Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Visualização de dados multi-dimensionais por meio de Mapas de Difusão

Lucas Mota Ribeiro

Monografia apresentada como requisito parcial
para conclusão do Bacharelado em Ciência da Computação

Orientador
Prof. Dr. Vinicius Ruela Pereira Borges

Brasília
2019

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Visualização de dados multi-dimensionais por meio de Mapas de Difusão

Lucas Mota Ribeiro

Monografia apresentada como requisito parcial
para conclusão do Bacharelado em Ciência da Computação

Prof. Dr. Vinicius Ruela Pereira Borges (Orientador)
CIC/UnB

Prof. Dr. Guilherme Novaes Ramos
CIC/UnB

Prof. Dr. Donald Matthew Pianto
EST/UnB

Prof. Dr. Edison Ishikawa
Coordenador do Bacharelado em Ciência da Computação

Brasília, 18 de Janeiro de 2019

Dedicatória

Dedico esse trabalho a meus pais e irmã pelo constante apoio ao meu crescimento e decisões, aos amigos pelos momentos compartilhados sendo eles bons ou ruins. E por fim, à toda humanidade.

Agradecimentos

Agradeço ao Dr. Geovan Tavares pelo grande auxílio nos direcionamentos do trabalho e ao meu orientador, Dr. Vinícius Borges, pelo constante suporte e apoio durante todo o desenvolvimento dessa monografia. Também agradeço todos os professores, amigos e familiares pelo suporte de cada um deles que acrescentou no meu crescimento pessoal.

Resumo

A visualização de dados multi-dimensionais é um tópico relevante na área de Visualização da Informação, nos quais alguns dos desafios mais recorrentes ocorre devido ao crescente volume de dados a ser tratado. Além disso, a dimensionalidade desses dados têm aumentado consideravelmente. Visualizações baseadas em projeções multi-dimensionais se tornaram alternativas importantes às técnicas tradicionais, pois conseguem gerar efetivamente representações gráficas 2D ou 3D de dados definidos em um espaço de alta dimensionalidade. Uma projeção multi-dimensional tem a complexa tarefa de mapear tais dados para um espaço de baixa dimensionalidade preservando seus padrões, suas estruturas e as relações de similaridade. Devido à grande variedade de tipos de dados (imagens, texto, geográficos e etc) e as diferentes maneiras em que podem ser caracterizados, a tarefa de selecionar uma técnica para gerar visualizações representativas dos dados não é trivial. Desta maneira, esta monografia propõe estudar o emprego do algoritmo Mapas de Difusão como uma técnica de visualização baseada em projeção multi-dimensional. Na literatura, tal algoritmo originalmente proposto como uma técnica de redução de dimensionalidade, foi empregado com sucesso em tarefas de segmentação e agrupamento de dados devido à sua formulação probabilística e natureza espectral. Os experimentos foram conduzidos visando avaliar a precisão com que as relações de similaridade dos dados originais são preservadas no espaço de baixa dimensão, como também a qualidade das representações gráficas obtidas. Os resultados foram comparados com outras técnicas clássicas de projeção multi-dimensional na literatura (*MultiDimensional Scaling*, *Isometric Feature Mapping* e *Locally Linear Embedding Standard*) e mostraram que dado uma configuração de parâmetros correta, a técnica Mapas de Difusão é apropriada para realizar a visualização de dados multi-dimensionais.

Palavras-chave: Redução de dimensionalidade, Mapas de Difusão

Abstract

Multidimensional visualization is a relevant topic in information visualization which some of the biggest recurring callenges are caused by the increasing amount of data available. Another contributing factor is the increasing dimensionality of the data. Projection-based visualizations became a pertinent choice over the traditional projetions because they can generate 2D or 3D graphical representations of high dimensional data. A multidimensional projection performs a complex task of mapping high dimensional data into a lower dimensional space preserving its original patterns, structures and similarity relations. Given the high number of types of data (text, images, geographic, etc.) and the different strategies for its characterization, the selection of a specific technique for data visualization is considered a non-trivial task. This project proposes to study and evaluate the use of Diffusion Maps as a projection-based visualization technique. Originally proposed as a dimensionality reduction technique, the Diffusion Maps algorithm has been successfully applied in previous works to segment and clusterize data due to its probabilistic and spectral nature. Experiments were conducted aiming to evaluate the proposed visualization technique concerning the preservation of the similarity relations of the original data in the respective lower dimensional space and also the quality of the layouts. The results were compared to other classic techniques (MultiDimensional Scaling, Isometric Feature Mapping e Locally Linear Embedding Standard) which showed that Diffusion Maps is appropriate to be used as a visualization technique for multidimensional data, when setting up appropriate parameters.

Keywords: Dimension Reduction, Diffusion Mapss

Sumário

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

Analisar dados é uma tarefa ampla com um grande número de abordagens e técnicas que podem ser geralmente definidas como uma sequência de passos, que vão desde a coleta até a avaliação dos resultados. Para cada uma dessas etapas, os resultados são repassados para as etapas seguintes onde a avaliação da qualidade dos dados que transitam entre cada etapa são, em geral, feitos por meio de uma visualização de dados. Essa visualização traz uma percepção diferenciada de como os dados se relacionam.

A visualização de dados tem o objetivo de gerar uma representação gráfica deles. Ela busca facilitar a identificação de padrões e estruturas [7] presentes nos dados em sua forma gráfica com o auxílio de estruturas visuais. Desde cartas cartográficas até os grafos de relações em redes sociais, a visualização de dados é uma abordagem que tem sido empregada com o objetivo de apresentar estruturas implícitas em dados complexos de uma maneira simples e compreensível de acordo com o sistema perceptual humano [8].

Há uma grande gama de técnicas de visualização de dados, na qual cada uma tem um foco diferente. As técnicas variam desde gráficos de dispersão até um gráfico de barras paralelas. Essas técnicas mais tradicionais são recorrentemente utilizadas em vários tipos de análises de dados tanto direta quanto indiretamente.

As técnicas tradicionais citadas acima para a visualização de dados podem apresentar problemas na geração das representações gráficas à medida que a dimensionalidade dos dados aumenta [9]. Devido ao espaço visual ser limitado, informações correspondentes aos padrões relevantes e estruturas dos dados podem ser perdidas, afetando a performance das visualizações. Ademais, o uso de dados com alta dimensionalidade pode afetar o desempenho de algoritmos de aprendizado de máquina e de análise de dados, conforme relatado no estudo sobre o processo conhecido *a maldição da dimensionalidade* [10], de Richard E. Bellman. Ele estudou problemas de otimização em sistemas de grande dimensionalidade e identificou alguns fatores inerentes aos dados multi-dimensionais, tais como:

- Número de combinações a serem feitas aumentam exponencialmente em relação ao número de dimensões;
- Dimensões ficam irrelevantes em comparação com outras;
- Distâncias euclideanas se tornam irrelevantes.

Na última década, a comunidade de Visualização da Informação propôs diversas técnicas de visualização para dados multi-dimensionais [11]. Uma categoria de técnicas de visualizações que se tornou popular foram as visualizações baseadas no posicionamento de pontos no espaço [12]. Basicamente, essas técnicas produzem uma representação gráfica (*layout*) ao associar elementos visuais (por exemplo, pontos, círculos, objetos geométricos etc) às instâncias de dados em um espaço visual de duas ou três dimensões. Como resultado, o *layout* apresenta os pontos distribuídos nesse espaço (sem indicação de eixos ou atributos), em que a proximidade desses pontos indicam alguma relação de similaridade das respectivas instâncias de dados no espaço original. De acordo com Paulovich et al. [12], visualizações baseadas no posicionamento de pontos no espaço podem ser categorizadas em projeções multi-dimensionais [13] ou árvores de similaridade [14].

Devido às limitações das visualizações tradicionais, o uso e o desenvolvimento de projeções multi-dimensionais se popularizaram na última década nas tarefas de análise e visualização de dados [15]. As projeções multi-dimensionais realizam um mapeamento de dados originalmente definidos em um espaço de alta dimensionalmente para um espaço de dimensão reduzida preservando ao máximo seus padrões e estruturas implícitas, isto é, minimizando a perda de informação. As técnicas de visualização baseadas em projeções multi-dimensionais produzem uma representação gráfica a partir do espaço reduzido, de forma a retratar no espaço visual as relações de similaridade entre as instâncias de dados e a formação de grupos que podem expressar estruturas interessantes.

Uma grande gama de técnicas de projeções multi-dimensionais foram propostas nos últimos anos, sendo divididas em três grupos, seguindo a taxonomia definida por Paulovich [3]: *multidimensional scaling*, *force-directed placement* e redução de dimensionalidade. Os focos desta pesquisa são as projeções multi-dimensionais baseadas em redução de dimensionalidade. A técnica *Principal Component Analysis* (PCA) [16] é uma projeção multi-dimensional de grande relevância na literatura que realiza uma transformação ortogonal nos dados de forma a representá-los por componentes que explicam a maior variância dos dados. Devido à existência de várias projeções multi-dimensionais [17], a escolha de uma técnica é uma tarefa difícil que não depende apenas do problema a ser tratado, mas também dos tipos de dados considerados.

A técnica de redução de dimensionalidade conhecida como Mapas de Difusão (*Diffusion Maps*) [18] estará em foco nesse projeto por ter se mostrado robusta em mapear as

proximidades dos dados em relação aos parâmetros de mudança entre eles [6, 19]. Embora a formulação da técnica Mapas de Difusão considere as estruturas geométricas lineares e não-lineares para realizar a redução de dimensionalidade, ainda não foram conduzidos estudos detalhados relativos ao seu emprego como uma técnica de visualização multi-dimensional. Além disso, essa técnica não foi aplicada de forma categórica na literatura para propósitos de análise e visualização de conjuntos de dados reais (imagens e texto), restringindo-se muitas vezes a bases de dados artificiais.

O presente trabalho tem como objetivo investigar se a técnica de Mapas de Difusão pode ser empregada como uma técnica de visualização multi-dimensional. Nesse estudo, serão considerados conjuntos de dados provenientes de medições, imagens e textos para analisar os *layouts* gerados pelos Mapas de Difusão. Além disso, os *layouts* gerados pela técnica Mapas de Difusão serão qualitativamente comparados com os *layouts* obtidos por visualizações baseadas em outros algoritmos de redução de dimensionalidade na literatura. Para esse propósito, serão consideradas duas métricas que avaliam a preservação das estruturas locais e globais dos dados no espaço original em relação ao espaço reduzido.

Considerando que o algoritmo Mapas de Difusão já foi empregado na literatura em tarefas de agrupamento de dados e segmentação [19, 20], a principal contribuição desse trabalho é um estudo de um método de visualização multi-dimensional baseado no posicionamento de pontos que mapeia os dados para um espaço de dimensões reduzidas por meio do algoritmo Mapas de Difusão. Nesse sentido, investiga-se a capacidade desse método em gerar representações gráficas intuitivas que expressem as relações de similaridade e estruturas relevantes das instâncias de dados. Espera-se que essa proposta seja uma alternativa viável em tarefas de análise visual de dados, uma vez que a escolha de uma técnica de visualização depende do tipo de dados em questão.

1.1 Estrutura da Monografia

Esta monografia está organizada da seguinte maneira: O Capítulo 2 se refere à fundamentação teórica, assim nela é discutida a definição dos dados a serem tratados, visualização de dados, projeções multi-dimensionais e as métricas para avaliação da qualidade de visualizações; o Capítulo 3 detalha o método Mapas de Difusão e seu uso em outros trabalhos relacionados na literatura, assim descrevendo a metodologia envolvida para realizar sua aplicação para a visualização de dados. O Capítulo 4 mostra os resultados obtidos aplicando o método de visualização proposto sobre uma variedade de conjuntos de dados. Por fim o Capítulo 5 descreve as considerações finais, as possibilidades de trabalhos futuros e aprimoramentos que podem ser feitos nessa pesquisa.

Capítulo 2

Fundamentos

2.1 Dados

Dados são a forma como a informação é registrada e quais regras ela obedece. Cada variedade de dados têm suas peculiaridades e formas específicas de serem analisadas, uma vez que podem ser provenientes de fontes distintas como textos, imagens, sensores etc. Conhecer tais técnicas e abordagens é de grande importância para escolher os algoritmos de aprendizado de máquina e os processos de mineração de dados mais apropriados para extrair informações implícitas.

Um conjunto de dados $\mathbf{X}$ é definido por um conjunto de instâncias $\{\mathbf{x_1}, ..., \mathbf{x_N}\}$ em que cada instância $\mathbf{x_i} = \{x_{i,1}, ...x_{i,m}\}$ é descrita por m atributos que definem um valor para uma medição ou característica. Essa forma dos dados é abstraída como uma tabela onde as linhas são instâncias e as colunas são atributos. A Tabela 2.1 ilustra um conjunto de dados contendo três instâncias, cada uma em uma linha, que são caracterizadas por três atributos ("Altura", "Peso" e "Idade"). A dimensionalidade dos dados é associada ao seu número de atributos, assim uma instância com m atributos têm m dimensões.

Tabela 2.1: Modelo tabular.

Altura (m)	**Peso (Kg)**	**Idade**
1.51	55	21
1.72	81	31
1.66	60	27

Uma tarefa muito importante no aprendizado de máquina se refere ao cálculo da similaridade entre instâncias de dados, que podem ser geralmente comparadas por meio de suas representações vetoriais no espaço euclideano. Assim, pode-se calcular a distância entre os pontos associados com essas instâncias utilizando uma métrica. Para medir a

similaridade (ou dissimilaridade) entre instâncias de dados, pode-se utilizar as métricas baseadas em funções de distância, como as distâncias euclideana e cosseno [21].

Considerando os vetores $\mathbf{x} = \{x_1, x_2, \ldots, x_m\}$ e $\mathbf{y} = \{y_1, y_2, \ldots, y_m\}$, a distância euclideana d_E é definida como a norma do vetor diferença como mostra a Eq. 2.1:

$$d_E(\mathbf{x}, \mathbf{y}) = \left(\sum_{i=1}^{m} |x_i - y_i|^2 \right)^{\frac{1}{2}}. \tag{2.1}$$

Já a distância de cosseno d_C é o cosseno do ângulo entre os dois vetores, como mostra a Eq. 2.2:

$$d_C(\mathbf{x}, \mathbf{y}) = \left(\frac{\mathbf{x} \cdot \mathbf{y}}{\|\mathbf{x}\|\|\mathbf{y}\|} \right). \tag{2.2}$$

2.1.1 Textos

Textos digitais representam uma grande parte da massa de dados presente no mundo devido ao seu baixo custo e por serem empregados em várias formas de comunicação, como é o caso e-mail, canais mensagem instantâneas, redes sociais e portais de notícias, entre outros exemplos. Um texto digital é definido por um conjunto de caracteres no qual podem variar de um a quatro bytes em tamanho considerando a formatação UTF-8. Todos os caracteres da língua latina têm, no máximo, dois bytes. Com isso, o processamento de texto pode ser feito em uma massa de dados grande com certa facilidade devido ao seu tamanho digital. Textos são de fácil acesso e por isso têm sido bastante empregados no campo da análise de dados.

Para a análise textual, muitas vezes é necessário aplicar algum pré-processamento para transformar o texto em uma estrutura mais apropriada para processamento computacional. O pré-processamento retira informações que são irrelevantes, redundantes e que causam ambiguidade em textos [22]. O pré-processamento de texto pode compreender a aplicação de algumas técnicas, como:

- Tokenização: é o conceito de separar o texto em unidades para serem analisadas como palavras separadas por espaço. Muitas vezes essa separação depende da língua sendo avaliada, pois alguns símbolos também podem ser considerados separadores.
- Remoção de *stopwords*: consiste em remover palavras de baixa influência semântica no texto como artigos, conectivos e pontuação. Um exemplo são os termos 'o', 'a', 'e', 'de', 'da' e 'do'.
- *Stemming*: tem o objetivo de reduzir as palavras para a sua forma raiz. Isso é feito removendo sua conjugação, prefixos e sufixos, e outras formatações a título de que

estas palavras sejam identificadas igualmente. Por exemplo as palavras “avião”, “aviões”, “aviação” são reduzidas para ’avião’. Muitas vezes, a técnica de *stemming* reduz palavras com significados diferentes em seus respectivos contextos para o mesmo significado.

- *Lematização:* utiliza um vocabulário e a análise morfológica da palavra para deflexioná-la e assim encontrar o lema da palavra. Essa técnica é complexa pois requer um conhecimento linguístico profundo do idioma sendo analisado. No entanto, pode ser útil para minimizar alguns problemas causados pelo *stemming*.

Outras técnicas de pré-processamento de texto podem incluir lowercase, pontuação e urls, mas variam conforme a origem do texto, isto é, se foram obtidos de redes sociais, coleção de documentos etc.

Após o pré-processamento dos dados, deve-se extrair as características do texto com o intuito de obter uma representação apropriada para que os textos sejam processados pelas técnicas de aprendizado de máquina. Um dos algoritmos mais relevantes da análise textual é o *Bag of Words* (BoW), que consiste na quantificação da frequência das palavras de um texto. Considera-se que cada palavra é representada por uma posição em um vetor numérico, inicialmente, zerado. Percorrendo o texto, a cada palavra encontrada se incrementa o valor no índice do vetor associada a essa palavra. Desta maneira, dois textos com seus respectivos vetores de frequência de palavras podem ser comparados calculando-se a distância entre seus vetores normalizados. Tal processo pode refletir o grau de similaridade dos textos associados, como também verificar se referem-se ao mesmo assunto.

O resultado da aplicação da técnica *Bag of Words* em uma coleção de textos digitais pode ser apresentado utilizando a técnica de WordCloud [23]. A representação gráfica gerada mostra cada palavra do vetor em tamanho proporcional a sua frequência no texto. A Figura 2.1 ilustra a geração de uma WordCloud a partir de uma notícia sobre um incêndio no estado da Califórnia, EUA [24]. Para criar a imagem, foi gerado um vetor de frequências após remover as *stopwords*. A partir desse vetor, uma imagem é criada utilizando uma aplicação *Python* [1] para organizar as palavras em um dado espaço gráfico com seus respectivos tamanhos e assim gerar a visualização baseada em WordCloud.

