOGH; PAZZANI, 2000), *Singular Value Decomposition* (SVD) (KLEMA; LAUB, 1980), *Symbolic Aggregate Approximation* (SAX e iSAX) (KEOGH; LIN; FU, 2005; SHIEH; KEOGH, 2008), *Piecewise Linear Representation* (PLA) (DUNHAM, 1986), *Adaptive Piecewise Constant Approximation* (APCA) (KEOGH *et al.*, 2001) e *Hidden Markov Models* (HMM) (RABINER; JUANG, 1986). Vale mencionar que na etapa de Transformação, pode-se adotar como representação das séries temporais seu formato original.

3.3 Tarefas de mineração de séries temporais

Após as etapas de Pré-processamento e Transformação, é aplicado um método de mineração de dados para análise do conjunto de séries temporais. As principais tarefas de interesse para mineração de séries temporais são sintetizadas a seguir (MAIMON; ROKACH, 2010):

- **Indexação:** dada uma série temporal de consulta Q e uma função de dissimilaridade $Dist(Q, S)$, o objetivo é encontrar a série temporal mais similar a Q em uma base de séries temporais.
- **Agrupamento:** o objetivo é encontrar agrupamentos em um conjunto de séries temporais, de modo que séries temporais de um mesmo grupo são mais similares entre si, do que em relação a séries pertencentes a outros grupos, de acordo com uma função de dissimilaridade $Dist(S, T)$.
- **Classificação:** dada uma série temporal S não rotulada, utilizar um classificador f treinado usando um conjunto de séries temporais previamente rotuladas, para associar S a uma das classes do problema.
- **Predição:** dada uma série temporal S contendo m observações, o objetivo é prever o valor da observação futura $m + 1$.
- **Sumarização:** dada uma série temporal S contendo m observações, em que m é muito grande, o objetivo é criar uma aproximação de S que seja capaz de manter suas principais características mas apresente um tamanho menor.
- **Deteção de anomalias:** dada uma série temporal S considerada normal e uma série temporal T , o objetivo é encontrar qualquer parte de T que seja considerada anômala, inesperada ou de interesse.
- **Segmentação:** dada uma série temporal S contendo m observações, construir um modelo $\bar{S}$ a partir de K segmentos ($K \ll m$), de modo que $\bar{S}$ se aproxima precisamente de S .

Em mineração de séries temporais, para tarefas como classificação, agrupamento e indexação, é importante definir a função de distância usada para calcular a dissimilaridade entre pares de séries temporais. A seguir, são apresentadas as principais funções de distância para séries temporais propostas na literatura.

3.3.1 Medidas de dissimilaridade

Uma das principais funções de distância utilizadas para análise de séries temporais, e em diversos domínios de aplicação, é a distância Euclidiana. Dada duas séries temporais de valores reais $A = \{a_1, a_2, \dots, a_m\}$ e $B = \{b_1, b_2, \dots, b_m\}$, a distância Euclidiana é definida pela Equação 3.1:

$$L_2(A, B) = \sqrt{\sum_{i=1}^m (a_i - b_i)^2} \quad (3.1)$$

A distância Euclidiana faz parte da família de distâncias Minkowski:

$$L_p(A, B) = \sqrt[p]{\sum_{i=1}^m (a_i - b_i)^p} \quad (3.2)$$

em que, para $p = 2$, tem-se a distância Euclidiana. Outras duas distâncias de destaque da família Minkowski são as distâncias L_1 e L_∞ , definidas pelas Equações 3.3 e 3.4, respectivamente:

$$L_1(A, B) = \sum_{i=1}^m |a_i - b_i| \quad (3.3)$$

$$L_\infty(A, B) = \max_{i=1}^m |a_i - b_i| \quad (3.4)$$

Em (DING *et al.*, 2008), são realizados experimentos com diversos conjuntos de séries temporais com diferentes funções de distância. Nesse estudo, os autores verificam que os resultados de classificação obtidos pela distância Euclidiana, especialmente para conjuntos volumosos, se equiparam aos obtidos por outras funções de distâncias mais complexas e que apresentam maior custo computacional. Uma desvantagem da distância Euclidiana é a limitação no cálculo de distância entre séries temporais com mesmo comprimento (número de observações). Na Figura 8, são mostradas as associações feitas entre cada par de observações de duas séries temporais para o cálculo da distância Euclidiana.

Pode-se notar que as séries temporais mostradas na Figura 8 apresentam comportamento muito similar, no entanto a série temporal B está deslocada uma unidade de tempo em relação à série temporal A . O uso da distância Euclidiana não permite o reconhecimento de tal similaridade, pois são calculadas as diferenças entre pares de observações sempre no mesmo instante de tempo.

Figura 8 – A distância Euclidiana calcula a diferença entre pares de observações no mesmo instante de tempo, para as duas séries temporais A e B .

Fonte: Elaborada pelo autor.

Nesse caso, uma alternativa é a *Dynamic Time Warping* (DTW) (BERNDT; CLIFFORD, 1994; KEOGH; PAZZANI, 2000), que permite realizar o alinhamento entre duas séries temporais e, assim, reconhecer a similaridade entre pares de séries, mesmo se elas apresentarem comprimentos diferentes e/ou se elas não estiverem alinhadas em relação à escala temporal. Sejam duas séries temporais $A = \{a_1, a_2, \dots, a_m\}$ e $B = \{b_1, b_2, \dots, b_n\}$ de tamanhos m e n respectivamente. No cálculo da DTW, o objetivo é resolver o problema:

$$DTW(A, B) = \min \sqrt{\sum_{k=1}^K w_k} \quad (3.5)$$

em que $w_k = (i, j)$ representa a associação entre as observações a_i e b_j das séries temporais A e B e equivale à distância Euclidiana $d(i, j) = \sqrt{(a_i - b_j)^2}$. A sequência $w_1, w_2, \dots, w_K$ representa a sequência de associações entre pares de observação das duas séries, denominado o caminho de ajuste. A Equação 3.5 é restrita às seguintes condições:

1. é necessário que $w_1 = (1, 1)$ e $w_K = (m, n)$, ou seja, o primeiro elemento do caminho de ajuste corresponde à associação entre as observações iniciais das duas séries temporais, e o último elemento do caminho de ajuste é a associação entre as observações finais de A e B ;
2. dado $w_k = (i, j)$, então o elemento seguinte $w_{k+1} = (i', j')$, em que $i' - i \leq 1$ e $j' - j \leq 1$;
3. dado $w_k = (i, j)$ e $w_{k+1} = (i', j')$, é necessário que $i' - i \geq 0$ e $j' - j \geq 0$.

É possível computar o caminho de ajuste usando programação dinâmica, para resolver a

seguinte recorrência (RATANAMAHATANA; KEOGH, 2004b):

$$\gamma(i, j) = \begin{cases} 0, & \text{se } i = 0 \text{ e } j = 0 \\ \infty, & \text{se } i = 0 \text{ ou } j = 0 \\ dist(i, j) + \min\{\gamma(i-1, j), \gamma(i, j-1), \gamma(i-1, j-1)\}, & \text{caso contrário} \end{cases} \quad (3.6)$$

em que $d(i, j)$ representa a distância Euclidiana entre as observações a_i e b_j . O resultado da distância DTW entre A e B é equivalente a $\sqrt{\gamma(m, n)}$.

Na Figura 9, é exibido o resultado da aplicação da DTW para calcular a distância entre as séries temporais da Figura 8. Pode-se notar que a DTW realiza o alinhamento entre as duas séries temporais, de modo a reconhecer a similaridade entre seus comportamentos, mesmo havendo um deslocamento na escala temporal.

Figura 9 – A similaridade entre as séries temporais A e B é reconhecida pela DTW.

Fonte: Elaborada pelo autor.

O custo computacional da DTW é $O(mn)$. Pode-se adicionar uma restrição de janela de ajuste, em que $|i - j| \leq \delta$, em que δ é o tamanho da janela. Para calcular a DTW com a janela de ajuste, basta definir $d(i, j) = \infty$, para $|i - j| > \delta$, na Equação 3.6. Com o uso da janela de ajuste é possível acelerar a computação da DTW e ainda obter resultados de classificação superiores aos produzidos sem o uso da janela (RATANAMAHATANA; KEOGH, 2004a). A DTW tem sido utilizada para tarefas de classificação (RATANAMAHATANA; KEOGH, 2004a; XI *et al.*, 2006), agrupamento (OATES; FIROIU; COHEN, 1999; ZHU *et al.*, 2012), indexação (RAKTHANMANON *et al.*, 2012), entre outras. Diversas variações e aproximações da DTW foram propostas na literatura (SALVADOR; CHAN, 2004; CHEN *et al.*, 2013; SHOKOOHI-YEKTA; WANG; KEOGH, 2015).

Uma característica comum às distâncias Euclidiana e DTW é que todas as observações das duas séries temporais são consideradas para o cálculo da função de distância. No caso em que uma ou mais observações das séries temporais analisadas correspondem a ruídos, ele inevitavelmente é levado em conta para o cálculo da distância. Na Figura 10 são ilustradas as

mesmas séries temporais das Figuras 8 e 9, porém com a presença de ruído na observação do instante 6 da série temporal A. É possível notar que a presença de uma observação correspondente a ruído interfere significativamente no resultado do alinhamento entre as séries A e B usando a DTW, quando comparado ao resultado mostrado na Figura 9. Uma alternativa nesse caso, é a utilização de uma função de distância menos sensível à presença de ruídos, como a *Longest Common Subsequence* (LCSS) (VLACHOS; KOLLIOS; GUNOPULOS, 2002; VLACHOS *et al.*, 2003).

Figura 10 – A presença de ruído interfere no cálculo de dissimilaridade usando a DTW.

Fonte: Elaborada pelo autor.

A LCSS (RISTAD; YIANILOS, 1998) visa encontrar a maior subsequência comum a duas séries temporais A e B, de tamanhos m e n , respectivamente. Ela é mais robusta em relação à presença de ruídos, pois algumas observações, possivelmente ruídos, podem ser ignoradas no cálculo da LCSS. Quanto maior for a subsequência encontrada, menos dissimilares são as séries temporais. Como as observações das séries temporais são valores reais, é necessário definir um parâmetro ϵ , tal que duas observações a_i e b_j são consideradas comuns apenas se $|a_i - b_j| < \epsilon$. Assim pode-se definir o comprimento da maior subsequência comum entre duas séries temporais A e B como:

$$LCSS_{compr}(i, j) = \begin{cases} 0, & \text{se } i = 0 \text{ ou } j = 0 \\ 1 + LCSS_{compr}(i - 1, j - 1), & \text{se } |a_i - b_j| < \epsilon \\ & \text{e } |i - j| < \delta \\ \max\{LCSS_{compr}(i - 1, j), LCSS_{compr}(i, j - 1)\}, & \text{caso contrário} \end{cases} \quad (3.7)$$

em que δ corresponde ao parâmetro de janela, similar à utilizada no cálculo da DTW. Após calcular o comprimento da maior subsequência comum, o valor de distância normalizado é dado pela Equação 3.8:

$$LCSS_{Dist}(A, B) = \frac{LCSS_{compr}(m, n)}{\min(m, n)} \quad (3.8)$$

em que $LCSS_{Dist}(A, B) = 0$ a maior similaridade possível entre A e B , e $LCSS_{Dist}(A, B) = 1$ indica a maior dissimilaridade possível entre as séries temporais.

Um exemplo da aplicação da LCSS é ilustrado na Figura 11. Pode-se notar que, ao utilizar a LCSS, a observação da série temporal A correspondente ao ruído não é associada a nenhuma observação da série temporal B , o que permite o reconhecimento da similaridade das duas séries temporais, mesmo com a presença de ruído e de deslocamento na escala temporal. Variações da LCSS incluem a *Edit distance with Real Penalty* (ERP) (CHEN; NG, 2004) e a *Edit Distance on Real sequence* (EDR) (CHEN; ÖZSU; ORIA, 2005).

Figura 11 – Reconhecimento de similaridade entre séries temporais com ruído usando LCSS.

Fonte: Elaborada pelo autor.

3.4 Considerações finais

Neste capítulo foram apresentados conceitos e trabalhos relacionados à mineração de séries temporais. Métodos de substituição de valores faltantes, normalização, e modelos de representação foram apresentados como parte da etapa de pré-processamento e representação de séries temporais. Foram sintetizadas as principais tarefas de mineração de séries temporais, e descritas as funções de distância apropriadas para séries temporais mais utilizadas em trabalhos publicados na literatura de mineração de séries temporais.

FRAMEWORK SITSMINING

Avanços nas tecnologias de coleta e geração de dados têm levado a um rápido crescimento no volume de dados armazenados, e portanto, disponíveis para análise. Em sensoriamento remoto, imagens de satélite são coletadas diariamente, e a análise manual desses dados é praticamente impossível, nos dias de hoje. Além disso, dados como séries temporais de imagens de satélites ou mesmo de valores reais apresentam alta complexidade, o que torna imprescindível o uso de recursos computacionais capazes de analisar e retornar informações relevantes para o usuário, em tempo hábil.

Uma abordagem comum na análise de imagens de satélite consiste em classificar cada pixel ou um subconjunto de pixels de apenas uma imagem, usando técnicas de mineração de dados como agrupamento, classificação, segmentação, entre outros.

O foco da metodologia proposta neste trabalho de mestrado é diferente: dada uma série temporal de imagens de satélite (SITS – do inglês *Satellite Image Time Series*), o objetivo é analisar cada área real, de acordo com o seu comportamento ao longo de um período de tempo. Assim, ao utilizar imagens cujos pixels correspondem a valores de um índice de vegetação, por exemplo, pode-se aplicar métodos de mineração de dados para encontrar padrões de interesse, considerando a evolução da vegetação de cada área ao longo de um período de interesse.

Nesse contexto, o *framework* SITSMining consiste em uma metodologia de análise temporal de dados obtidos a partir de uma ou mais SITS. O *framework* compreende tarefas que incluem:

1. extração de dados a partir de uma ou mais SITS;
2. escolha de uma representação dos dados extraídos adequada para análise de dados temporais;
3. pré-processamento dos dados, incluindo tarefas de tratamento de ruídos e valores faltantes, e normalização de séries temporais;

4. aplicação de tarefas de mineração de dados e uso de funções de distâncias adequadas para séries temporais;
5. representação dos resultados usando um formato que permita a visualização dos padrões obtidos de modo adequado para a análise do especialista em agrometeorologia.

Os componentes e o fluxo de dados do *framework* SITSMining são ilustrados na Figura 12. O *framework* é dividido em 3 camadas: 1) de Extração e Pré-processamento; 2) de Mineração de Dados; e 3) de Saída. Cada camada possui um ou mais módulos, nos quais são executadas as tarefas relacionadas à análise dos dados obtidos a partir de SITS. A seguir, os dados recebidos como entrada e todo o processo executado nas camadas do *framework* SITSMining são descritos em detalhes.

Figura 12 – *Framework* SITSMining e seus componentes.

Fonte: Adaptada de [Amaral et al. \(2014\)](#).

4.1 Dados de entrada

O primeiro item recebido como entrada pelo *framework* é um conjunto de séries temporais de imagens de satélite (SITS) $T = \{T_1, T_2, \dots, T_D\}$, em que $T_i = \{I_{i1}, I_{i2}, \dots, I_{im}\}$ corresponde a uma SITS de m imagens, tal que I_{ij} é a imagem de satélite j da SITS i do conjunto T . Para o *framework* SITSMining, é necessário que T obedeça às seguintes restrições:

1. o número de imagens m deve ser o mesmo para todas as SITS de T ;
2. o período de tempo compreendido por cada SITS deve ser o mesmo, ou seja, a primeira imagem de cada SITS $\{I_{i1}\}_{i=1}^D$ deve corresponder ao mesmo instante de tempo inicial, e a diferença de tempo entre cada par de imagens subsequentes I_{ij} e $I_{i(j+1)}$ deve ter o mesmo valor, para todo $i \in \{1, 2, \dots, D\}$ e $j \in \{1, 2, \dots, m-1\}$;
3. as imagens de satélite de todas as SITS devem corresponder à mesma cena e possuir a mesma resolução espacial. Isso implica que, para uma dada SITS, seja p_{kij} o pixel de linha i e coluna j na imagem I_k , é necessário que todos os pixels do conjunto $\{p_{kij}\}_{k=1}^m$ referenciem a mesma área real;
4. para uma SITS, todas as imagens de satélite devem carregar informações do mesmo tipo. Por exemplo, não é permitido que os pixels de uma imagem indiquem medidas de temperatura de superfície, enquanto os pixels de outra imagem de T correspondam a medidas de um índice de vegetação.

O segundo item recebido como entrada é a região de interesse (ROI – do inglês *Region of Interest*), utilizada para definir um subconjunto de pixels nas imagem de satélite, denominado conjunto de pixels de interesse. A região de interesse corresponde a um conjunto de pares de coordenadas $ROI = \{(lat_1, long_1), (lat_2, long_2), \dots, (lat_p, long_p)\}$, em que cada par $(lat_i, long_i)$ representa uma área real referenciada por um pixel em cada imagem. Dependendo da resolução espacial das imagens de satélite, é possível que mais de um par de coordenadas referenciem diferentes áreas reais que incidem em um mesmo pixel. Nesse caso, por padrão, a estratégia é incluir esse pixel ao conjunto de pixels de interesse da mesma maneira que os demais. No entanto, é possível, por exemplo, atribuir maior peso ou alguma medida de importância para pixels que incluam mais de uma área de interesse.

O terceiro item de entrada somente é utilizado para tarefas supervisionadas de mineração de dados e corresponde ao conjunto de treinamento. As instâncias do conjunto de treinamento carregam informação de rótulo ou classe, e são necessárias para tarefas como a classificação. O conjunto de treinamento, na maioria dos casos, é produzido por especialistas em agrometeorologia como resultado de um processo de supervisão de dados extraídos de imagens de satélite. A estrutura do conjunto de treinamento é descrita na próxima seção, pois é semelhante à estrutura

do conjunto de instâncias criado após extração dos dados a partir de T , na Camada de Extração e Pré-processamento.

