Mapeamento de dados multi-dimensionais – integrando mineração e visualização

Fernando Vieira Paulovich

Mapeamento de dados multi-dimensionais – integrando mineração e visualização

Fernando Vieira Paulovich

Orientadora: ***Profa. Dra. Rosane Minghim***

Tese apresentada ao Instituto de Ciências Matemáticas e de Computação - ICMC-USP, como parte dos requisitos para obtenção do título de Doutor em Ciências - Ciências de Computação e Matemática Computacional.

USP – São Carlos
Setembro de 2008

À Débora, sem você esse trabalho
não teria acontecido.

Agradecimentos

a Deus por todas as bençãos a mim concedidas;

à minha amada e venerada esposa, que aceitou casar com alguém que estava começando o doutorado, decidindo viver monetariamente sustentada por uma bolsa (um dia te compro um chinelo novo). Eu nunca vou esquecer o que você tem feito por mim;

aos meus pais, os grandes responsáveis pelo que sou, queiram eles ou não, e às minhas irmãs Cristiane, Adriane e Fabiana, que sempre me ajudaram nunca reclamando se eu comia o último Bis da caixa (depois de ter comido os outros 19). Não agradeço meus cunhados porque isso não existe, vai soar falso, e as outras pessoas vão achar que estou agradecendo por mera formalidade;

à minha orientadora, Dra. Rosane Minghim, que me abriu as portas do ICMC-USP, acreditando na minha capacidade e esforço, nunca se negando a me ajudar na busca de meus objetivos pessoais. Saiba que em muitos aspectos você é um exemplo que vou seguir, porém numa versão mais calma;

à todos os outros professores e funcionários do ICMC-USP, pela competência e suporte nas horas em que mais precisei, especialmente à Dra. Maria Cristina Ferreira de Oliveira, ao Dr. Luis Gustavo Nonato, ao Dr. Alneu de Andrade Lopes, ao Dr. Guilherme Pimentel Telles (o desertor), ao Dr. Antonio Castelo Filho, e ao Dr. João do Espirito Santo Batista Neto;

aos meus colegas de laboratório pela ajuda indispensável nesses anos de doutorado, mesmo que não intencionalmente, esse trabalho tem muito de vocês;

aos meus colegas da TU-Delft e da *Treparel Information Solutions*, especialmente aos senhores Dr. Anton Heijs, Dr. Charl P. Botha e Dr. Frits H. Post que tão bem me receberam durante meu estágio de doutoramento em Delft, Holanda (o lugar mais lindo do mundo);

à banca que gentilmente aceitou o convite para essa defesa;

e à FAPESP e Capes pelo apoio financeiro nesses anos.

Resumo

As técnicas de projeção ou posicionamento de pontos no plano, que servem para mapear dados multi-dimensionais em espaços visuais, sempre despertaram grande interesse da comunidade de visualização e análise de dados por representarem uma forma útil de exploração baseada em relações de similaridade e correlação. Apesar disso, muitos problemas ainda são encontrados em tais técnicas, limitando suas aplicações. Em especial, as técnicas de projeção multi-dimensional de maior qualidade têm custo computacional proibitivo para grandes conjuntos de dados. Adicionalmente, problemas referentes à escalabilidade visual, isto é, à capacidade da metáfora visual empregada de representar dados de forma compacta e amigável, são recorrentes. Esta tese trata o problema da projeção multi-dimensional de vários pontos de vista, propondo técnicas que resolvem, até certo ponto, cada um dos problemas verificados. Também é fato que a complexidade e o tamanho dos conjuntos de dados indicam que a visualização deve trabalhar em conjunto com técnicas de mineração, tanto embutidas no processo de mapeamento, como por meio de ferramentas auxiliares de interpretação. Nesta tese incorporamos alguns aspectos de mineração integrados ao processo de visualização multi-dimensional, principalmente na aplicação de projeções para visualização de coleções de documentos, propondo uma estratégia de extração de tópicos. Como suporte ao desenvolvimento e teste dessas técnicas, foram criados diferentes sistemas de software. O principal inclui as técnicas desenvolvidas e muitas das técnicas clássicas de projeção, podendo ser usado para exploração de conjuntos de dados multi-dimensionais em geral, com funcionalidade adicional para mapeamento de coleções de documentos. Como principal contribuição desta tese propomos um entendimento mais profundo dos problemas encontrados nas técnicas de projeção vigentes e o desenvolvimento de técnicas de projeção (ou mapeamento) que são rápidas, tratam adequadamente a formação visual de grupos de dados altamente similares, separam satisfatoriamente esses grupos no layout, e permitem a exploração dos dados em vários níveis de detalhe.

Palavras-chave: Visualização de informação; Mineração visual de dados; Projeção multi-dimensional; Mapa de documentos.

Abstract

Projection or point placement techniques, useful for mapping multidimensional data into visual spaces, have always risen interest in the visualization and data analysis communities because they can support data exploration based on similarity or correlation relations. Regardless of that interest, various problems arise when dealing with such techniques, impairing their widespread application. In particularly the projections that yield highest quality layouts have prohibitive computational cost for large data sets. Additionally, there are issues regarding visual scalability, i.e., the capability of visually fit the individual points in the exploration space as the data set grows large. This thesis treats the problems of projections from various perspectives, presenting novel techniques that solve, to certain extent, several of the verified problems. It is also a fact that size and complexity of data sets suggest the integration of data mining capabilities into the visualization pipeline, both during the mapping process and as a tools to extract additional information after the data have been layed out. This thesis also add some aspects of mining to the multidimensional visualization process, mainly for the particular application of analysis of document collections, proposing and implementing an approach for topic extraction. As supporting tools for testing these techniques and comparing them to existing ones different software systems were written. The main one includes the techniques developed here as well as several of the classical projection and dimensional reduction techniques, and can be used for exploring various kinds of data sets, with addition functionality to support the mapping of document collections. This thesis contributes to the understanding of the projection or mapping problem and develops new techniques that are fast, treat adequately the visual formation of groups of highly related data items, separate those groups properly and allow exploration of data in various levels of detail.

Keywords: Information visualization; Visual data mining; Multidimensional projection; Documents map.

Sumário

Lista de Figuras

Lista de Algoritmos

Lista de Abreviaturas e Siglas

ALS	*Anchored Least Stress*
CBR	*Case-Based Reasoning*
CCA	*Curvilinear Component Analysis*
CPs	Componentes Principais
FA	*Factor Analysis*
FDP	*Force-Directed Placement*
FS	*Force Scheme*
IDH	Índice de Desenvolvimento Humano
IDMAP	*Interactive Document Map*
ILP	*Inductive Logic Programming*
IR	*Information Retrieval*
ISOMAP	*Isometric Feature Mapping*
KDD	*Knowledge Discovery in Databases*
LLE	*Local Linear Embedding*
LSI	*Latent Semantic Index*
LSP	*Least Square Projection*
LSS	*Least Squares Scaling*
MDS	*Multidimensional Scaling*
MM	*Modelos baseados em Molas*
NCD	*Normalized Compression Distance*
ONU	Organização das Nações Unidas
PCA	*Principal Components Analysis*
PP	*Projection Pursuit*
ProjClus	*Projection by Clustering*
SA	*Simulated Annealing*
SM	*Sammon's Mapping*
SOM	*Self-Organizing Maps*
SON	*Sonification*
SVD	*Singular Value Decomposition*

Lista de Artigos Publicados

1. Paulovich, F. V.; Minghim, R. Text Map Explorer: a tool to create and explore document maps. In: *Proceedings of 10th International Conference on Information Visualisation 2006 (IV06)*, Londres. IEEE Computer Society Press, 2006, p. 245–251.