Uma técnica relacionada com o *Bag of Words*, mas que considera a importância global de cada palavra em um conjunto de documentos, é a *Term Frequency-Inverse Document Frequency* (TF-IDF), que quantifica o termo t no documento d e no corpus de documentos D identificando assim sua relevância local e global. A obtenção do vetor TF-IDF de uma coleção de documentos compreende algumas etapas. Inicialmente, deve-se encontrar a

[1] Código baseado na implementação `https://github.com/amueller/word_cloud`

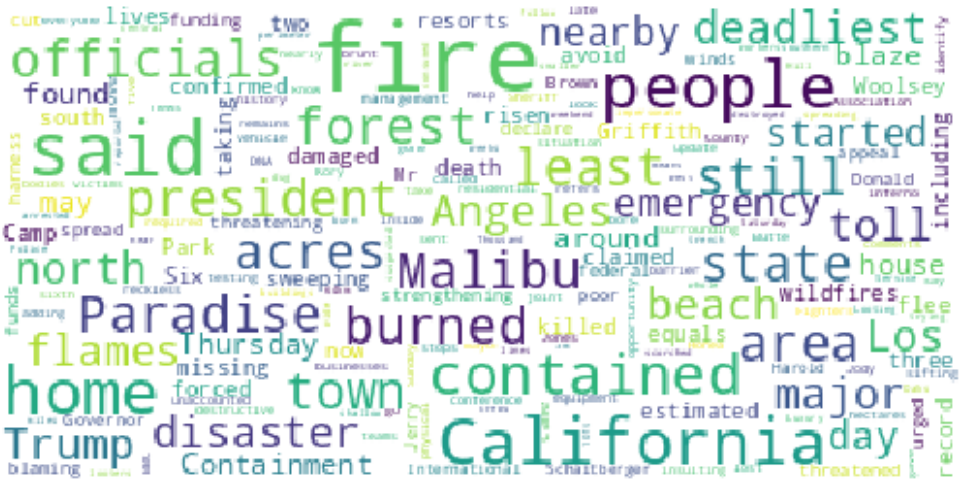


Figura 2.1: WordCloud de uma notícia sobre um incêndio no estado da Califórnia, EUA.

frequência de um termo t no documento d, como ocorre na técnica *Bag of Words* conforme mostra a Eq. 2.3:

$$tf(t,d) = f_{t,d}, \tag{2.3}$$

Em seguida, calcula-se o logaritmo da diferença do número de documentos (N) pelo número de vezes que o termo t aparece no corpus D, de acordo com a Eq. 2.4:

$$idf(t,D) = \log \frac{N}{|d \in D : t \in d|}, N = |D|. \tag{2.4}$$

Finalmente, a Eq. 2.5 computa os resultados obtidos nas equações anteriores como um produto das duas métricas:

$$tfidf(t,d,D) = tf(t,d) \cdot idf(t,d), \tag{2.5}$$

2.1.2 Imagens

Uma imagem digital é a representação numérica discreta de uma imagem analógica. Esse processo de conversão é formado por dois subprocessos conhecidos como *sampling* e quantização. O primeiro indica o número de subdivisões que será feita no sinal, assim, ditando a resolução da imagem digital. O segundo é o número de bits presentes para representar uma cor, ou seja, a profundidade cromática. Uma imagem digital é constituída por *pixels*, que são posições que armazenam um valor de intensidade dependendo do tipo de imagem.

Em geral, existem imagens baseadas em níveis de cinza ou em cores. Nas imagens em níveis de cinza, os pixels armazenam valores de intensidade por meio de uma única

banda. Já nas imagens coloridas, utilizam-se três canais de cores para descrever uma cor para cada pixel. Um modelo de cor tradicional em dispositivos de exibição é o *Red Green Blue* (RGB), em que cada canal é responsável pelas cores primárias Vermelho, Verde e Azul, respectivamente.

Para analisar uma imagem digital de forma original, é necessário avaliar cada *pixel* dessa imagem individualmente. Uma imagem de 500×500 *pixels* com 3 canais de cores de 8 bits cada, tem um tamanho de 750 Kb. A análise da imagem em sua forma original não é viável, de forma que se faz necessário extrair informações úteis de formas pontual, assim como é feito para os textos.

Uma das formas de extrair características relevantes para identificar uma imagem (ou alguma de suas regiões) é utilizando descritores de imagens [25]. Um descritor é composto por um algoritmo que caracteriza uma imagem ao calcular valores numéricos de suas propriedades visuais, produzindo um vetor de características. Desta maneira, um conjunto de imagens pode ser definido em um espaço vetorial (denominado espaço de características) em uma dimensão reduzida ao invés de utilizar a imagem original em sua forma original, isto é, como uma matriz de *pixels*.

Uma abordagem bem conhecida para a descrição de imagens se baseia na detecção de pontos locais de interesse, como os tradicionais *Scale Invariant Feature Transform* (SIFT) [26] e *Speeded-up Robust Features* (SURF) [27]. Essa abordagem detecta os pontos de alta relevância na imagem como locais de alto contraste, brilho ou forma bem definida para caracterizar uma vizinhança de *pixels* associada. Os pontos de interesse detectados em uma imagem podem ser comparados com os pontos detectados em outra imagem, com o objetivo de calcular a similaridade entre elas. A Figura 2.2 ilustra esse processo de correspondência de pontos locais de interesse, obtidos a partir de duas imagens, caracterizadas utilizando o descritor ORB [1].

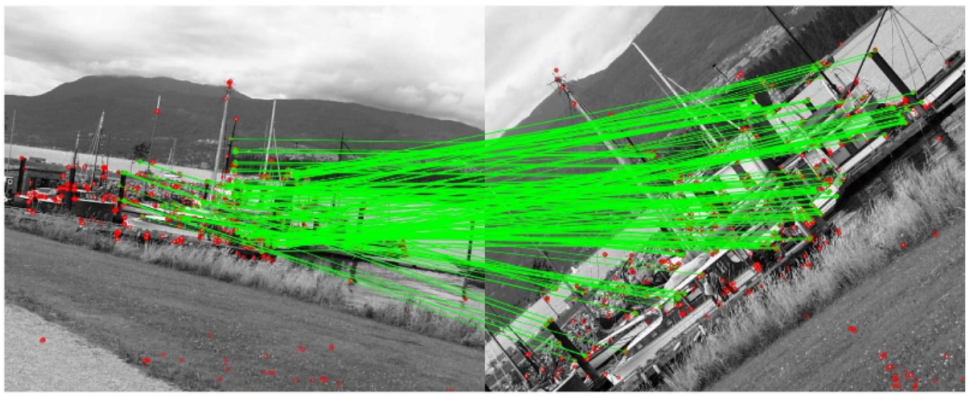


Figura 2.2: Utilização de descritores ORB [1] para associação de pontos locais de interesse em imagens.

Outra estratégia consiste em segmentar a imagem em suas regiões constituintes e assim obter um descritor de características para cada região. Para isso, deve-se utilizar um algoritmo para processar as imagens e identificar apropriadamente as regiões e objetos de interesse da imagem. Em seguida, deve-se utilizar o descritor de características nessas regiões, obtendo-se os vetores de características. A Figura 2.3 apresenta uma aplicação utilizando o descritor *Histograms of Oriented Gradients* (HOG) para detecção de placas de trânsito em imagens de rodovias [2].

Figura 2.3: Utilização de descritores HOG [2] para a detecção de placas de trânsito.

Em imagens digitais, também é possível identificar a textura de suas regiões ou objetos constituintes. Nesse caso, a partir de um conjunto de segmentos de textura da imagem, é possível utilizar a técnica *Bag of Visual Words* [28], que é análoga à técnica *Bag of Words*. A técnica *Bag of Visual Words* contabiliza a ocorrência de determinados segmentos de textura em uma imagem, produzindo um vetor de ocorrências de texturas para cada imagem.

2.2 Visualização de Dados

A visualização é um processo que transforma os dados, a informação e o conhecimento em representações gráficas intuitivas para o sistema visual humano, visando facilitar a interpretação e a identificação de padrões implícitos [29, 30]. As representações gráficas, também conhecidas por *layouts*, podem ser feitas por meio de diagramas, gráficos ou mapas. O processo de visualização de dados é estruturado na sequência de passos pré-processamento, mapeamento e renderização [31] como mostra a Figura 2.4.

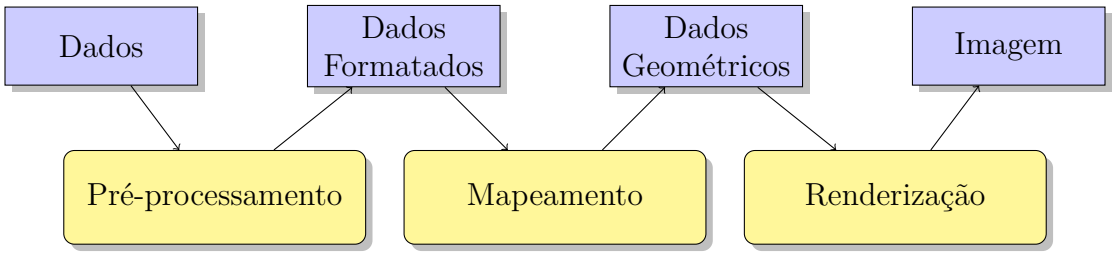


Figura 2.4: Sequência de passos (amarelo) da visualização de dados.

A etapa de pré-processamento é responsável por transformar os dados para uma formatação possível de ser utilizada pela abordagem escolhida na etapa de mapeamento. Esse

pré-processamento pode incluir sub-etapas individuais como alterar formatação, seleção dos dados, limpeza dos dados e mapeamento para dimensionalidades menores. A etapa de mapeamento consiste em mapear os dados formatados para dados geométricos como pontos, linhas e superfícies com os seus respectivos atributos como cores, posicionamento e rotação. Dadas as informações geométricas, é possível gerar a imagem resultante da visualização na etapa de renderização.

Geralmente, técnicas tradicionais de mapeamento e renderização não se comportam bem com dados de dimensionalidade muito grande, conforme abordado no Capítulo 1. Assim, as técnicas de projeções multi-dimensionais podem ser consideradas alternativas viáveis uma vez que suas formulações incorporam uma estratégia para transformar os dados em um espaço de alta dimensionalidade para um espaço de dimensionalidade reduzida.

2.2.1 Visualizações Clássicas

Uma das visualizações de dados mais intuitiva e presente na literatura é o gráfico de dispersão [32], no qual cada instância dos dados é associada com um ponto de um sistema cartesiano de duas ou três dimensões. Em seguida, os pontos são posicionados no sistema de coordenadas conforme os valores em cada uma dessas dimensões. A Figura 2.5 exemplifica um gráfico de dispersão mostrando a base de dados Íris utilizando as dimensões de comprimento e largura das sépalas. Gráficos de dispersões apresentam limitações na visualização de dados cuja dimensionalidade seja maior do que três.

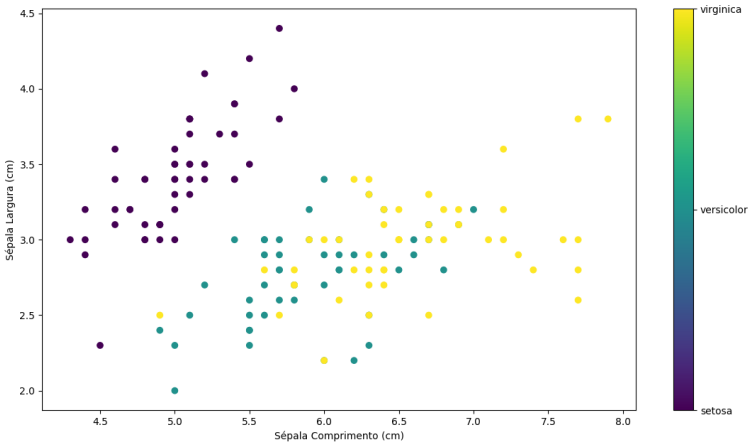


Figura 2.5: Gráfico de dispersão da base de dados Íris sob as dimensões Comprimento da Sépala e Largura da Sépala.

Nesses casos, é possível ignorar atributos relativos à certas dimensões, porém, informações sobre os dados são perdidas. Essa limitação pode ser minimizada utilizando uma matriz de gráficos de dispersão (*Scatterplot matrix*). Nessa estrutura definida por gráficos uniformemente distribuídos em linhas e colunas, a diagonal principal define a dimensão que é representada em um dos dois eixos. Assim, uma posição na matriz tem seu eixo da abcissa definida pela dimensão representada na mesma linha e o eixo da ordenada pela dimensão representada na mesma coluna. Essa representação pode ser considerada complexa, pois é necessário interpretar vários gráficos da matriz de modo a visualizar como os dados se relacionam em cada dimensão. A Figura 2.6 mostra uma matriz de gráficos de dispersão da base de dados Íris, em que a diagonal principal apresenta um histograma de cada dimensão.

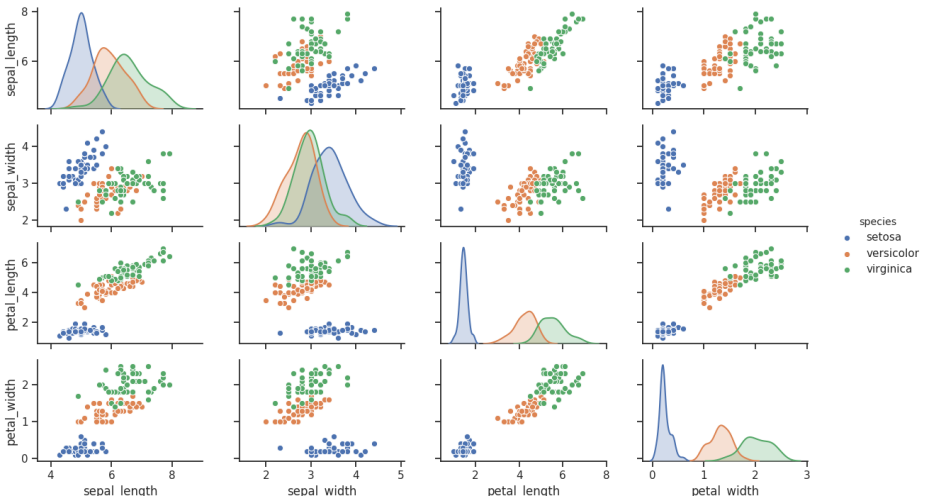


Figura 2.6: Matriz de gráficos de dispersão da base de dados Íris .

Outra alternativa para visualização de dados multi-dimensionais é técnica de visualização por coordenadas paralelas [33], que consiste em representar dados multi-dimensionais por um conjunto de eixos verticais (representando os atributos) e linhas (representando as instâncias). A visualização gerada associa cada dimensão por um eixo vertical no qual uma instância de dados é representada por uma poli-linha, que pode ser definida por um conjunto de semi-retas, em que cada uma das semi-retas conecta duas coordenadas adjacentes. A Figura 2.7 mostra como a base de dados Íris de quatro dimensões com três classes se comporta em uma visualização utilizando a técnica coordenadas paralelas.

Apesar de suportar conjuntos de dados com poucas dimensões, o uso de coordenadas paralelas tem algumas limitações práticas. Uma delas se refere à maneira com que as variáveis correlacionadas não podem ser necessariamente representadas em coordenadas adjacentes, o que requer a ordenação dos eixos, um processo que não é trivial para todos os tipos de dados. Alguns dados podem conter um número excessivo de dimensões, como

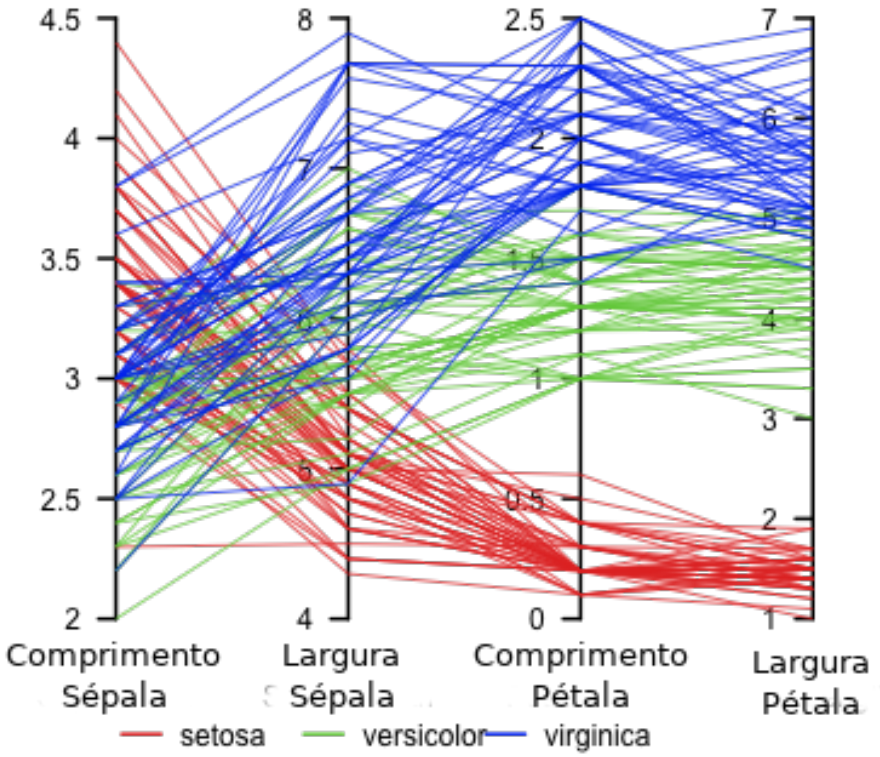


Figura 2.7: Gráfico de coordenadas paralelas da base de dados Íris.

centenas ou milhares, inviabilizando o uso de coordenadas paralelas por se tornar complexo abstrair informação em um espaço visual limitado. Isso mostra que aplicar outras técnicas em etapas anteriores à visualização, muitas vezes, é indispensável para simplificar e visualizar dados multi-dimensionais.

2.2.2 Projeções multi-dimensionais

Uma projeção multi-dimensional é definida por uma função que mapeia os dados definidos em um espaço de alta dimensionalidade m para um espaço de dimensionalidade p, em que $m > p$ e tipicamente $p = 1, 2, 3$ para viabilizar a visualização. Formalmente definida por Paulovich [3] como: Seja $\mathbf{X}$ um conjunto de pontos em $\mathbb{R}^m$, $\delta : \mathbb{R}^m \times \mathbb{R}^m \mapsto \mathbb{R}$ uma medida de proximidade dos pontos em $\mathbb{R}^m$, $\mathbf{Y}$ um conjunto de pontos em $\mathbb{R}^p$ e $d : \mathbb{R}^p \times \mathbb{R}^p \mapsto \mathbb{R}$ uma medida de proximidade dos pontos em $\mathbb{R}^p$. Uma projeção multi-dimensional pode ser

descrita como uma função $f : \mathbf{X} \mapsto \mathbf{Y}$ que tenta aproximar $|\delta(\boldsymbol{x}_i, \boldsymbol{x}_j) - d(f(\boldsymbol{x}_i), f(\boldsymbol{x}_j))|$ de zero, $\forall \boldsymbol{x}_i, \boldsymbol{x}_j \in \mathbf{X}$.

As técnicas de projeções multi-dimensionais são vastas na literatura, sendo possível elaborar uma taxonomia. Neste trabalho, considera-se a taxonomia definida por Paulovich [3]. Ilustrada na Figura 2.8, essa taxonomia define três vertentes principais das técnicas de projeções multi-dimensionais: *Multidimensional Scaling*, *Force-Directed Placement* e Redução de dimensionalidade. Além disso, cada uma dessas categorias possui suas respectivas subdivisões.