4.2 Camada de Extração e Pré-processamento

Na Camada de Extração e Pré-processamento, o principal objetivo é gerar um conjunto de instâncias a partir dos dados de entrada, que será utilizado na camada de mineração de dados do *framework*. As tarefas realizadas nessa camada são divididas em dois módulos: 1) módulo de extração e 2) módulo de pré-processamento.

No módulo de extração, são realizadas duas principais tarefas. A primeira é a extração do valor de cada pixel que referencia uma ou mais áreas reais da região de interesse. Assim, a partir da região de interesse, é obtido um conjunto de pixels de interesse para cada imagem de satélite de cada SITS de T . Para essa tarefa, é necessário que as imagens de satélite já tenham sido pré-processadas. A extração do valor de cada pixel pode ser realizada usando softwares como o ENVI¹ ou bibliotecas que trabalham com dados geoespaciais, como a GDAL².

Após extração do valor de cada pixel de interesse, é gerado o conjunto de instâncias $X = \{x_1, x_2, \dots, x_n\}$, em que cada instância x_i armazena as informações de uma única área real. As instâncias de X possuem os seguintes valores de atributo:

- **id:** número inteiro usado para identificar cada instância.
- **latitude:** coordenada de latitude da área real referenciada pela instância.
- **longitude:** coordenada de longitude da área real referenciada pela instância.
- **ts:** série temporal multidimensional $S = \{s_1, s_2, \dots, s_m\}$, em que cada observação $s_i \in \mathbb{R}^D$ é um vetor D -dimensional de valores reais $s_i = (s_{i1}, s_{i2}, \dots, s_{iD})$, no qual s_{ij} é o valor do pixel correspondente à instância na imagem de satélite i na SITS j , com $i \in \{1, 2, \dots, m\}$ e $j \in \{1, 2, \dots, D\}$.

A extração do valor de pixel a partir de cada imagem de satélite para geração da série temporal multidimensional S de uma instância é exibida na Figura 13. Nesse processo, dada uma instância x_i correspondente a uma dada área real, o primeiro passo é localizar o pixel que referencia tal área em cada imagem de satélite das SITS. Em seguida, o valor de cada pixel é extraído e usado para compor uma observação s_j da série temporal S gerada para a instância x_i . O conjunto de instâncias X , gerado pelo módulo de extração, é ilustrado na Figura 14.

Após a geração do conjunto X , ele é encaminhado para o módulo de pré-processamento. No contexto de imagens de satélite, a presença de ruídos e valores faltantes devido a fatores

¹ <http://www.exelisvis.com/>

² <http://www.gdal.org/>

Figura 13 – Processo de extração de valores das SITS para geração da série temporal multidimensional. Dado um par de coordenadas, são encontrados os pixels correspondentes a essas coordenadas em cada imagem de cada SITS. Em seguida, os valores são extraídos, organizados por dimensão e ordenados pelo tempo, na série temporal multidimensional.

Fonte: Elaborada pelo autor.

como falha nos sensores, ou presença de nuvens é comum e até esperado pelos especialistas. Para a análise do *framework* SITSMining, não há intenção na remoção desses valores, e sim na sua substituição por valores adequados de modo a tornar a série temporal S completa. Em geral, valores faltantes ou ruídos podem ser indicados por valores muito elevados, muito reduzidos ou algum valor pré-definido e usado para indicar erro. Para substituição desses valores, pode-se utilizar técnicas de estimativas de valores, como interpolação linear ou imputação múltipla (ROYSTON *et al.*, 2004).

Outra tarefa no módulo de pré-processamento é a normalização de cada dimensão de

Figura 14 – Módulos e conjunto de instância da Camada de Extração e Pré-processamento.

Fonte: Adaptada de [Amaral et al. \(2014\)](#).

S. Dado que cada SITS contém imagens de satélite correspondentes a um tipo diferente de medida ou índice, o intervalo de valores dos pixels em cada SITS pode ser diferente. Para uso de uma função de distância compatível com séries temporais multidimensionais, é indicado que os valores de cada dimensão (correspondente a uma SITS) estejam normalizados para que todas as dimensões apresentem contribuições com mesmo peso para o cálculo da distância ([SHOKOOHI-YEKTA; WANG; KEOGH, 2015](#)). Nessa tarefa, podem ser utilizadas técnicas de normalização para séries temporais como *min-max* e *z-score* ([MITSA, 2010](#)).

4.3 Camada de Mineração de Dados

A Camada de Mineração de Dados recebe o conjunto de instâncias X pré-processado e em formato adequado para a análise nessa camada. No módulo de mineração de dados, são consideradas duas principais tarefas, correspondentes ao submódulos de classificação e agrupamento, como ilustrado na Figura 15.

Para o submódulo de classificação, é necessário receber o conjunto X e o conjunto de treinamento fornecido como entrada para o *framework*. O conjunto de treinamento deve possuir os mesmos atributos de X (id, latitude, longitude, S) e um atributo adicional y , que corresponde ao valor de classe. O conjunto de treinamento recebido como entrada pelo *framework* SITSMining deve ser composto apenas por instâncias rotuladas, portanto todas as instâncias devem apresentar valor para o atributo y . No módulo de classificação, são incluídos algoritmos de classificação supervisionada e semisupervisionada. Apesar das diferenças entre as duas abordagens de

Figura 15 – Camada de Mineração de Dados.

Fonte: Adaptada de [Amaral et al. \(2014\)](#).

classificação, ambas recebem um conjunto de instâncias rotuladas (conjunto de treinamento) e um conjunto de instâncias não rotuladas X , embora X seja usado apenas para teste no caso supervisionado, e usado para treinamento e teste no caso semissupervisionado.

No caso do submódulo de agrupamento, não é preciso fornecer o conjunto de treinamento, já que o agrupamento é uma tarefa não supervisionada de mineração de dados. Como na maioria dos algoritmos de classificação e agrupamento é necessário o uso de uma função de distância, deve ser fornecida uma função compatível com séries temporais quando uma única SITS é recebida como entrada pelo *framework* ($D = 1$), ou uma função apropriada para séries temporais multidimensionais, no caso de múltiplas SITS ($D > 1$). Alguns exemplos de funções de distância que são adequadas para séries temporais unidimensionais ou multidimensionais são a Euclidiana, DTW e LCSS, mencionadas no Capítulo 3.

4.4 Camada de Saída

Após a etapa de mineração de dados, cada instância do conjunto X é associada a um rótulo, que pode significar sua classe, no caso de classificação, ou seu agrupamento, na análise de agrupamento. Na Camada de Saída, esses resultados são submetidos a dois módulos: 1) espacialização e 2) de perfil. No módulo de espacialização, os resultados são processados para permitir a geração de um mapa geográfico, em que cada instância de X é ilustrada de acordo com suas coordenadas (latitude, longitude) e uma coloração distinta é utilizada para cada valor possível de rótulo atribuído pelo algoritmo de mineração de dados. Para geração desse mapa, são selecionados apenas 3 atributos de X : latitude, longitude e o recente atributo de rótulo, conforme exibido na Figura 16.

Um exemplo de mapa produzido pelo módulo de espacialização é ilustrado na Figura 17a, resultado de uma análise de agrupamento usando o algoritmo K-Means para o estado de São Paulo, de abril/2005 a março/2006, adaptado do trabalho no qual foi publicado o *framework* SITSMining ([AMARAL et al., 2014](#)). Nesse experimento, foram analisadas duas SITS, uma obtida usando um índice de reflexibilidade da luz solar (Albedo) ([CSISZAR; GUTMAN, 1999](#)), e outra com valores de temperatura de superfície. Pode-se notar que, para cada agrupamento, é utilizada

Figura 16 – Camada de Saída.

Fonte: Adaptada de [Amaral et al. \(2014\)](#).

uma cor distinta, o que permite ao especialista em agrometeorologia identificar quais áreas da região de interesse apresentaram comportamento similares, de acordo com os índices utilizados.

O outro módulo de Camada de Saída é o de perfil, no qual são calculadas séries temporais representativas de cada classe ou agrupamento encontrado na análise de mineração de dados. No caso da classificação, a série temporal representativa pode ser simplesmente a média das séries temporais de todas as instâncias associadas a uma dada classe. Na análise de agrupamento, uma série temporal representativa adequada é o centróide do agrupamento, ou no caso de algoritmos como K-Medoids e CLARANS (NG; HAN, 2002), o medóide. Assim, é gerado um conjunto de perfis, em que cada perfil corresponde a uma série temporal representativa e o valor de classe ou agrupamento ao qual ela corresponde, como mostrado na Figura 16. O conjunto de perfis permite criar um gráfico que exibe as séries temporais representativas de cada classe ou agrupamento com colorações distintas, como ilustrado na Figura 17b. Nessa figura são mostrados resultados da mesma análise de agrupamentos usando Albedo e temperatura de superfície (AMARAL et al., 2014), em que os centróides de cada agrupamento encontrado pelo algoritmo K-Means são exibidos, para cada índice de medida.

4.5 Ferramenta SatImagExplorer

Como parte deste trabalho de mestrado, o *framework* SITSMining foi implementado e incorporado ao sistema da SatImagExplorer (CHINO; ROMANI; TRAINA, 2010), uma ferramenta desenvolvida para extração de dados a partir de séries temporais de imagens de satélite e que permite a geração de gráficos para visualização das séries temporais extraídas. Dada uma sequência de imagens de satélite e definida uma região de interesse, é feita a extração e pré-processamento de séries temporais de maneira similar à descrita na Camada de Extração

Figura 17 – Gráficos ilustrados usando os resultados produzidos pela Camada de Saída do *framework*.

(a) Visualização geográfica dos resultados obtidos após análise de agrupamentos.

(b) Visualização dos centróides de cada agrupamento.

Fonte: Elaborada pelo autor.

e Pré-processamento do *framework*. Para incorporação total do *framework* SITSMining, a ferramenta SatImagExplorer foi estendida com as seguintes funcionalidades:

- **análise de mineração de dados:** foram implementados dois módulos de mineração de dados: classificação e agrupamento. No módulo de classificação foram implementados os algoritmos de classificação *K-Nearest-Neighbors*, *Linear Neighborhood Propagation* e a técnica CSPG, proposta neste trabalho de mestrado (Capítulo 5). No módulo de agrupamento, foram implementados os algoritmos K-Means, K-Medoids, CLARANS e BIRCH. Além disso, para os algoritmos que necessitam de uma função de distância apropriada para séries temporais, foram implementadas as funções da família Minkowski e a *Dynamic Time Warping*.
- **visualização de resultados:** foram implementados os módulos de visualização geoespacial e de perfil. O módulo de visualização geoespacial recebe os resultados da análise de mineração de dados no formato produzido pelo módulo de especialização da Camada

de Saída do *framework* SITSMining, o que permite ilustrar a localização geográfica das instâncias de acordo com sua classe ou agrupamento. No módulo de visualização de perfil, são recebidas as séries temporais representativas de cada classe ou agrupamento e ilustradas com colorações diferentes em um gráfico de perfis.

Dessa maneira, as tarefas das camadas de Extração e Pré-processamento, Mineração de dados e Saída do *framework* SITSMining podem ser executadas de modo automatizado usando apenas a ferramenta SatImagExplorer. Toda a implementação para integração do *framework* ao sistema da SatImagExplorer foi feita usando a linguagem de programação C++, usando o *framework* multiplataforma Qt³. Na Figura 18 é ilustrada a interface gráfica correspondente às etapas de seleção da região de interesse (Figura 18a), parametrização do algoritmo de mineração de dados (Figura 18b) e visualização dos resultados (Figura 18c). Uma vantagem importante da SatImagExplorer é sua organização modular, que permite a adição de novos métodos de pré-processamento de séries temporais, algoritmos de classificação, agrupamento, funções de distâncias, a qualquer momento, de maneira eficiente.

Para o especialista em agrometeorologia, é de grande valia a possibilidade de realizar de modo automático a análise de dados temporais obtidos a partir de imagens de satélite usando métodos de mineração de dados, e poder visualizar os resultados de maneira gráfica e intuitiva em um único sistema integrado. A ferramenta SatImagExplorer vem sendo utilizada para produção de trabalhos científicos pelos pesquisadores da Embrapa Informática Agropecuária e Cepagri/Unicamp.

4.6 Considerações Finais

Neste capítulo foi apresentado o *framework* SITSMining, uma metodologia de análise de dados obtidos a partir de séries temporais de imagens de satélite que compreende atividades desde a extração de séries temporais, pré-processamento, análise usando mineração de dados e representação dos resultados em um formato adequado para interpretação e avaliação do especialista no domínio de aplicação. Uma implementação do *framework* SITSMining foi incorporada à ferramenta SatImagExplorer, o que representa uma importante contribuição para pesquisas em agrometeorologia, pois permite ao especialista realizar o processo completo de análise de séries temporais extraídas de imagens de satélite de maneira automática em um único ambiente integrado. Além disso, a possibilidade de incorporar novos algoritmos e tarefas de mineração de dados à qualquer momento e de maneira eficiente, torna a SatImagExplorer um importante recurso para suporte a pesquisas na área de aplicação.

³ <http://www.qt.io/>

Figura 18 – Interface da ferramenta SatImagExplorer.

(a) Seleção da ROI.

(b) Parametrização do algoritmo.

(c) Visualização dos resultados.

Fonte: Elaborada pelo autor.

TÉCNICA CSPG

A técnica proposta, denominada “Classificação Semissupervisionada usando Proximidade Geoespacial” (CSPG), foi desenvolvida para classificação semissupervisionada de dados obtidos a partir de imagens de satélite. O objetivo da CSPG é classificar áreas reais referenciadas por pixels de imagens de satélite considerando duas principais informações:

1. **Evolução temporal:** é analisado o comportamento apresentado por uma área real ao longo de um período de tempo. Por exemplo, utilizando um índice de vegetação, é possível analisar qual foi a evolução do comportamento da vegetação em uma determinada área ao longo de um ano. Para a tarefa de classificação, o objetivo é poder discriminar áreas com diferentes conteúdos, como florestas, rios, áreas urbanas e agrícolas, a partir da análise do comportamento de cada uma delas, de acordo com o tipo de informação (índice) representado pelos pixels de imagens de satélite.
2. **Proximidade geoespacial:** a informação das coordenadas (latitude, longitude) das áreas reais analisadas é usada para construção do classificador da CSPG, tendo como embasamento a suposição de que áreas próximas **tendem** a apresentar o mesmo conteúdo. Por exemplo, considere que uma região ocupada por um mesmo conteúdo como floresta, cidade ou plantações agrícolas apresente uma área física de akm^2 . Se, de acordo com a resolução espacial da imagem de satélite, a região de tamanho a é representada por um conjunto de b pixels, então todos esses pixels devem ser rotulados com o mesmo valor de classe. Por exemplo, áreas de plantio de culturas agrícolas como a cana-de-açúcar são organizadas em áreas vastas e adjacentes. Nesse contexto, a suposição de que áreas próximas **tendem** a pertencer à mesma classe é adequada ao problema analisado. Para o algoritmo da CSPG, assume-se que instâncias representadas por pixels adjacentes na imagem de satélite **tendem** a possuir o mesmo valor do atributo y .

A estratégia usada para incorporar as informações de evolução temporal e proximidade

geoespacial ao algoritmo da CSPG é a abordagem de classificação semissupervisionada baseada em grafos. Como mencionado na Seção 2.3, a suposição baseada em grafos assume que vértices conectados por arestas que indicam alta similaridade tendem a pertencer à mesma classe. Por isso, a abordagem de grafos foi escolhida para o algoritmo da CSPG, de modo que as instâncias que apresentam comportamento similar da série temporal e/ou estão próximas geograficamente são representadas por vértices conectados por arestas que indicam sua similaridade, e portanto tendem a pertencer à mesma classe.

Em uma primeira etapa, as instâncias de treinamento, correspondentes a pixels das imagens de satélite, são modeladas como vértices do grafo, e arestas são criadas para representação da similaridade existente entre pares de vértices. São criadas arestas para conectar dois vértices de acordo com duas condições:

1. se as séries temporais das instâncias representadas por eles são muito similares;
2. se os vértices representam dois pixels adjacentes na imagem de satélite.

Após a construção do grafo, é aplicada a técnica de propagação de rótulos no grafo, em que cada vértice propaga seu valor de rótulo, ou classe, para todos seus vértices vizinhos. Ao final desse processo, cada vértice é rotulado de acordo com o rótulo recebido em maior quantidade, que corresponde a uma das classe do problema. Usando a abordagem de propagação de rótulos, de acordo com a suposição baseada em grafos adotada pela CSPG, vértices conectados por arestas de maior valor propagam seus rótulos com mais intensidade, de modo que, ao final do processo de propagação, vértices vizinhos com alta similaridade sejam classificados usando o mesmo rótulo. A CSPG é uma técnica de classificação semissupervisionada transdutiva, pois o objetivo é a classificação das mesmas instâncias não rotuladas utilizadas no treinamento do classificador.

A seguir, cada passo do algoritmo da CSPG é apresentado e discutido em detalhes.

5.1 Dados de entrada

Como entrada, é dado um conjunto de instâncias $X = X_R \cup X_{NR}$, união do conjunto de instâncias rotuladas $X_R = \{(x_1, y_1), (x_2, y_2), \dots, (x_r, y_r)\}$ e o conjunto de instâncias não rotuladas $X_{NR} = \{x_{r+1}, x_{r+2}, \dots, x_n\}$ para as quais o atributo y não é conhecido. Para o algoritmo da CSPG, X é o conjunto de treinamento.