2. Paulovich, F. V.; Nonato, L. G.; Minghim, R.; Levkowitz, H. Visual mapping of text collections through a fast high precision projection technique. In: *Proceedings of 10th International Conference on Information Visualisation 2006 (IV06)*, Londres. IEEE Computer Society Press, 2006, p. 282–290.

3. Paulovich, F. V.; Oliveira, M. C. F.; Minghim, R. The Projection Explorer: a flexible tool for projection-based multidimensional visualization. *In: Proceedings of XX Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI 2007)*, Belo Horizonte. IEEE Computer Society Press, 2007. p. 27–36.

4. Paulovich, F. V.; Nonato, L. G.; Minghim, R.; Levkowitz, H. Least Square Projection: a fast high precision multidimensional projection technique and its application to document mapping. *IEEE Transactions on Visualization and Computer Graphics*, v. 14, p. 564–575, 2008.

5. Paulovich, F. V.; Pinho, R.; Botha, C. P.; Heijs, A.; Minghim, R. PEx-WEB: content-based visualization of web search results. In: *Proceedings of 12th International Conference on Information Visualisation 2008 (IV08)*, Londres. IEEE Computer Society Press, 2008, p. 208–214.

6. Paulovich, F. V.; Minghim, R. HiPP: a novel hierarchical point placement strategy and its application to the exploration of document collections. *IEEE Transactions on Visualization and Computer Graphics (Proceedings of Information Visualization 2008)*, vol. 14, no. 6, pp. 1229–1236, 2008.

7. Minghim, R.; Paulovich, F. V.; Lopes, A. A. Content-based text mapping using multidimensional projections for exploration of document collections. *In: Proceedings of Visualization and Data Analysis*, San Jose/California. IS&T/SPIE Symposium on Electronic Imaging, 2006. v. 6060. p. S1–S12.

8. Lopes, A. A.; Minghim, R.; Melo, V.; Paulovich, F. V. Mapping texts through dimensionality reduction and visualization techniques for interactive exploration of document collections. *In: Proceedings of Visualization and Data Analysis*, San Jose/California. IS&T/SPIE Symposium on Electronic Imaging, 2006. v. 6060. p. T1–T12.

9. Telles, G. P.; Minghim, R.; Paulovich, F. V. Normalized compression distance for visual analysis of document collections. *Computers & Graphics*, v. 31, p. 327–337, 2007.

10. Lopes, A. A.; Pinho, R.; Paulovich, F. V.; Minghim, R. Visual text mining using association rules. *Computers & Graphics*, v. 31, p. 316–326, 2007.

11. Muniz, M.; Paulovich, F. V.; Minghim, R.; Infante, K.; Muniz, F.; Vieira, R.; Aluisio, S. Taming the tiger topic: a xces compliant corpus portal to generate sub corpus based on automatic text topic identification. In: *Proceedings of Corpus Linguistics 2007 (CL 2007)*, Birmingham, 2007.

12. Cuadros, A. M.; Paulovich, F. V.; Minghim, R.; Telles, G. P. Point placement by phylogenetic trees and its application for visual analysis of document collections. In: *Proceedings of IEEE Symposium on Visual Analytics Science and Technology 2007 (VAST 2007)*, Sacramento, Califórnia. IEEE Computer Society Press, 2007. p. 99–106.

13. Alencar, A. B.; Andrade Filho, M. G.; Paulovich, F. V.; Minghim, R.; Oliveira, M. C. F. Mineração visual de séries temporais: um estudo de caso com séries de vazões de usinas hidrelétricas. In: *Proceedings of XVII Simpósio Brasileiro de Recursos Hídricos*, São Paulo, 2007. p. 1–20.

14. Alencar, A. B.; Paulovich, F. V.; Minghim, R.; Andrade Filho, M. G.; Oliveira, M. C. F. Similarity-based visualization of time series collections: an application to analysis of streamflows. In: *Proceedings of 12th International Conference on Information Visualisation 2008 (IV08)*, Londres. IEEE Computer Society Press, 2008. p. 280–286.

15. Eler, D. M.; Paulovich, F. V.; Oliveira, M. C. F.; Minghim, R. Coordinated and multiple views for visualizing text collections. In: *Proceedings of 12th International Conference on Information Visualisation 2008 (IV08)*, Londres. IEEE Computer Society Press, 2008. p. 246–251.

16. Eler, D. M.; Nakazaki, M.; Paulovich, F. V.; Santos, D. P.; Oliveira, M. C. F.; Batista Neto, J. E. S.; Minghim, R. Multidimensional visualization to support analysis of image collections. *In: Proceedings of XXI Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI 2008)*, Campo Grande. IEEE Computer Society Press, 2008. p. 289–296.

Tabela 1: Artigos publicados como resultado desse doutorado e pela cooperação com outros alunos e pesquisadores como parte de seus respectivos trabalhos.

Capítulo

1

Introdução

1.1 Contextualização

Com a diminuição do custo e a melhoria das tecnologias para armazenamento, distribuição e recuperação de dados, a quantidade de informação produzida, ou disseminada, tem crescido substancialmente tanto em volume quanto em complexidade. Um exemplo concreto desse aumento pode ser verificado no registro de certidões de nascimento nos Estados Unidos. Historicamente, certidões de nascimento continham de 7 a 15 campos de informação. Porém, com o tempo, o estado de Illinois, assim como outros estados, passou a coletar mais de 100 campos de informação (Sweeney, 2001).

Apesar desse aumento de coleta e armazenamento de informação parecer positivo, excesso de informação pode resultar em um efeito conhecido como "sobrecarga de informação". A sobrecarga de informação ocorre quando o volume de informação é tão grande que uma pessoa não é capaz de localizar e fazer uso do que é necessário (Christian et al., 2000). Assim, como bem apontado por Wurman (1989), "Uma das grandes ironias da era da informação é que conforme a tecnologia de distribuição da informação se torna mais sofisticada, a possibilidade de que nós possamos processar tudo se torna mais remota".

Apesar de muita informação não relevante poder ser filtrada, por exemplo, e-mails não desejados, a geração de informação útil ainda continua sendo muito maior do que a capacidade das ferramentas de interpretação vigentes. Dessa forma, apesar da grande capacidade de armazenamento e distribuição de informação, se métodos eficazes não forem desenvolvidos de apoio à sua interpretação, esse armazenamento acaba sendo pouco útil. Daí a necessidade de

métodos e ferramentas capazes de sintetizar esse aglomerado de dados para que seja possível interpretá-lo, apresentando-o de forma simples e amigável.

Dentre as possíveis áreas que estudam métodos para apoiar a interpretação de tal informação, duas têm despertado grande interesse da comunidade científica: a *Mineração de Dados* e a *Visualização de Informação*. A mineração de dados busca extrair padrões úteis de conjuntos de dados ou criar modelos de previsão (Tan et al., 2005), usualmente como parte de um processo mais genérico de extração de conhecimento denominado *Knowledge Discovery in Databases (KDD)* (Fayyad et al., 1996; Fayyad, 1998), que é o processo completo de converter dados não processados em informação útil. Já a visualização de informação é mais direcionada a apoiar esse processo por meio da elaboração de métodos visuais de apresentação e interação com dados abstratos (Card et al., 1999). Como tais estratégias têm objetivos similares e são complementares, a fusão de ambas em uma área onde a mineração e a visualização co-existem na busca de soluções para interpretação de conjuntos de dados complexos é natural. Tal área é denominada *Mineração Visual de Dados* (Wong, 1999).