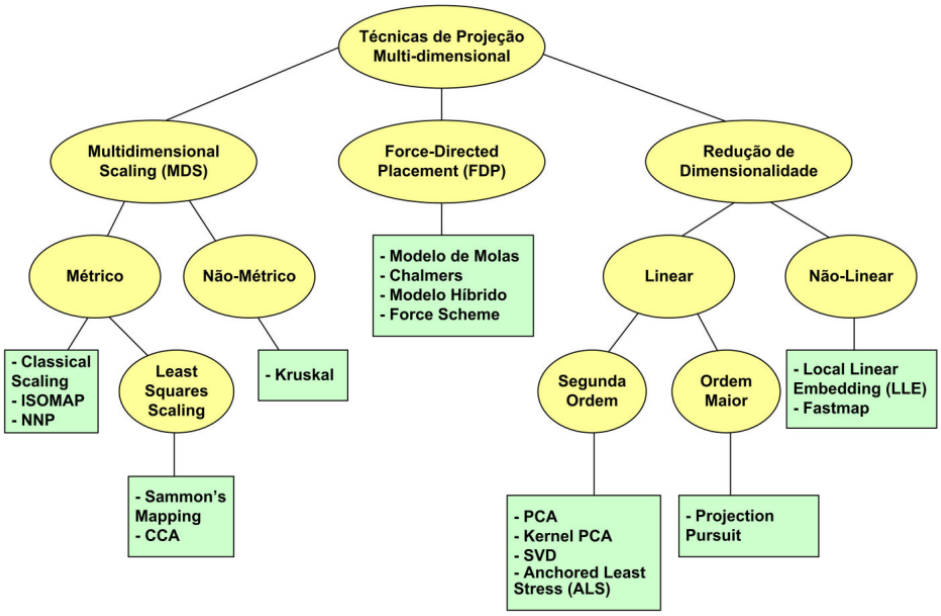


Figura 2.8: Taxonomia das projeções multi-dimensionais [3] .

Multidimensional Scaling

Multidimensional Scaling (MDS) [34] é uma abordagem para a projeção orientada às distâncias entre pares de instância de dados. Dada uma matriz de dissimilaridades associada a um conjunto de dados (por exemplo, uma estrutura matricial que contém todas as distâncias entre todos os pares de instâncias), uma técnica MDS gera um mapeamento tentando preservá-las e ignorando o posicionamento inicial dos pontos no espaço original (de alta dimensão).

O MDS é uma técnica espectral aplicada sobre a matriz de dissimilaridades associada ao conjunto de dados. Essa técnica determina uma projeção que define um mapeamento dos dados no espaço original para as posições dos pontos no espaço reduzido. O Algoritmo 1 descreve as etapas que constituem o princípio MDS.

Algoritmo 1: Multidimensional Scaling Clássico

1) Entre com os dados $\mathbf{X}$ de m dimensões e p dimensões desejadas no mapeamento
2) Construa a matriz $D^{(2)} = [d_{ij}^2]$ utilizando a Eq. 2.6

$$d_{ij} = \sqrt{\sum_{k=1}^{m}(x_{i,k} - x_{j,k})^2} \tag{2.6}$$

3) A partir da matriz $D^{(2)}$, calcule a matriz $A = -\frac{1}{2}JD^{(2)}J$ onde J é uma matriz de centralização definida por $J = I_m - \frac{1}{m}11'$
4) Calcule os p autovalores $\lambda_1, \lambda_2, \ldots, \lambda_p$ de A associados aos seus autovetores $e_1, e_2, \ldots, e_p$
5) A projeção é definida por $\mathbf{X'} = E_m\Lambda_m^{1/2}$ onde E_m é a matriz de autovetores e Λ_m é a matriz diagonal de autovalores.

Uma projeção multi-dimensional desta categoria é o *Isometric Mapping* (ISOMAP) [35], que primeiramente representa as instâncias de dados como um grafo construído com base k vizinhos mais próximos de cada nó. Tal grafo se assemelha a uma matriz de distâncias geodésicas, sendo passada como entrada para a técnica MDS como mostra o Algoritmo 2.

Algoritmo 2: ISOMAP

1) Entre com os dados $\mathbf{X}$ e o número de k de vizinhos a serem avaliados
2) Calcular todos os k vizinhos mais próximos de cada ponto de $\mathbf{X}$
3) Construir um grafo a partir da vizinhança de cada ponto
4) Computar o mapeamento a partir das distâncias geodésicas de cada ponto no grafo para um espaço de dimensão reduzida por meio do MDS

Force-directed Placement

Force-directed Placement (FDP) [36] é um princípio que simula um esquema de atração e repulsão entre as instâncias de dados visando determinar encontrar um estado de equilíbrio. Conhecido como análogo a um sistemas de molas, o FDP recebe como entrada um grafo que é abstraído como um sistema físico de atração e repulsão. A ideia é aplicar modelos físicos reais desse sistema, como a lei de Hooke para modelo de molas, ou as leis da eletrodinâmica como lei de Coulomb. Após atingir o equilíbrio entre as forças de

atração e repulsão, a projeção dos dados no espaço de alta dimensão é realizada para uma dimensão reduzida.

Redução de Dimensionalidade

As projeções multi-dimensionais baseadas na redução de dimensionalidade são caracterizadas por uma função objetivo f, que recebe um conjunto $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ de variáveis aleatórias com dimensionalidade m. Essa função realiza um mapeamento para um conjunto associado $Y = \{\mathbf{y}_1, \ldots, \mathbf{y}_N\}$ de dimensionalidade p, em que $p < m$ [37]. A função objetivo que caracteriza esse mapeamento deve ser definida de acordo com algum critério e deve gerar um espaço de baixa dimensão, preservando ao máximo os padrões e estruturas nos dados originais. Uma projeção multi-dimensional pode ser obtida a partir de uma técnica de redução de dimensionalidade ao considerar $p = \{1, 2, 3\}$ e ao formular f com um critério para preservar as relações de similaridade nos dados [3].

As técnicas de redução de dimensionalidade podem ser categorizadas em lineares e não-lineares [3], devido à natureza do mapeamento ser baseada em formulação linear. Uma das técnicas lineares para redução de dimensionalidade mais populares é a *Principal Component Analysis* (PCA) [38] que surgiu em 1901 e se popularizou ao ser aplicada em várias áreas como química e geologia [39], além de ter se tornando parte do currículo de algumas dessas áreas não diretamente relacionadas com a análise de dados. Essa técnica realiza uma transformação ortogonal nos dados para apresenta-los por suas componentes que os representam melhor, chamadas de componentes principais.

Um exemplo de técnica não-linear é o Local Linear Embbeding (LLE) Standard, que foi desenvolvido para abordar o problema de redução de dimensionalidades não lineares [40] que pode ser descrito pelo Algoritmo 3.

$$\mathcal{E}(W) = \sum_i |X_i - \sum_j W_{ij} X_j|^2 \tag{2.7}$$

$$\Phi(Y) = \sum_i |Y_i - \sum_j W_{ij} Y_j|^2 \tag{2.8}$$

2.3 Avaliação das Projeções Multi-dimensionais

A avaliação de uma projeção multi-dimensional requer a definição de um critério capaz de medir a preservação dos padrões e estruturas presentes nos dados originais na projeção obtida. O critério para avaliação verifica se as relações de vizinhança nos dados originais são preservadas nos pontos associados no espaço reduzido, obtido como resultado da

Algoritmo 3: Local Linear Embedding Standard

1) Entre com os dados $\mathbf{X}$
2) Computar os vizinhos de cada X_i
3) Computar os pesos W_{ij} que reconstroem a instância X_i a partir dos vizinhos minimizando o custo da Eq. 2.7 com certas regras de limite.
4) Computar os vetores Y_i que melhor reconstroem W_{ij} minimizando a forma quadrática da Eq. 2.8.

projeção. Na literatura, as técnicas comumente empregadas para avaliar as projeções são a *Neighborhood Preservation* e a *Neighborhood Hit* [5].

2.3.1 Neighborhood Preservation

A *Neighborhood Preservation* avalia a preservação de vizinhança dos pontos no espaço m-dimensional no espaço reduzido, que é considerado na geração do *layout* pela visualização baseada em projeção. Mais detalhadamente, considerando os k vizinhos de cada ponto m-dimensional, calcula-se a proporção dos n vizinhos mais próximos que permanecem vizinhos no espaço reduzido. O processo é definido pelo Algoritmo 4.

Algoritmo 4: Neighborhood Preservation

1) Entre com os dados $\mathbf{X}$ representando os dados originais, $\mathbf{Y}$ representando os dados mapeados e k o número de vizinhos;
2) Calcule a matriz de distâncias entre todos os pontos para os dados originais e também para os projetados gerando as matrizes A e A', respectivamente;
3) Ordene de forma crescente as linhas das matrizes de distâncias A e A';
4) Para cada um dos k primeiros pontos em cada linha de A, verifique se ele se encontra entre os k primeiros pontos da mesma linha de A'. Se sim, incremente i;
5) i é número de instâncias que se encontram na vizinhança k de cada ponto após a projeção;
6) Calcula-se a média da preservação de vizinhança para todas as instâncias do conjunto na projeção, dada por $i/(k * N)$, em que N é o número de instâncias de $\mathbf{X}$.

O processo da avaliação via *Neighborhood Preservation* varia o valor k sobre um intervalo crescente de vizinhanças. Como resultado, é gerado um gráfico *Número de vizinhos*

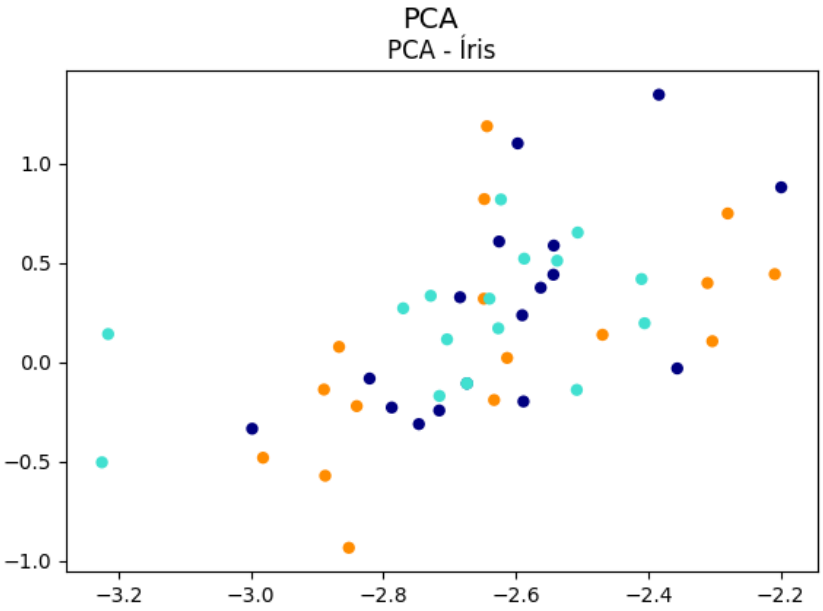


Figura 2.9: Projeção PCA sobre a base de dados [4].

× *Precisão* como mostra a Figura 2.10, que representa a comparação da precisão de preservação de vizinhanças utilizando as técnicas PCA sobre uma curva em 3 dimensões onde seu mapeamento é visualizável pela Figura 2.9.

A Figura 2.10 mostra uma curva que indica a preservação de vizinhanças num intervalo do tamanho dos k vizinhos mais próximos preservados. Realizando a média desse valor sobre todos os pontos é gerado o gráfico. Quanto mais próximo da direita, maior é a vizinhança k avaliada mostrando que é preservada as vizinhanças globais na projeção.

2.3.2 Neighborhood Hit

Já a *Neighborhood Hit* realiza um processo similar a *Neighborhood Preservation*, mas considerando as classes do conjunto de dados. Calcula-se para cada ponto na projeção a proporção dos k vizinhos que se encontram na mesma classe do objeto no espaço m-dimensional. O processo é definido pelo Algoritmo 5.

De modo similar a *Neighborhood Preservation*, gera-se um gráfico *Número de vizinhos* × *Precisão*, em que k assume valores em um intervalo crescente de vizinhanças. A Figura 2.11 exemplifica esse gráfico sobre a base de dados Íris [4].

A Figura 2.11 mostra uma curva que indica a equivalência de classes vizinhas na projeção num intervalo do tamanho dos k vizinhos mais próximos. Assim como na Figura 2.10,

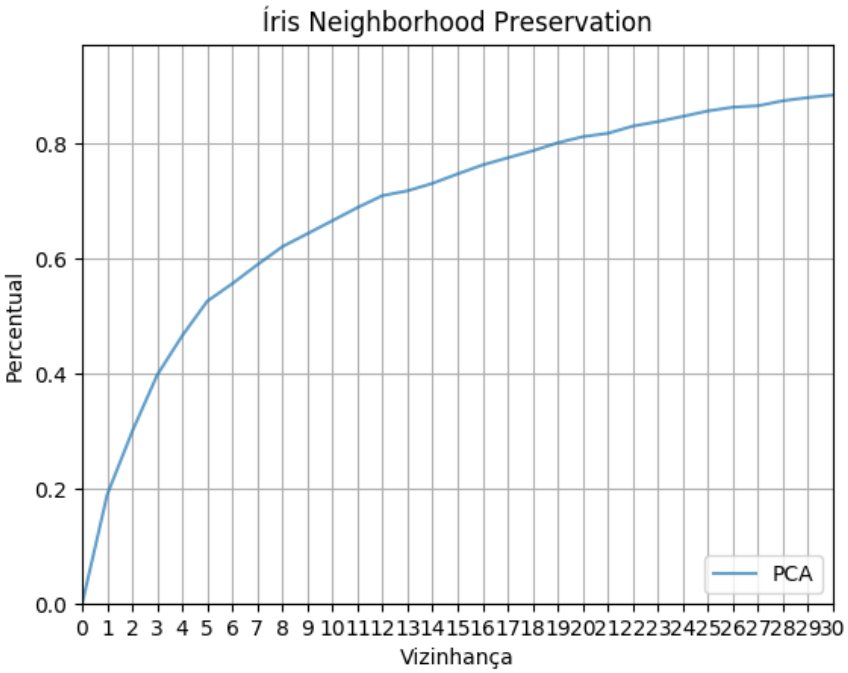


Figura 2.10: Exemplo de *Neighborhood Preservation* [5] uma projeção PCA.

Algoritmo 5: Neighborhood Hit

1) Entre com os dados $\mathbf{X}$ representando os dados projetado e k o número de vizinhos;
2) Calcule a matriz de distância entre todos os objetos associados aos dados originais e também para os pontos na projeção, gerando a matriz A;
3) Ordene de forma crescente as linhas das matrizes de distâncias A;
4) Para cada um dos k primeiros itens em cada linha de A, verifique se ele é da mesma classe que o elemento da linha. Se sim, incremente i;
5) i é número de instâncias que são da mesma classe na vizinhança k de cada ponto após a projeção;
6) A média de classes preservadas dada uma vizinhança k após a projeção é $i/(k * N)$, em que N é o número de instâncias em $\mathbf{X}$.

quanto mais próximo da direita, maior é a vizinhança k avaliada mostrado que a projeção mantém a proximidade das classes após a projeção.

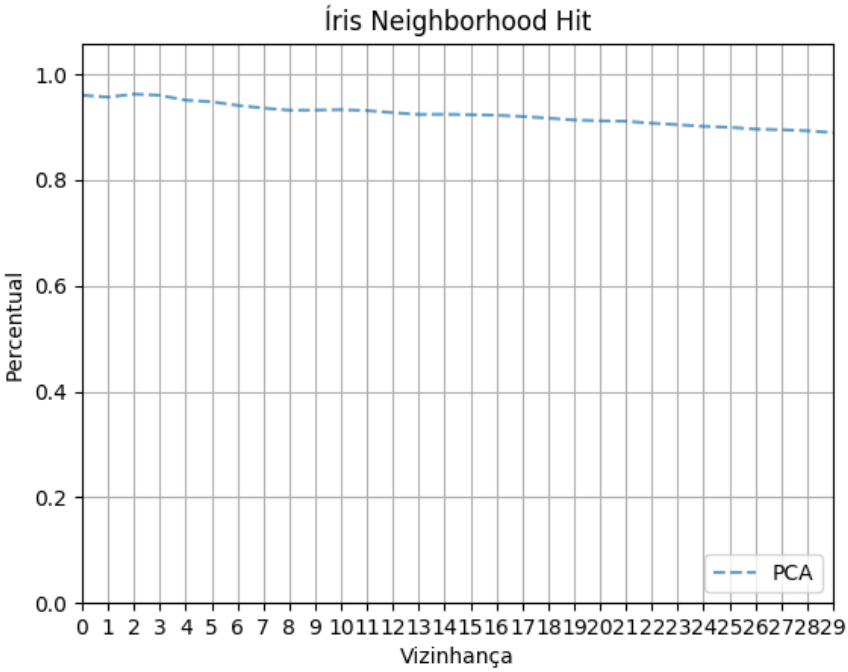


Figura 2.11: Exemplo da *Neighborhood Hit* [5] uma projeção utilizando PCA .

Capítulo 3

Metodologia

Esse capítulo descreve o processo para a utilização de Mapas de Difusão como uma técnica de visualização de dados. Para esse propósito, foi desenvolvido um método que consiste nas etapas de entrada dos dados, pré-processamento com redução de dimensionalidade via Mapas de Difusão e a saída do processo de renderização, que juntamente com as especificações geométricas associadas aos dados, produzirá a visualização.

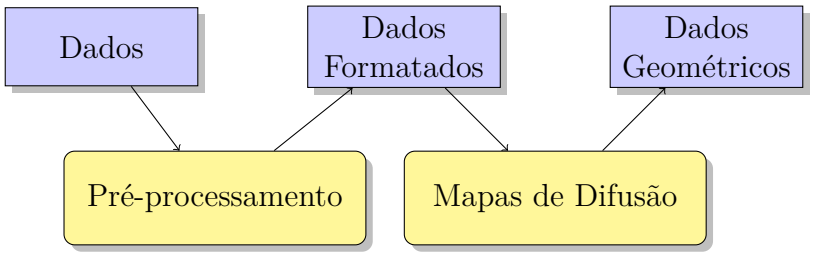


Figura 3.1: Sequência de passos (amarelo) do uso de técnica Mapas de Difusão para visualização de dados.

3.1 Pré-processamento

A etapa de pré-processamento tem como objetivo transformar os dados para um formato vetorial de relevância para em seguida serem processados. Para imagens, são extraídos descritores SIFT [41] que em seguida formam um único vetor de 150 dimensões que representa a imagem. Para textos, é utilizado o algoritmo de vetorização TF-IDF para gerar um vetor do tamanho do vocabulário do corpus indicando a relevância de cada palavra do vocabulário no texto. Os outros tipos de dados abordados já estão em formato vetorial podendo assim serem tratados de forma crua.

3.2 Mapas de Difusão

Mapas de Difusão [18] é uma técnica de redução de dimensionalidade desenvolvida por Coifman e Lafon. Essa técnica utiliza a ideia de passeio aleatório em dados a fim de encontrar estruturas geométricas de baixa dimensionalidade que representem as relações de similaridade existentes nos dados. Esse método não-linear utiliza uma série de procedimentos já fundamentados em outras técnicas caracterizando os dados sobre uma nova perspectiva.

3.2.1 Cadeias de Markov e Passeios Aleatórios

Uma cadeia de Markov é um modelo estocástico, ou seja, com estados indeterminados por terem origem em eventos aleatórios. Esse modelo representa a probabilidade de transição entre estados que depende somente do estado anterior. Assim, pode-se prever a probabilidade de estados futuros somente baseado no estado atual, com resultados semelhantes a predições que utilizam dados históricos sobre a transição. Um exemplo de uma cadeia de Markov com dois estados A e B pode ser visto na Figura 3.2. Pode-se perceber que o estado A possui 40% de chance de continuar no mesmo estado e 60% de ir para o estado B. Já o estado B têm 30% de chance de continuar no mesmo estado e 70% de ir pro estado A.

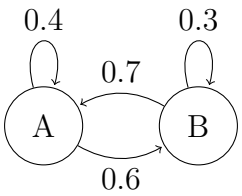


Figura 3.2: Representação gráfica de uma cadeia de Markov.