Esse conjunto de treinamento é formado utilizando a mesma estratégia descrita no *framework* SITSMining, em que $S = \{I_1, I_2, \dots, I_m\}$ é uma série temporal de imagens de satélite e $P_i = \{p_{i1}, p_{i2}, \dots, p_{in}\}$ é o conjunto dos n pixels da imagem I_i correspondentes a uma dada região de interesse. Pode-se notar que o número n de pixels em uma imagem de satélite, que por sua vez é o número de áreas reais analisadas, corresponde ao número de instâncias no conjunto de

treinamento. Os atributos do conjunto de treinamento X são $\{lat, long, ts, y\}$, em que, para cada instância $x_i \in X$, tem-se:

- $x_i.lat$: coordenada de latitude da área real representada pela instância x_i .
- $x_i.long$: coordenada de longitude da área real representada pela instância x_i .
- $x_i.ts$: série temporal que corresponde ao comportamento da área real representada por x_i .
- $x_i.y$: é o valor de rótulo da instância x_i , em que $y = \{1, 2, \dots, c\}$ e c é o número de classes do problema. Inicialmente, para $x_i \in X_{NR}$, $x_i.y$ não é conhecido.

Para o caso de uma única SITS S , a série temporal é definida como $ts = \{ts_1, ts_2, \dots, ts_m\}$, em que $ts_j \in \mathbb{R}$, para $j \in \{1, 2, \dots, m\}$, corresponde ao valor do pixel p_{ji} da imagem de satélite I_j .

No caso de múltiplas SITS $S = \{S_1, S_2, \dots, S_D\}$, a série temporal é definida como $ts = \{ts_1, ts_2, \dots, ts_m\}$, em que a observação $ts_j \in \mathbb{R}^D$, para $j \in \{1, 2, \dots, m\}$, corresponde a um vetor D -dimensional $ts_j = \{ts_{j1}, ts_{j2}, \dots, ts_{jD}\}$, no qual o valor ts_{jk} corresponde ao valor do pixel p_{ji} da imagem de satélite I_j da SITS S_k .

O valor de cada observação da série temporal ts é dependente da imagem de satélite utilizada e de todo o pré-processamento aplicado antes da extração de valores.

5.2 Criação do grafo

O processo de classificação da técnica CSPG é transdutivo, em que, dado o conjunto de treinamento $X = X_R \cup X_{NR}$, o objetivo é treinar o classificador para inferir o valor do rótulo y para as instâncias de X_{NR} . Para a rotulação de tais instâncias, a estratégia usada no algoritmo da CSPG é modelar todas as instâncias de X , rotuladas e não rotuladas, como vértices de um grafo e então realizar a rotulação dos vértices não rotulados inicialmente. A construção do grafo é descrita a seguir.

Seja $G(V, E)$ um grafo não direcionado simples, no qual cada vértice $v_i \in V$, para $i \in 1, 2, \dots, n$ corresponde à instância $x_i \in X$. Um aresta $(i, j) \in E$ conecta os vértices v_i e v_j e é ponderada pelo valor w_{ij} , que representa a similaridade entre os vértices conectados. A criação de arestas está sujeita pelo menos uma das condições:

1. se o vértice v_j corresponde a um dos K vértices menos dissimilares a v_i , de acordo com uma função de distância $Dist(x_i.ts, x_j.ts)$, então diz-se que v_i e v_j são conectados por uma aresta pela série temporal. O valor K é definido pelo usuário e passado como parâmetro para o algoritmo da CSPG. A função de distância $Dist$ deve ser compatível com séries temporais ts do conjunto de treinamento, sejam elas uni ou multidimensionais.

2. se os pixels p_{li} e p_{lj} , pertencentes à imagem de satélite I_l , forem adjacentes, então se diz que v_i e v_j (que representam as instâncias x_i e x_j) são conectados por uma aresta pela proximidade geoespacial.

Em um primeiro passo, é necessário definir o conjunto $\mathcal{N}_i$, que correspondem às K instâncias menos dissimilares à instância x_i , de acordo com a função de distância $Dist$. Para isso, são realizados os cálculos de dissimilaridade necessários, e são criadas arestas de acordo com a condição da similaridade pela série temporal que conectam v_i aos vértices que representam as instâncias de $\mathcal{N}_i$. Vale mencionar que v_i e v_j são conectados se $x_i \in \mathcal{N}_j$ ou $x_j \in \mathcal{N}_i$ ou ($x_i \in \mathcal{N}_j$ e se $x_j \in \mathcal{N}_i$).

Após o cálculo das arestas geradas pela condição de similaridade da série temporal, são calculadas as arestas de acordo com a condição de proximidade geográfica, com uma simples checagem de adjacência de pixels nas imagens de satélite. Nessa etapa, há a possibilidade de que um par de vértices v_i e v_j seja conectado pelas duas condições. É importante ressaltar que isso significa que duas áreas reais são vizinhas geograficamente e apresentam comportamentos similares ao longo de um período de tempo. Por isso, vértices conectados e que satisfazem as duas condições de conexão têm maior probabilidade de pertencerem à mesma classe do que vértices que são conectados apenas por uma das condições.

Vale mencionar que, como o grafo não possui arestas múltiplas, só é possível haver uma aresta que conecte cada par de vértices, portanto o termo “conectado pelas duas condições” é usado para indicar que as instâncias representadas por esses vértices apresentam similaridade tanto pela série temporal quanto pela proximidade geoespacial.

Para cada caso de conexão entre os vértices, a aresta que conecta um dado par de vértices v_i e v_j recebe uma ponderação distinta W_{ij} , dada pela Equação 5.1:

$$W_{ij} = \begin{cases} w_{st}, & \text{se } v_i \text{ e } v_j \text{ são conectados apenas pela série temporal} \\ w_{prox}, & \text{se } v_i \text{ e } v_j \text{ são conectados apenas pela proximidade geoespacial} \\ w_{st} + w_{prox}, & \text{se } v_i \text{ e } v_j \text{ são conectados pelas duas condições} \\ 0, & \text{caso contrário} \end{cases} \quad (5.1)$$

dado que

$$w = \frac{w_{prox}}{w_{st}} \quad (5.2)$$

em que w é um parâmetro definido pelo usuário e equivale à razão entre w_{prox} e w_{st} . Dado o valor w , qualquer solução que satisfaça a Equação 5.2 produzirá o mesmo resultado no algoritmo da CSPG.

Analisando a modelagem do grafo G , pode-se afirmar que um dado vértice v_i pode ser conectado a, no máximo, $2K + 8$ vértices. Considerando as arestas pela proximidade geoespacial, v_i pode estar conectado a no máximo 8 vértices, que corresponde ao número máximo de pixels adjacentes a um dado pixel. Analisando as arestas pela série temporal, as arestas (v_i, v_j) , ou (v_j, v_i) (por ser grafo não direcionado), são criadas para os vértices $v_j \in \mathcal{N}_i$ e para o caso em que $v_i \in \mathcal{N}_j$, resultando assim em, no máximo, $2K$. Portanto, considerando o número máximo de vizinhos por vértice $2K + 8$, é concluído que o número máximo de arestas do grafo é $n(2K + 8)$.

Para representação do grafo é usada uma matriz de adjacência W , tal que se $W(i, j) = W_{ij}$ corresponde ao peso da aresta que conecta os vértices v_i e v_j , e $W(i, j) = 0$ indica ausência de aresta entre o par de vértices.

5.3 Rotulação do grafo

Após a construção da matriz de adjacência, é utilizado um processo similar à propagação de rótulos, baseado na abordagem definida em (ZHOU *et al.*, 2004). A matriz W precisa ser normalizada simetricamente para convergência no processo de propagação de rótulos (ZHOU *et al.*, 2004). Assim, a matriz normalizada S é definida pela Equação 5.3:

$$S = D^{-\frac{1}{2}} W D^{-\frac{1}{2}} \quad (5.3)$$

em que D é uma matriz diagonal, na qual $D(i, i) = \sum_{j=0}^n W(i, j)$. Dessa maneira, cada elemento $W(i, j)$ é normalizado de acordo com a soma das linhas i e j , $S(i, j) = W(i, j) / \sqrt{D(i, i)D(j, j)}$. A Equação 5.3 é a mesma utilizada para normalização simétrica da matriz de adjacência em (NG *et al.*, 2002; ZHOU *et al.*, 2004).

Para o processo de propagação de rótulos, cada vértice recebe os rótulos propagados por seus vizinhos através das arestas. Ao final do processo, cada vértice armazena a quantidade recebida de cada rótulo, correspondente a cada classe do problema. Assim, é definida uma matriz F de tamanho $n \times c$, em que n é o número de instâncias de treinamento e c é o número de classes. O elemento $F(i, j)$ corresponde à quantidade de rótulo classe j recebida pelo vértice v_i . Além de F , é definida uma matriz constante Y , que corresponde aos rótulos iniciais do conjunto de treinamento, na qual $Y(i, j) = 1$ se $x_i \in X_R$ e $x_i.y = j$, e $Y(i, j) = 0$, caso contrário. O processo iterativo de propagação de rótulos é dado pela Equação 5.4 (ZHOU *et al.*, 2004):

$$F^{t+1} = \alpha S F^t + (1 - \alpha) Y \quad (5.4)$$

em que α é um parâmetro definido pelo usuário e indica a porcentagem dos rótulos que um vértice recebe através de seus vizinhos, e $(1 - \alpha)$ é a porcentagem do rótulo inicial que é mantido pelos vértices.

Seja F^* a matriz de rótulos armazenados quando há convergência do processo iterativo, cada vértice correspondente a uma instância não rotulada $x_i \in X_{NR}$ é associado à classe cujo rótulo está armazenado em maior quantidade, ou seja:

$$x_i.y = \underset{1 \leq j \leq c}{\operatorname{argmax}} F^*(i, j). \quad (5.5)$$

para toda instância $x_i \in X_{NR}$.

Na Figura 19, é ilustrado o funcionamento da CSPG. Os vértices rotulados e não rotulados do grafo, sem arestas, são mostrados na Figura 19a. Na Figura 19b, são mostradas apenas algumas arestas que conectam os vértices, por limitação de espaço. Após criação do grafo, é iniciado o processo de propagação de rótulos, em que cada vértice não rotulado, em uma determinada iteração armazena uma quantidade de rótulo referente a cada classe do problema, como exibido na Figura 19c. Finalmente, os vértices não rotulados são associados à classe cujo rótulo foi recebido em maior quantidade (Figura 19d).

Figura 19 – Exemplo funcionamento da técnica CSPG, desde a criação do grafo até a propagação de rótulos.

Fonte: Elaborada pelo autor.

O comportamento e influência de cada parâmetro no processo de classificação da CSPG são discutidos na Seção 6.3. O algoritmo da CSPG é apresentado no Algoritmo 1.

Algoritmo 1: CSPG

Input: Conjunto de instâncias rotuladas $X_R = \{x_1, \dots, x_r\}$, conjunto de instâncias não rotuladas $X_{NR} = \{x_{r+1}, \dots, x_n\}$, conjunto de treinamento $X = X_R \cup X_{NR}$, com atributos $A = (lat, long, st, y)$

C : número de classes do problema;

w : parâmetro relacionado ao peso das arestas;

k : número de vizinhos pela série temporal;

α : parâmetro da propagação de rótulos;

$erroLim$: valor de erro limitante;

Y : matriz de rótulos iniciais.

$Dist$: função de distância para séries temporais

Output: Rótulo das instâncias de X_{NR} .

```

1 calculaArestasSerieTemporal( $X, k, w, Dist, W$ ) // Adiciona pesos  $w_{st}$  à matriz  $W$ 
2 calculaArestasProximidade( $X, w, W$ )           // Adiciona pesos  $w_{prox}$  à matriz  $W$ 
3 for  $i = [1..n]$  do
4    $D(i, i) \leftarrow \sum_j^n W(i, j)$ 
5  $S \leftarrow D^{-1/2} W D^{-1/2}$  //  $S$  é matriz normalizada
6 repeat
7    $F^{t+1} \leftarrow \alpha S F + (1 - \alpha) Y$ 
8    $e \leftarrow calculaErroQuadratico(F)$ 
9 until  $e < erroLim$ ;
10 for  $i = [(r+1)..n]$  do
11    $rotulo(i) = \operatorname{argmax}_{1 \leq j \leq C} F(i, j)$ 
12 return  $rotulo$ 

```

Custo computacional

O espaço utilizado pelo algoritmo consiste na quantidade de memória alocada para as matrizes S , Y e F , já que o espaço da matriz W é o mesmo usado após a normalização para armazenar a matriz S . As matrizes F e Y possuem $n \times c$ e $r \times c$, respectivamente. No entanto, como no processo iterativo de propagação de rótulos é necessário calcular $F^{(t+1)}$ com base nos valores de F^t , é necessário armazenar uma matriz auxiliar com mesmo número de elementos de F . Como o número máximo de arestas do grafo é $n(2K + 8)$, tem-se que a matriz S possui, no máximo, $n(2K + 8)$ elementos diferentes de zero. Como elementos de valor zero indicam ausência de aresta, não é necessário armazená-los em memória, reduzindo o tamanho da matriz S de n^2 elementos para $n(2K + 8)$, no caso em que $n \gg K$.

Portanto, o espaço necessário é a soma $2nc + rc + 2nK + 8n$. Dado que na maioria dos casos, $n \gg K$, $n \gg c$ e $n \gg r$, tem-se que o custo de utilização de memória é de $O(n)$.

Para analisar a complexidade algorítmica da CSPG, são calculados os custos de 3 passos do algoritmo: 1) encontrar K vizinhos mais próximos pela série temporal; 2) calcular proximidade

geoespacial e 3) processo iterativo de propagação de rótulos. Para encontrar os K vizinhos mais próximos sem a utilização de aproximações, é necessário calcular a distância entre as séries temporais de cada par de instâncias no conjunto de treinamento. O uso da função de distância $Dist$ apresenta custo c_{dist} para cálculo da distância entre duas séries temporais de valores reais. Seja m o comprimento das séries temporais, $c_{dist} = O(m)$ para distâncias como a Euclidiana e $c_{dist} = O(m^2)$ para funções como a DTW ou LCSS, utilizando programação dinâmica. Portanto o custo total do cálculo de todas as distâncias entre pares de instâncias é $O(n^2m)$ ou $O(n^2m^2)$. No caso de séries temporais multidimensionais, com D dimensões, c_{dist_mult} pode variar dependendo da adaptação de função de distância utilizada para o caso multidimensional, mas pode ser aproximado para $c_{dist_mult} = d c_{dist}$. Nesse caso, o custo do cálculo das distâncias seria de $O(n^2md)$ ou $O(n^2m^2d)$.

Para encontrar a vizinhança pela proximidade geoespacial tem-se a resolução espacial da imagem de satélite e as coordenadas do pixel extremo superior esquerdo. Com isso, pode-se fazer a conversão de (latitude, longitude) para (linha, coluna) do pixel na imagem de satélite, para todas as n instâncias de treinamento. Uma maneira de encontrar a vizinhança pela proximidade é alocar uma matriz auxiliar para mapear as instâncias assim como os pixels na imagem de satélite, encontrando assim os vizinhos adjacentes a cada instância. A conversão de coordenadas apresenta custo $O(n)$ e a localização dos vizinhos $O(8n) = O(n)$.

Para o processo de propagação de rótulos, em cada iteração é necessário calcular os elementos da matriz $F^{(t+1)}$, que corresponde à multiplicação das matrizes S e F^t , seguida da soma dos elementos de Y . A multiplicação das matrizes apresenta custo $(2nK + 8n)c = 2nKc + 8nc$, e a soma dos elementos possui custo rc . Portanto, o custo combinado em uma iteração é de $2nKc + 8nc + rc$, de modo que o custo até a convergência do processo iterativo é de $2nKct + 8nct + rct$, em que t é o número de iterações. Nesse caso, se $Kct \approx n$ tem-se um custo de $O(n^2)$, porém os valores de K e c são, em geral, muito inferiores a n . Uma estimativa do número de iterações foi obtida na realização dos experimentos apresentados no Capítulo 6, para os quais com $n \approx 200.000$, $K \approx 30$ e $c = 2$, a média do número de iterações foi de $t \approx 200$.

Por fim, pode-se concluir que dentre os custos apresentados, de maneira geral, o cálculo de distâncias entre as séries temporais das instâncias do conjunto de treinamento é o que apresenta maior custo computacional (caso de menor custo $O(n^2m)$, com distância Euclidiana e série temporal unidimensional). Em relação ao tamanho do conjunto de treinamento n , pode-se afirmar que o custo da técnica CSPG é de $O(n^2)$.

5.4 Considerações finais

Neste capítulo foi apresentada a técnica “Classificação Semissupervisionada usando Proximidade Geoespacial” (CSPG), que utiliza a abordagem baseada em grafos para classificação semissupervisionada de séries temporais obtidas a partir de imagens de satélite. Na

CSPG, é construído um grafo que incorpora as informações de proximidade geográfica e das séries temporais das instâncias analisadas. O processo de propagação de rótulos é aplicado no grafo, para classificação dos vértices não rotulados de acordo com as arestas que representam a similaridade entre pares de vértices, usando as informações de proximidade geoespacial e temporal. É importante mencionar que a técnica CSPG foi incorporada à implementação do *framework* SITSMining na ferramenta SatImagExplorer. No Capítulo 6, são apresentados experimentos envolvendo a análise dos parâmetros da CSPG, análise comparativa entre a CSPG e técnicas propostas na literatura de classificação supervisionada e semissupervisionada, e o uso da técnica CSPG com o *framework* SITSMining para classificação de áreas reais usando o sistema SatImagExplorer.