No processo de mineração visual de dados, os algoritmos de mineração podem ser assistidos por técnicas de visualização de informação de três diferentes maneiras: (1) na interpretação da estrutura original dos dados; (2) na ajuda à interpretação do resultado dos algoritmos de mineração; ou (3) na intervenção do usuário em estágios intermediários no processo de mineração (Ankerst, 2001). Na primeira, a visualização é empregada no início do processo de mineração visual para ajudar no entendimento dos dados antes dos algoritmos de mineração serem aplicados, já na segunda, as técnicas de visualização são empregadas somente no final do processo para apoiar a interpretação dos resultados obtidos, existindo nesses dois casos pouco acoplamento entre as técnicas de visualização e os algoritmos de mineração. Na última, uma abordagem de interação entre o usuário e o computador é fornecida, permitindo que os parâmetros dos algoritmos de mineração sejam controlados ao longo do processo, de forma que a capacidade humana de tomada de decisão, baseada em conhecimento específico do domínio, seja empregada no lugar de decisões controladas por alguma heurística[1] (Wong, 1999).

Embora a união de mineração e visualização em um único processo seja promissora, com o aumento substancial de tamanho e complexidade dos conjuntos de dados atualmente disponíveis, a extração de informação relevante desses ainda permanece um desafio. Uma medida da complexidade dos dados é o número de atributos associados a cada instância de dados. Considere, por exemplo, dados de censo demográfico: uma instância pode ser descrita por atributos tais como idade, sexo, educação, ocupação, renda, e assim por diante. Considerando cada atributo como uma dimensão, se tivermos m atributos então cada instância de dados pode ser interpretada como um vetor m-dimensional em um espaço m-dimensional. Em análise estatística tradicional, instâncias de dados com quatro ou mais dimensões são conhecidas como multi-variadas ou

[1] De acordo com ANSI/IEEE STD 100–1984 (1984), uma heurística é um método ou algoritmo para resolução de problemas onde as soluções são buscadas por aproximações sucessivas, avaliando-se empiricamente os progressos alcançados até que o problema seja solucionado.

hiper-variadas. A bibliografia atual em visualização de informação refere-se a esse tipo de dados como multi-dimensional.

Muitas técnicas de visualização de informação para dados multi-dimensionais têm sido propostas. A maioria mapeia cada item de dado em um elemento gráfico correspondente, que pode ser um pixel, uma linha, um ícone ou outro marcador gráfico. De acordo com Grinstein et al. (2001), alguns exemplos mais clássicos seriam: *scatterplots 2D e 3D* (Cleveland, 1993), *Matrizes de scatterplots* (Andrews, 1972), *Table lens* (Rao e Card, 1994), técnicas *Iconográficas* (Chernoff, 1973; Pickett e Gristein, 1988; Beddow, 1990; Rose e Wong, 2000), *Empilhamento Dimensional* (LeBlanc et al., 1990), *Coordenadas Paralelas* (Inselberg, 1985; Inselberg e Dimsdale, 1990; Inselberg, 1997), *Técnicas Orientadas a Pixel* (Keim e Kreigel, 1994; Keim e Kriegel, 1996; Keim, 2000), *Segmentos de Círculo* (Ankerst et al., 1996), *RadViz* (Hoffman, 1999), *Grand Tour* (Asimov, 1985), *Mapas Auto-organizáveis de Kohonen* (Kohonen, 1990), etc. Sendo que muitas dessas são similares ou estão relacionadas (Grinstein et al., 2001). Um estudo sobre essas técnicas pode ser encontrado em (Oliveira e Levkowitz, 2003).

Embora tais técnicas possam ser normalmente empregadas com sucesso em pequenos conjuntos de dados, para conjuntos maiores com alta dimensionalidade elas tendem a apresentar problemas, não conseguindo identificar satisfatoriamente grupos de dados correlacionados, determinar *outliers*[2] ou tendências. Isso é apenas uma das razões para a falta de capacidade das ferramentas de exploração visual de dados atuais em lidar com bases de dados de grande magnitude (Oliveira e Levkowitz, 2003).

Uma abordagem alternativa aos métodos de visualização tradicionais, que tem sido aplicada com sucesso na visualização de dados com alta dimensionalidade, são as técnicas de projeção de dados multi-dimensionais, ou simplesmente técnicas de *Projeção Multi-dimensional* (Tejada et al., 2003). Uma técnica de projeção multi-dimensional mapeia as instâncias de dados em um espaço uni-, bi- ou tri-dimensional, preservando alguma informação sobre as relações de distância ou similaridade entre elas de forma a revelar o máximo possível as estruturas existentes. Assim, uma representação gráfica pode ser criada de forma a tirar proveito da habilidade visual humana para reconhecer estruturas ou padrões baseados em similaridade, tais como grupos de elementos similares ou relações entre grupos ou entre elementos de grupos diferentes.

Tipicamente, o resultado de uma projeção multi-dimensional é um conjunto de pontos no plano, numa reta ou em um volume, cada elemento representando uma instância de dados. Numa projeção de boa qualidade, pontos projetados próximos indicam instâncias similares, e os distantes, as não correlacionadas de acordo com a alguma medida de similaridade. Assim, uma representação visual empregando informação geométrica – familiar à maioria das pessoas – pode ser utilizada como apoio à interpretação de conjuntos de dados multi-dimensionais. A aplicação de projeções para a análise de dados não é nova, existindo aplicações em diferentes campos, desde análise estatística à psico-física (Cox e Cox, 2000). Dentre as vantagens desse tipo

[2]Segundo Tan et al. (2005), um *outlier* é (1) um objeto de dados que, em algum sentido, tem características que são diferentes da maioria dos outros objetos do conjunto de dados, ou (2) são valores de um atributo que são incomuns com respeito a valores típicos para o mesmo.

de abordagem, destacam-se a possibilidade de empregar a habilidade humana de identificar e extrair, com ajuda de ferramentas gráficas, padrões que não são detectáveis por métodos clássicos de análise e fornecer ao usuário maior confiança nos resultados obtidos se comparado com outras técnicas (Siedlecki et al., 1988). Além disso, os resultados de uma projeção podem ser usados para se identificar possíveis grupos de elementos similares, e diferentemente da maioria das técnicas de agrupamento atuais, a partir de uma projeção é possível verificar o relacionamento entre diferentes grupos por meio das fronteiras compartilhadas, bem como identificar as instâncias que podem eventualmente pertencer a mais de um grupo, já que essas fronteiras não são rígidas. Por fim, uma projeção pode ser classificada como uma representação que dá suporte a abordagem de exploração "focus+context" (Card et al., 1999), significando que essa fornece a possibilidade de se obter tanto uma visão geral do espaço de informação completo quanto uma visão detalhada dos grupos de instâncias de dados.

Recentemente, uma das aplicações que tem despertado interesse da comunidade científica é a exploração de coleções de documentos. O emprego de projeções multi-dimensionais pode ser bastante efetivo nesse processo, resultando no que é conhecido como "mapa de documentos" (Paulovich et al., 2008a). Um mapa de documentos é um espaço de informação visual do conjunto de dados que permite a navegação do usuário e reflete espacialmente uma ou mais propriedades dos textos. Mapas podem suportar uma variedade de tarefas exploratórias, conectando usuários com seus respectivos mapas cognitivos do espaço de informação visual (Chen, 2006) enquanto evitam algumas das complexidades inerentes ao espaço de informação subjacente. Assim, eles facilitam o acesso a coleções complexas de documentos e melhoram a efetividade de resolver problemas reais de tarefas de gerenciamento de conhecimento (Becks et al., 2002).