Como a técnica Mapas de Difusão se baseia na teoria da Cadeias de Markov, deve-se representar os dados sob a forma de um grafo de probabilidades. Um grafo G é definido por um conjunto não-vazio V chamado de vértices ou nós, um conjunto E que forma as arestas do grafo G e uma função w que define os pesos dessas arestas.

$$G = (V, E, w) \tag{3.1}$$

$$V = \{v_1, v_2, \ldots, v_N\} \tag{3.2}$$

$$E = \{e_{ij} = (v_i, v_j) | v_i, v_j \in V\} \tag{3.3}$$

$$w : V \times V \to W \subset \mathbb{R} \tag{3.4}$$

Como as cadeias de Markov representam conectividade entre estados, pode-se modelar um grafo de probabilidades em que os estados são os vértices e as probabilidades de transição são as arestas.

Um conceito utilizado sobre as cadeias de Markov é o problema de passeios aleatórios. Considerando um estado inicial, é avaliada qual a probabilidade de chegar a um estado final após t transições aleatórias. Essa abordagem permite não só avaliar a probabilidade do próximo estado, mas qualquer probabilidade de chegar a um dado estado após t passos.

3.2.2 Conectividade

Grafos é uma estrutura de dados que se aplica naturalmente a problemas que envolvem relações e conectividade, porém contém uma série de desafios inerentes à sua estrutura. Esses problemas são de grande relevância para aplicações reais evidenciado pela fundamentação da área de teoria dos grafos que é especializada no estudo de tais problemas.

Associado a outras técnicas como cadeias de Markov, diversos trabalhos na literatura demonstraram que é também possível empregar grafos em problemas de classificação [42], ranqueamento [43], agrupamento [44] e redução de dimensionalidade. Técnicas de redução de dimensionalidade baseadas em grafos são muito poderosas por serem simples, efetivas e representarem satisfatoriamente associações complexas nos dados [45].

Na técnica Mapas de Difusão, os dados são representados em um grafo, porém a medida de conectividade entre eles não é implícita. Assim, a conectividade entre as instâncias dos dados é abstraída por um *kernel* aplicado a todas as combinações das instâncias. Isso representa a probabilidades de um único passeio aleatório entre duas instâncias. O *kernel* k deve ser simétrico e não-negativo, sendo restrito pelas equações Eq. 3.5 e Eq. 3.6:

$$k(\mathbf{x_1}, \mathbf{x_2}) = k(\mathbf{x_2}, \mathbf{x_1}), \tag{3.5}$$

$$k(\mathbf{x_1}, \mathbf{x_2}) \geq 0 \tag{3.6}$$

É comum na literatura utilizar o *kernel* gaussiano (Eq. 3.7) por ter a propriedade $k(x, x) > 0, \forall x \in V$, conforme mostra a Eq. 3.7

$$k(\mathbf{x_1}, \mathbf{x_2}) = exp\left(- \frac{\|\mathbf{x_1} - \mathbf{x_2}\|^2}{\epsilon} \right), \tag{3.7}$$

em que ϵ é um parâmetro do *kernel* gaussiano que define sua escala, ou seja, seu valor pode aumentar ou diminuir o peso das arestas do grafo. Em outras palavras, o valor de ϵ é fundamental para modelar as relações de vizinhança do grafo. Por isso, o resultado do *kernel* gaussiano não é generalizável, uma vez que os dados possuem características

e particularidades próprias. Logo, o valor do parâmetro ϵ deve ser obtido por meio de experimentações ou alguma heurística sobre os dados.

Todas as combinações de vértices sobre o *kernel* podem ser armazenadas em uma matriz simétrica não-negativa $K = [k_{ij}]_{N\times N}, k_{ij} = k(\mathbf{x_i}, \mathbf{x_j})$. Porém, para poder passar pelo processo de passeios aleatórios é necessário ocorrer uma normalização para que a soma das probabilidades de transição de cada estado tenha a soma igual a 1. Uma matriz de normalização D^{-1} pode ser gerada da seguinte forma:

$$D_{N\times N}, D_{ii} = \sum_{j=1}^{N} k(\mathbf{x_i}, \mathbf{x_j}). \tag{3.8}$$

É notável que a matriz D é uma matriz diagonal onde cada valor da diagonal representa a soma de todos os valores da mesma linha na matriz K. Assim, a matriz P de transição para realizar o passeio aleatório pode ser decomposta como:

$$P = D^{-1}K. \tag{3.9}$$

3.2.3 Processo de Difusão

Dada a matriz P de transição, é possível verificar um processo de difusão no grafo, ou seja, o passeio aleatório em dada t transições, isto é, eleva-se a matriz P à potência t (P^t).

A probabilidade do caminho entre as instâncias percorrerem a estrutura geométrica natural dos dados aumenta proporcionalmente ao número de etapas t. Isso ocorre devido à alta conectividade dos dados em locais densos, e analogamente, à baixa conectividade em locais esparsos, cujas características são atenuadas a cada nova etapa.

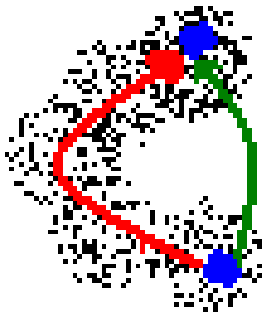


Figura 3.3: Caminho pela estrutura geométrica dos dados [6].

É visível na imagem Figura 3.3 que o caminho verde teria sua probabilidade reduzida ao longo do processo por consistir em poucos pulos mais longos, enquanto o caminho vermelho seria um caminho de maior probabilidade por possuir vários pulos curtos.

Esse processo revela a estrutura geométrica dos dados a medida que as etapas vão ocorrendo, além de se mostrar resiliente a ruido nos dados já que instâncias que pouco representam a estrutura geométrica têm pouca relevância no decorrer do processo.

3.2.4 Distância de Difusão

Dado o processo de difusão, é necessário mapear os dados para uma dimensionalidade menor. Para esse propósito, é introduzido o conceito da distância de difusão, que é uma métrica denotada por:

$$D_t^2(\mathbf{x_i}, \mathbf{x_j}) = \|p_t(\mathbf{x_i}, \cdot) - p_t(\mathbf{x_j}, \cdot)\|_\xi^2, \tag{3.10}$$

em que $\cdot$ representa as instâncias em $\mathbf{X}$, $p(\mathbf{x_i}, \cdot)$ determina a probabilidade da transição para o ponto $\cdot$ e ξ é um fator de ponderação. É perceptível que $\|p_t(\mathbf{x_i}, \cdot) - p_t(\mathbf{x_j}, \cdot)\|$ é a diferença da linhas i e j da matriz P^t ponderada por ξ que define a distância de difusão.

As distâncias de difusão D_t ainda contém a mesma dimensionalidade dos dados originais. Para reduzir, é necessário descartar dimensões no espaço de difusão e mapear para o espaço euclidiano. É provado por Coifman e Lafon [18] que utilizando uma abordagem espectral, $D_t(x_i, x_j)$ pode ser definido por:

$$D_t^2(\mathbf{x_i}, \mathbf{x_j}) = \sum_{l \geq 1} \lambda_l^{2t} (\psi_l(\mathbf{x_i}) - \psi_l(\mathbf{x_j}))^2 = \|\Psi_t(\mathbf{x_i}) - \Psi_t(\mathbf{x_j})\|^2, \tag{3.11}$$

em que λ_l são os autovalores e ψ_l são autovetores de P.

O mapeamento $\Psi_t : \mathbf{X} \to \mathbb{R}^p$ é definido pela Eq. 3.12:

$$\Psi_t(\mathbf{x_i}) = \begin{bmatrix} \lambda_0^t \psi_0(\mathbf{x_i}) \\ \lambda_1^t \psi_1(\mathbf{x_i}) \\ \vdots \\ \lambda_{p-1}^t \psi_{p-1}(\mathbf{x_i}) \end{bmatrix} \tag{3.12}$$

onde $\psi_0(\mathbf{x_i})$ representa o i-ésimo valor do vetor ψ_0.

Vale ressaltar que existe um mapeamento das distâncias de difusão para o espaço euclideano utilizando métricas de difusão. Como o processamento da distância de difusão tem um custo computacional alto, o mapeamento proposto permite simplificar essa tarefa.

A matriz de distâncias D_t e a função de mapeamento Ψ_t possibilitam reduzir a dimensionalidade dos dados para um espaço de distância entre as instâncias por meio da decomposição espectral realizada na matriz P como mostra o mapeamento Ψ. Essa abordagem é conhecida como uma abordagem espectral utilizada por outros algoritmos como

PCA, MDS e ISOMAP. A técnica espectral utiliza autovalores e autovetores para descrever os dados de forma a selecionar as componentes que melhor o descrevem.

A técnica Mapas de Difusão não utiliza o primeiro fator de mapeamento $\lambda_0^t \psi_0(x_i)$ por este ser constante, assim utilizando somente os seguintes $p-1$ fatores para o mapeamento. Além disso, os novos eixos que definirão o espaço de baixa dimensionalidade estão associados com a ordenação dos autovalores extraídos da decomposição espectral da matriz P.

De maneira resumida, o Algoritmo 6 descreve as etapas que definem a técnica Mapas de Difusão:

Algoritmo 6: Algoritmo descrevendo Mapas de Difusão

1) Entre com os dados $\mathbf{X_{N\times d}}$ (Dados), ϵ, p (Dimensionalidade do mapeamento), t (Tempo) onde d é o número de dimensões dos dados X e N é o número de instâncias
2) Computar a matriz $K_{N\times N}$ utilizando o *kernel* $k(\mathbf{x_i}, \mathbf{x_j})$
3) Computar a matriz diagonal $D_{N\times N}, D_{ii} = \sum_{j=1}^{N} k(x_i, x_j)$
4) Computar a matriz de transição $P = D^{-1}K$
5) Calcular autovalores e autovetores de P, obtendo-se o conjunto de autovetores e autovalores.
6) Construir o mapeamento $\Psi_t(\mathbf{x_i}) = \begin{bmatrix} \lambda_1^t \psi_1(\mathbf{x_i}) \\ \lambda_2^t \psi_2(\mathbf{x_i}) \\ \vdots \\ \lambda_p^t \psi_p(\mathbf{x_i}) \end{bmatrix}, \forall i \in \{1, 2, \cdots, N\}$

3.2.5 Mapas de Difusão na Literatura

Mapas de Difusão é uma técnica de redução de dimensionalidade que não tem grande presença na literatura em comparação com outras técnicas como PCA, MDS e LLE. Os principais trabalhos utilizaram o Mapas de Difusão em experimentos para comparar com outros métodos de redução de dimensionalidade ou realizaram aplicações em problemas específicos.

Uma pesquisa focada no processo dos Mapas de Difusão [6] foi conduzida por De la Porte *et al.*, que avalia as vantagens da técnica em relação ao MDS e PCA. Os autores verificaram a preservação de grupos de uma base de dados tridimensional em um formato de "C" após a aplicação do mapeamento. Os resultados mostraram que, apesar do MDS e do PCA terem mapeado os dados de maneira semelhante, a ordenação dos grupos foi somente preservada na técnica Mapas de Difusão.

Um avaliação comparativa entre técnicas de redução de dimensionalidade foi feita por Maaten e Postma [46]. O estudo reuniu 13 algoritmos de projeção multi-dimensional, que foram aplicados em cinco conjuntos de dados artificiais e cinco conjunto de dados reais. Os experimentos consideraram a preservação de 1 vizinho mais próximo em relação a cada objeto multi-dimensional de acordo com uma medida de confiança e uma medida de continuidade. A medida de confiança avalia se os pontos da projeção estão próximos entre si, sendo calculada pela Eq. 3.13:

$$T(k) = 1 - \frac{1}{nk(2n - 3k - 1)} \sum_{i=1}^{n} \sum_{j \in U_i^{(k)}} (r(i,j) - k), \tag{3.13}$$

em que $r(i,j)$ representa o posto do ponto j no espaço projetado de acordo com as distâncias entre todos os pares de pontos do espaço de projeção. A variável $U_i^{(k)}$ indica o conjunto de pontos que estão na vizinhança k na projeção, mas não na vizinhança k de um objeto multi-dimensional. Já a medida de continuidade verifica a extensão da preservação das vizinhanças de acordo com a Eq. 3.14:

$$C(k) = 1 - \frac{1}{nk(2n - 3k - 1)} \sum_{i=1}^{n} \sum_{j \in V_i^{(k)}} (\hat{r}(i,j) - k), \tag{3.14}$$

em que $\hat{r}(i,j)$ representa o posto do ponto j no espaço original de acordo com as distâncias entre todos os pares de pontos do espaço original. A variável $V_i^{(k)}$ indica o conjunto de pontos que estão na vizinhança k no dado original mas não na vizinhança k da projeção.

Além disso, o referido estudo explorou um possível intervalo de valores para o parâmetro t de entrada para a técnica Mapas de Difusão como $[10, 100]$ e considera $\epsilon = 2$. Essa generalização do parâmetro ϵ para todos os conjuntos de dados apresenta uma limitação na experimentação. Os resultados em bases de dados artificiais da avaliação de medida de 1 vizinho mais próximo e confiança para a técnica foi muito inferior em relação às restantes, porém a medida de continuidade obteve bons resultados, mostrando sua capacidade de preservar regiões com geometria bem definida. Para conjuntos de dados reais, o resultado foi inferior para todas as medidas. O próprio trabalho relata que a motivação da baixa performance é que a seleção do kernel não deve ser generalizada para todos os conjuntos de dados e sua escolha depende fortemente da natureza dos dados.

Em sua dissertação de mestrado, Leandro [19] aplicou a técnica do Mapas de Difusão para detectar zonas de singularidade em campos vetoriais planares. A pesquisa utilizou o campo vetoriais artificiais como a entrada do algoritmo e avaliou quais regiões de singularidade apresentam baixa similaridade causando "furos" nas coordenadas de difusão de forma global, facilitando a identificação dessas regiões.

Rajpoot *et al.* [47] utilizaram Mapas de Difusão para encontrar a correspondência de formas visuais em silhuetas em conjuntos de imagens reais e binárias. Após gerar descritores multi-dimensionais conforme as características da forma de regiões das imagens, a técnica Mapas de Difusão foi aplicada no conjunto de descritores com o objetivo de avaliar a preservação de classes. Para esse propósito, uma medida chamada *class separability index* (índice de separabilidade de classes) definida pela Eq. 3.15, foi empregada para analisar cada agrupamento i.

$$c_i = \frac{\bar{d}_i}{\sqrt{\bar{\sigma}_i^2}}, \tag{3.15}$$

Na Eq. 3.15 acima, $\bar{d}_i$ é a distância média do centroide do *cluster* i para todos os outros agrupamentos e $\bar{\sigma}_i$ é definido como a medida do espalhamento médio pela média das variâncias da componente principal do agrupamento. Os resultados do trabalho avaliam como efetivo o uso da técnica mostrando que somente ocorreu ambiguidade em formas naturalmente semelhantes como as classes "cachorro" e "cavalo".

Xu *et al.* [20] aplicaram Mapas de Difusão para analisar genes cancerígenos e assim determinar sua origem. A análise é complexa devido a baixa quantidade de amostras de cada tipo de tumor, além da alta dimensionalidade das instâncias. O trabalho utilizou Mapas de Difusão para reduzir a dimensionalidade dos genes originais e em seguida empregaram a técnica Fuzzy ART [48] para realizar um agrupamento dos pontos do espaço de baixa dimensionalidade. Os experimentos com Mapas de Difusão consideraram o parâmetro $t = 1$ e variou-se o valor de ϵ. Os resultados mostraram que a técnica discriminou tipos diferentes de câncer de acordo com os grupos obtidos e sugere, como trabalhos futuros, avaliar configurações alternativas em sua formulação.

Os trabalhos descritos mostraram o potencial da técnica Mapas de Difusão quando aplicada em problemas reais de algumas áreas do conhecimento. Os trabalhos mencionaram que sua utilização para a preservação de vizinhanças e medidas de continuidade têm resultados superiores quando o *kernel* é configurado especificamente para o conjunto de dados desejado.

3.3 Implementação

Para realizar o experimento, a aplicação foi desenvolvida na linguagem de programação *Python*[1] devido a sua simplicidade e ativa comunidade que disponibilizou uma série de ferramentas, *frameworks* e bibliotecas que auxiliam no desenvolvimento de algoritmos e abordagens, evitando o retrabalho.

[1] `https://www.python.org`https://www.python.org

Essa aplicação receberá como entrada as bases de dados pré-processadas, ou seja, em formato vetorial pronto para aplicação dos algoritmos MDS, ISOMAP, LLE Standard e Mapas de Difusão [2], em que para cada base de dados, será considerada uma configuração específica do *kernel* gaussiano nos Mapas de Difusão. A definição da variável ϵ do *kernel* será realizada empiricamente e também utilizando o método BGH [49], para verificar se essa estratégia automática é uma alternativa robusta para este fim. A técnica BGH realiza regressões não-paramétricas sobre os dados utilizando modelos probabilísticos para estimar o valor de ϵ supondo um *kernel* gaussiano. A variável t do processo de difusão será estimada a partir da avaliação de qual configuração representa melhor o dado.

3.3.1 Algoritmos

Após a seleção da configuração do *kernel* dos Mapas de Difusão, será aplicado os algoritmos LLE Standard [50], MDS Clássico [34] e ISOMAP [35], além dos Mapas de Difusão com suas respectivas configurações. O resultado gerado por cada projeção será avaliado comparativamente utilizando as técnicas *Neighborhood Preservation* e *Neighborhood Hit*.

Para realizar a execução desse processo de forma pragmática, é recomendado utilizar abordagens já fundamentadas e aplicadas por outros trabalhos trazendo assim uma cadeia de confiança ao processo. Para isso é utilizada a linguagem de programação *Python* em conjunto com a biblioteca *scikit-learn*[3] [51]. Esse conjunto fornece as ferramentas necessárias para implementar e avaliar os resultados dos algoritmos de projeção multidimensional.

Python é uma linguagem de programação para uso geral na qual sua principal característica é ser linguagem multi-paradigma com total suporte à programação estruturada e orientação a objeto. Funcionalidades de outros paradigmas como programação funcional e orientada a aspectos também são suportadas na linguagem. Os fatores que tornaram a linguagem *Python* tão popular são as suas simplicidades, extensibilidade e comunidade ativa. Isso proporcionou à linguagem o patamar de uma das principais opções para desenvolvimento de software experimental. Em função da fácil extensibilidade, é simples reutilizar, avaliar e integrar aplicações de terceiros.

Scikit-learn é uma biblioteca desenvolvida em *Python* para auxílio no desenvolvimento de técnicas de aprendizado de máquina. Essa biblioteca contém vários algoritmos de aprendizado já implementados e ferramentas de auxílio de visualização com suporte a gráficos como o de dispersão e barras paralelas. Devido ao grande número de contribuidores e pelo fato de ter seus algoritmos auditados recorrentemente pela comunidade, a

[2]Implementação baseada no código `https://github.com/satra/mapalign`

[3]Código fonte e documentação se encontram no endereço `https://scikit-learn.org`

biblioteca *scikit-learn* foi escolhida para suportar a implementação e a fase de experimentos.

Os algoritmos MDS Clássico [34], ISOMAP [35] e LLE Standard [50] já estão implementados na biblioteca *scikit-learn*. Sua implementações seguem essa estrutura dos Algoritmos 1, 2 e 3, respectivamente detalhadas no Capítulo 2.