EXPERIMENTOS

Neste capítulo, é detalhado o processo de validação da técnica proposta, por meio da realização de estudo experimental. A técnica CSPG foi utilizada para classificação semissupervisionada de dados reais obtidos a partir de séries temporais de imagens de satélite, visando a identificação de áreas de plantio de cana-de-açúcar. Os principais objetivos do estudo experimental são:

- Definir valores adequados e analisar o impacto dos parâmetros da técnica CSPG para a classificação dos dados obtidos a partir de imagens de satélite.
- Realizar uma análise comparativa entre a técnica CSPG e algoritmos propostos na literatura de classificação supervisionada e semissupervisionada. Avaliar o potencial de identificação de áreas de cana-de-açúcar da técnica proposta em comparação aos outros algoritmos envolvidos na análise, quando utilizados para classificação de conjuntos de dados reais extraídos de imagens de satélite.
- Estudar a aplicabilidade da técnica proposta na classificação de dados reais, com objetivo de dar suporte a pesquisas em agricultura.

Os experimentos apresentados neste capítulo podem ser divididos em quatro grupos:

1. Experimentos para definição de parâmetros dos algoritmos de classificação envolvidos na análise comparativa, de funções de distância e parâmetros relacionados aos conjuntos de dados usados nos experimentos dos outros três grupos.
2. Experimentos envolvendo a técnica CSPG, para avaliação do impacto de seus parâmetros para a classificação dos conjuntos de dados utilizados.
3. Experimentos para análise comparativa entre a CSPG e algoritmos de classificação propostos na literatura.

4. Experimentos de classificação de dados reais obtidos a partir de imagens de satélite usando a implementação do *framework* SITSMining , utilizando a CSPG como método de classificação.

Nas seções seguintes, é definida a metodologia usada para o estudo experimental, são detalhados o planejamento e a execução dos quatro grupos de experimentos, e são analisados e discutidos os principais resultados obtidos.

6.1 Séries Temporais de Imagens de Satélite

Os dados reais utilizados no estudo experimental consistem em séries temporais de imagens capturadas por sensores de dois satélites: 1) TERRA¹, usando o sensor *Moderate Resolution Imaging Spectroradiometer* (MODIS) e 2) *National Oceanic and Atmospheric Administration* (NOAA)², usando o sensor *Advanced Very High Resolution Radiometer* (AVHRR). As imagens dos satélites TERRA e NOAA foram submetidas a métodos de pré-processamento para correções radiométrica, geométrica e atmosférica, posteriormente foram transformadas em imagens NDVI (*Normalized Difference Vegetation Index*) (ROUSE *et al.*, 1973). O NDVI é um índice de vegetação amplamente usado em pesquisas de agricultura e uso de solo, e está relacionado a medidas de biomassa e quantidade de vegetação (JUSTICE; HIERNAUX, 1986; PRICE, 1993).

As imagens TERRA utilizadas nos experimentos são produto do método denominado Composição de Máximo Valor (MVC – do inglês *Maximum Value Composite*) (HOLBEN, 1986). O método MVC analisa 16 imagens TERRA diárias e seleciona apenas o melhor valor NDVI para cada pixel, produzindo uma única imagem contendo os melhores valores NDVI para o período de tempo analisado. A produção da imagem NDVI pelo método MVC prioriza a qualidade de imagem, em troca da diminuição da resolução temporal. Assim, as imagens NDVI do satélite TERRA apresentam resolução espacial de 1km × 1km e resolução temporal de 16 dias. Isso significa que cada pixel de uma imagem corresponde ao valor NDVI de uma área real de 1km² para um período de 16 dias (Tabela 1).

As imagens capturadas pelo satélite NOAA possuem resolução espacial de 1,1km × 1,1km e resolução temporal de 30 dias (Tabela 1). Tais imagens também são produto do método MVC, resultado da análise de 30 imagens NOAA diárias.

Devido à diferença na resolução temporal dos satélites TERRA e NOAA, a quantidade de imagens disponíveis para um dado período de tempo é diferente para cada satélite. Por exemplo, o período de um ano da safra cana-de-açúcar é representado por 12 imagens NOAA e 23 imagens TERRA (Tabela 1). A safra de cana tem início no mês de abril de um ano e término em março do ano seguinte.

¹ http://www.nasa.gov/mission_pages/terra/index.html

² <http://www.noaa.gov/satellites.html>

Tabela 1 – Informações das imagens dos satélites TERRA e NOAA.

Satélite	Resolução Espacial	Resolução Temporal	Imagens por Safra de Cana
TERRA	1km × 1km	16 dias	23 imagens
NOAA	1,1km × 1,1km	30 dias	12 imagens

Fonte: Dados da pesquisa.

A região de interesse corresponde ao conjunto de coordenadas (latitude, longitude) que referenciam as áreas reais de interesse. Para os experimentos realizados neste trabalho, a região de interesse corresponde às áreas do estado de São Paulo, ilustradas em azul na Figura 20. São Paulo é principal estado produtor de cana-de-açúcar no país.

Figura 20 – Região de interesse para geração do conjunto de dados.

Fonte: Elaborada pelo autor.

É possível notar que a região próxima ao litoral, localizada ao sudeste do estado de São Paulo, não pertence à região de interesse. Essas áreas não foram consideradas, pois não se tem informações relativas ao plantio de cana-de-açúcar nessa região, o que inviabiliza a avaliação de resultados dos experimentos para essas áreas.

Para a região de interesse, as informações de plantio de cana-de-açúcar foram obtidas a partir da máscara desenvolvida no projeto Canasat/INPE³. A máscara corresponde ao mapeamento das áreas de cultivo de cana-de-açúcar desenvolvido por especialistas no domínio de agrometeorologia. O processo de geração da máscara consiste em duas etapas: 1) classificação de imagens de satélite de média resolução espacial, gerando um mapeamento inicial das culturas de cana-de-açúcar; e 2) refinamento do mapeamento por meio de ajustes e correções realizadas manualmente pelos especialistas.

A máscara de cana-de-açúcar é formada por pares de coordenadas (latitude, longitude) que indicam a localização dos campos de cultivo da cana em uma determinada safra, por

³ <http://www.dsr.inpe.br/laf/canasat>

exemplo 2004/2005. O uso da máscara de cana-de-açúcar, assim como a metodologia aplicada nos experimentos, são detalhados na próxima seção.

6.2 Metodologia do Estudo Experimental

Para avaliação e validação da técnica CSPG, foram realizadas duas análises principais:

1. **Análise de parâmetros:** 1) definição dos melhores valores para parâmetros dos algoritmos usados na análise comparativa; 2) estudo da sensibilidade dos parâmetros da técnica CSPG e seu impacto na classificação de dados reais obtidos a partir de séries temporais de imagens de satélite (Seção 6.3).
2. **Análise comparativa:** comparação dos resultados de classificação obtidos pela CSPG em relação aos obtidos por técnicas propostas na literatura para classificação supervisionada e semisupervisionada (Seção 6.4).

Para as duas análises, a abordagem usada nos experimentos consiste na classificação de cada área real da região de interesse em **Cana** ou **NãoCana**, baseada no comportamento de sua vegetação durante o período de uma safra de cana-de-açúcar. Assim, em cada experimento, são classificadas séries temporais de NDVI correspondentes ao período de abril de um ano até março do ano seguinte. A seguir, é detalhado o processo de geração dos conjuntos de dados usados nos experimentos.

6.2.1 Conjunto de dados

Cada conjunto de dados utilizado nos experimentos de classificação foi obtido a partir de uma SITS do satélite TERRA ou do NOAA. Além disso, um conjunto de dados está relacionado a apenas uma safra de cana-de-açúcar. Por exemplo, um conjunto de dados referente à safra de 2004/2005 contém instâncias que armazenam informações das áreas reais da região de interesse para o período de abril de 2004 a março de 2005.

Um conjunto de dados contendo n instâncias é definido por $X = \{x_1, x_2, \dots, x_n\}$, em que uma instância $\mathbf{x}$ possui os seguintes atributos:

- **lat:** coordenada latitude da área real referenciada pela instância $\mathbf{x}$.
- **long:** coordenada longitude da área real referenciada pela instância $\mathbf{x}$.
- **ts:** série temporal de NDVI dada por $ts = (s_1, s_2, \dots, s_m)$, e correspondente ao comportamento do índice de vegetação na área de coordenadas (lat, long), ao longo de uma safra de cana-de-açúcar.

É importante notar que ts é gerada a partir de uma SITS, usando o mesmo procedimento descrito na camada de Extração e Pré-processamento do *framework* SITSMining (Capítulo 4). Por isso, devido à diferença na resolução temporal dos satélites TERRA e NOAA, são obtidos diferentes valores de m : para o TERRA, o comprimento da série temporal de NDVI é $m = 23$, e para o NOAA é $m = 12$.

- y : rótulo da instância x , em que $y = \mathbf{Cana}$ se a área referenciada por x é de cana-de-açúcar, e $y = \mathbf{NãoCana}$, caso contrário.

A definição completa dos conjuntos de dados usados nos experimentos é mostrada na Tabela 2.

Tabela 2 – Definição do conjunto de dados.

Satélite	NOAA ou TERRA
Safra	ano safra de cana-de-açúcar
$ X $ (NOAA)	174.156
$ X $ (TERRA)	201.157
Atributo 1	$lat \in \mathbb{R}$
Atributo 2	$long \in \mathbb{R}$
Atributo 3	$ts = (s_1, s_2, \dots, s_m), s_i \in [-1, +1]$
Atributo 4	$y \in \{\mathbf{Cana}, \mathbf{NãoCana}\}$

Fonte: Dados da pesquisa.

Para definição do rótulo de cada instância, é usada a máscara de cana-de-açúcar descrita na Seção 6.1. A máscara consiste em um conjunto de pares de coordenadas que referenciam as áreas reais de plantio de cana em uma determinada safra. Assim, se o par de coordenadas (lat , $long$) de uma instância x estiver presente na máscara, $x.y = \mathbf{Cana}$, caso contrário, $x.y = \mathbf{NãoCana}$. Na Figura 21, são ilustradas a máscara de cana-de-açúcar para a safra 2004/2005 e a coloração da região de interesse de acordo com o rótulo de cada área.

Após a obtenção dos rótulos, o conjunto de dados X é particionado em dois conjuntos:

- X_R : conjunto de instâncias rotuladas.
- X_{NR} : conjunto de instâncias não rotuladas.

Seja $|X| = n$, as instâncias de X_R e X_{NR} são amostradas aleatoriamente a partir de X , em que $|X_R| = s$ e $|X_{NR}| = n - s$. Para os algoritmos de classificação, os rótulos y das instâncias de X_{NR} são omitidos e usados posteriormente para avaliação do resultado da classificação.

Nos experimentos, os conjuntos X_R e X_{NR} são usados nas etapas de treinamento e teste dos algoritmos de classificação. Entretanto o uso dos conjuntos está associado ao tipo de classificação utilizada:

Figura 21 – Máscara da safra 2004/2005 da cana-de-açúcar (a) e aplicação da máscara para rotulação dos pixels da região de interesse.

Fonte: Elaborada pelo autor.

1. **Caso supervisionado:** o treinamento do classificador é realizado usando apenas as instâncias de X_R . Para a etapa de teste, o objetivo é inferir o rótulo das instâncias de X_{NR} .
2. **Caso semissupervisionado:** o classificador é treinado usando as instâncias de X_R e X_{NR} . Para o teste, assim como no caso supervisionado, são classificadas as instâncias de X_{NR} .

O processo de criação dos conjuntos de treinamento e teste a partir dos conjuntos X_R e X_{NR} é ilustrado na Figura 22.

Figura 22 – Processo de geração do conjunto de instâncias rotuladas e não rotuladas, e definição dos conjuntos de treinamento e teste para os casos supervisionado e semissupervisionado.

Fonte: Elaborada pelo autor.

Em um cenário real de classificação de séries temporais de NDVI, na ausência da máscara de cana-de-açúcar, a criação do conjunto de instâncias rotuladas X_R provavelmente é realizada pelo especialista no domínio de aplicação. Devido à complexidade dos dados analisados, o tempo e esforço humano necessários para rotulação de uma grande quantidade de instâncias são muito elevados, o que resulta na criação de um conjunto X_R muito reduzido em relação ao tamanho

de X . Por isso, para simular esse cenário real, os experimentos foram realizados com valores de amostragem $s = \{100, 200, 300, 400\}$ para o conjunto X_R , o que corresponde a porcentagens inferiores a 0.3% dos conjuntos de dados X usados nos experimentos.

Outro parâmetro, que mede o balanceamento entre as classes no conjunto X_R , é definido como a razão r entre a quantidade n_{RC} de instâncias rotuladas da classe **Cana** e a quantidade n_{RNC} de instâncias rotuladas da classe **NãoCana**:

$$r = \frac{n_{RC}}{n_{RNC}} \quad (6.1)$$

em que $n_{RC} + n_{RNC} = s$.

Um experimento preliminar foi realizado para verificar o valor de r para o qual os algoritmos usados na análise comparativa apresentam os melhores resultados de classificação. Os detalhes de execução desse experimento são apresentados na Seção 6.3.

6.2.2 Algoritmos de classificação

Para análise comparativa com a técnica CSPG, foram usados algoritmos consolidados na literatura de classificação supervisionada e semissupervisionada. Foram selecionados três algoritmos de classificação supervisionada e dois de classificação semissupervisionada:

Classificação supervisionada	Classificação semissupervisionada
<ul style="list-style-type: none"> • <i>K Nearest Neighbors</i> (KNN) • <i>Support Vector Machines</i> (SVM) • <i>Naive Bayes</i> (NB) 	<ul style="list-style-type: none"> • <i>Transductive Support Vector Machines</i> (TSVM) • <i>Linear Neighborhood Propagation</i> (LNP)

Os algoritmos KNN e LNP foram implementados na linguagem C++, assim como a técnica proposta CSPG. Para SVM e TSVM, foi utilizado o software *SVMLight*⁴ (JOACHIMS, 1999) e para o algoritmo Naive Bayes foi usado o pacote 'e1071' (DIMITRIADOU *et al.*, 2008), desenvolvido na linguagem R. Os melhores valores de parâmetros para cada algoritmo foram encontrados empiricamente e são apresentados na Seção 6.3.

6.2.3 Avaliação dos resultados

A medida para avaliação dos resultados de classificação utilizada nos experimentos foi o MCC (*Matthew's Correlation Coefficient*) (MATTHEWS, 1975). O MCC foi adotado por ser mais robusto ao desbalanceamento de classes do que outras medidas tradicionais usadas em trabalhos de aprendizado de máquina e mineração de dados, como a acurácia, por exemplo.

⁴ <http://svmlight.joachims.org/>

Considerando a classe **Cana** como positiva, e **NãoCana** como negativa, a medida MCC é definida como:

$$MCC = \frac{VP \times VN - FP \times FN}{\sqrt{(VP + FP)(VP + FN)(VN + FP)(VN + FN)}} \quad (6.2)$$

em que:

- VP (Verdadeiro positivo): número de instâncias de **Cana** classificadas como **Cana**.
- VN (Verdadeiro negativo): número de instâncias de **NãoCana** classificadas como **NãoCana**.
- FP (Falso positivo): número de instâncias de **NãoCana** classificadas como **Cana**.
- FN (Falso negativo): número de instâncias de **Cana** classificadas como **NãoCana**.

A medida MCC avalia a qualidade de um resultado de classificação (quádrupla (VP, VN, FP, FN)) e está compreendida no intervalo $[-1.0; +1.0]$. O melhor e pior caso são ilustrados na Tabela 3.

Tabela 3 – Melhor e pior caso na avaliação de um resultado de classificação usando MCC.

	MCC	VP + VN	FP + FN
Melhor caso	+1.0	$ X_{NR} $	0
Pior caso	-1.0	0	$ X_{NR} $

É importante destacar que, para todos os experimentos apresentados neste capítulo, cada valor de MCC corresponde à média de 10 valores MCC produzidos por execuções de classificação usando diferentes particionamentos de X em X_R e X_{NR} , amostrados aleatoriamente.

6.3 Análise de parâmetros

Nesta seção, são detalhados experimentos para definição dos valores de parâmetros relacionados ao 1) conjunto de dados, 2) aos algoritmos de classificação usados na análise comparativa e 3) à técnica CSPG.

Vale mencionar que toda a análise de parâmetros foi baseada em uma série de experimentos realizados com diversas configurações e combinações de valores para esses parâmetros em todos os algoritmos. No entanto, visando uma melhor apresentação desses experimentos, os resultados finais foram sintetizados e organizados nas subseções seguintes.

Em todos os experimentos realizados, foram utilizados conjuntos de dados extraídos a partir de séries de imagens NDVI dos satélites TERRA e NOAA, e a medida de avaliação usada foi o MCC.

6.3.1 Parametrização do conjunto de dados

Para cada experimento de classificação apresentado nas seções seguintes, foi utilizado o mesmo valor do parâmetro r , que mede a razão entre o número de instâncias de classe positiva (**Cana**) e o número de instâncias de classe negativa (**NãoCana**). Para encontrar o melhor valor de r , foram realizados experimentos com diferentes valores de r para cada algoritmo de classificação mencionado na Seção 6.2.2. Para tanto, foi utilizado o conjunto de séries temporais extraídas de imagens do TERRA. Na Tabela 4, são mostrados os parâmetros para execução desse experimento. Os conjuntos X_{Cana} e $X_{NaoCana}$ correspondem aos subconjuntos de instâncias de classe **Cana** e **NãoCana**, respectivamente, em X .

Tabela 4 – Parametrização do experimento para análise do parâmetro r .

Safra da cana	2004/2005
Satélite	TERRA
$ X $	201.157
$ X_{Cana} $	23.543
$ X_{NaoCana} $	177.614
s	{100, 200, 300, 400}

Fonte: Dados da pesquisa.

Analizando os resultados do experimento, ilustrados na Tabela 5, pode-se concluir que os algoritmos de classificação apresentam os maiores valores de MCC, na maioria dos casos, para o valor $r = 0,5$.