Entre as abordagens mais conhecidas para a criação de mapas de documentos baseadas em projeções multi-dimensionais podemos destacar os *Mapas Cartográficos* de Skupin (2002), cuja maior qualidade está no fato de usar uma metáfora familiar a maioria dos usuários: os mapas geográficos; a técnica proposta por Wise (1999), conhecida como *Galaxies*, que emprega uma "abordagem ecológica" para a visualização de texto, a qual utiliza visualizações análogas ao céu noturno e modelos de terreno cuja interpretação é facilitada pela capacidade inerente em nosso cérebro como um resultado de nossa herança biológica. Elas são incorporadas aos sistemas IN-SPIRE^{TM} (PNNL, 2008) e *Infosky* (Andrews et al., 2002), respectivamente.

Assim, projeções multi-dimensionais tem se mostrado como uma solução efetiva à interpretação de conjuntos multi-dimensionais de dados, com resultados bastantes expressivos para coleções de documentos, dando apoio e suporte à rápida extração de informação relevante desse tipo de dado, configurando, portanto, um mecanismo que pode efetivamente contribuir para diminuir os problemas associados à sobrecarga de informação.

1.2 Motivação

Embora as projeções multi-dimensionais tenham se mostrado um mecanismo eficiente para a análise de dados, principalmente para a extração de informação de interesse de grandes coleções

de documentos, tendo apresentado diversos benefícios se comparado com os modelos vigentes de apresentação baseados em listas textuais (Becks et al., 2002), alguns problemas nesse tipo de abordagem são recorrentes, sendo o objetivo central desse trabalho de doutorado tratar alguns desses problemas.

Das várias técnicas de projeção multi-dimensional atualmente disponíveis, as que apresentam melhor resultado em termos da preservação das relações de similaridade e vizinhança, são as de complexidade computacional mais alta, normalmente quadrática ou maior (ver Capítulo 2). Dessa forma, sua aplicação a grandes bases de dados é inviável, ou mesmo impossível. Por outro lado, as técnicas que visam reduzir essa complexidade por meio de aproximações de modelos mais precisos geralmente definem resultados pouco úteis. Assim, apesar da diminuição da complexidade possibilitar o tratamento de bases de dados maiores, pouco sentido existe nessa redução se a técnica resultante não conseguir gerar layouts que reflitam as estruturas existentes nos dados. Portanto, é necessário um melhor compromisso entre escalabilidade computacional e precisão das projeções geradas.

Apesar da popularidade das técnicas de projeção, atualmente nenhuma técnica disponível foi projetada levando em consideração características específicas do domínio a ser tratado, como por exemplo o tipo de distribuição de distâncias definido a partir do espaço a ser projetado e da similaridade empregada. Um exemplo típico seria a projeção de coleções de documentos. Nesse caso, normalmente cada documento é convertido em um vetor onde as dimensões são os termos e as coordenadas refletem suas freqüências de ocorrência. Isso leva à definição de um espaço de alta dimensão e esparso, de forma que as instâncias de dados, nesse caso os documentos, estão somente relacionados a pequenos sub-grupos de documentos, definindo sub-espaços dentro do espaço original (Martín-Merino e Muñoz, 2004). Apesar desse fato ser conhecido e facilmente verificável usando-se histogramas das distâncias entre os vetores que representam os documentos, nenhuma das técnicas existentes para mapas de documentos teve seu desenvolvimento pautado nessa característica. E, como será discutido no restante desta tese, não considerar esse fato acaba levando a projeções distorcidas com relações de similaridade e vizinhança pouco preservadas.

Em geral, é possível dividir as técnicas de visualização de informação em dois grandes grupos: um que busca definir representações gráficas que consigam refletir os relacionamentos entre os atributos dos dados; e outro que visa a criação de representações onde seja possível extrair informação sobre os relacionamentos entre as instâncias de dados. As projeções multi-dimensionais se encaixam no último grupo. Assim, como nenhuma informação sobre os atributos é revelada, não é possível somente com o uso de uma projeção entender o motivo da formação dos grupos no layout final, tornando o processo de exploração de projeções mais difícil e demorado. Este fato é contrário à proposição da redução da sobrecarga de informação, já que não é possível, sem analisar as instâncias de dados, entender o relacionamento entre essas.

Por fim, embora uma projeção consiga efetivamente sintetizar informação de conjuntos de dados de forma que seja mais fácil a interpretação de uma projeção do que de cada instância de dados individualmente, se conjuntos de dados muito grandes forem considerados, problemas podem ocorrer. Como muita informação simultânea será apresentada ao usuário, pode ser

causada uma sobrecarga cognitiva na interpretação da representação visual resultante. Além disso, considerando que uma projeção será desenhada em um espaço limitado, normalmente o monitor de um computador, problemas podem ocorrer devido à metáfora visual empregada: a utilização de um elemento gráfico para cada instância de dados. Para bases grandes de dados isso pode resultar em um layout com sobreposição dos elementos gráficos o que pode levar à difícil distinção dos grupos de elementos e a problemas na avaliação da densidade dos grupos formados. Assim, um componente importante para as técnicas de visualização de informação, apesar de muitas vezes não devidamente observado, deve ser levado em consideração: a *escalabilidade visual*, ou seja, a capacidade da representação visual de efetivamente representar grandes conjuntos de dados (Eick e Karr, 2002).

1.3 Objetivos

De forma a prover uma solução para os problemas citados anteriormente, os objetivos desse trabalho de doutorado são:

> *Definir novas técnicas e abordagens, baseadas no conceito de projeções multi-dimensionais, que possam, de fato, lidar com grandes bases de dados de alta dimensionalidade, mantendo o compromisso com a qualidade dos layouts gerados. Para isso elas devem tratar os problemas clássicos das projeções, isto é, a tendência de misturar grupos ou vizinhanças diferentes de dados na mesma região e o aspecto de alta complexidade computacional. Em outras palavras, o presente trabalho busca oferecer soluções de projeções que forneçam um maior compromisso entre custo computacional e qualidade do layout final. Além disso, elas devem ser capazes de prover exploração de conjuntos de dados em diferentes níveis de detalhamento, reduzindo assim os problemas relacionados à escalabilidade visual das projeções. Busca-se também definir uma técnica de suporte à interpretação de projeções multi-dimensionais para o caso específico de coleções de documentos, de forma a ser possível determinar os motivos que levaram à formação dos grupos em uma determinada projeção. O objetivo principal é tornar uma projeção multi-dimensional um mecanismo efetivo para o processo de mineração visual de dados com especial atenção à mineração visual de textos.*

A seguir é fornecido um resumo dos resultados alcançados ao longo desse projeto de doutorado.

1.4 Resultados

De forma a cumprir os objetivos propostos para essa tese, inicialmente foi feito um estudo das técnicas de projeção multi-dimensionais mais relevantes para esse trabalho de doutorado,

identificando-se as características e os problemas das abordagens vigentes. Dessa forma compôs-se a base teórica para as técnicas e abordagens aqui desenvolvidas.

Como resultado, três diferentes técnicas para projeção multi-dimensional foram criadas, cada uma com características específicas que as tornam mais indicadas a diferentes domínios. Uma das técnicas, denominada *Projection By Clustering (ProjClus)* (Paulovich e Minghim, 2006), é uma aproximação de uma técnica já existente a fim de reduzir sua complexidade computacional, mantendo a qualidade dos layouts gerados. Outra, conhecida como *Least Square Projection (LSP)* (Paulovich et al., 2008a), se baseia na preservação das relações de vizinhança locais e similaridades ente grupos de instâncias, sendo indicada para a projeção de espaços esparsos de alta dimensão, como a representação vetorial de coleções de documentos. A última, denominada *Hierarchial Point Placement (HiPP)* (Paulovich e Minghim, 2008), define uma abordagem hierárquica que dá suporte a exploração e reorganização de conjuntos de dados em diferentes níveis de detalhamento, reduzindo os problemas inerentes às projeções multi-dimensionais com relação a escalabilidade visual. Além dessas, foi criada uma abordagem para extração dos assuntos, aqui denominados tópicos, tratados dentro de subconjuntos de documentos, para apoio à interpretação das projeções para esse tipo de dado.