Capítulo 4

Resultados

Este capítulo descreve os experimentos realizados para validar e avaliar o método proposto para visualização de dados multi-dimensionais utilizando a técnica Mapas de Difusão. São descritos os conjuntos de dados reais e artificiais empregados e os seus respectivos pré-processamentos necessários para aplicar os algoritmos de redução de dimensionalidade. Como o algoritmo Mapas de Difusão demanda o ajuste de parâmetros, experimentos foram realizados para determinar seus valores apropriados.

Por fim, os *layouts* da projeção baseada em Mapas de Difusão e de outras projeções multi-dimensionais da literatura são gerados, com o intuito de viabilizar a interpretação de padrões implícitos nos dados, como também comparar suas qualidades. Nesse sentido, consideram-se as métricas *Neighborhood Preservation* e *Neighborhood Hit* para avaliar numericamente a qualidade dos *layouts* produzidos pelas projeções consideradas.

4.1 Conjuntos de Dados

A validação e avaliação do método proposto considera conjuntos de dados de diferentes naturezas, que podem ser divididos inicialmente em dois grupos: dados reais e artificiais. Os conjuntos de dados reais utilizados são Íris [4], COREL [52] e *20 News Groups* [53]. Os conjuntos de dados artificiais foram gerados automaticamente, sendo um conjunto de dados formado por pontos randomicamente sobrepostos a uma forma "S" em 3 dimensões e outro conjunto caracterizado por pontos randomicamente sobrepostos a dois círculos concêntricos sendo cada um de classes diferentes. Maiores detalhes sobre esses conjuntos de dados são fornecidos nas próximas sub-seções.

4.1.1 Conjuntos de Dados Reais

O primeiro conjunto de dados avaliado é o Íris [4], que é composto por dados relacionados de três espécies de flores: *Iris Setosa*, *Iris Versicolour* e *Iris Virginica*. A base Íris é formada por 150 instâncias sendo 50 para cada uma das 3 classes na qual cada classe representa uma das espécies de flor citadas. Cada instância contém 4 parâmetros: largura da sépala (cm), altura da sépala (cm), largura da pétala (cm) e altura da pétala (cm). Nessa pesquisa, o conjunto de dados Íris foi escolhido devido à sua grande aplicabilidade na literatura de reconhecimento de padrões, pois apresenta um padrão de classes linearmente segmentadas ou outras não-linearmente segmentadas.

O segundo conjunto de dados se refere à uma coleção de imagens conhecida como COREL [52]. Essa coleção fornece imagens reais com diversas propriedades diferentes como cores, formas e padrões. A coleção contém mil fotografias igualmente distribuídas dentre 10 classes: tribos africanas, praia, prédios, ônibus, dinossauros, elefantes, flores, cavalos, paisagens e comida. As imagens são representadas por descritores SIFT [41], que descrevem cada ponto de interesse da imagem como um vetor de alta dimensionalidade. Em conjunto, esses vetores descrevem a imagem como um único vetor de 150 dimensões. A escolha do conjunto COREL nos experimentos se deve ao fato de que as instâncias são classificadas em dez categorias, igualmente balanceadas, e apresentam alta dimensionalidade, tornando mais complexa a tarefa de distinguir as características que discriminam essas classes entre si.

O terceiro conjunto de dados é o *20 News Groups* [53], uma base textual formada por aproximadamente 20000 notícias divididas em 20 categorias diferentes correspondendo a certos tópicos. Essa base contém uma série de textos reais sobre uma gama variada de assuntos, permitindo verificar a performance dos algoritmos sob textos de temas diversificados. Esse conjunto de textos é dividido em árvores de conhecimento, proporcionando avaliar a proximidade de subtópicos relacionados e não-relacionados conforme a definição dos temas, que podem ser visualizadas pelas árvores na Figura 4.1. Os experimentos consideram uma amostra aleatória de 3000 notícias devido às limitações computacionais do algoritmo LLE Standard, resultado do seu alto custo de memória.

Para a base *20 News Groups* é necessário aplicar pré-processamento em cada documento de texto. Para esse propósito, se utiliza um histograma de *tokens* para identificar cada documento. A matriz de ocorrência do corpus é utilizada como entrada para a técnica TF-IDF, descrita no Capítulo 2. A ideia é representar cada documento como um vetor multi-dimensional, possibilitando seus processamentos pelas técnicas de projeções multi-dimensionais. O critério para considerar o conjunto *20 News Groups* se baseia na existência de relações de similaridade entre os tópicos e as classes dos textos.

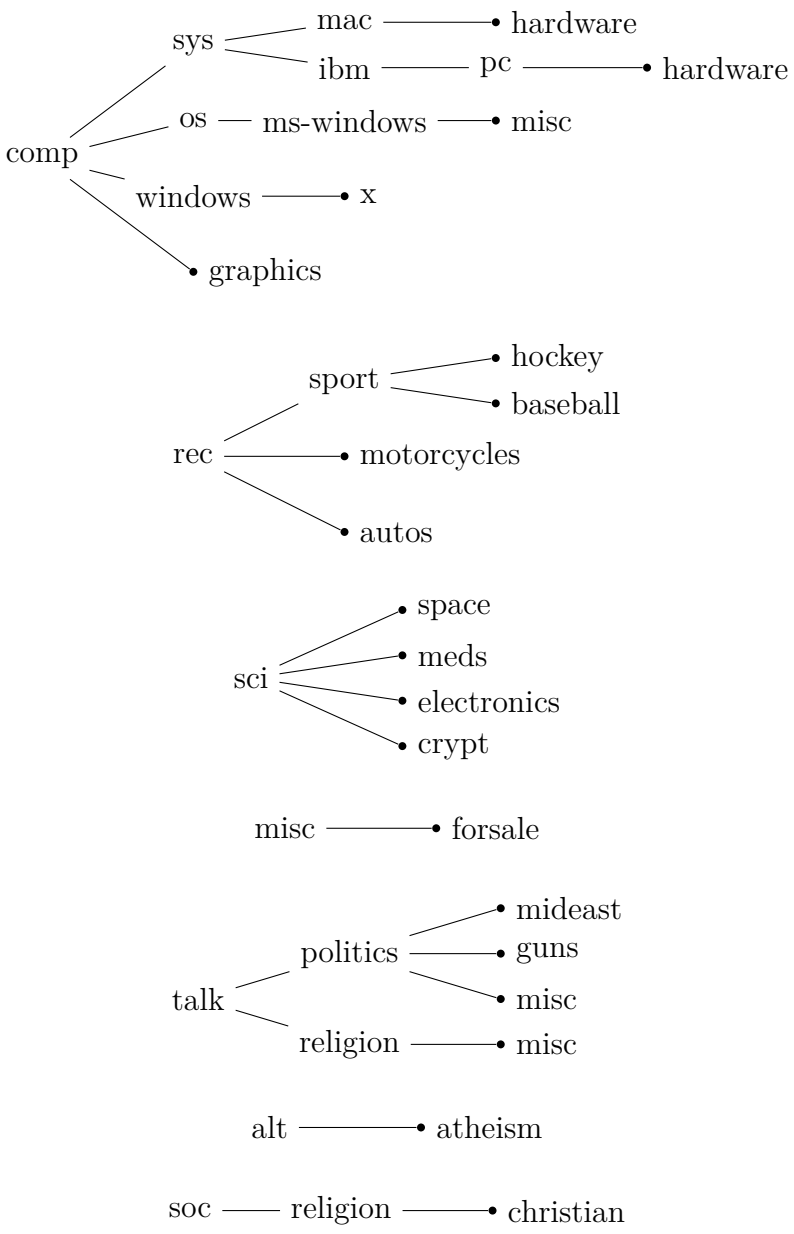


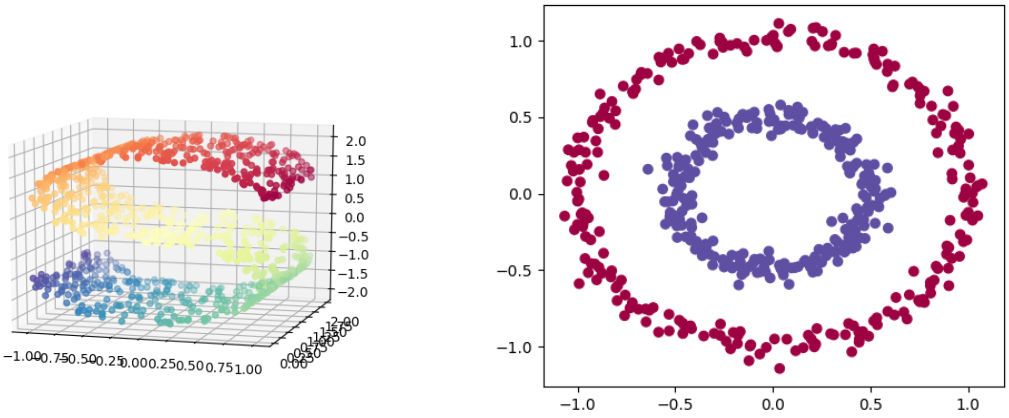
Figura 4.1: Arvore de tópicos da base de dados *20 News Groups*

4.1.2 Conjuntos de Dados Artificiais

É comum em pesquisas que avaliam e validam técnicas de redução de dimensionalidade considerar o uso de dados artificiais por serem facilmente modelados conforme estruturas geométricas pré-definidas. A biblioteca *python scikit-learn* [51] facilita o procedimento de geração de dados artificiais disponibilizando funções com suporte para representação de dados sobre uma grande variedade de estruturas geométricas e combinações possíveis.

Os experimentos nessa pesquisa contemplam dois conjuntos de dados artificiais. O primeiro conjunto é denominado simplesmente por "Curva S", sendo definido por 1000

pontos associados que modelam uma curva que simula uma silhueta “S” de três dimensões, como ilustrado na Figura 4.2a. Por sua vez, o segundo conjunto artificial de dados é chamado de “Círculos” e é formado por 500 pontos associados a dois círculos concêntricos de raios diferentes em duas dimensões, como mostra a Figura 4.2b. Esses conjuntos são automaticamente criados por um método que recebe como entrada um conjunto de pontos gerados randomicamente sobrepostos às estruturas geométricas.



(a) Curva S.

(b) Círculos concêntricos.

Figura 4.2: Conjuntos de dados artificias empregados nos experimentos.

4.2 Seleção dos parâmetros

Os algoritmos avaliados contem vários parâmetros de entrada para realizar o mapeamento dos dados. Essa seção, por sua vez, tem o objetivo de definir a abordagem para a seleção desses parâmetros. Especificamente, a técnica Mapas de Difusão considera em sua formulação dois parâmetros de entrada: t e ϵ (eps). Por isso, é necessário investigar a influência dos valores desses parâmetros na geração do *layout* final.

Primeiramente, avalia-se os resultados do mapeamento da técnica Mapas de Difusão fixando-se o valor $\epsilon = 0.5$ e variando-se os valores $t = \{0.01, 0.1, 1, 10, 100, 1000\}$ considerando o conjunto de dados Corel. O *layout* produzido na Figura 4.3 exemplifica as variações do parâmetro t, que representa a escala do mapeamento no espaço de difusão como avaliado por Lafon e Coifman [18]. De acordo com a Eq. 3.12 e com as visualizações realizadas, pode-se interpretar que o parâmetro t afeta o *layout* principalmente na escala das coordenadas. Logo, os experimentos seguintes consideram o valor fixo $t = 10$.

A experimentação considerando o parâmetro ϵ (eps) leva em conta os seguintes valores $\epsilon = \{0.1, 0.5, 1, 5, 10, 50\}$, mantendo-se $t = 10$. A variação do ϵ causa grande mudança no

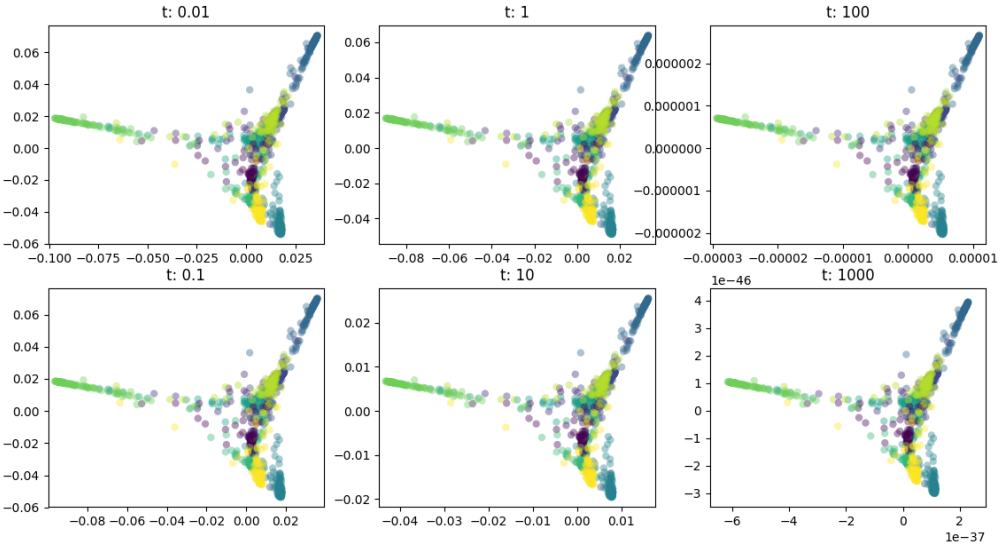


Figura 4.3: Exemplo do comportamento de projeções sobre $t = \{0.01, 0.1, 1, 10, 100, 1000\}$ da base de dados Corel.

processo de difusão como mostra os layouts gerados pela Figura 4.4, uma vez que altera as escalas das distâncias iniciais dentre os dados, como também afeta a geometria do posicionamento das pontos no *layout*.

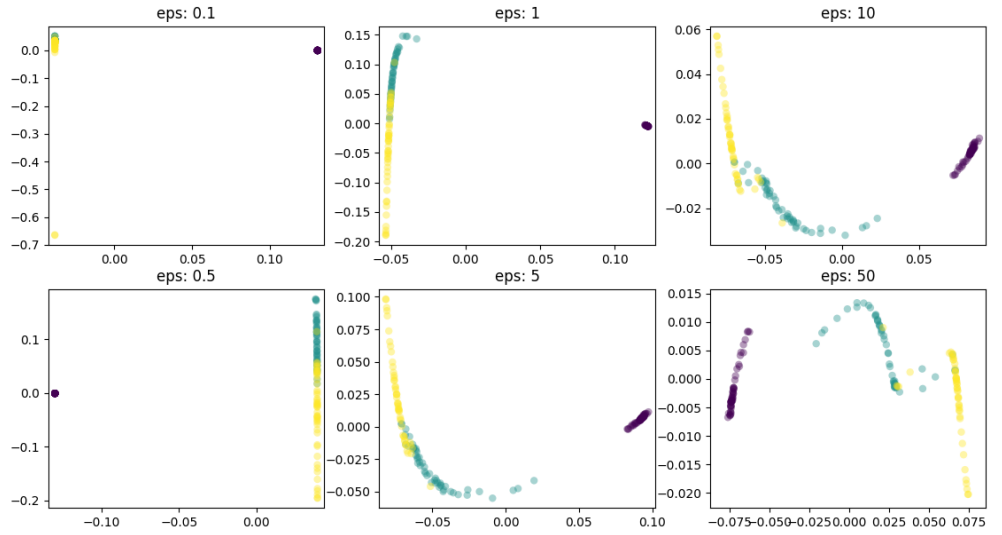


Figura 4.4: Exemplo do comportamento das projeções sob variação do $\epsilon = \{0.1, 0.5, 1, 5, 10, 50\}$ da base de dados Iris sobre .

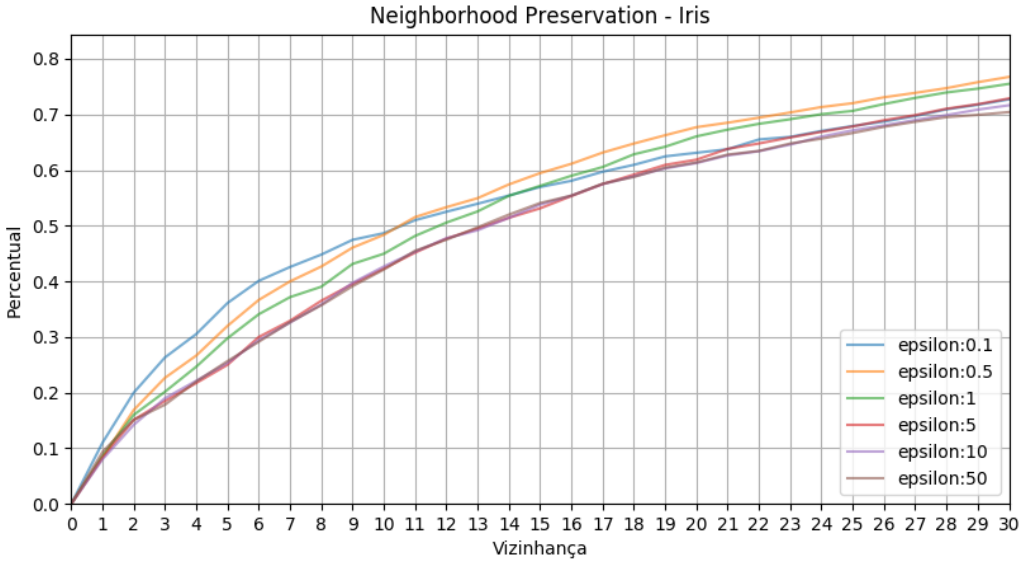


Figura 4.5: *Neighborhood Preservation* da base de dados Iris variando-se os valores para ϵ.

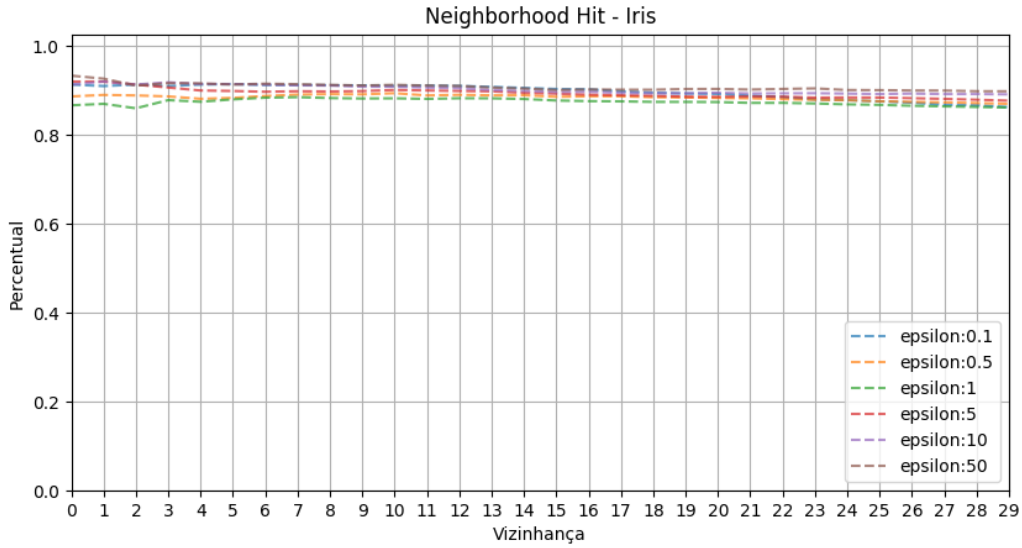


Figura 4.6: *Neighborhood Hit* da base de dados Iris sobre variando-se os valores para ϵ.