6.3.2 Parametrização dos algoritmos de classificação

Para os algoritmos de classificação utilizados na análise comparativa, foram realizados experimentos para definição dos melhores valores para os seguintes parâmetros:

- **KNN**: número de vizinhos k .
- **LNP**: número de vizinhos k e porcentagem de informação recebida dos vértices vizinhos na propagação de rótulos α .
- **SVM**: função de kernel e parâmetro γ usado no kernel RBF.
- **TSVM**: função de kernel e parâmetro γ usado no kernel RBF.

Os melhores valores para esses parâmetros foram definidos empiricamente e são exibidos na Tabela 6. Para os demais parâmetros de aprendizado da SVM e TSVM, foram usados os valores definidos por padrão na implementação do *SVMLight*. É importante ressaltar que os valores exibidos na Tabela 6 foram utilizados na parametrização dos algoritmos em todos os experimentos apresentados no restante deste capítulo.

Tabela 5 – Resultados de MCC para o experimento de análise do parâmetro r .

		$r = 0,3$	$r = 0,4$	$r = 0,5$	$r = 0,6$	$r = 0,7$
$s = 100$	KNN	0,1588	0,1850	0,1898	0,1700	0,1368
	LNP	0,0903	0,1627	0,2002	0,1878	0,1769
	SVM	0,0404	0,1242	0,1781	0,0978	0,0273
	TSVM	0,0207	0,0689	0,1732	0,1289	0,0629
	NB	0,0842	0,0868	0,0966	0,1078	0,1111
		$r = 0,3$	$r = 0,4$	$r = 0,5$	$r = 0,6$	$r = 0,7$
$s = 200$	KNN	0,1690	0,2047	0,2178	0,2016	0,1752
	LNP	0,0924	0,1686	0,2240	0,2343	0,2147
	SVM	0,0699	0,2019	0,2143	0,1622	0,0922
	TSVM	0,0210	0,0511	0,2238	0,1161	0,0033
	NB	0,1131	0,1119	0,1167	0,1264	0,1315
		$r = 0,3$	$r = 0,4$	$r = 0,5$	$r = 0,6$	$r = 0,7$
$s = 300$	KNN	0,1897	0,2098	0,2221	0,2063	0,1777
	LNP	0,0985	0,1742	0,2253	0,2372	0,2055
	SVM	0,1122	0,2298	0,2222	0,1798	0,0865
	TSVM	0,0000	0,0630	0,2336	0,1765	0,0721
	NB	0,1117	0,1274	0,1344	0,1213	0,1197
		$r = 0,3$	$r = 0,4$	$r = 0,5$	$r = 0,6$	$r = 0,7$
$s = 400$	KNN	0,1968	0,2164	0,2317	0,2094	0,1813
	LNP	0,1023	0,1828	0,2313	0,2494	0,2184
	SVM	0,1484	0,2404	0,2438	0,1956	0,1071
	TSVM	0,0001	0,1089	0,2365	0,2060	0,1062
	NB	0,1199	0,1252	0,1214	0,1223	0,1148

Fonte: Dados da pesquisa.

Dos algoritmos de classificação, apenas o KNN permite o uso de diferentes funções de distância. Logo, para definição de uma função de distância adequada para o KNN, no contexto deste trabalho, foi realizado um experimento envolvendo três tradicionais funções de distância compatíveis com séries temporais: 1) Euclidiana (L_2), 2) *Dynamic Time Warping* (DTW) e 3) *Longest Common Subsequence* (LCSS). A configuração do experimento é mostrada na Tabela 7. Na Tabela 8, são exibidos os melhores valores encontrados empiricamente para os parâmetros das funções de distância DTW e LCSS. O parâmetro δ corresponde à janela de ajuste para DTW e LCSS, e o parâmetro ε da LCSS é um valor limitante, tal que se a diferença entre dois valores de NDVI for inferior a ε , eles são considerados iguais.

Analisando os resultados apresentados na Tabela 9, para a classificação do conjunto da

Tabela 6 – Valores dos parâmetros para cada algoritmo de classificação.

Algoritmo	Parâmetro	Valor
KNN	k	7
LNP	k α	11 0,98
SVM	Kernel γ	RBF 15
TSVM	Kernel γ	RBF 15

Fonte: Dados da pesquisa.

Tabela 7 – Configuração do experimento de função de distância para KNN.

Safra da cana	2004/2005
Satélite	TERRA
$ X $	201.157
$ X_{Cana} $	23.543
$ X_{NaoCana} $	177.614
s	{100, 200, 300, 400}
r	0,5

Fonte: Dados da pesquisa.

Tabela 8 – Valores dos parâmetros das funções de distância DTW e LCSS.

Função de distância	Parâmetro	Valor
DTW	δ	2
LCSS	δ	2
LCSS	ε	0,06

Fonte: Dados da pesquisa.

safra 2004/2005 do satélite TERRA, o algoritmo KNN combinado com a função de distância DTW apresentou os melhores valores de MCC. No entanto, a diferença entre os valores de MCC obtidos usando a distância Euclidiana e a DTW, para todos os valores de s , não é maior que 0,003. Portanto, pode-se dizer que o algoritmo KNN combinado com as duas funções de distância apresenta praticamente a mesma qualidade de classificação, nesse contexto. Para os experimentos apresentados na análise comparativa, a função de distância utilizada para o algoritmo KNN foi a DTW.

6.3.3 Parametrização da técnica CSPG

Nesta seção, são analisados os três parâmetros da técnica CSPG:

- α : porcentagem de informação recebida dos vértices vizinhos na propagação de rótulos.

Tabela 9 – Valores de MCC obtidos no experimento de função de distância para KNN.

s	L_2	DTW	LCSS
100	0,1874	0,1898	0,1585
200	0,2173	0,2172	0,1952
300	0,2215	0,2221	0,2034
400	0,2307	0,2317	0,2081

Fonte: Dados da pesquisa.

- k : número de vizinhos de cada vértice, considerando a similaridade pela série temporal de NDVI.
- w : razão entre o peso w_{st} das arestas relacionadas às séries temporais, e o peso w_{prox} das arestas que conectam vértices pela proximidade geoespacial.

Foram realizados diversos experimentos a fim de analisar o impacto de cada parâmetro na classificação de dados obtidos a partir de imagens NDVI. É importante ressaltar que esses experimentos foram realizados usando a distância Euclidiana para definição da vizinhança pela série temporal, no grafo da CSPG. A escolha da função de distância é baseada nos resultados obtidos no experimento realizado com o algoritmo KNN. Nesse experimento, as funções Euclidiana e DTW apresentaram os melhores resultados, produzindo valores MCC muito próximos. No entanto, como o cálculo de distância entre as séries temporais corresponde à maior parte do custo computacional do algoritmo da CSPG ($O(n^2)$), foi adotada a distância Euclidiana, por apresentar custo linear em relação ao tamanho da série temporal.

Trabalhos da literatura de classificação semissupervisionada que utilizam a propagação de rótulos em grafos, em geral, definem o valor do parâmetro α empiricamente. Em sua maioria, esses trabalhos utilizam nos experimentos valores de α próximos a 1,0, como $\alpha = 0,99$ (ZHOU *et al.*, 2004; WANG; ZHANG, 2008). Para analisar os resultados produzidos pela CSPG com diferentes valores de α , foi realizado um experimento envolvendo a classificação do conjunto de dados TERRA para safra 2004/2005, variando os valores de α , k e w . A configuração para esse experimento é ilustrada na Tabela 10. Vale mencionar que a diferença entre cada valor de w foi definida como 10, pois para valores muito próximos, a diferença no MCC obtido pela CSPG é muito pequena.

Os resultados do experimento são apresentados na Tabela 11, em que cada valor corresponde à média dos MCC obtidos nas execuções variando os valores de k e w , para cada par de valores de α e s . É importante mencionar que dos 10 valores de w testados no experimento, são exibidos apenas os 5 melhores na Tabela 11. Os resultados mostram que, para $\alpha = 0.95$, a CSPG obteve os maiores valores médios de MCC. No entanto, é possível observar que a diferença de MCC de $\alpha = 0.95$ para $\alpha = 0.94$ e $\alpha = 0.96$ é muito pequena, o que indica que esses valores de α também são adequados para essa análise. No restante dos experimentos descritos neste

Tabela 10 – Configuração do experimento para análise do parâmetro α .

Safra da cana	2004/2005
Satélite	TERRA
$ X $	201.157
s	{100, 200, 300, 400}
α	{0, 90; 0, 91; 0, 92; 0, 93; 0, 94; 0, 95; 0, 96; 0, 97; 0, 98; 0, 99}
k	{1, 5, 9, 13, 17, 21, 25, 29, 33, 37}
w	{1, 11, 21, 31, 41}

Fonte: Dados da pesquisa.

capítulo, será usado $\alpha = 0.95$.

Tabela 11 – Resultados de MCC para o experimento de análise do parâmetro α .

	α				
	0.94	0.95	0.96	0.97	0.98
$s = 100$	0.2523	0.2546	0.2556	0.2534	0.2431
$s = 200$	0.2886	0.2889	0.2872	0.2816	0.2670
$s = 300$	0.2997	0.3000	0.2985	0.2927	0.2764
$s = 400$	0.2757	0.2762	0.2746	0.2690	0.2546

Fonte: Dados da pesquisa.

Com a definição de $\alpha = 0.95$, é necessário analisar o impacto causado pelo número e peso das arestas usados para construção do grafo no processo de propagação de rótulos. O número de arestas de proximidade geoespacial é limitado a 8, correspondente ao número máximo de pixels adjacentes a um dado pixel na imagem de satélite. Já o número de arestas definidas pela similaridade entre séries temporais de NDVI é determinado pelo parâmetro k . Por fim, para definir o peso de cada tipo de aresta, é usado o parâmetro w , de modo que:

- se $w < 1$, as arestas de série temporal têm maior peso ($w_{st} > w_{prox}$).
- se $w > 1$, as arestas de proximidade geoespacial têm maior peso ($w_{st} < w_{prox}$).
- se $w = 1$, as arestas dos dois tipos possuem o mesmo peso ($w_{st} = w_{prox}$).

Assim, pode-se dizer que a configuração $w = 1$ e k próximo a 8 representa um caso em que os dois tipos de arestas contribuem de maneira equivalente para a propagação de rótulos no grafo. Para se obter uma ideia preliminar a respeito do impacto de cada tipo de aresta nos resultados de classificação da CSPG, foi realizado um experimento com objetivo de comparar o algoritmo da original da CSPG com duas variações:

- **CSPG_{st}**: propagação de rótulos no grafo formado apenas por arestas que conectam vértices considerando a similaridade entre as séries temporais de NDVI. O número de vizinhos para cada vértice é definido pelo parâmetro k .
- **CSPG_{prox}**: propagação de rótulos em grafo formado apenas por arestas de proximidade geoespacial. O número de vizinhos de um vértice é definido apenas pelo número de pixels adjacentes ao pixel representado por ele, na imagem de satélite.

A configuração desse experimento é apresentada na Tabela 12. Como apenas a CSPG original utiliza dois tipos de arestas, o parâmetro w não é necessário para CSPG_{st} e CSPG_{prox}. Os valores de parâmetros para a CSPG foram definidos como $w = 1$ e $k = 8$, na tentativa de não favorecer nenhum dos tipos de arestas no processo de propagação de rótulos.

Tabela 12 – Configuração de experimento de comparação entre CSPG, CSPG_{st} e CSPG_{prox}.

Safra da cana	2004/2005
Satélite	TERRA
$ X $	201.157
s	$\{100, 200, 300, 400\}$
CSPG	$\alpha = 0.95 \quad w = 1 \quad k = 8$
CSPG _{ts}	$\alpha = 0.95 \quad k = 8$
CSPG _{prox}	$\alpha = 0.95$

Fonte: Dados da pesquisa.

Os resultados do experimento são ilustrados na Figura 23. Analisando o gráfico, pode ser observado que os resultados obtidos pelas variações da CSPG são muito próximos e ambos são inferiores aos obtidos pela CSPG original. Isso indica que os dois tipos de arestas provavelmente contribuem de maneira equilibrada na propagação de rótulos, para obtenção dos resultados de classificação da CSPG original. Além disso, pode-se dizer que a propagação de rótulos no grafo com apenas um dos tipos de arestas não é capaz de superar os resultados produzidos pelo grafo que combina similaridade das séries temporais de NDVI e a proximidade geoespacial.

Para analisar mais detalhadamente o comportamento dos parâmetros w e k no processo de classificação da CSPG, foram realizados experimentos variando os valores desses parâmetros, com $\alpha = 0.95$, para a safra de 2004/2005 do satélite TERRA. É importante mencionar que nesse experimento, assim como no experimento para definição do valor de α (Tabelas 10 e 11), não são ilustrados os testes para $w < 1$. Esses testes foram omitidos, pois com todas as combinações de α e k para $w < 1$, os resultados produzidos foram inferiores aos obtidos para $w \geq 1$. Por isso, optou-se por sintetizar os resultados, apresentando apenas os testes para os casos $w = 1$ e $w > 1$.

Os resultados do experimento para análise de w e k são apresentados na Tabela 13, em que os números destacados indicam o maior valor MCC para um dado k e s , considerando todos os valores de w . Para simplificar a análise dos resultados, eles podem ser divididos em três grupos:

Figura 23 – Resultados MCC da comparação entre CSPG e as variações $CSPG_{st}$ e $CSPG_{prox}$.

Fonte: Dados da pesquisa.

- **Alta prioridade para séries temporais:** quando k é elevado e w é próximo de 1, tem-se o mesmo peso para os dois tipos de arestas, porém um número muito maior de arestas que conectam vértices pela série temporal.
- **Alta prioridade para proximidade geoespacial:** quando k é baixo, e w é elevado, tem-se aproximadamente o mesmo número de arestas para os dois tipos, porém o peso das arestas de proximidade é muito maior.
- **Mesma prioridade para os dois tipos de aresta:** quando k e w são baixos, tem-se aproximadamente o mesmo número de arestas de cada tipo e mesmo peso para ambos. No caso em que k e w são elevados, há maior equilíbrio que nos dois outros grupos, pois apesar do aumento na quantidade de arestas pela série temporal, eleva-se o peso das arestas por proximidade, que são em menor número.

Analisando os resultados mostrados na Tabela 13, é possível observar que, de maneira geral, os resultados nos casos de alta prioridade para séries temporal e alta prioridade para proximidade geoespacial apresentam valores MCC inferiores aos que apresentam aproximadamente mesma prioridade para os dois tipos de aresta. Além disso, considerando apenas os valores MCC para mesma prioridade, observa-se que elevar w e k é mais benéfico para o processo de classificação dos dados analisados do que usar baixos valores para esses parâmetros. No entanto, é possível notar que a diferença entre cada valor MCC destacado diminui à medida que o valor de k é aumentado. Isso indica que a melhoria de MCC obtida ao elevar-se w e k possui uma tendência a se estabilizar para valores muito grandes desses parâmetros.

É fato que o aumento do número de vizinhos pela série temporal implica a elevação da ocorrência de casos de um vértice ser conectado tanto pela aresta de série temporal quanto

Tabela 13 – Resultados de MCC para o experimento de análise dos parâmetros w e k .

$s = 100$	k	$w = 1$	$w = 11$	$w = 21$	$w = 31$	$w = 41$
	1	0.2240	0.1976	0.1937	0.1921	0.1909
	5	0.2558	0.2416	0.2347	0.2321	0.2298
	9	0.2568	0.2592	0.2531	0.2498	0.2470
	13	0.2437	0.2666	0.2622	0.2598	0.2569
	17	0.2389	0.2715	0.2687	0.2668	0.2645
	21	0.2344	0.2743	0.2737	0.2719	0.2704
	25	0.2299	0.2749	0.2764	0.2753	0.2743
	29	0.2260	0.2750	0.2788	0.2787	0.2779
	33	0.2229	0.2745	0.2802	0.2809	0.2806
	37	0.2203	0.2733	0.2811	0.2825	0.2824
$s = 200$	k	$w = 1$	$w = 11$	$w = 21$	$w = 31$	$w = 41$
	1	0.2457	0.2222	0.2173	0.2144	0.2123
	5	0.2759	0.2709	0.2617	0.2557	0.2519
	9	0.2530	0.2877	0.2796	0.2738	0.2692
	13	0.2433	0.2960	0.2900	0.2849	0.2805
	17	0.2384	0.3000	0.2974	0.2929	0.2886
	21	0.2347	0.3017	0.3025	0.2987	0.2947
	25	0.2303	0.3022	0.3061	0.3033	0.3000
	29	0.2253	0.3001	0.3074	0.3064	0.3038
	33	0.2223	0.2981	0.3083	0.3089	0.3068
	37	0.2198	0.2957	0.3085	0.3103	0.3091
$s = 300$	k	$w = 1$	$w = 11$	$w = 21$	$w = 31$	$w = 41$
	1	0.2563	0.2332	0.2273	0.2240	0.2217
	5	0.2882	0.2831	0.2728	0.2663	0.2615
	9	0.2702	0.3022	0.2921	0.2851	0.2797
	13	0.2612	0.3122	0.3036	0.2969	0.2914
	17	0.2534	0.3165	0.3115	0.3053	0.2997
	21	0.2464	0.3181	0.3168	0.3112	0.3061
	25	0.2409	0.3184	0.3205	0.3168	0.3120
	29	0.2342	0.3156	0.3221	0.3201	0.3161
	33	0.2306	0.3130	0.3229	0.3222	0.3194
	37	0.2268	0.3094	0.3230	0.3236	0.3218
$s = 400$	k	$w = 1$	$w = 11$	$w = 21$	$w = 31$	$w = 41$
	1	0.2685	0.2440	0.2373	0.2339	0.2316
	5	0.3041	0.2930	0.2812	0.2741	0.2689
	9	0.2928	0.3126	0.3008	0.2927	0.2865
	13	0.2831	0.3227	0.3127	0.3051	0.2983
	17	0.2750	0.3279	0.3208	0.3134	0.3070
	21	0.2650	0.3305	0.3261	0.3193	0.3137
	25	0.2599	0.3304	0.3297	0.3245	0.3188
	29	0.2535	0.3286	0.3317	0.3279	0.3232
	33	0.2488	0.3265	0.3332	0.3308	0.3268
	37	0.2443	0.3233	0.3335	0.3326	0.3294

Fonte: Dados da pesquisa.

pela aresta de proximidade geoespacial (tendo assim peso $w_{st} + w_{prox}$). A tendência, nesse caso, é que duas áreas geograficamente vizinhas e que também apresentam alta similaridade pela

série temporal de NDVI, pertençam à mesma classe (**Cana** ou **NãoCana**). Vale mencionar que apenas esse caso não é o suficiente para melhorar o MCC, pois quando k é elevado e w é baixo, priorizando arestas de série temporal, também há a ocorrência de intersecção dos dois tipos de arestas, mas o MCC produzido é inferior ao caso com mesma prioridade. Na Figura 24, é ilustrado o aumento da ocorrência da intersecção dos tipos de aresta, em função do aumento de k .