Para a realização do estudo comparativo entre as técnicas de projeção vigentes dois métodos de avaliação foram também propostos, o *Neighborhood Hit* (Paulovich et al., 2008a) e o *Neighborhood Preservation* (Paulovich e Minghim, 2008). Um sistema, conhecido como *Projection Explorer (PEx)* (Paulovich et al., 2007), foi também desenvolvido, que é uma ferramenta para criação e exploração de projeções multi-dimensionais em geral. A PEx oferece diferentes técnicas para projeção multi-dimensional, para redução de dimensionalidade e para transformações de dados, além de uma série de medidas de dissimilaridade, técnicas para avaliação de projeções, mecanismos para exploração das projeções geradas, entre outras funcionalidades.

Além dessas técnicas e ferramentas, outros trabalhos foram desenvolvidos e publicados em cooperação com outros alunos e pesquisadores como parte de seus trabalhos, entre esses: (1) o emprego de uma aproximação da complexidade de Kolmogorov como forma de dissimilaridade entre documentos (Telles et al., 2007); (2) o uso de uma técnica de construção de árvores filogenéticas para a visualização de coleções de documentos (Cuadros et al., 2007); (3) o desenvolvimento de quatro extensões da PEx, a PEx-Web (Paulovich et al., 2008b), a PEx-Corpus (Muniz et al., 2007), a Temporal-PEx (Alencar et al., 2008) e a PEx-Image (Eler et al., 2008b), além da H-PEx (Paulovich e Minghim, 2008); (4) a definição de uma abordagem para a extração de tópicos baseada em regras de associação (Lopes et al., 2007); e (5) por fim, a definição de um arcabouço para a coordenação de múltiplas projeções de coleções de documentos (Eler et al., 2008a).

Os sistemas desenvolvidos formaram uma parte bastante significativa deste doutorado, e fizeram parte de uma série de projetos do grupo de visualização do ICMC-USP, como o Infovis2 (FAPESP), o MineVis (CNPq) e o MultiVis (CNPq). Os programas e a maioria do código produzido se encontram disponibilizados em `http://infoserver.lcad.icmc.usp.br/infovis2`.

Como resultado direto desse trabalho, foram publicados 4 artigos em periódicos internacionais da área de computação gráfica e visualização e 12 artigos em anais de congressos nacionais e internacionais. Uma lista completa e detalhada do que foi publicado pode ser encontrada na Tabela 1, no preâmbulo desta tese.

1.5 Organização

Neste texto foi decidido dar destaque às técnicas desenvolvidas, ao invés das ferramentas de software. Para a apresentação do que foi produzido nesse projeto de doutorado, essa tese está estruturada da seguinte maneira:

- No Capítulo 2 é apresentada uma revisão bibliográfica sobre as técnicas de projeção multi-dimensional mais relevantes para o nosso contexto. Empregando-se um estudo comparativo, as particularidades dessas técnicas são identificadas a fim de definir suas limitações, servindo como base para o desenvolvimento das técnicas desse trabalho de doutorado;

  Os demais Capítulos detalham as principais contribuições desta tese.

- No Capítulo 3 a primeira técnica de projeção multi-dimensional desenvolvida dentro desse projeto de doutorado, conhecida como ProjClus, é apresentada. Ela busca reduzir a complexidade computacional de uma técnica já existente sem comprometer a qualidade dos layouts gerados, sendo uma técnica de propósito geral;

- No Capítulo 4 outra técnica de projeção multi-dimensional, denominada LSP, é apresentada. Essa baseia-se em uma técnica para reconstrução e edição de malhas, mas definida de forma a ser possível trabalhar em espaços de alta-dimensionalidade enquanto evita a necessidade de empregar uma malha;

- No Capítulo 5 uma abordagem de apoio à análise de mapas documentos é apresentada. A partir de uma seleção realizada pelo usuário, é extraído um conjunto de palavras que descreva os assuntos mais relevantes tratados dentro dos documentos selecionados. Dessa forma, a tarefa de entendimento dos agrupamentos definidos em uma projeção é facilitada, provendo uma forma mais rápida de se extrair uma visão geral desse tipo de conjunto de dados;

- No Capítulo 6 é apresentada uma nova abordagem hierárquica para a criação, visualização e exploração de projeções multi-dimensionais, com ênfase em coleções de documentos, denominada HiPP. Uma estrutura hierárquica é definida possibilitando ao usuário navegar em diferentes níveis de abstração ou detalhamento, facilitando o processo de exploração, especialmente das coleções de documentos. Além disso, uma abordagem para a re-estruturação dessa hierarquia é provida possibilitando a re-organização dos dados conforme a representação visual é explorada.

- Por fim, no Capítulo 7 as conclusões desse trabalho são delineadas com um resumo do trabalho aqui apresentado, suas maiores contribuições para o processo de mineração visual de dados, e suas limitações, além da discussão de trabalhos futuros.

Capítulo

2

Técnicas para Projeção de Dados Multi-dimensionais

2.1 Considerações Iniciais

Uma técnica para projeção de dados multi-dimensionais, ou simplesmente técnica de *Projeção Multi-dimensional*, tipicamente mapeia dados m-dimensionais em um espaço p-dimensional com $p = \{1, 2, 3\}$, preservando alguma informação sobre as relações de distância entre as instâncias (objetos) de dados de forma a revelar o máximo possível as estruturas existentes. Assim, uma representação gráfica pode ser criada de forma a tirar proveito da habilidade visual humana em reconhecer estruturas ou padrões baseados em similaridade, tais como grupos de elementos similares ou relações entre grupos.

Formalmente uma técnica de projeção multi-dimensional pode ser definida como:

Definição 2.1 (Projeção Multi-dimensional (Tejada et al., 2003)) *Seja* $\mathbf{X}$ *um conjunto de objetos em* $\mathbb{R}^m$ *com* $\delta : \mathbb{R}^m \times \mathbb{R}^m \rightarrow \mathbb{R}$ *um critério de proximidade entre objetos em* $\mathbb{R}^m$*, e* $\mathbf{Y}$ *um conjunto de pontos em* $\mathbb{R}^p$ *para* $p = \{1, 2, 3\}$ *e* $d : \mathbb{R}^p \times \mathbb{R}^p \rightarrow \mathbb{R}$ *um critério de proximidade em* $\mathbb{R}^p$*. Uma técnica de projeção multi-dimensional pode ser descrita como uma função* $f : \mathbf{X} \rightarrow \mathbf{Y}$ *que visa tornar* $|\delta(\mathbf{x}_i, \mathbf{x}_j) - d(f(\mathbf{x}_i), f(\mathbf{x}_j))|$ *o mais próximo possível de zero,* $\forall \mathbf{x}_i, \mathbf{x}_j \in \mathbf{X}$*.*

Em geral, o resultado de uma projeção multi-dimensional é um conjunto de pontos no plano. O resultado também poderia ser pontos em uma reta ou em um volume, porém como as técnicas apresentadas aqui são caracterizadas por produzirem projeções bi-dimensionais, sempre

será considerada uma projeção no plano, sem perda de generalidade. Idealmente, se pontos forem posicionados próximos nesse layout, isso indica que os objetos que esses representam são similares de acordo com a distância (dissimilaridade) escolhida (δ) e, se pontos forem projetados distantes, isso indica que os objetos que os mesmos representam não são relacionados, ou são pouco relacionados.