Com os resultados das métricas *Neighborhood Preservation* e *Neighborhood Hit* sobre cada configuração de ϵ, conforme ilustrado na Figura 4.5 e na Figura 4.6, os valores de cada coordenada paralela são normalizados entre si. Os gráficos normalizados são apresentados na Figura 4.7 e na Figura 4.8, respectivamente. A configuração que tiver a maior soma de valores normalizados em cada coordenada será selecionada como mostra o Algoritmo

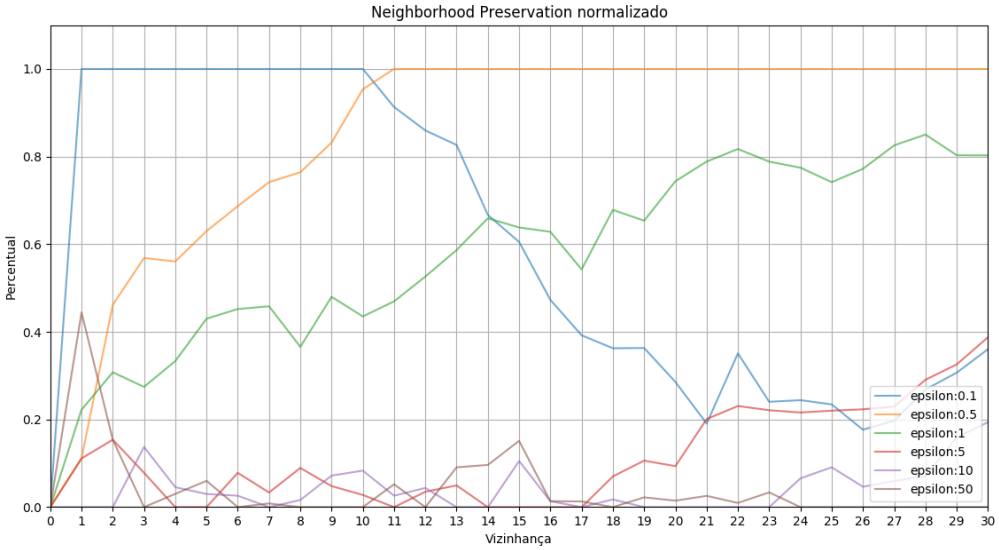


Figura 4.7: *Neighborhood Preservation* normalizado da base de dados Iris sobre vários ϵ (eps) diferentes.

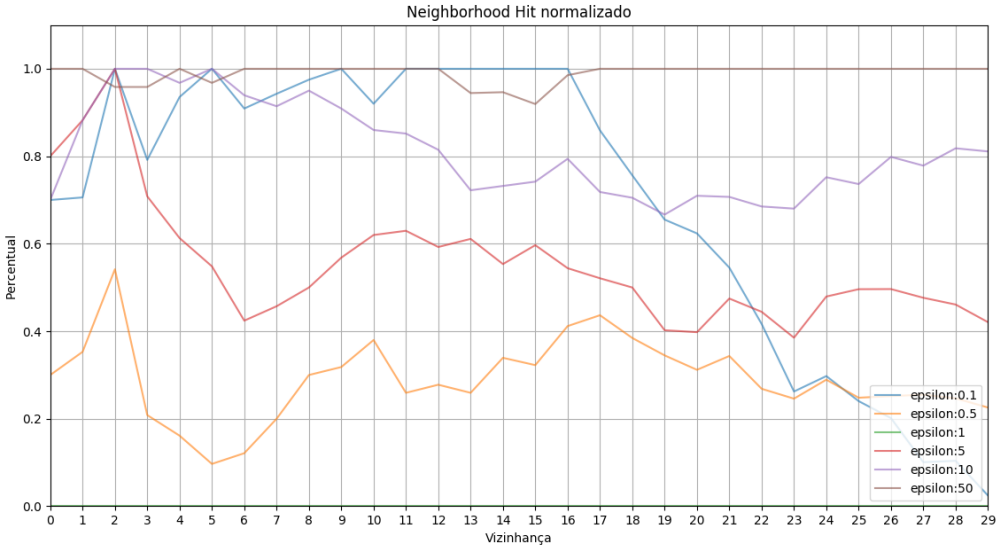


Figura 4.8: *Neighborhood Hit* normalizado da base de dados Iris sobre vários ϵ (eps) diferentes.

7. O objetivo da normalização dos gráficos das métricas *Neighborhood Preservation* e *Neighborhood Hit* é avaliar como cada configuração performa especificamente em relação às outras configurações. Ou seja, ao invés de uma avaliação global, avaliar as configurações relativamente.

Algoritmo 7: Algoritmo do processo de seleção do melhor valor de ϵ

1) Entre com a matriz $M_{n \times k}$ onde n é o número de *epsilons* avaliados e k é o número de vizinhanças avaliadas pela técnica *Neighborhood Preservation* ou *Neighborhood Hit*.
2) Normalize os valores cada coluna entre 0 e 1.
3) Some o valor de cada poli-linha.
4) Seleciona a poli-linha de maior valor da soma.

Conforme mencionado anteriormente, o parâmetro ϵ da técnica Mapas de Difusão depende das características do conjunto de dados a ser visualizado via Mapas de Difusão. Como o procedimento manual (exemplificado anteriormente) para determinar um valor para ϵ não é muito eficiente, a técnica BGH [49] foi utilizada para estimar um valor apropriado para ϵ. A Tabela 4.1 apresenta os valores ϵ determinados para cada conjunto de dados em que, em cada um, selecionou-se o melhor resultado nas métricas *Neighborhood Preservation* e *Neighborhood Hit*.

Tabela 4.1: Estimativa do ϵ para cada base de dados.

	BGH	*Neighborhood Preservation*	*Neighborhood Hit*
Curva S	0.0625	50	0.1
Círculos	0.0039	0.1	0.1
Corel	0.0156	0.1	0.1
Íris	0.125	0.5	50
20 News Groups	0.25	0.1	0.1

4.3 Visualização dos dados

Para cada conjunto de dados, foram gerados *layouts* utilizando as técnicas de visualização baseadas em projeções multi-dimensionais: Mapas de Difusão, ISOMAP e LLE Standard. O critério para seleção do ISOMAP e LLE Standard se deve ao emprego para propósitos de comparação em outros trabalhos na literatura [54] [15] e pelo fato de que suas implementações constam na biblioteca do *Python*, a *scikit-learn*.

Essas técnicas utilizam grafos completos como entrada e avaliar as distâncias entre todos os pares de pontos tem um custo computacional elevado. Para reduzir o custo de tal tarefa, considera-se que cada nó do grafo é somente conectado aos seus k-vizinhos mais próximos garantindo que o grafo seja conexo nessa configuração. O valor selecionado para os três algoritmos é $k = 100$ por manter os grafos conexos em todas as bases de dados avaliadas.

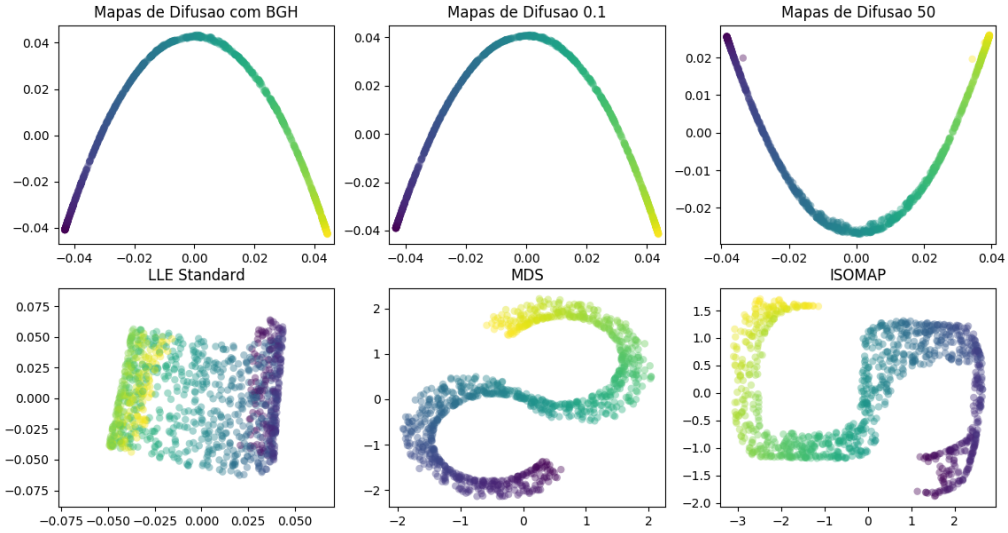


Figura 4.9: Projeção em duas dimensões da curva S aplicado sobre os algoritmos Mapas de Difusão, LLE Standard, MDS e ISOMAP.

No *layout* gerado referente ao conjunto de dados "Curva S" (Figura 4.9), é perceptível que em todas as configurações do Mapas de Difusão o formato das curvas da forma de "S" original não é preservado. Porém, verifica-se um padrão peculiar nos pontos da estrutura geométrica, interpretado pela transição das cores que representam a classe de cada ponto. A projeção baseada em LLE Standard realiza uma planificação da estrutura original com sobreposição nas extremidades, enquanto mantém as relações de vizinhança presentes nos dados originais. Os *layouts* produzidos pelas projeções MDS e ISOMAP são semelhantes, pois mantém a geometria da forma "S", ou seja, posiciona os pontos de acordo com o formato dessa estrutura geométrica.

Em relação ao conjunto de dados "Círculos" (Figura 4.10), o *layout* gerado pela visualização baseada em Mapas de Difusão, cujos parâmetros são obtidos utilizado a técnica BGH, separa os pontos conforme as duas classes definidas. No entanto, nota-se que as estruturas geométricas dos dados, isto é, o aspecto circular das classes não é preservado. Por sua vez, no *layout* associado ao Mapas de Difusão com os parâmetros empiricamente ajustados, percebe-se que as formas circulares das duas classes círculos estão com suas fronteiras bem definidas, apesar de posuírem pequenas deformidades. O restante das projeções geraram *layouts* semelhantes praticamente indistinguíveis dos dados originais.

A Figura 4.11 apresenta os *layouts* obtidos utilizando o conjunto de dados Íris. Primeiramente, os *layouts* gerados pela técnica Mapas de Difusão com BGH e Mapas de Difusão com o valor $\epsilon = 0.5$ indicam relações de proximidade entre os pontos pertencentes às

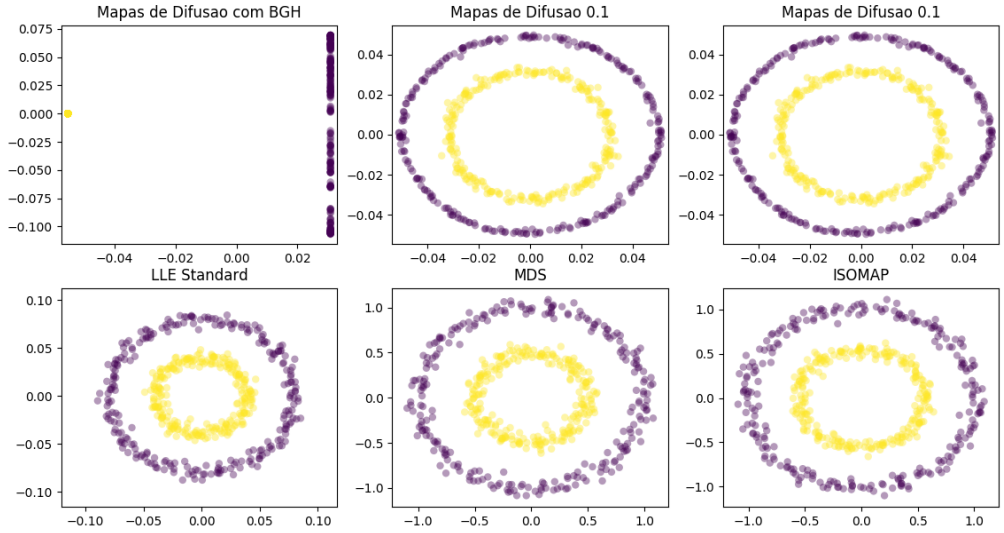


Figura 4.10: Projeção em duas dimensões dos Círculos Concêntricos aplicado sobre os algoritmos Mapas de Difusão, LLE Standard, MDS e ISOMAP.

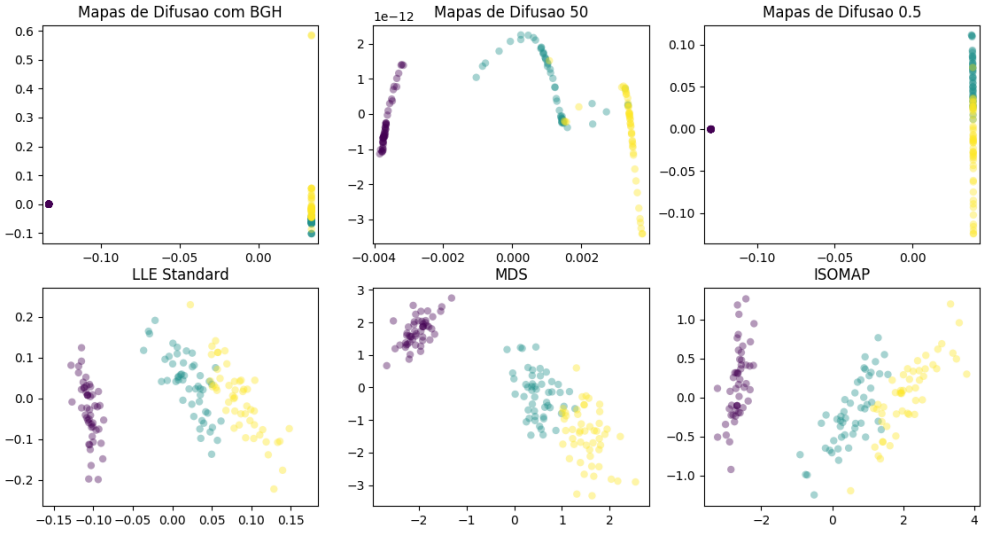


Figura 4.11: Projeção em duas dimensões da base de dados Íris aplicado sobre os algoritmos Mapas de Difusão, LLE Standard, MDS e ISOMAP.

classes amarela e azul, não sendo possível identificar separabilidade entre as classes. No *layout* gerado pela projeção Mapas de Difusão com $\epsilon = 50$, percebe-se um melhor espalhamento dos pontos e uma clara divisão das classes. As técnicas LLE Standard, MDS e ISOMAP geraram *layouts* semelhantes, pois os pontos foram distribuídos no espaço

reduzido de forma a identificar uma separação quase linear entre as classes de flor, com pouca sobreposição de pontos nas fronteiras.

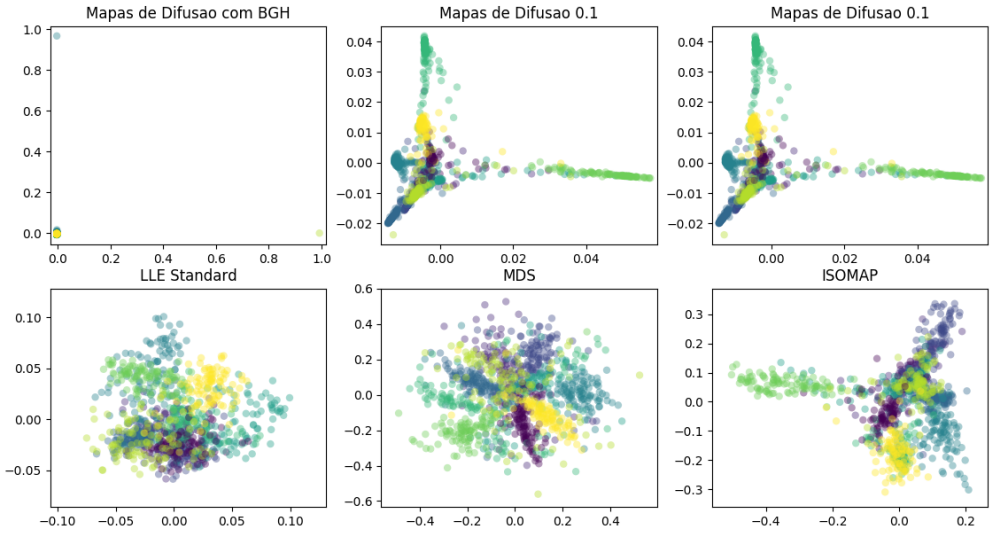


Figura 4.12: *Layouts* gerados a partir da base de dados Corel aplicado sobre as técnicas de visualização Mapas de Difusão, LLE Standard, MDS e ISOMAP.

A Figura 4.12 ilustra os *layouts* obtidos pelas projeções consideradas nos experimentos para a base de dados Corel. Pode-se notar que o *layout* Mapas de Difusão com BGH separou as classes em somente três regiões e com extrema proximidade e sobreposição de pontos. O *layout* associado ao Mapas de Difusão considerando $\epsilon = 0.1$ revela agrupamentos de pontos de acordo com as classes com estruturas geométricas em formato de curva. As representações gráficas obtidas pelas técnicas LLE Standard, MDS e ISOMAP posicionaram os pontos no espaço visual de maneira mais esparsa. No entanto, pode-se visualizar a formação de alguns grupos de pontos cujas instâncias associadas são da mesma classe.

Os *layouts* resultantes das visualizações da base de dados 20 News Groups são apresentados na Figura 4.13. Um aspecto comum observado nos referidos *layouts* é a impossibilidade de analisar as relações de similaridade e de vizinhança entre os textos, uma vez que não há formação de grupos de pontos associado às categorias do conjunto de dados. Os *layouts* correspondentes às configurações BGH e empírica da técnica Mapas de Difusão apresentam diferentes densidades no posicionamento de pontos no espaço visual. O *layout* obtido pela técnica LLE Standard distribuiu os pontos em uma geometria de característica espicular, mas com uma região de alta densidade de pontos no centro. O *layout* gerado pela visualização baseada em MDS posicionou os pontos conforme uma ge-

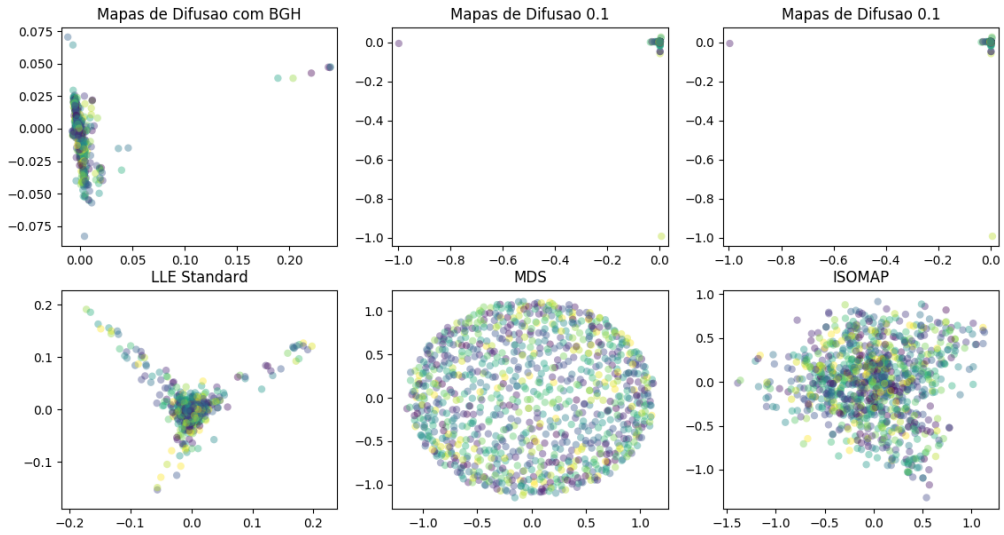


Figura 4.13: *Layouts* obtidos pelas técnicas de projeção Mapas de Difusão, LLE Standard, MDS e ISOMAP utilizando o conjunto de dados *20 News Groups* .

ometria circular, enquanto que a técnica ISOMAP produziu um *layout* que não apresenta nenhuma estrutura geométrica implícita.