Figura 24 – Número de intersecções dos tipos de aresta em função de k .

Fonte: Dados da pesquisa.

Considerando os resultados apresentados na Tabela 13, para os experimentos da análise comparativa, o algoritmo da CSPG é usado para classificação com $\alpha = 0.95$, $w = 31$ e $k = 37$, pois essa configuração de parâmetros foi a que obteve maior MCC médio para os valores de s utilizados no experimentos apresentados.

6.4 Análise comparativa

Para validação da técnica proposta, foram realizados experimentos visando a análise comparativa entre a CSPG e algoritmos de classificação supervisionada e semissupervisionada mencionados na Seção 6.2.2: KNN, SVM, Naive bayes, LNP e TSVM. Os valores de parâmetros dos demais algoritmos de classificação utilizados são os apresentados na Tabela 6, e a técnica CSPG foi parametrizada com $\alpha = 0.95$, $w = 31$ e $k = 37$. Os conjuntos de dados utilizados nos experimentos foram obtidos a partir de séries de imagens NDVI dos satélites TERRA e NOAA, para diferentes safras de cana-de-açúcar. Foram realizados dois principais experimentos para análise comparativa:

- **Experimento 1:** classificação de conjuntos de dados dos satélites TERRA e NOAA para a safra de 2004/2005. O objetivo é comparar os resultados de classificação obtidos pela

CSPG e os algoritmos de classificação, na análise de conjuntos de dados extraídos de imagens de satélites distintos, para o mesmo período. Os conjuntos de dados e parâmetros usados no Experimento 1 são mostrados na Tabela 14.

- **Experimento 2:** classificação de conjuntos de dados do satélite TERRA para as safras de 2004/2005, 2011/2012. O objetivo é comparar os resultados obtidos pela CSPG e os algoritmo de classificação para diferentes safras de cana-de-açúcar no período de 2004 a 2012.

Tabela 14 – Configuração do experimento de análise comparativa para TERRA e NOAA, safra 2004/2005.

	Conjunto 1	Conjunto 2
Safra da cana	2004/2005	2004/2005
Satélite	TERRA	NOAA
$ X $	201.157	174.156
$ X_{Cana} $	23.543	22.574
$ X_{NaoCana} $	177.614	151.402
s	{100, 200, 300, 400}	
r	0.5	

Fonte: Dados da pesquisa.

Nas Figuras 29a e 29b, são mostrados os resultados obtidos por cada algoritmo de classificação para os conjuntos de dados dos satélites TERRA e NOAA, respectivamente.

Os resultados exibidos na Figura 25 mostram que a técnica CSPG apresenta valores médios de MCC superiores aos dos outros algoritmos de classificação, considerando os dois conjuntos de dados. No conjunto de dados do satélite NOAA, é possível notar que a SVM é superior à TSVM e os outros algoritmos, exceto a CSPG. Nesse caso, pode-se concluir que a suposição de separação por baixa densidade, adotada pela TSVM não é verdadeira para esse conjunto de dados, pois o uso de dados não rotulados resultou em um valor de MCC inferior ao da SVM, que considera apenas dados rotulados.

Ao comparar a técnica CSPG e LNP, deve-se analisar o impacto da construção do grafo usando cada algoritmo, pois a etapa de rotulação no grafo é a mesma (propagação de rótulos) para as duas técnicas. Como a CSPG se mostrou superior para os dois conjuntos de dados, pode-se concluir que a suposição adotada pela CSPG reflete mais precisamente do que a suposição da LNP a maneira como as classes do problema são organizadas e qual é a relação entre as instâncias de cada classe, sejam rotuladas ou não rotuladas. Em outras palavras, pode-se dizer que é menos provável que as instâncias de cada classe possam ser reconstruídas linearmente pelas vizinhas considerando a série temporal (suposição da LNP) do que as instâncias de mesma classe estejam conectadas caso elas apresentem mesmo comportamento de vegetação ou estejam próximas geograficamente (suposição da CSPG).

Figura 25 – Resultados de classificação (MCC) para a safra 2004/2005 e satélites TERRA e NOAA.

(a) Conjunto de dados TERRA.

(b) Conjunto de dados NOAA.

Fonte: Dados da pesquisa.

É possível observar que, para todos os algoritmos de classificação, são obtidos valores MCC mais elevados com o aumento do número s de instâncias rotuladas utilizadas para treinamento dos classificadores. Para verificar a significância estatística dos resultados, foi aplicado o Teste-t pareado. A utilização do Teste-t pareado é comum em trabalhos das áreas de mineração de dados e aprendizado de máquina, incluindo classificação semissupervisionada (ZHU *et al.*, 2006; LI *et al.*, 2011).

O Teste-t pode ser utilizado para verificar se a diferença entre os resultados obtidos por dois algoritmos de classificação é considerada estatisticamente significativa. Para os experimentos descritos nesta seção, o Teste-t foi aplicado para comparar a técnica CSPG com cada algoritmo de classificação envolvido na análise comparativa. O cálculo do Teste-t é exibido na Equação

6.3:

$$t_{N,k-1} = \frac{\bar{d}}{s_d/\sqrt{k}} \quad (6.3)$$

em que:

- N : nível de confiança desejado.
- k : número de execuções de classificação para obtenção de cada valor MCC ($k = 10$). O subscrito $k - 1$ corresponde aos graus de liberdade para o Teste-t.
- $\bar{d}$: média das diferenças entre os valores MCC obtidos por dois algoritmos de classificação nas k execuções.
- s_d : desvio padrão de MCC nas k execuções.

Como o valor t possui uma distribuição *t-Student*, é possível afirmar, para um nível de confiança de 95%, que:

- se $t > 2,62$: valores MCC obtidos pela CSPG são superiores aos do algoritmo concorrente;
- se $t < -2,62$: valores MCC obtidos pela CSPG são inferiores aos do algoritmo concorrente;
- se $-2,62 \leq t \leq 2,62$: não há diferença estatística entre os valores obtidos pelos dois algoritmos.

Na Figura 26, são ilustrados os resultados da aplicação do Teste-t pareado aos resultados do experimento com dados dos satélites TERRA e NOAA. A realização do teste de significância estatística permite afirmar que a CSPG apresenta superioridade em relação aos algoritmos concorrentes, para classificação dos dados referentes à safra de 2004/2005 do satélite TERRA. Para os dados do satélite NOAA, nota-se que não há diferença significativa entre os resultados obtidos pela CSPG e os algoritmos SVM e Naive Bayes, para $s = 100$. Nos demais casos, os resultados obtidos pela CSPG são superiores.

No Experimento 2, são classificados dados das safras 2004/2005 e 2011/2012, obtidos a partir de séries de imagens NDVI, do satélite TERRA. Os conjuntos de dado e a configuração desse experimento são apresentados na Tabela 15. Os resultados obtidos, exibidos na Figura 27, permitem afirmar que os valores médios apresentados pela CSPG para as duas safras são superiores aos dos outros algoritmos de classificação. No entanto, em especial para a safra 2011/2012, é possível notar uma diminuição dos valores MCC obtidos por todos os algoritmos considerados, inclusive a CSPG, para esse conjunto de dados.

Figura 26 – Teste-t pareado dos resultados de classificação da safra 2004/2005 dos satélites TERRA e NOAA.

Fonte: Dados da pesquisa.

Tabela 15 – Configuração do experimento de função de distância para CSPG.

	Conjunto 1	Conjunto 2
Safra da cana	2004/2005	2011/2012
Satélite	TERRA	TERRA
$ X $	201.157	201.157
$ X_{Cana} $	23.543	36.717
$ X_{NaoCana} $	177.614	164.440
s	{100, 200, 300, 400}	
r	0.5	

Fonte: Dados da pesquisa.

Para verificar a significância estatística dos resultados obtidos, foi realizado o Teste-t pareado, assim como no primeiro experimento da análise comparativa. Os resultados do Teste-t são exibidos na Figura 28. Diferentemente dos resultados para a safra 2004/2005, em que a CSPG apresentou superioridade em relação aos outros algoritmos para todos os valores de s , em 2011/2012, não há diferença entre a CSPG e SVM, para $s = 100$ e $s = 200$, e entre CSPG e TSVM, para $s = 100$. A partir desses resultados e considerando a redução dos valores MCC obtidos por todos os algoritmos, pode-se concluir que os dados da safra 2011/2012 apresentam maior dificuldade de análise em relação aos da safra 2004/2005, para classificação visando identificação das áreas de cana-de-açúcar. Além disso, é esperada uma diminuição nos valores de MCC, pois os valores para os parâmetros dos algoritmos foram definidos a partir de experimentos com o conjunto TERRA para 2004/2005 e provavelmente não são os valores ótimos para classificação do conjunto 2011/2012.

Figura 27 – Resultados de classificação (MCC) para as safras 2004/2005 e 2011/2012 do satélite TERRA.

(a) Conjunto de dados TERRA – safra 2004/2005.

(b) Conjunto de dados TERRA – safra 2011/2012.

Fonte: Dados da pesquisa.

6.5 CSPG e *framework* SITSMining

Com objetivo de ilustrar a aplicabilidade da CSPG como método de classificação na Camada de Mineração de Dados do *framework* SITSMining (detalhado no Capítulo 4), foi realizado um experimento para classificação do conjunto de dados do satélite TERRA na safra de 2004/2005, para a região do estado de São Paulo. Utilizando a implementação do *framework* na ferramenta SatImagExplorer, é possível obter um gráfico do mapeamento das áreas reais da região de interesse associadas ao plantio cana-de-açúcar, de acordo com a predição realizada pelo algoritmo CSPG. Os parâmetros e dados de entrada do *framework* SITSMining para realização desse experimento são apresentados na Tabela 16.

O *framework* recebe três itens de entrada: 1) Uma série temporal de 23 imagens NDVI

Figura 28 – Teste-t pareado dos resultados de classificação das safras 2004/2005 e 2011/2012 do satélite TERRA.

Fonte: Dados da pesquisa.

Tabela 16 – Configuração do experimento usando *framework* SITSMining e CSPG.

Camada	Parâmetros	Valores
Entrada	Séries de imagens	TERRA (2004/2005)
	Região de interesse	Estado de SP sem litoral
	Conjunto de dados rotulados	300 instâncias
Extração e Pré-processamento	Normalização	Não
	Tratamento de valores faltantes	Interpolação linear
Mineração de dados	Tarefa	Classificação Semissupervisionada
	Método	CSPG ($w = 31, k = 37, \alpha = 0,95$)
	Função de distância	Euclidiana

Fonte: Dados da pesquisa.

do satélite TERRA, correspondentes à safra de 2004/2005; 2) Região de interesse do estado de São Paulo, exceto as áreas litorâneas e 3) Um conjunto de dados previamente rotulados. O conjunto de dados rotulados foi gerado de maneira similar ao método descrito na Seção 6.2, a partir de amostragem aleatória do conjunto de instâncias gerado na Camada de Extração de Pré-processamento do *framework*. A geração do conjunto de dados rotulados pode ser feita após a extração dos dados, pois ele é necessário apenas para o processo de classificação, na Camada de Mineração de Dados. Das 300 instâncias rotuladas, 150 pertencem à classe **Cana** e as demais 150 à classe **NãoCana**.

Na Camada de Extração de Pré-processamento, o conjunto de dados foi gerado a partir da extração de valores NDVI das imagens do satélite TERRA, para os pixels correspondentes à região de interesse fornecida. O conjunto de dados é formado por 200.857 instâncias, sendo 23.393 de **Cana** e 177.464 de **NãoCana**. Os valores faltantes foram substituídos usando o cálculo de interpolação linear. Como os valores NDVI são obtidos no intervalo $[-1.0; +1.0]$, não é necessário realizar sua normalização. Para a Camada de Mineração de Dados, foi escolhida a tarefa de classificação semissupervisionada usando a técnica CSPG combinada com a distância

Euclidiana.

Os resultados de classificação produzidos, em formato de tabela verdade, são ilustrados na Tabela 17. O valor MCC obtido é de 0,343. Na Camada de Saída do *framework*, os resultados foram interpretados para geração do gráfico de visualização geoespacial, ilustrado na Figura 29.

Tabela 17 – Tabela verdade para experimento usando *framework* SITSMining e CSPG.

Classe Real \ Classe Predita	Classe Predita	
	Cana	NãoCana
Cana	78.07%	21.93%
NãoCana	27.61%	72.39%

Fonte: Dados da pesquisa.

Figura 29 – Visualização geoespacial dos resultados de classificação da CSPG e *framework* SITSMining (a), e a ilustração das áreas de cana-de-açúcar obtidas a partir de máscara de cana, para safra de 2004/2005 (b).

Fonte: Dados da pesquisa.

De acordo com especialistas em agrometeorologia, os resultados apresentados usando visualização geoespacial permitem afirmar que a técnica CSPG é capaz de classificar corretamente as áreas das principais regiões produtoras de cana-de-açúcar do estado de São Paulo. Além disso, mesmo nas regiões classificadas como **Cana**, é possível notar a presença de áreas pequenos conjuntos de áreas classificados como **NãoCana**, que correspondem a áreas urbanas de municípios.

Ao analisar os resultados ilustrados na Figura 29, pode-se notar que, em geral, há agrupamentos de áreas rotuladas com mesma classe. Isso ocorre porque os vértices são conectados, em parte, por arestas de proximidade geoespacial. Assim a localização geográfica tem grande impacto na propagação rótulos. Por outro lado, a presença de arestas que conectam vértices pela similaridade das séries temporais de NDVI permite que vértices recebam informações de rótulos propagados por vértices que não sejam próximos geograficamente próximos a ele.

Provavelmente, como nas áreas urbanas a série temporal de NDVI é muito distinta das de áreas de cana-de-açúcar, embora os vértices que representam cana e cidades estejam próximos geograficamente, seus vizinhos pela série temporal de NDVI serão distintos. Como há a tendência de que os vizinhos pela série de NDVI das áreas de cana propaguem o rótulo **Cana**, e os vizinhos de cidade propaguem o rótulo **NãoCana**, a discriminação desses dois tipos de áreas é facilitada. No entanto, a discriminação entre áreas de cana-de-açúcar e de pastagens ou outras culturas agrícolas é dificultada, pois os vizinhos pela série de NDVI podem ser os mesmos, segundo a distância Euclidiana. Assim, além de estarem possivelmente próximos geograficamente, esses vértices podem estar conectados aos mesmos vizinhos pela série de NDVI, o que pode resultar na rotulação dessas áreas com a mesma classe.

Além do experimento usando a técnica CSPG, foram realizados diversos estudos voltados a pesquisas na área de aplicação utilizando a implementação do *framework* SITSMining na ferramenta SatImagExplorer. A tarefa de agrupamento de séries temporais extraídas de imagens de satélite foi utilizada na análise de uso de terra (GONCALVES *et al.*, 2014) e de mudança de uso de terra (SCRIVANI *et al.*, 2014), e na análise comparativa do índice NDVI para os sensores AVHRR/NOAA e MODIS (GONCALVES *et al.*, 2013). A SatImagExplorer também foi utilizada para classificação semissupervisionada de séries temporais multidimensionais, obtidas a partir de SITS de NDVI, Albedo e temperatura de superfície (ROMANI *et al.*, 2013). Em todos esses trabalhos, as tarefas de planejamento de experimentos e avaliação dos resultados foram realizadas com apoio dos pesquisadores da Embrapa Informática Agropecuária e Cepagri/Unicamp.

6.6 Considerações finais

Neste capítulo, foi apresentado o estudo experimental realizado para validação da técnica de classificação semissupervisionada CSPG. Nos experimentos, foram utilizados dados reais correspondentes a séries de imagens de dois satélites, com imagens do índice de vegetação NDVI. Em um dos experimentos, foram discutidos os impactos da variação de cada parâmetro da técnica CSPG, e concluiu-se que tanto a informação geográfica quanto a série temporal de NDVI apresentam importante contribuição para o classificador da CSPG, para identificação de áreas de plantio de cana-de-açúcar. Além disso, para os dados analisados, os resultados obtidos usando a distância Euclidiana são equiparados aos da DTW, apesar da simplicidade do algoritmo e baixo custo computacional.

Também foi realizada uma análise comparativa envolvendo a CSPG, os métodos de classificação supervisionada KNN, SVM e Naive Bayes; e de classificação semissupervisionada LNP e TSVM. Para mais de um conjunto de dados reais, a CSPG mostrou superioridade na maioria dos casos em relação aos outros métodos de classificação. No entanto, é importante destacar a dificuldade de classificação dos conjuntos apresentados, especialmente para o satélite TERRA na safra 2011/2012, em que é notável o decréscimo no valor médio de MCC de todos os

algoritmos analisados, em relação à safra de 2004/2005.