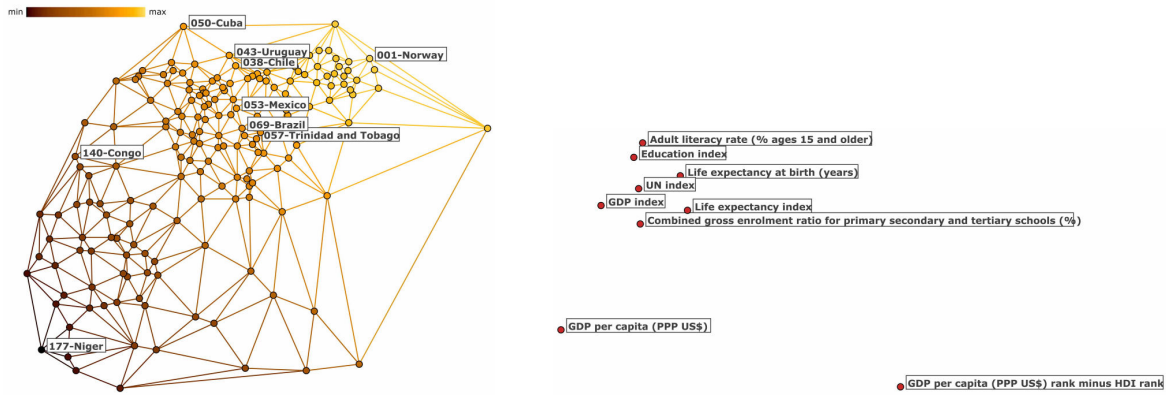
A Figura 2.1 apresenta dois exemplos diferentes visando indicar como projeções podem auxiliar a interpretação de dados multi-dimensionais. Essas projeções foram geradas a partir de um conjunto de dados contendo informação sobre o *Índice de Desenvolvimento Humano (IDH)* de diferentes países no ano de 2006. Os dados foram disponibilizados no *Relatório de Desenvolvimento Humano* do ano de 2006[1] do programa de desenvolvimento da *Organização das Nações Unidas (ONU)*. O IDH inclui medidas comparativas da expectativa de vida, alfabetização, educação e padrão de vida, em um total de oito variáveis medidas para 177 países. Indica se um país é desenvolvido, está em desenvolvimento ou é sub-desenvolvido, e também provê uma medida para avaliar o impacto das políticas econômicas na qualidade de vida da população.

Para se criar a projeção da Figura 2.1(a) todos os atributos foram empregados, com exceção do índice de desenvolvimento atribuído pela ONU, que foi utilizado para colorir os pontos (países). Essa projeção claramente reflete esse índice de desenvolvimento, agrupando os países com qualidade de vida similar. Nela é possível verificar que em 2006 o status de desenvolvimento do Brasil estava tão próximo ao da Noruega quanto ao do Congo, e que, comparado com nossos vizinhos latinos americanos, o Brasil ocupava uma posição inferior de qualidade de vida (apesar do índice atribuído pela ONU não conseguir refletir tão claramente essa disparidade por ser uma projeção uni-dimensional dos atributos). Nessa figura as arestas são definidas criando-se uma triangulação Delaunay (Edelsbrunner, 2001) para ajudar na percepção de cores de certas regiões. A projeção da Figura 2.1(b) foi criada projetando-se as variáveis ao invés dos países. Assim, é possível analisar a correlação dessas com o índice gerado. Com essa análise nota-se que a expectativa de vida da população (*Life expectancy at birth (years)*), ou mesmo o próprio índice de desenvolvimento (*UN index*), estão mais relacionados ao grau de escolaridade e educação de um país do que ao seu produto interno bruto per capita (*GDP per capita (PPP US$)*).

A técnica empregada para se criar essas projeções, conhecida como *Sammnon's Mapping* (Sammon, 1969), e outras relevantes ao contexto desse projeto de doutorado são apresentadas e comparadas no restante desse capítulo. Inicialmente, uma breve discussão sobre distâncias, similaridades e transformação de dados é apresentada. Após isso, as técnicas de projeção multi-dimensionais são detalhadas, mostrando o resultado dessas quando aplicadas a dois conjuntos de dados selecionados. Por fim, as conclusões desse estudo são delineadas, apontando os problemas e deficiências das técnicas analisadas formando a base teórica para o desenvolvimento das técnicas definidas nessa tese.

Para facilitar o entendimento, os símbolos mais freqüentemente empregados nesse estudo e suas descrições podem ser encontrados na Tabela 2.1.

[1] veja `http://hdr.undp.org/hdr2006/statistics/`



(a) Semelhanças entre países com base no IDH.

(b) Correlação entre as medidas do IDH e o valor final do índice.

Figura 2.1: Exemplos de projeções multi-dimensionais para os dados do *Índice de Desenvolvimento Humano (IDH)* de 2006 da *Organização das Nações Unidas (ONU)*.

Tabela 2.1: Símbolos mais freqüentes e seus significados.

Símbolo	Significado
$\mathbf{X}$	conjunto de objetos no espaço original m-dimensional.
m	dimensão do espaço original.
$\mathbf{x}_i$	i-ésimo objeto do espaço original. Quando esse admitir uma representação vetorial, $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{im})$ representam suas coordenadas.
$\mathbf{Y}$	conjunto de pontos no espaço projetado p-dimensional.
p	dimensão do espaço projetado.
$\mathbf{y}_i$	i-ésimo ponto do espaço projetado. Quando esse admitir uma representação vetorial, $\mathbf{y}_i = (y_{i1}, y_{i2}, \ldots, y_{ip})$ representam suas coordenadas.
n	número de objetos no espaço original e pontos no projetado.
$\delta(\mathbf{x}_i, \mathbf{x}_j)$	dissimilaridade entre os objetos i e j no espaço original.
$d(\mathbf{y}_i, \mathbf{y}_j)$	distância entre os pontos i e j no espaço projetado.

2.2 Distâncias, Similaridades e Transformação dos Dados

Na aplicação das técnicas de projeção multi-dimensional, a forma como a distância entre os objetos multi-dimensionais $\mathbf{X}$ é calculada desempenha papel central. Assim, o primeiro passo para a aplicação efetiva dessas técnicas é entender o que é mensurado quando uma distância ($\delta(\mathbf{x}_i, \mathbf{x}_j)$) é aplicada, e buscar encontrar uma que melhor reflita a diferença entre os objetos em um determinado domínio. Por exemplo, dado um conjunto de séries temporais, se a distância Euclideana for aplicada o que se estará medindo é a diferença de intensidade entre as séries, ignorando o formato global das mesmas. Caso se deseje comparar as séries com base em seus formatos, outra métrica deve ser utilizada (Alencar et al., 2008). Entretanto, na ausência de uma forma de medida típica, pode-se iniciar o processo de análise de um determinado domínio ou aplicação usando-se medidas conhecidas de distância ou dissimilaridade.

Dentre os diferentes tipos de distâncias existentes, a distância de *Minkowski* é uma das mais conhecidas. Na verdade, ela forma toda uma família de métricas de distância denominadas normas L_p, onde p é um parâmetro que define a métrica, que não deve ser confundido com a dimensão do espaço projetado. A Equação (2.1) apresenta sua definição.

$$L_p(\mathbf{x}_i, \mathbf{x}_j) = (\sum_{k=1}^{m} |x_{ik} - x_{jk}|^p)^{\frac{1}{p}} \tag{2.1}$$

Com $p = 1$ obtém-se a distância *Manhattan*, também conhecida como *City Block*, com $p = 2$ tem-se a distância Euclideana, e com $p = \infty$ obtém-se a distância do infinito, também conhecida como norma L_{max}, calculada como $L_\infty(\mathbf{x}_i, \mathbf{x}_j) = \max_{k=1}^{m} |x_{ik} - x_{jk}|$.