4.4 Neighborhood Hit e Neighborhood Preservation

A Figura 4.14 ilustra a performance da métrica *Neighborhood Preservation* ao avaliar as técnicas de projeção multi-dimensional Mapas de Difusão (três configurações), ISOMAP, MDS e LLE Standard utilizando a base de dados Curva S. Os resultados indicam que a técnica ISOMAP preservou melhor as relações locais de similaridade entre os dados em relação às técnicas MDS e LLE Standard, de acordo com o padrão das respectivas curvas no gráfico. É possível observar que a técnica Mapas de Difusão não preservou satisfatoriamente as relações de vizinhança, obtendo-se o pior desempenho em relação às outras projeções. As projeções consideradas nesse experimento relativas à base de dados Curva S não foram avaliadas pela métrica *Neighborhood Hit*, pois essa métrica não se aplica à essa base uma vez que não há classes associadas aos dados.

A avaliação da qualidade das projeções de acordo com a métrica *Neighborhood Preservation* e utilizando a base de dados Círculos é apresentada na Figura 4.15. Pode-se verificar que as técnicas preservaram as vizinhanças dos dados de maneira similar, o que está de acordo com a estrutura circular dos pontos nos respectivos *layouts*. Somente a

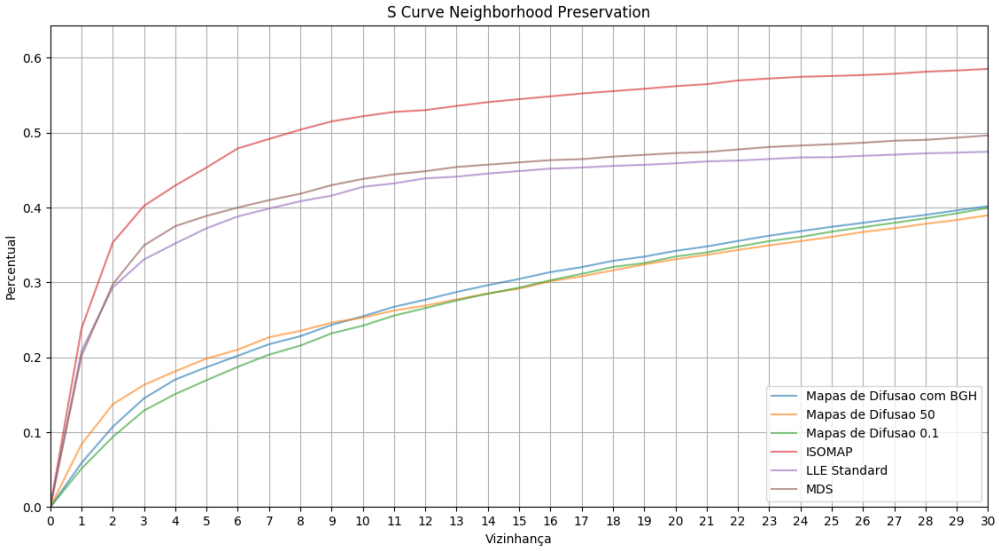


Figura 4.14: Comparação da *Neighborhood Preservation* da base de dados da curva S aplicado sobre os algoritmos ISOMAP, MDS, LLE Standard e Mapas de Difusão.

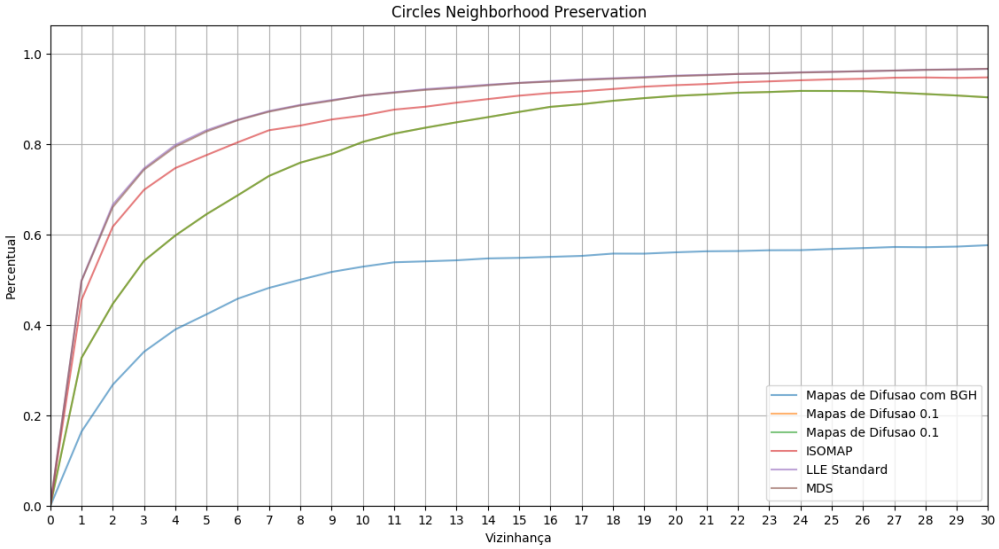


Figura 4.15: Comparação da *Neighborhood Preservation* da base de dados de Círculos Concêntricos aplicado sobre os algoritmos ISOMAP, MDS, LLE Standard e Mapas de Difusão onde a curva MDS sobrepõe à LLE Standard .

técnica Mapas de Difusão com BGH obteve uma baixa performance pois não preservou a estrutura circular dos dados na representação gráfica.

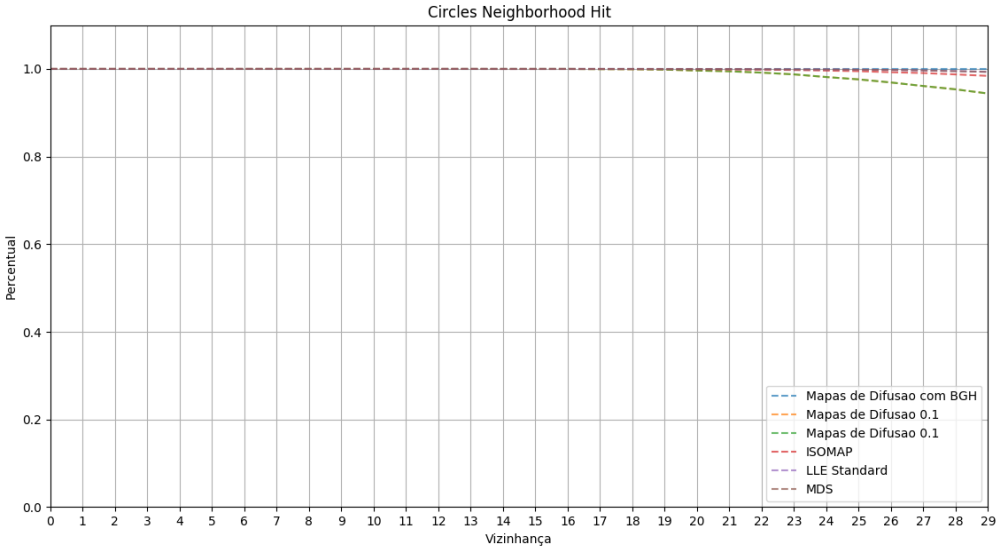


Figura 4.16: Comparação da *Neighborhood Hit* da base de dados de Círculos Concêntricos aplicado sobre os algoritmos ISOMAP, MDS, LLE Standard e Mapas de Difusão.

Em relação à avaliação por meio da técnica *Neighborhood Hit* da base de dados Círculos, mostrada na Figura 4.16, todas as projeções obtiveram preservação de vizinhança próximos de 100%, considerando até os 30 vizinhos mais próximos. Tal fato pode ser explicado pela fronteira entre as classes ser significativamente grande em todos os *layouts*.

Na Figura 4.17, apresentam-se os resultados da métrica *Neighborhood Preservation* para a base de dados Íris. A análise do gráfico evidencia performances similares entre os algoritmos ISOMAP e MDS que contém os melhores resultados em relação às demais. As variações da técnica Mapas de Difusão não apresentam grandes diferenças entre si em conjunto com a técnica LLE Standard. O gráfico associado com a métrica *Neighborhood Hit*, ilustrado na Figura 4.18, mostrou que as técnicas obtiveram uma boa proximidade das classes, em que somente a técnica LLE Standard não apresenta boa preservação para grandes vizinhanças em relação às as outras técnicas.

A avaliação da qualidade das projeções na base de dados Corel por meio da métrica *Neighborhood Preservation* pode ser verificada na Figura 4.19. É importante notar que todas as estimativas se encontram abaixo dos 40% de preservação com a técnica Mapas de Difusão de $\epsilon = 0.1$ tendo o melhor resultado seguida das técnicas MDS, Mapas de Difusão com BGH e LLE Standard, respectivamente.

Os resultado da métrica *Neighborhood Hit*, ilustrado na Figura 4.20, apresenta uma superioridade da classe do Mapas de Difusão com o parâmetro $\epsilon = 0.1$ se mantendo em 70% em todas as vizinhanças avaliadas assim mostrando uma boa separação de classes

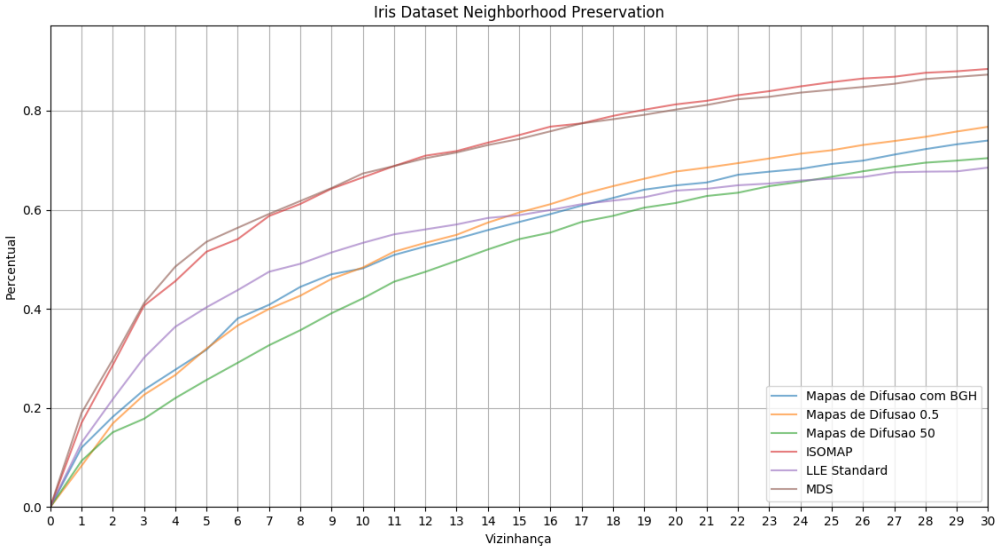


Figura 4.17: Comparação da *Neighborhood Preservation* da base de dados Íris aplicado sobre os algoritmos ISOMAP, MDS, LLE Standard e Mapas de Difusão.

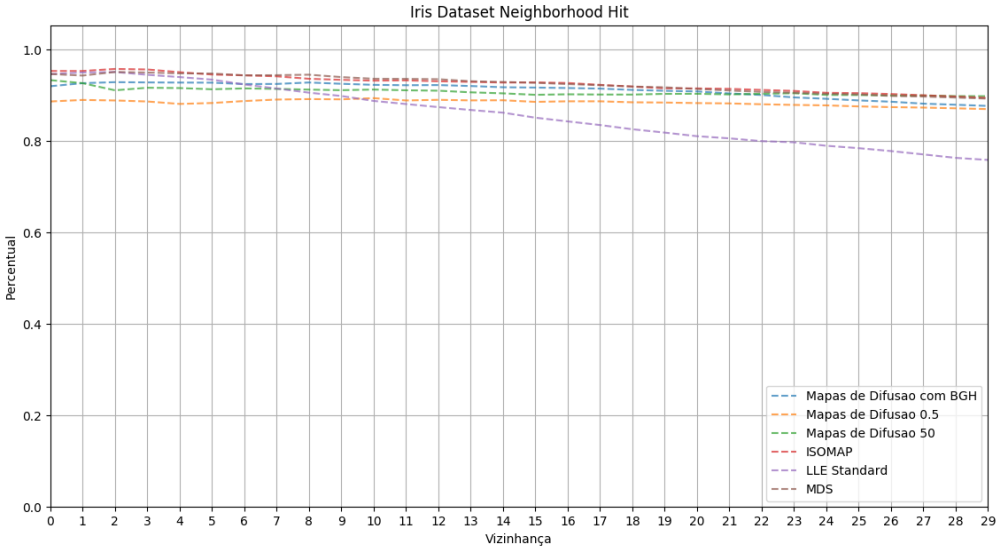


Figura 4.18: Comparação do *Neighborhood Hit* da base de dados Íris aplicado sobre os algoritmos ISOMAP, MDS, LLE Standard e Mapas de Difusão.

no espaço reduzido. As avaliações Mapas de Difusão com BGH, ISOMAP e MDS tendem a 55% quanto maior a vizinhança, mostrando pouca distinção em seus desempenhos em relação à separação de classes, apesar dos *layouts* serem distintos estruturalmente. A

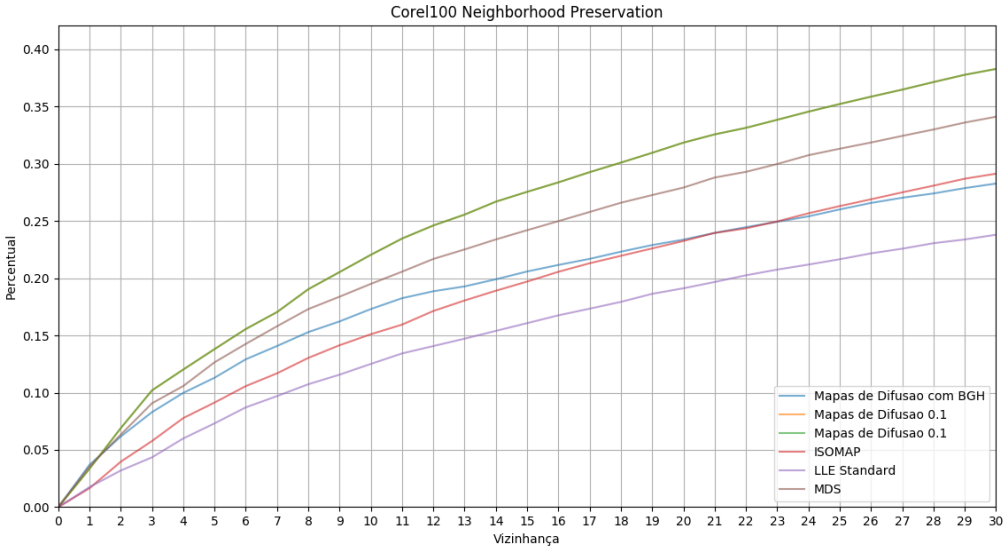


Figura 4.19: Comparação da *Neighborhood Preservation* da base de dados Corel aplicado aplicado sobre os algoritmos ISOMAP, MDS, LLE Standard e Mapas de Difusão.

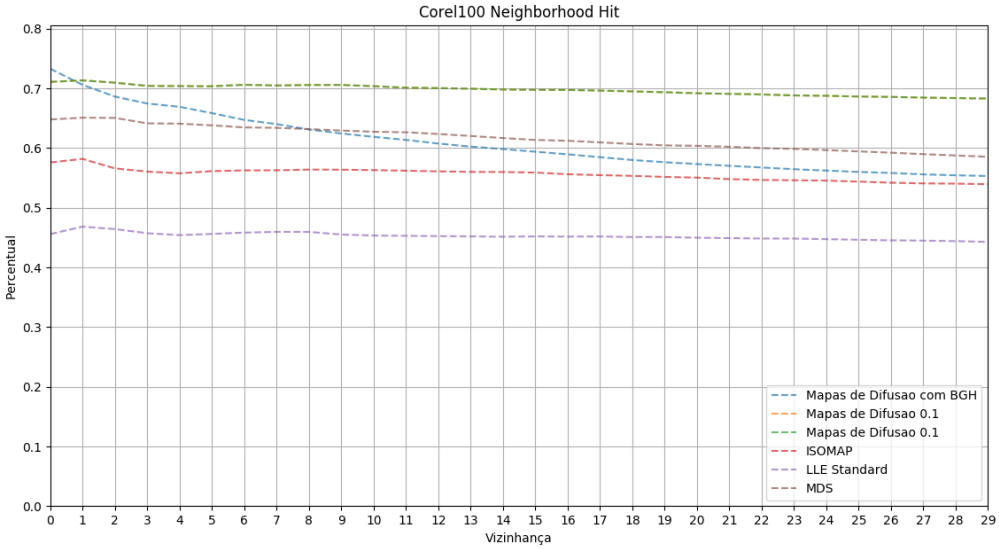


Figura 4.20: Comparação da *Neighborhood Hit* da base de dados Corel aplicado sobre os algoritmos ISOMAP, MDS, LLE Standard e Mapas de Difusão.

técnica LLE Standard se estagna em 45% confirmado pela maior sobreposição de classes em seu *layout*.

A Figura 4.21 apresenta a performance da avaliação das projeções utilizando a métrica *Neighborhood Preservation* e a base de dados *20 News Groups*. O gráfico indica que todas

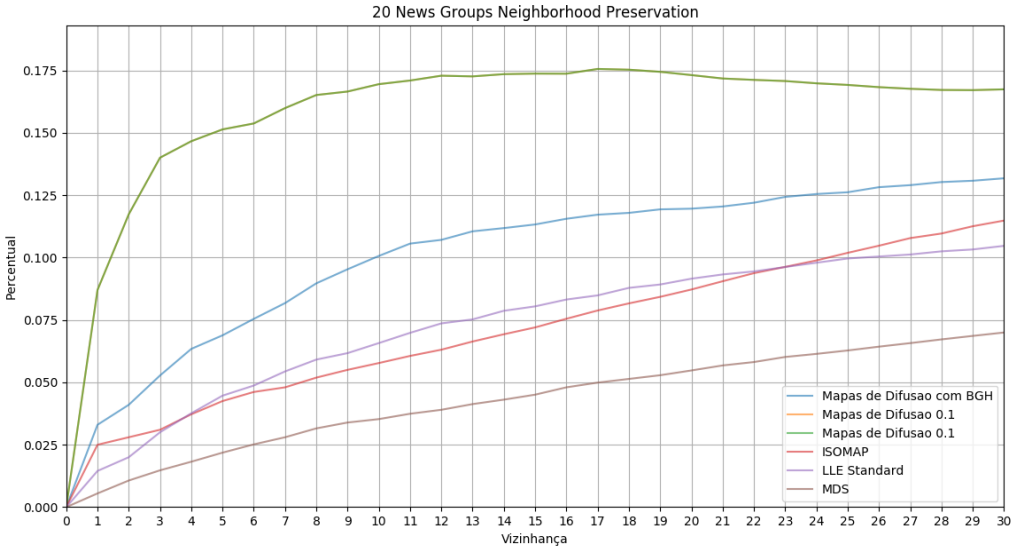


Figura 4.21: Comparação da *Neighborhood Preservation* da base de dados 20 News Groups aplicado sobre os algoritmos ISOMAP, MDS, LLE Standard e Mapas de Difusão.