A partir desses resultados, pode-se concluir que a técnica CSPG, juntamente com o *framework* SITSMining, representam um importante recurso para suporte em análises de dados extraídos de séries temporais de imagens de satélite. A implementação da CSPG no sistema SatImagExplorer oferece ao especialista no domínio de aplicação um importante recurso computacional para apoio a pesquisas no área de agrometeorologia.

CONCLUSÃO

Neste trabalho de mestrado foram propostas uma metodologia de análise e uma técnica de mineração de dados obtidos a partir de séries temporais de imagens de satélite. As contribuições apresentadas nesta dissertação são ferramentas úteis para o endereçamento de problemas da área de agrometeorologia de maneira automatizada, minimizando o esforço requerido do especialista e ampliando a possibilidade de realização de estudos que requerem suporte computacional.

Os estudos experimentais realizados indicam o potencial de aplicação da metodologia proposta e da técnica desenvolvida, principalmente na análise de grandes quantidades de dados temporais extraídos de imagens de satélite de baixa resolução espacial, produzindo resultados promissores, de acordo com especialistas nesse domínio.

7.1 Principais contribuições

Este trabalho apresentou as seguintes contribuições principais:

1. metodologia de análise de dados extraídos a partir de séries temporais de imagens de satélite, denominada *framework* SITSMining. Essa abordagem é útil ao especialista em agrometeorologia, pois oferece uma metodologia adequada para análise de dados obtidos a partir de sensores orbitais. O *framework* define um processo de análise que compreende as tarefas de extração de séries temporais a partir de sequências de imagens de satélite, pré-processamento das série temporais e transformação para um formato adequado ao processo de mineração de dados, aplicação de análise de classificação ou agrupamento, e retorno dos resultados para o usuário em um formato adequado que facilita a interpretação e reconhecimento dos padrões encontrados na análise. Os experimentos realizados com o *framework* SITSMining e a técnica de classificação semissupervisionada proposta visando o mapeamento das áreas de plantio de cana-de-açúcar mostram a aplicabilidade

da metodologia, pois incorpora os passos essenciais da análise de mineração de séries temporais obtidas a partir de imagens de satélite.

2. técnica CSPG (Classificação Semissupervisionada usando Proximidade Geoespacial), que utiliza uma abordagem semissupervisionada baseada em grafos para classificação de séries temporais obtidas a partir de sequências de imagens de satélite. A principal característica da técnica proposta é a modelagem dos dados extraídos de imagens de satélites utilizando as informações de proximidade geoespacial e da série temporal do índice da imagem. Essa estratégia é útil na modelagem de dados referentes a áreas de plantio de culturas agrícolas, como a de cana-de-açúcar. Como existe a tendência da plantação de culturas em áreas vastas e próximas, a utilização da informação de vizinhança geográfica e da semelhança entre o comportamento da vegetação de cada área ao longo de uma safra de cana-de-açúcar favorece o processo de classificação dessas áreas. Os resultados dos experimentos de análise comparativa com métodos propostos na literatura de classificação supervisionada e semissupervisionada mostram que a técnica CSPG é superior aos demais métodos, para classificação de séries temporais visando identificação de áreas de plantio de cana-de-açúcar.
3. extensão da ferramenta SatImagExplorer com a incorporação do *framework* SITSMining e a técnica CSPG. Com isso, todo o processo de mineração de dados extraídos de séries temporais de imagens de satélite é realizado de forma automática na ferramenta, o que é de grande valia para o especialista no domínio de aplicação. A SatImagExplorer vem sendo utilizada em trabalhos científicos da área de agrometeorologia por pesquisadores da Embrapa Informática Agropecuária e Cepagri/Unicamp.
4. conjunto de estudos experimentais visando análises de uso de terra e identificação de áreas de cana-de-açúcar usando tarefas de mineração de dados. Em parceria com especialistas em agrometeorologia, foram desenvolvidos trabalhos utilizando a implementação do *framework* SITSMining na ferramenta SatImagExplorer para análise de dados obtidos a partir de SITS, usando métodos de classificação e agrupamento. Os experimentos realizados nesses trabalhos foram planejados e avaliados com apoio dos pesquisadores da Embrapa Informática Agropecuária e Cepagri/Unicamp.
5. publicações resultantes dos trabalhos realizados:
 - AMARAL, B. F.; GONCALVES, R. R. V.; ROMANI, L. A. S.; SOUSA, E. P. M. . Aprimorando a classificação semissupervisionada de séries temporais extraídas de imagens de satélite. In: Proceedings of the II Symposium on Knowledge Discovery, Mining and Learning KDMILE, São Carlos – SP, 2014, 8p.
 - AMARAL, B. F.; CHINO, D. Y. T.; ROMANI, L. A. S.; GONCALVES, R. R. V.; TRAINA, A. J. M.; SOUSA, E. P. M. . The SITSMining Framework: a Data Mining

Approach for Satellite Image Time Series. In: Proceedings of the 16th International Conference on Enterprise Information Systems (ICEIS 2014) Lisboa, Portugal, 2014, p. 225–232.

- GONCALVES, R. R. V.; ZULLO JR., J.; AMARAL, B. F.; COLTRI, P. P.; SOUSA, E. P. M.; ROMANI, L. A. S. . Land use temporal analysis through clustering techniques on satellites image time series. In: IGARSS IEEE International Geoscience and Remote Sensing Symposium, Québec, Canadá, 2014, p. 2173–2176.
- SCRIVANI, R.; AMARAL, B. F.; GONCALVES, R. R. V.; SOUSA, E. P. M.; ZULLO JR., J.; ROMANI, L. A. S. . Identificação da mudança de uso da terra usando técnicas de agrupamento de séries temporais de imagens de satélite. In: 5º Geopantanal - Simpósio de Geotecnologias na Pantanal, Campo Grande – MS, 2014, p. 554–563.
- GONCALVES, R. R. V.; ZULLO JR., J.; MARQUEZINI, O.; AMARAL, B. F.; SOUSA, E. P. M.; ROMANI, L. A. S. . Análise da relação entre os perfis de NDVI obtidos dos sensores AVHRR/NOAA e MODIS nas áreas produtoras de cana-de-açúcar em São Paulo. In: Simpósio Brasileiro de Sensoriamento Remoto, Foz do Iguaçu – PR, 2013, p. 640–647.
- ROMANI, L. A. S.; AMARAL, B. F.; GONCALVES, R. R. V.; ZULLO JR., J.; SOUSA, E. P. M. . Aplicação de técnicas de classificação semissupervisionada para análise de séries multitemporais de imagens de satélite. In: Anais do XVI Simpósio Brasileiro de Sensoriamento Remoto, Foz do Iguaçu – PR, 2013, p. 1750–1757.

Está em andamento a elaboração de um trabalho a ser submetido ao periódico IEEE Transactions on Geoscience and Remote Sensing (TGRS).

7.2 Trabalhos futuros

O trabalho desenvolvido neste projeto de mestrado representa um ponto de partida para a realização de pesquisa em diferentes vertentes de trabalhos futuros. As principais propostas são:

1. desenvolvimento de uma função de distância capaz de discriminar o conteúdo de diferentes áreas reais de acordo com a dissimilaridade entre suas séries temporais de um índice de vegetação;
2. utilização, adaptação ou desenvolvimento de uma representação de séries temporais que evidencie características relevantes para identificação de culturas agrícolas como a cana-de-açúcar;

3. extensão do *framework* SITSMining para adição de novas tarefas de mineração de dados, como regras de associação, detecção de anomalias e novos recursos para apresentação dos resultados, de acordo com a necessidade do especialista em agrometeorologia;
4. aprimoramento da técnica CSPG, explorando alternativas tanto no processo de modelagem de dados em grafo quanto na etapa de rotulação. A incorporação de conhecimento aprofundado no domínio de aplicação pode resultar na criação de um grafo que represente de maneira mais fiel o relacionamento entre as áreas reais referenciadas pelas imagens de satélite;
5. aplicação e, se necessário, adaptação da técnica CSPG para classificação de dados obtidos a partir imagens de satélite de média e alta resolução espacial;
6. aplicação da técnica CSPG para identificação de outros alvos, como diferentes culturas agrícolas, lagos, florestas, entre outros.

REFERÊNCIAS

- AMARAL, B. F.; CHINO, D. Y. T.; ROMANI, L. A. S.; GONCALVES, R. R. V.; TRAINA, A. J. M.; SOUSA, E. P. M. The sitsmining framework: a data mining approach for satellite image time series. In: **16th International Conference on Enterprise Information Systems**. Lisboa, Portugal: SCITEPRESS, 2014. p. 225–232. Citado 4 vezes nas páginas [36](#), [40](#), [41](#) e [42](#).
- BATISTA, G. E.; WANG, X.; KEOGH, E. J. A complexity-invariant distance measure for time series. In: **Proceedings of the 2011 SIAM International Conference of Data Mining**. Mesa, USA: SIAM, 2011. (SDM '11, v. 11), p. 699–710. Citado na página [27](#).
- BELKIN, M.; MATVEEVA, I.; NIYOGI, P. Regularization and semi-supervised learning on large graphs. In: **Proceedings of the 17th Annual Conference on Learning Theory**. Banff, Canada: Springer, 2004. (COLT '04, v. 3120), p. 624–638. Citado na página [24](#).
- BERNDT, D.; CLIFFORD, J. Using dynamic time warping to find patterns in time series. In: **AAAI Workshop on Knowledge Discovery in Databases**. Seattle, USA: AAAI Press, 1994. p. 359–370. Citado na página [31](#).
- BIRANT, D.; KUT, A. St-dbscan: An algorithm for clustering spatial–temporal data. **Data & Knowledge Engineering**, Elsevier, v. 60, n. 1, p. 208–221, 2007. Citado na página [9](#).
- BLUM, A.; CHAWLA, S. Learning from labeled and unlabeled data using graph mincuts. In: **Proceedings of the 18th International Conference on Machine Learning**. Williamstown, USA: Morgan Kaufmann, 2001. (ICML '01), p. 19–26. Citado na página [24](#).
- BLUM, A.; MITCHELL, T. Combining labeled and unlabeled data with co-training. In: **ACM. Proceedings of the 11th Annual Conference on Computational Learning Theory**. Madison, USA, 1998. (COLT '98), p. 92–100. Citado na página [21](#).
- CAMPS-VALLS, G.; MARSHEVA, T. V. B.; ZHOU, D. Semi-supervised graph-based hyperspectral image classification. **Geoscience and Remote Sensing, IEEE Transactions on**, IEEE, v. 45, n. 10, p. 3044–3054, 2007. Citado na página [24](#).
- CASTELLI, V.; COVER, T. M. On the exponential value of labeled samples. **Pattern Recognition Letters**, Elsevier, v. 16, n. 1, p. 105–111, 1995. Citado na página [17](#).
- CHAPELLE, O.; SCHÖLKOPF, B.; ZIEN, A. (Ed.). **Semi-Supervised Learning**. Cambridge, MA: MIT Press, 2006. Citado 6 vezes nas páginas [9](#), [10](#), [12](#), [13](#), [19](#) e [20](#).
- CHAPELLE, O.; SINDHWANI, V.; KEERTHI, S. S. Optimization techniques for semi-supervised support vector machines. **The Journal of Machine Learning Research**, JMLR.org, v. 9, p. 203–233, 2008. Citado na página [18](#).
- CHEN, L.; NG, R. On the marriage of lp-norms and edit distance. In: **Proceedings of the Thirtieth International Conference on Very Large Data Bases**. Toronto, Canada: Morgan Kaufmann, 2004. v. 30, p. 792–803. Citado na página [34](#).

- CHEN, L.; ÖZSU, M. T.; ORIA, V. Robust and fast similarity search for moving object trajectories. In: **Proceedings of the 2005 ACM SIGMOD international conference on Management of data**. Baltimore, USA: ACM, 2005. (SIGMOD '05), p. 491–502. Citado na página 34.
- CHEN, Y.; HU, B.; KEOGH, E.; BATISTA, G. E. Dtw-d: time series semi-supervised learning from a single example. In: **Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data mining**. Chicago, USA: ACM, 2013. (KDD '13), p. 383–391. Citado na página 32.
- CHINO, D. Y. T.; ROMANI, L. A. S.; TRAINA, A. J. M. Construindo séries temporais de imagens de satélite para sumarização de dados climáticos e monitoramento de safras agrícolas. **Revista Eletrônica de Iniciação Científica**, v. 10, p. 1–16, 2010. Citado na página 42.
- COLLINS, M.; SINGER, Y. Unsupervised models for named entity classification. In: **Proceedings of the joint SIGDAT conference on empirical methods in natural language processing and very large corpora**. College Park, USA: [s.n.], 1999. p. 100–110. Citado na página 21.
- COZMAN, F. G.; COHEN, I.; CIRELO, M. C. *et al.* Semi-supervised learning of mixture models. In: **Proceedings of the 20th International Conference on Machine Learning**. Washington D.C., USA: AAAI Press, 2003. (ICML '03), p. 99–106. Citado na página 17.
- CSISZAR, I.; GUTMAN, G. Mapping global land surface albedo from noaa avhrr. **Journal of Geophysical Research: Atmospheres (1984–2012)**, Wiley Online Library, v. 104, n. D6, p. 6215–6228, 1999. Citado na página 41.
- DAI, W.; XUE, G.-R.; YANG, Q.; YU, Y. Co-clustering based classification for out-of-domain documents. In: **Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. San Jose, USA: ACM, 2007. (KDD '07), p. 210–219. Citado na página 18.
- DARA, R.; KREMER, S. C.; STACEY, D. *et al.* Clustering unlabeled data with soms improves classification of labeled real-world data. In: IEEE. **Proceedings of the 2002 International Joint Conference on Neural Networks**. Honolulu, Hawaii, 2002. (IJCNN '02), p. 2237–2242. Citado na página 18.
- DEMIRIZ, A.; BENNETT, K. P.; EMBRECHTS, M. J. Semi-supervised clustering using genetic algorithms. **Artificial neural networks in engineering**, p. 809–814, 1999. Citado na página 18.
- DEMPSTER, A. P.; LAIRD, N. M.; RUBIN, D. B. Maximum likelihood from incomplete data via the em algorithm. **Journal of the royal statistical society. Series B (methodological)**, JSTOR, p. 1–38, 1977. Citado na página 16.
- DIMITRIADOU, E.; HORNIK, K.; LEISCH, F.; MEYER, D.; WEINGESSEL, A. Misc functions of the department of statistics (e1071), tu wien. **R package**, p. 1–5, 2008. Citado na página 63.
- DING, H.; TRAJCEVSKI, G.; SCHEUERMANN, P.; WANG, X.; KEOGH, E. Querying and mining of time series data: experimental comparison of representations and distance measures. **Proceedings of the VLDB Endowment**, VLDB Endowment, v. 1, n. 2, p. 1542–1552, 2008. Citado na página 30.

DUNHAM, J. G. Optimum uniform piecewise linear approximation of planar curves. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, IEEE, n. 1, p. 67–75, 1986. Citado na página 29.

FALOUTSOS, C.; RANGANATHAN, M.; MANOLOPOULOS, Y. Fast subsequence matching in time-series databases. **SIGMOD Rec.**, ACM, New York, USA, v. 23, n. 2, p. 419–429, 1994. Citado na página 28.

FAYYAD, U.; PIATETSKY-SHAPIO, G.; SMYTH, P. From data mining to knowledge discovery in databases. **AI magazine**, v. 17, n. 3, p. 37, 1996. Citado 2 vezes nas páginas 7 e 8.

FUJINO, A.; UEDA, N.; SAITO, K. A hybrid generative/discriminative approach to semi-supervised classifier design. In: **Proceedings of the National Conference on Artificial Intelligence**. Pittsburgh, USA: AAAI Press, 2005. v. 20, n. 2, p. 764. Citado na página 17.

GOLDBERG, A. B.; ZHU, X. Seeing stars when there aren't many stars: graph-based semi-supervised learning for sentiment categorization. In: ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. **Proceedings of the First Workshop on Graph Based Methods for Natural Language Processing**. [S.l.], 2006. p. 45–52. Citado na página 24.

GOLDMAN, S.; ZHOU, Y. Enhancing supervised learning with unlabeled data. In: **Proceedings of the Seventeenth International Conference on Machine Learning**. Stanford, USA: Morgan Kaufmann, 2000. (ICML '00), p. 327–334. Citado na página 21.

GONCALVES, R. R. V.; ZULLO, J.; AMARAL, B. F.; COLTRI, P. P.; SOUSA, E. P. M.; ROMANI, L. A. S. Land use temporal analysis through clustering techniques on satellite image time series. In: **Proceedings of the 2014 IEEE International Geoscience and Remote Sensing Symposium**. Quebec, Canada: [s.n.], 2014. (IGARSS '14), p. 2173–2176. Citado na página 81.

GONCALVES, R. R. V.; ZULLO, J.; MARQUEZIN, O.; AMARAL, B. F.; SOUSA, E. P. M.; ROMANI, L. A. S. Análise da relação entre os perfis de ndvi obtidos dos sensores avhrr/noaa e modis nas áreas produtoras de cana-de-açúcar em são paulo. In: **Anais do XVI Simpósio Brasileiro de Sensoriamento Remoto**. Foz do Iguaçu – PR: [s.n.], 2013. p. 640–647. Citado na página 81.

HAN, J.; KAMBER, M.; PEI, J. **Data mining: concepts and techniques: concepts and techniques**. [S.l.]: Elsevier, 2011. Citado 5 vezes nas páginas 2, 7, 9, 10 e 11.

HARRIS, F. J. On the use of windows for harmonic analysis with the discrete fourier transform. **Proceedings of the IEEE**, IEEE, v. 66, n. 1, p. 51–83, 1978. Citado na página 29.

HEIN, M.; AUDIBERT, J.-Y.; LUXBURG, U. V. Graph laplacians and their convergence on random neighborhood graphs. **Journal of Machine Learning Research**, v. 8, p. 1325–1368, Jun 2007. Citado na página 24.