Para que uma medida possa ser considerada uma métrica, quatro propriedades devem ser obedecidas. Suponha uma espaço métrico $\mathcal{M} = (\mathbf{X}, \delta)$ definido sobre o domínio dos objetos $\mathbf{X}$ com a função de distância δ. As propriedades da função $\delta : \mathbf{X} \times \mathbf{X} \mapsto \mathbb{R}$, freqüentemente conhecidas como postulados do espaço métrico (Zezula et al., 2005), são:

1. **Não-Negatividade:** $\forall \mathbf{x}_i, \mathbf{x}_j \in \mathbf{X}$, $\delta(\mathbf{x}_i, \mathbf{x}_j) \geq 0$
2. **Identidade:** $\forall \mathbf{x}_i, \mathbf{x}_j \in \mathbf{X}$, $\mathbf{x}_i = \mathbf{x}_j \Leftrightarrow \delta(\mathbf{x}_i, \mathbf{x}_j) = 0$
3. **Simetria:** $\forall \mathbf{x}_i, \mathbf{x}_j \in \mathbf{X}$, $\delta(\mathbf{x}_i, \mathbf{x}_j) = \delta(\mathbf{x}_j, \mathbf{x}_i)$
4. **Desigualdade Triangular:** $\forall \mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k \in \mathbf{X}$, $\delta(\mathbf{x}_i, \mathbf{x}_k) \leq \delta(\mathbf{x}_i, \mathbf{x}_j) + \delta(\mathbf{x}_j, \mathbf{x}_k)$

Apesar dessas propriedades serem matematicamente interessantes, por exemplo possibilitando o aumento da eficiência de certas técnicas (no caso da **Desigualdade Triangular**), existem algumas medidas importantes que não obedecem essas propriedades. A essas é dado o nome de dissimilaridade[2]. As dissimilaridades formam um grupo maior de medidas, onde as métricas de distância são um sub-grupo específico, e podem ser qualquer tipo de função que consiga definir numericamente o quão diferente um objeto é de outro (Tan et al., 2005).

Dentre as possíveis formulações de dissimilaridade, é comum encontrar definições que são o inverso da similaridade $s(\mathbf{x}_i, \mathbf{x}_j)$ entre objetos – similaridade sendo uma medida numérica de quão semelhante dois objetos são. Dois exemplos de inversão são, $\delta(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{s(\mathbf{x}_i, \mathbf{x}_j)+1}$ e $\delta(\mathbf{x}_i, \mathbf{x}_j) = e^{-s(\mathbf{x}_i, \mathbf{x}_j)}$. Caso os limites inferiores e superiores da similaridade sejam conhecidos, é comum mapeá-las para o intervalo $[0, 1]$ usando-se $s'(\mathbf{x}_i, \mathbf{x}_j) = \frac{s(\mathbf{x}_i, \mathbf{x}_j) - s_{min}}{s_{max} - s_{min}}$, onde s_{min} denota a menor similaridade possível e s_{max} a maior, e aplicar a transformação $\delta(\mathbf{x}_i, \mathbf{x}_j) = 1 - s'(\mathbf{x}_i, \mathbf{x}_j)$. De forma geral, qualquer função monotônica decrescente pode ser usada para se transformar similaridades em dissimilaridades.

Um exemplo conhecido de dissimilaridade é a inversão da similaridade do cosseno entre dois vetores usando-se $1 - cos(\mathbf{x}_i, \mathbf{x}_j)$. Para espaços de alta dimensão e esparsos, essa medida

[2]O termo dissimilaridade não ocorre no português, mas não foi encontrado termo que pudesse ser melhor aplicado.

é preferível à Euclideana uma vez que a mesma não considera em seu resultado comparações $0 - 0$. Conceitualmente, isso reflete o fato de que a similaridade (dissimilaridade) entre dois objetos depende do número de características que os mesmos compartilham ao invés do número que ambos não compartilham. No caso de espaços de alta dimensão e esparsos isso é uma característica desejável uma vez que os objetos terão apenas um conjunto limitado de características que os descrevem (atributos diferentes de zero) e portanto serão muito similares em termos das características que os mesmos não apresentam (Tan et al., 2005). Um outro exemplo de similaridade com as mesmas características é conhecida como coeficiente de *Tanimoto* (ou *Jaccard* extendida), dada pela seguinte equação:

$$s(\mathbf{x}_i, \mathbf{x}_j) = \frac{\mathbf{x}_i \cdot \mathbf{x}_j}{||\mathbf{x}_i||^2 + ||\mathbf{x}_j||^2 - \mathbf{x}_i \cdot \mathbf{x}_j} \tag{2.2}$$

nesse caso a dissimilaridade pode ser calculada usando a mesma abordagem aplicada ao cosseno, fazendo-se $1 - s(\mathbf{x}_i, \mathbf{x}_j)$.

No cálculo das dissimilaridades (ou similaridades), dois diferentes cenários podem distorcer os resultados ou torná-los tendenciosos: (1) quando os vetores que representam os objetos têm normas Euclideanas muito diferentes; e (2) quando uma ou mais coordenadas dos vetores está em uma escala diferente das outras coordenadas, levando a resultados que tenderão à diferença entre essas coordenadas, ignorando-se a representatividade das demais.

Para evitar o primeiro cenário um processo de **normalização** pode ser aplicado nos vetores, fazendo que todos os vetores tenham norma Euclideana unitária, isto é $||\mathbf{x}_i|| = \sum_{k=1}^{m} x_{ik}^2 = 1$. Isto é feito dividindo cada coordenada do vetor i por sua norma, $x'_{ij} = x_{ij}/||\mathbf{x}_i||$ para $1 \leq j \leq m$. O segundo cenário pode ser evitado aplicando-se um processo conhecido como **padronização** ou ***standardization*** (Tan et al., 2005). Nesse, se $\bar{x_j} = \frac{1}{n}\sum_{i=1}^{n} x_{ij}$ é a média da coordenada j e $\sigma_j = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_{ij} - \bar{x_j})^2}$ seu desvio padrão, essa transformação é obtida fazendo-se $x'_{ij} = (x_{ij} - \bar{x_j})/\sigma_j$ para $1 \leq i \leq n$ e $1 \leq j \leq m$, criando novas coordenadas que têm média igual a 0 e desvio padrão igual a 1.

Existem diversas outras formas de se transformar os dados e de se calcular as dissimilaridades entre objetos multi-dimensionais. Aqui, somente um resumo dessas possibilidades foi apresentado. Para cada domínio de dados, pode ser preferível o emprego de medidas e transformações próprias, dependendo do que se deseja mensurar. Para mais informações, uma discussão mais detalhada sobre esse assunto pode ser encontrada em (Tan et al., 2005; Zezula et al., 2005).

2.2.1 Dissimilaridades entre Documentos

Nem todos os conjuntos de dados apresentam um formato vetorial de forma a ser possível calcular a dissimilaridade entre os objetos que os compõem usando as medidas apresentadas na Seção 2.2. Para eles, ou outras métricas não baseadas em vetores são usadas ou então deve ser empregada uma forma de se transformar os objetos desses conjuntos em vetores. Um caso típico da transformação em uma representação vetorial é o de coleções de documentos.