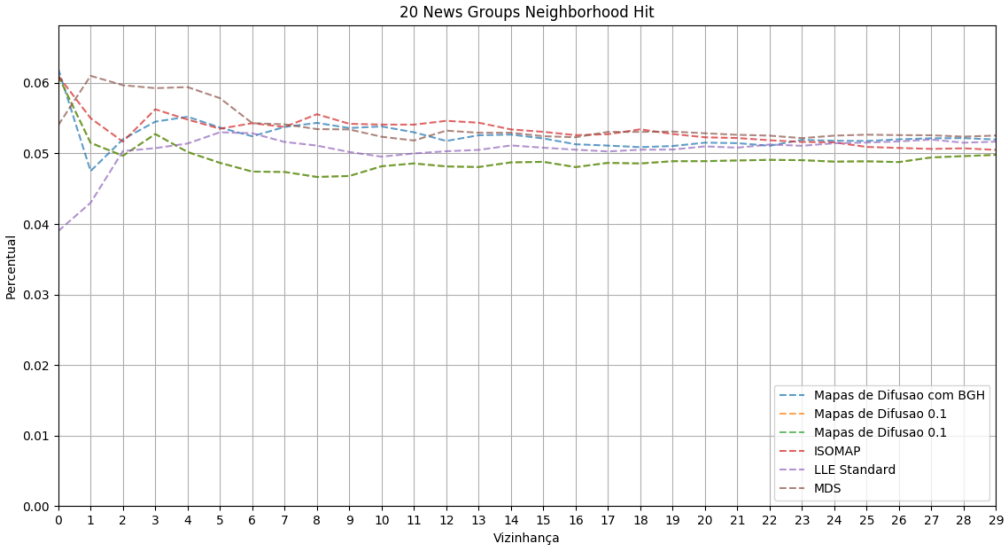


Figura 4.22: Comparação da *Neighborhood Hit* da base de dados 20 News Groups aplicado sobre os algoritmos ISOMAP, MDS, LLE Standard e Mapas de Difusão.

as abordagens se mantiveram próximas ou abaixo dos 15% de preservação, mostrando que os dados possuem uma estrutura complexa de ser mapeada por tais métodos. A técnica de Mapas de Difusão considerando $\epsilon = 0.1$ obteve o melhor desempenho em relação às demais técnicas de projeção multi-dimensional. Além disso, a métrica *Neighborhood Hit* mostrou

que todas as projeções obtiveram separações de classes com padrão regular variando-se o tamanho da vizinhança.

4.5 Discussão

O uso do BGH associado ao algoritmo Mapas de Difusão se tornou relevante por retirar o processo manual e empírico de estimar um valor apropriado para a variável ϵ. De acordo com a métrica *Neighborhood Preservation*, a visualização baseada em Mapas de Difusão não obteve a melhor performance nos conjuntos de dados considerados nos experimentos, porém a preservação das relações locais de similaridade nos conjuntos de dados reais foram próximas às técnicas ISOMAP, MDS e LLE Standard. Com relação aos resultados obtidos pela métrica *Neighborhood Hit*, pode-se analisar que a performance da técnica Mapas de Difusão foi similar em relação às outras técnicas consideradas em todas as bases de dados.

Os *layouts* gerados pela técnica Mapas de Difusão com BGH ao visualizar os conjuntos de dados *Círculos, Íris, Corel e 20 News Groups* muitas vezes não aproveitam adequadamente o espaço visual, dificultando tarefas de interpretação de padrões e relações de similaridade entre as instâncias de dados. Tal fato leva ao uso de recursos de interação com o *layout* para manipular a escala, enfatizando determinadas regiões ou áreas. Outro aspecto observado nos *layouts* gerados foi a sobreposição de pontos no espaço visual, sendo uma limitação de visualizações baseadas no posicionamento de pontos já identificadas em outros trabalhos na literatura [52]. Tal limitação pode ser minimizada ao ajustar empiricamente o valor ϵ, como puderam ser observados nos resultados da aplicação da técnica Mapas de Difusão nos conjuntos de dados Círculos, Íris e COREL.

Capítulo 5

Conclusão

5.1 Considerações Finais

Mapas de Difusão é uma técnica de redução de dimensionalidade ainda pouco explorada na literatura, que foi empregada em algumas tarefas como segmentação e redução de dimensionalidade. Tal fato motivou o estudo do Mapas de Difusão como uma técnica de visualização baseada em projeção multi-dimensional devido às suas características probabilísticas e espectrais. A técnica para esse fim aplica a teoria de caminhos aleatórios sobre um grafo probabilístico de transição que modela os dados e, a partir de suas coordenadas de difusão, determina um espaço de dimensões reduzidas para a projeção.

O método de visualização proposto é composto pelas seguintes etapas: pré-processamento, redução de dimensionalidade e renderização. O pré-processamento prepara os dados realizando sua formatação, limpeza e ajustes para a etapa seguinte. A redução de dimensionalidade realiza uma projeção dos dados para um espaço representativo com menos dimensões do que o espaço original e, por fim, a renderização considera as duas principais dimensões do espaço reduzido para gerar uma representação gráfica (*layout*). Para avaliar a qualidade dos resultados, as métricas *Neighborhood Preservation* e *Neighborhood Hit* foram utilizadas para mensurar a preservação das relações de similaridade entre os dados perante diferentes configurações de vizinhança. Nos experimentos, as representações gráficas obtidas pela visualização baseada em Mapas de Difusão foram comparadas também em relação às técnicas MDS, ISOMAP e LLE Standard.

Como recomendado por vários trabalhos, a seleção do parâmetro ϵ, específico para cada conjunto de dados, é essencial para a sua performance. Foram selecionados empiricamente dois valores que obtiveram as melhores performances a partir de um conjunto de valores possíveis, avaliados de acordo com as métricas *Neighborhood Preservation* e *Neighborhood Hit*. Os experimentos também levaram em conta uma estratégia alternativa baseada na técnica BGH [49] para estimação automática do parâmetro ϵ.

A técnica Mapas de Difusão não obteve performance superior para bases de dados artificiais (Curva S e Círculos) considerando a métrica *Neighborhood Preservation* em relação às outras projeções multi-dimensionais. Já em relação à métrica *Neighborhood Hit*, a técnica Mapas de Difusão gerou *layouts* representativos para a base de dados Círculos, apesar de não preservar suas características geométricas originais. Em conjuntos de dados reais, ocorreram casos em que ao menos uma configuração de parâmetros da técnica Mapas de Difusão foi superior em relação a alguma das técnicas MDS, ISOMAP e LLE Standard com base nas duas métricas.

5.2 Limitações

A técnica Mapas de Difusão possui algumas limitações relacionadas ao ajuste do parâmetro ϵ, uma vez que os seus valores afetam diretamente a qualidade dos *layouts* gerados. O parâmetro ϵ deve ser ajustado de acordo com o conjunto de dados e é importante para a definição das relações de vizinhança entre as instâncias de dados. Com o uso de técnicas de seleção BGH, os resultados gerados obtêm bons resultados nas métricas *Neighborhood Preservation* e *Neighborhood Hit*. O *layout*, porém, produzido não é informativo. Como exemplo, temos o bom resultado nas métricas da base de dados *20 News Groups*, mas é impossível interpretar padrões relevantes enquanto outras técnicas mostraram um melhor aproveitamento no espaço reduzido. Isso mostra que as métricas *Neighborhood Preservation* e *Neighborhood Hit* não estão relacionadas a uma boa visualização de dados. Assim, é necessário ainda recorrer à estimação empírica do valor de ϵ para identificar um valor apropriado para o processo de visualização dos dados.

Em relação à qualidade dos *layouts* gerados, observou-se a alta proximidade de pontos em certas regiões, fenômeno conhecido como *visual cluttering*. No entanto, esta limitação é comum às visualizações baseadas no posicionamento de pontos no espaço visual, conforme reportado na literatura [52]. A sobreposição de pontos no *layout* ocorre devido ao alto grau de similaridade entre as instâncias associadas no espaço de alta dimensão.

5.3 Trabalhos Futuros

Como trabalhos futuros, sugere-se avaliar em maior profundidade a influência de diferentes modelos de *kernel* e as suas respectivas configurações em determinados tipos de dados. A pesquisa envolveu apenas o uso do *kernel* gaussiano por ser um modelo com boa capacidade de generalização na determinação da vizinhança das instâncias de dados.

Apesar de que as medidas *Neighborhood Preservation* e *Neighborhood Hit* tenham avaliado com sucesso a preservação das relações de similaridade e de classes na vizinhança

de pontos, outras estratégias podem ser consideradas para garantir uma visualização que melhor aproveite o espaço de projeção. Por exemplo, no trabalho de Maaten e Postma [46], as métricas de confiança e de continuidade foram empregadas para avaliar a qualidade de agrupamentos gerados pelos Mapas de Difusão.

O uso da técnica BGH [49] para estimar o parâmetro ϵ do Mapas de Difusão pode não ser ideal para o uso geral, pois o BGH assume a existência de estruturas geométricas implícitas de baixa dimensionalidade nos dados e também sendo suscetível à ruídos no processo de estimação do parâmetro ϵ [55].

Referências

[1] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *International Conference on Computer Vision (ICCV), 2011 IEEE*, pages 2564–2571. IEEE, 2011.

[2] Gary Overett and Lars Petersson. Large scale sign detection using hog feature variants. In *Intelligent Vehicles Symposium (IV), 2011*, pages 326–331. IEEE, 2011.

[3] Fernando Vieira Paulovich. *Mapeamento de dados multi-dimensionais-integrando mineração e visualização*. PhD thesis, Universidade de São Paulo, 2008.

[4] JJ Freeman. Experiments in discrimination and classification. *Pattern Recognition*, 1(3):207–218, 1969.

[5] Jorge Poco, Ronak Etemadpour, Fernando Vieira Paulovich, TV Long, Paul Rosenthal, Maria Cristina Ferreira de Oliveira, Lars Linsen, and Rosane Minghim. A framework for exploring multidimensional data with 3d projections. In *Computer Graphics Forum*, volume 30, pages 1111–1120. Wiley Online Library, 2011.

[6] J De la Porte, BM Herbst, W Hereman, and SJ van Der Walt. An introduction to diffusion maps. In *Proceedings of the 19th Symposium of the Pattern Recognition Association of South Africa (PRASA 2008), Cape Town, South Africa*, pages 15–25, 2008.

[7] Will J Schroeder, Bill Lorensen, and Ken Martin. *The visualization toolkit: an object-oriented approach to 3D graphics*. Kitware, 2004.

[8] Michael Friendly. A brief history of data visualization. In *Handbook of data visualization*, pages 15–56. Springer, 2008.

[9] Sándor Kromesch and Sándor Juhász. High dimensional data visualization. In *6th International Symposium of Hungarian Researchers on Computational Intelligence*, pages 1–12. Citeseer, 2005.

[10] Arthur Zimek, Erich Schubert, and Hans-Peter Kriegel. A survey on unsupervised outlier detection in high-dimensional numerical data. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 5(5):363–387, 2012.

[11] Martin Theus. High-dimensional data visualization. In *Handbook of data visualization*, pages 151–178. Springer, 2008.

[12] Fernando V Paulovich and Rosane Minghim. Hipp: A novel hierarchical point placement strategy and its application to the exploration of document collections. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1229–1236, 2008.

[13] Eduardo Tejada, Rosane Minghim, and Luis Gustavo Nonato. On improved projection techniques to support visual exploration of multi-dimensional data sets. *Information Visualization*, 2(4):218–231, 2003.

[14] Ana M Cuadros, Fernando V Paulovich, Rosane Minghim, and Guilherme P Telles. Point placement by phylogenetic trees and its application to visual analysis of document collections. In *IEEE Symposium on Visual Analytics Science and Technology (VAST 2007)*, pages 99–106. IEEE, 2007.

[15] Elisa Portes dos Santos Amorim, Emilio Vital Brazil, Joel Daniels, Paulo Joia, Luis Gustavo Nonato, and Mario Costa Sousa. ilamp: Exploring high-dimensional spacing through backward multidimensional projection. In *Visual Analytics Science and Technology (VAST), 2012 IEEE*, pages 53–62. IEEE, 2012.

[16] Karl Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901.

[17] Luis Gustavo Nonato and Michael Aupetit. Multidimensional projection for visual analytics: Linking techniques with distortions, tasks, and layout enrichment. *IEEE Transactions on Visualization and Computer Graphics*, 2018.

[18] Ronald R. Coifman and Stéphane Lafon. Diffusion maps. *Applied and Computational Harmonic Analysis*, 21(1):5 – 30, 2006. Special Issue: Diffusion Maps and Wavelets.

[19] Leandro Aparecido Simal Moreira. Campos de vetores planares. Master's thesis, PUC-Rio, 2010.

[20] Rui Xu, Steven Damelin, and Donald C Wunsch. Applications of diffusion maps in gene expression data-based cancer diagnosis analysis. In *2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 4613–4616. IEEE, 2007.

[21] Gang Qian, Shamik Sural, Yuelong Gu, and Sakti Pramanik. Similarity between euclidean and cosine angle distance for nearest neighbor queries. In *Proceedings of the 2004 ACM Symposium on Applied Computing*, pages 1232–1237. ACM, 2004.

[22] V Srividhya and R Anitha. Evaluating preprocessing techniques in text categorization. *International Journal of Computer Science and Application*, 47(11):49–51, 2010.

[23] Florian Heimerl, Steffen Lohmann, Simon Lange, and Thomas Ertl. Word cloud explorer: Text analytics based on word clouds. In *System Sciences (HICSS), 2014 47th Hawaii International Conference on*, pages 1833–1842. IEEE, 2014.

[24] California wildfires: Death toll reaches grim milestone. BBC News, 2018. Acessado na data 20/11/2018.

[25] E. Kasutani and A. Yamada. The mpeg-7 color layout descriptor: a compact image feature description for high-speed image/video segment retrieval. In *Proceedings 2001 International Conference on Image Processing (Cat. No.01CH37205)*, volume 1, pages 674–677 vol.1, Oct 2001.

[26] David G Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[27] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer Vision and Image Understanding*, 110(3):346–359, 2008.

[28] Jun Yang, Yu-Gang Jiang, Alexander G Hauptmann, and Chong-Wah Ngo. Evaluating bag-of-visual-words representations in scene classification. In *International Workshop on Multimedia Information Retrieval*, pages 197–206. ACM, 2007.

[29] Nahum Gershon and Ward Page. What storytelling can do for information visualization. *Communications of the ACM*, 44(8):31–37, 2001.

[30] Matthew O Ward, Georges Grinstein, and Daniel Keim. *Interactive data visualization: foundations, techniques, and applications*. AK Peters/CRC Press, 2015.

[31] A Endert, W Ribarsky, Cagatay Turkay, B.L. Wong, Ian Nabney, Ignacio Díaz Blanco, and Fabrice Rossi. The state of the art in integrating machine learning into visual analytics: Integrating machine learning into visual analytics. *Computer Graphics Forum*, 03 2017.

[32] Michael Friendly and Daniel Denis. The early origins and development of the scatterplot. *Journal of the History of the Behavioral Sciences*, 41(2):103–130, 2005.

[33] M. Ocagne. *Coordonnées parallèles & axiales: méthode de transformation géométrique et procédé nouveau de calcul graphique déduits de la considération des coordonnées parallèles*. Gauthier-Villars, 1885.

[34] Florian Wickelmaier. An introduction to mds. *Sound Quality Research Unit, Aalborg University, Denmark*, 46(5), 2003.

[35] Joshua B. Tenenbaum, Vin de Silva, and John C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

[36] Peter D Eades. A heuristic for graph drawing. *Congressus Numerantium*, 42:149–160, 1984.

[37] John P Cunningham and Zoubin Ghahramani. Linear dimensionality reduction: Survey, insights, and generalizations. *The Journal of Machine Learning Research*, 16(1):2859–2900, 2015.

[38] Ian Jolliffe. Principal component analysis. In *Series in Statistics*, pages 1–487. Springer, 2002.

[39] Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52, 1987.

[40] Lawrence K Saul and Sam T Roweis. An introduction to locally linear embedding. *unpublished. Available at: http://www. cs. toronto. edu/˜ roweis/lle/publications. html*, 2000.

[41] David G Lowe. Object recognition from local scale-invariant features. In *Computer vision, 1999. IEEE International Conference on*, volume 2, pages 1150–1157. Ieee, 1999.

[42] Martin Szummer and Tommi Jaakkola. Partially labeled classification with markov random walks. In *Advances in neural information processing systems*, pages 945–952, 2002.

[43] Ronny Lempel and Shlomo Moran. The stochastic approach for link-structure analysis (salsa) and the tkc effect1. *Computer Networks*, 33(1-6):387–401, 2000.

[44] Stijn Dongen. Performance criteria for graph clustering and markov cluster experiments. Technical report, Amsterdam, The Netherlands, The Netherlands, 2000.

[45] S. Yan, D. Xu, B. Zhang, H. Zhang, Q. Yang, and S. Lin. Graph embedding and extensions: A general framework for dimensionality reduction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(1):40–51, Jan 2007.

[46] Laurens Van Der Maaten, Eric Postma, and Jaap Van den Herik. Dimensionality reduction: a comparative review. *Journal of Machine Learning Research*, 10:66–71, 2009.

[47] N. Rajpoot, M. Arif, and A. H. Bhalerao. Unsupervised learning of shape manifolds. In *Proc. BMVC*, pages 90.1–90.10, 2007.

[48] Gail A Carpenter, Stephen Grossberg, and David B Rosen. Fuzzy art: Fast stable learning and categorization of analog patterns by an adaptive resonance system. *Neural networks*, 4(6):759–771, 1991.

[49] Tyrus Berry, Dimitrios Giannakis, and John Harlim. Nonparametric forecasting of low-dimensional dynamical systems. *Physical Review E*, 91(3):032915, 2015.

[50] Sam T Roweis and Lawrence K Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

[51] Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122, 2013.

[52] J. G. Paiva, L. Florian, H. Pedrini, G. Telles, and R. Minghim. Improved similarity trees and their application to visual data classification. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2459–2468, Dec 2011.

[53] Ken Lang. Newsweeder: Learning to filter netnews. In *Proceedings of the Twelfth International Conference on Machine Learning*, pages 331–339, 1995.

[54] Fernando V Paulovich, Luis G Nonato, Rosane Minghim, and Haim Levkowitz. Least square projection: A fast high-precision multidimensional projection technique and its application to document mapping. *IEEE Transactions on Visualization and Computer Graphics*, 14(3):564–575, 2008.

[55] Zhizhen Zhao and Dimitrios Giannakis. Analog forecasting with dynamics-adapted kernels. *Nonlinearity*, 29(9):2888, 2016.