HODGE, V. J.; AUSTIN, J. A survey of outlier detection methodologies. **Artificial Intelligence Review**, Springer, v. 22, n. 2, p. 85–126, 2004. Citado na página 10.

HOLBEN, B. N. Characteristics of maximum-value composite images from temporal avhrr data. **International Journal of Remote Sensing**, v. 7, n. 11, p. 1417–1434, 1986. Citado na página 58.

- JAAKKOLA, M. S. T.; SZUMMER, M. Partially labeled classification with markov random walks. **Advances in Neural Information Processing Systems 15 (NIPS'02)**, The MIT Press, Vancouver, Canada, v. 14, p. 945–952, 2002. Citado na página 24.
- JOACHIMS, T. Advances in kernel methods. In: SCHÖLKOPF, B.; BURGESS, C. J. C.; SMOLA, A. J. (Ed.). Cambridge, USA: MIT Press, 1999. cap. Making large-scale support vector machine learning practical, p. 169–184. Citado 2 vezes nas páginas 18 e 63.
- JOACHIMS, T. *et al.* Transductive learning via spectral graph partitioning. In: **Proceedings of the 20th International Conference on Machine Learning**. Washington D.C., USA: AAAI Press, 2003. (ICML '03, v. 3), p. 290–297. Citado na página 24.
- JUSTICE, C. O.; HIERNAUX, P. Monitoring the grasslands of the sahel using noaa avhrr data: Niger 1983. **International Journal of Remote Sensing**, Taylor & Francis, v. 7, n. 11, p. 1475–1497, 1986. Citado na página 58.
- KARYPIS, G.; HAN, E.-H.; KUMAR, V. Chameleon: Hierarchical clustering using dynamic modeling. **Computer**, IEEE, v. 32, n. 8, p. 68–75, 1999. Citado na página 9.
- KAUFMAN, L.; ROUSSEEUW, P. J. Partitioning around medoids (program pam). **Finding groups in data: an introduction to cluster analysis**, Wiley Online Library, p. 68–125, 1990. Citado na página 9.
- KEOGH, E.; CHAKRABARTI, K.; PAZZANI, M.; MEHROTRA, S. Locally adaptive dimensionality reduction for indexing large time series databases. **ACM SIGMOD Record**, ACM, v. 30, n. 2, p. 151–162, 2001. Citado 2 vezes nas páginas 28 e 29.
- KEOGH, E.; LIN, J.; FU, A. Hot sax: Efficiently finding the most unusual time series subsequence. In: IEEE. **Proceedings of the 5th IEEE International Conference on Data Mining**. Tokyo, Japana, 2005. (ICDE '05), p. 8–pp. Citado na página 29.
- KEOGH, E. J.; PAZZANI, M. J. Scaling up dynamic time warping for datamining applications. In: **Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. Boston, USA: ACM, 2000. (KDD '00), p. 285–289. Citado 2 vezes nas páginas 29 e 31.
- KLEMA, V. C.; LAUB, A. J. The singular value decomposition: Its computation and some applications. **IEEE Transactions on Automatic Control**, IEEE, v. 25, n. 2, p. 164–176, 1980. Citado na página 29.
- KONDOR, R. I.; LAFFERTY, J. Diffusion kernels on graphs and other discrete structures. In: **Proceedings of the 19th International Conference on Machine Learning**. Sydney, Australia: [s.n.], 2002. (ICML '02), p. 315–322. Citado na página 24.
- LEVIN, A.; VIOLA, P.; FREUND, Y. Unsupervised improvement of visual detectors using cotraining. In: **Proceedings of the 9th IEEE International Conference on Computer Vision, 2003**. Nice, France: IEEE, 2003. p. 626–633. Citado na página 21.
- LI, S.; WANG, Z.; ZHOU, G.; LEE, S. Y. M. Semi-supervised learning for imbalanced sentiment classification. In: **Proceedings of the 22nd International Joint Conference on Artificial Intelligence**. Barcelona, Spain: [s.n.], 2011. (IJCAI '11), p. 1826. Citado na página 75.

- LIU, W.; HE, J.; CHANG, S.-F. Large graph construction for scalable semi-supervised learning. In: **Proceedings of the 27th International Conference on Machine Learning**. Haifa, Israel: [s.n.], 2010. (ICML'10), p. 679–686. Citado na página 24.
- LUCAS, A. A.; SCHULER, C. A. B. Análise do ndvi/noaa em cana-de-açúcar e mata atlântica no litoral norte de pernambuco, brasil. **Revista Brasileira de Engenharia Agrícola e Ambiental**, v. 11, p. 607–614, 2007. Citado na página 4.
- MAIMON, O.; ROKACH, L. **Data mining and knowledge discovery handbook**. [S.l.]: Springer, 2010. v. 2. Citado 6 vezes nas páginas 7, 9, 10, 11, 27 e 29.
- MATTHEWS, B. Comparison of the predicted and observed secondary structure of {T4} phage lysozyme. **Biochimica et Biophysica Acta (BBA) - Protein Structure**, v. 405, n. 2, p. 442 – 451, 1975. ISSN 0005-2795. Citado na página 63.
- MITCHELL, T. M. **Machine Learning**. 1. ed. New York, NY, USA: McGraw-Hill, Inc., 1997. Citado na página 11.
- MITSA, T. **Temporal data mining**. [S.l.]: CRC Press, 2010. Citado 2 vezes nas páginas 28 e 40.
- MORETTIN, P. A.; TOLOI, C. M. C. **Análise de séries temporais**. São Paulo: Editora Edgard Blucher Ltda, 2006. Citado na página 27.
- NG, A. Y.; JORDAN, M. I.; WEISS, Y. *et al.* On spectral clustering: Analysis and an algorithm. **Advances in Neural Information Processing Systems 15 (NIPS'02)**, The MIT Press, Vancouver, Canada, v. 2, p. 849–856, 2002. Citado na página 51.
- NG, R. T.; HAN, J. Clarans: A method for clustering objects for spatial data mining. **IEEE Transactions on Knowledge and Data Engineering**, IEEE, v. 14, n. 5, p. 1003–1016, 2002. Citado na página 42.
- NGUYEN, M. N.; LI, X.-L.; NG, S.-K. Positive unlabeled learning for time series classification. In: **Proceedings of the 22nd International Joint Conference on Artificial Intelligence**. Barcelona, Spain: AAAI Press, 2011. (IJCAI '11), p. 1421–1426. Citado na página 20.
- NIGAM, K.; GHANI, R. Analyzing the effectiveness and applicability of co-training. In: **ACM. Proceedings of the ninth international conference on Information and knowledge management**. [S.l.], 2000. p. 86–93. Citado na página 21.
- NIGAM, K.; MCCALLUM, A. K.; THRUN, S.; MITCHELL, T. Text classification from labeled and unlabeled documents using em. **Machine learning**, Springer, v. 39, n. 2-3, p. 103–134, 2000. Citado 2 vezes nas páginas 16 e 17.
- OATES, T.; FIROIU, L.; COHEN, P. R. Clustering time series with hidden markov models and dynamic time warping. In: **CITeseer. Proceedings of the IJCAI-99 workshop on neural, symbolic and reinforcement learning methods for sequence learning**. [S.l.], 1999. p. 17–21. Citado na página 32.
- PRICE, J. C. Estimating leaf area index from satellite data. **IEEE Transactions on Geoscience and Remote Sensing**, IEEE, v. 31, n. 3, p. 727–734, 1993. Citado na página 58.
- RABINER, L. R.; JUANG, B.-H. An introduction to hidden markov models. **ASSP Magazine, IEEE**, IEEE, v. 3, n. 1, p. 4–16, 1986. Citado na página 29.

- RAKTHANMANON, T.; CAMPANA, B.; MUEEN, A.; BATISTA, G.; WESTOVER, B.; ZHU, Q.; ZAKARIA, J.; KEOGH, E. Searching and mining trillions of time series subsequences under dynamic time warping. In: **ACM. Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. Beijin, China, 2012. p. 262–270. Citado na página 32.
- RAO, D.; RAVICHANDRAN, D. Semi-supervised polarity lexicon induction. In: **ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics**. [S.l.], 2009. p. 675–682. Citado na página 24.
- RATANAMAHATANA, C.; KEOGH, E. Making Time-Series Classification More Accurate Using Learned Constraints. In: **Proceedings of the 2004 SIAM International Conference on Data Mining**. Lake Buena Vista, USA: [s.n.], 2004. (SDM '04), p. 11–22. Citado na página 32.
- RATANAMAHATANA, C. A.; KEOGH, E. Everything you know about dynamic time warping is wrong. In: **Third Workshop on Mining Temporal and Sequential Data**. Seattle, USA: ACM, 2004. Citado na página 32.
- RATANAMAHATANA, C. A.; WANICHSAN, D. Stopping criterion selection for efficient semi-supervised time series classification. In: **Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing**. [S.l.]: Springer, 2008. p. 1–14. Citado na página 20.
- RATSABY, J.; VENKATESH, S. S. Learning from a mixture of labeled and unlabeled examples with parametric side information. In: **Proceedings of the eighth annual conference on Computational learning theory**. [S.l.]: ACM, 1995. p. 412–417. Citado na página 17.
- REN, Z.; YUAN, J.; ZHANG, Z. Robust hand gesture recognition based on finger-earth mover's distance with a commodity depth camera. In: **Proceedings of the 19th ACM international conference on Multimedia**. Scottsdale, USA: ACM, 2011. (MM '11), p. 1093–1096. Citado na página 27.
- RISTAD, E. S.; YIANILOS, P. N. Learning string-edit distance. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, IEEE, v. 20, n. 5, p. 522–532, 1998. Citado na página 33.
- ROMANI, L. A. S.; AMARAL, B. F.; GONÇALVES, R. R. V.; ZULLO, J.; SOUSA, E. P. M. Aplicação de técnicas de classificação semissupervisionada para análise de séries multitemporais de imagens de satélite. In: **Anais do XVI Simpósio Brasileiro de Sensoriamento Remoto**. Foz do Iguaçu – PR: [s.n.], 2013. p. 1750–1757. Citado na página 81.
- ROUSE, J. W.; HAAS, R. H.; SCHELL, J. A.; DEERING, D. W. Monitoring vegetation systems in the great plains with erts. In: FREDEN, S. C.; BECKER, M. A. (Ed.). **Third ERTS Symposium**. [S.l.]: NASA, 1973. v. 1, p. 309–317. Citado 2 vezes nas páginas 4 e 58.
- ROYSTON, P. *et al.* Multiple imputation of missing values. **Stata journal**, STATA PRESS, v. 4, n. 3, p. 227–41, 2004. Citado na página 39.
- SALVADOR, S.; CHAN, P. Fastdtw: Toward accurate dynamic time warping in linear time and space. In: **KDD workshop on mining temporal and sequential data**. Seattle, USA: ACM, 2004. Citado na página 32.

SCRIVANI, R.; AMARAL, B. F.; GONÇALVES, R. R. V.; SOUSA, E. P. M.; ZULLO, J.; ROMANI, L. A. S. Identificação de mudança de uso da terra usando técnicas de agrupamento de séries temporais de imagens de satélite. In: **Anais do 5o Simpósio de Geotecnologias na Pantanal**. Campo Grande – MS: [s.n.], 2014. (Geopantanal '14), p. 554–563. Citado na página 81.

SHENSA, M. J. The discrete wavelet transform: wedding the a trous and mallat algorithms. **IEEE Transactions on Signal Processing**, IEEE, v. 40, n. 10, p. 2464–2482, 1992. Citado na página 29.

SHIEH, J.; KEOGH, E. isax: indexing and mining terabyte sized time series. In: **Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining**. Las Vegas, USA: ACM, 2008. p. 623–631. Citado 2 vezes nas páginas 28 e 29.

SHOKOOHI-YEKTA, M.; WANG, J.; KEOGH, E. On the non-trivial generalization of dynamic time warping to the multi-dimensional case. In: **Proceedings of the 2015 SIAM International Conference on Data Mining**. Vancouver, Canada: SIAM, 2015. (SDM '15). Citado 3 vezes nas páginas 27, 32 e 40.

SMOLA, A. J.; KONDOR, R. Kernels and regularization on graphs. In: **Learning theory and kernel machines**. [S.l.]: Springer, 2003. p. 144–158. Citado na página 24.

SZLAM, A. D.; MAGGIONI, M.; COIFMAN, R. R. Regularization on graphs with function-adapted diffusion processes. **The Journal of Machine Learning Research**, JMLR. org, v. 9, p. 1711–1739, 2008. Citado na página 24.

VAPNIK, V. N.; VAPNIK, V. **Statistical learning theory**. [S.l.]: Wiley New York, 1998. v. 1. Citado na página 11.

VLACHOS, M.; HADJIELEFTHERIOU, M.; GUNOPULOS, D.; KEOGH, E. Indexing multi-dimensional time-series with support for multiple distance measures. In: **ACM. Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining**. [S.l.], 2003. p. 216–225. Citado na página 33.

VLACHOS, M.; KOLLIOS, G.; GUNOPULOS, D. Discovering similar multidimensional trajectories. In: **Proceedings of the 18th International Conference on Data Engineering**. [S.l.]: IEEE, 2002. (ICDE '02), p. 673–684. Citado na página 33.

WAN, X. Co-training for cross-lingual sentiment classification. In: **ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1**. [S.l.], 2009. p. 235–243. Citado na página 21.

WANG, F.; ZHANG, C. Label propagation through linear neighborhoods. **IEEE Transactions on Knowledge and Data Engineering**, IEEE, v. 20, n. 1, p. 55–67, 2008. Citado 3 vezes nas páginas 24, 25 e 68.

WANG, J.; RICH, P. M.; PRICE, K. P. Temporal responses of ndvi to precipitation and temperature in the central great plains, usa. **International Journal of Remote Sensing**, v. 24, n. 11, p. 2345–2364, 2003. Citado na página 4.

WANG, W.; YANG, J.; MUNTZ, R. *et al.* Sting: A statistical information grid approach to spatial data mining. In: **Proceedings of the 23rd International Conference on Very Large Data Bases**. Athens, Greece: Morgan Kaufmann, 1997. (VLDB '97), p. 186–195. Citado na página 9.

WEI, L.; KEOGH, E. Semi-supervised time series classification. In: **Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. Philadelphia, USA: ACM, 2006. (KDD '06), p. 748–753. Citado 2 vezes nas páginas 20 e 27.

XI, X.; KEOGH, E.; SHELTON, C.; WEI, L.; RATANAMAHATANA, C. A. Fast time series classification using numerosity reduction. In: **Proceedings of the 23rd International Conference on Machine Learning**. Pittsburgh, USA: ACM, 2006. (ICML '06), p. 1033–1040. Citado na página 32.

ZENG, H.-J.; WANG, X.-H.; CHEN, Z.; LU, H.; MA, W.-Y. Cbc: Clustering based text classification requiring minimal labeled data. In: **Proceedings of the Third IEEE International Conference on Data Mining**. Melbourne, USA: IEEE Computer Society, 2003. (ICDM '03), p. 443–450. Citado na página 18.

ZHANG, T.; RAMAKRISHNAN, R.; LIVNY, M. Birch: an efficient data clustering method for very large databases. In: **ACM SIGMOD Record**. [S.l.]: ACM, 1996. (SIGMOD Rec. '96, 2), p. 103–114. Citado na página 9.

ZHOU, D.; BOUSQUET, O.; LAL, T. N.; WESTON, J.; SCHÖLKOPF, B. Learning with local and global consistency. **Advances in Neural Information Processing Systems 17 (NIPS'04)**, The MIT Press, Vancouver, Canada, v. 16, n. 16, p. 321–328, 2004. Citado 3 vezes nas páginas 24, 51 e 68.

ZHOU, Z.-H.; LI, M. Tri-training: Exploiting unlabeled data using three classifiers. **IEEE Transactions on Knowledge and Data Engineering**, IEEE, v. 17, n. 11, p. 1529–1541, 2005. Citado na página 21.

ZHU, Q.; BATISTA, G.; RAKTHANMANON, T.; KEOGH, E. A novel approximation to dynamic time warping allows anytime clustering of massive time series datasets. In: **Proceedings of the 2012 SIAM International Conference on Data Mining (SDM'12)**. Anaheim, USA: SIAM, 2012. p. 999–1010. Citado na página 32.

ZHU, X. **Semi-Supervised Learning Literature Survey**. [S.l.], 2005. Citado 2 vezes nas páginas 16 e 17.

ZHU, X.; GHAMRANI, Z.; LAFFERTY, J. *et al.* Semi-supervised learning using gaussian fields and harmonic functions. In: **Proceedings of the 20th International Conference on Machine Learning (ICML'03)**. Washington D.C., USA: AAAI Press, 2003. v. 3, p. 912–919. Citado na página 24.

ZHU, X.; GOLDBERG, A. B.; BRACHMAN, R.; DIETTERICH, T. **Introduction to Semi-Supervised Learning**. [S.l.]: Morgan and Claypool Publishers, 2009. Citado 10 vezes nas páginas 2, 9, 12, 13, 16, 17, 18, 20, 21 e 23.

ZHU, X.; KANDOLA, J.; GHAMRANI, Z.; LAFFERTY, J. D. Nonparametric transforms of graph kernels for semi-supervised learning. In: **Advances in Neural Information Processing Systems 17 (NIPS'04)**. Vancouver, Canada: The MIT Press, 2004. p. 1641–1648. Citado na página 24.

ZHU, X.; KANDOLA, J.; LAFFERTY, J.; GHAHRAMANI, Z. Graph kernels by spectral transforms. In: CHAPELLE, O.; SCHÖLKOPF, B.; ZIEN, A. (Ed.). **Semi-supervised learning**. Cambridge, USA: MIT Press Cambridge, 2006. p. 277–292. Citado na página [75](#).