Existem diversas abordagens para se transformar um conjunto de documentos em vetores. Aqui, é adotada uma abordagem que consiste em primeiro converter cada documento em um vetor cujas coordenadas refletem a freqüência de ocorrência de certos termos presentes nos documentos, e depois unir esses vetores em uma matriz de *documentos x termos* na qual cada termo é ponderado de acordo com alguma medida. Os passos para se criar tal matriz são:

1. Eliminação de palavras não representativas (*stop words*), como artigos, preposições, etc.;
2. Conversão das palavras restantes em seus radicais (lematização) - no caso de documentos em inglês o algoritmo de Porter (1997) é empregado;
3. Contagem das palavras dentro de cada documento para determinar suas freqüências de ocorrência;
4. Aplicação dos cortes superiores e inferiores de Luhn (1968) para eliminar palavras muito freqüentes ou raras. É importante observar que os valores de corte de Luhn que devem ser utilizados são estabelecidos de forma empírica, já que não existe uma base teórica para sua determinação (van Rijsbergen, 1979);
5. Ponderação das freqüências dos termos em cada documento de acordo com alguma medida, por exemplo, *term-frequency inverse document-frequency (tf-idf)* (Salton e Buckley, 1988). No caso do *tf-idf* se um termo aparecer em muitos documentos, seu "poder de representação" é diminuindo, uma vez que supostamente o mesmo não serve para distinguir documentos individuais da coleção.

O *tf-idf* pode ser calculado usando:

$$tf - idf(t_i, d_j) = freq(t_i, d_j) \times \log\left(\frac{n}{dfreq(t_i)}\right) \tag{2.3}$$

onde t_i é o i-ésimo termo do documento d_j, $freq(t_i, d_j)$ é a freqüência do termo t_i no documento d_j, n é a quantidade de documentos na coleção e $dfreq(t_i)$ é a quantidade de documentos onde o termo t_i ocorre pelo menos uma vez.

Na matriz criada, cada linha é um vetor representando um documento e cada termo final é uma dimensão, com suas coordenadas definidas por meio do *tf-idf*.

Nesta tese, com base nesses vetores a dissimilaridade entre os documentos é calculada usando-se uma estratégia baseada no cosseno $(1 - cos(\mathbf{x}_i, \mathbf{x}_j))$ uma vez que essa se mostra bastante apropriada para espaços esparsos, como é o caso da representação vetorial de coleções de documentos (Tan et al., 2005).

Uma segunda possibilidade para coleções de documentos é aplicar uma medida de dissimilaridade não baseada em vetores, conhecida como *Normalized Compression Distance (NCD)* (Cilibrasi e Vitányi, 2005). Seja $C(x)$ o tamanho em bytes da compressão da cadeia de caracteres

x por um dado compressor, a dissimilaridade NCD entre duas cadeias de caracteres, ou dois documentos, x e y é dada por

$$NCD(x,y) = \frac{C(xy) - \min\{C(x), C(y)\}}{\max\{C(x), C(y)\}} \tag{2.4}$$

onde xy é a concatenação de x e y.

A NCD é uma aproximação da complexidade de Kolmogorov e foi originalmente definida como uma medida de dissimilaridade entre seqüencias de DNA. Posteriormente, a mesma foi modificada para acomodar problemas relacionados com a precisão do compressor utilizado, e apresentada como uma medida de dissimilaridade entre documentos em (Telles et al., 2005, 2007), com resultados similares aos alcançados se usada a dissimilaridade baseada no cosseno. A Equação(2.5) apresenta essa modificação, conhecida como NCD_S ou *Scaled NCD*.

$$NCD_S(x,y) = NCD(x,y) - \frac{NCD(x,x) + NCD(y,y)}{2}. \tag{2.5}$$

As técnicas que nessa tese denominamos projeções multi-dimensionais podem ser alimentadas pela matriz de atributos de um conjunto de dados ou diretamente pelas relações de (dis)similaridade.

A seguir diferentes técnicas de projeção multi-dimensional são detalhadas, apresentando os pontos negativos e positivos de cada técnica com exemplos de aplicação.

2.3 Técnicas de Projeção de dados Multi-dimensionais

É possível identificar na literatura diversas técnicas que podem ser empregadas para se realizar projeções multi-dimensionais. Aqui são apresentadas as técnicas mais relevantes para a análise visual de dados a que se refere este trabalho, que são divididas em três grandes grupos: (1) *Force-Direct Placement (FDP)*; (2) *Multidimensional Scaling (MDS)*; e (3) técnicas para redução de dimensionalidade.

O primeiro grupo é formado por técnicas baseadas na idéia de um sistema composto por objetos que estão sob a ação de forças de atração e repulsão. O segundo grupo é composto por técnicas que visam mapear o espaço original em um espaço visual (1D, 2D ou 3D) buscando preservar relações de distância; esse grupo de técnicas vem sendo empregado em diferentes áreas do conhecimento, tais como psicologia experimental, arqueologia, desenho de grafos, etc. (uma lista parcial pode ser encontrada em (Buja et al., 1998)). Por fim, o último grupo visa transformar o espaço original em um espaço com dimensão reduzida preservando algum tipo de característica dos objetos originais nos reduzidos. No caso específico desta tese, as relações de similaridade. Nesse caso, para que essas técnicas possam ser aplicadas no contexto da visualização, o espaço reduzido deverá ser uni-, bi- ou tri-dimensional. A Figura 2.2 apresenta uma classificação das técnicas apresentadas a seguir – as elipses representam as (sub-)classes de técnicas e os retângulos contém os nomes das técnicas em cada (sub-)classe.

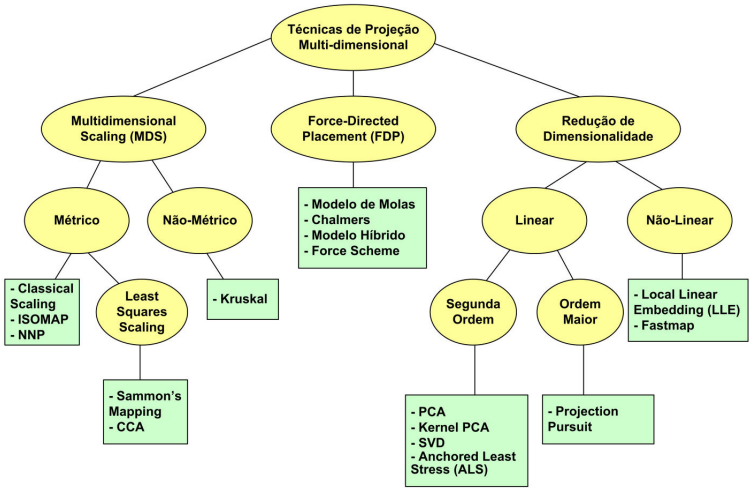


Figura 2.2: Classificação das técnicas de projeção apresentadas nesse capítulo.

Antes de descrever as técnicas, a próxima seção apresenta os conjuntos de dados que serão utilizados para exemplificá-las nas seções seguintes.

2.3.1 Conjuntos de Dados

De forma a se realizar a análise comparativa das técnicas de projeção apresentadas nesse Capítulo, dois conjuntos de dados diferentes foram selecionados, cada um apresentando características específicas. O primeiro conjunto, aqui chamado de **Superfície-S**, é uma superfície bi-dimensional retorcida no formato de um "S" com um furo ao meio, embutida em um espaço tri-dimensional. Com esse conjunto buscamos verificar a efetividade das técnicas de projeção em lidar com dados que apresentem relações não-lineares entre seus atributos, definindo se as mesmas são capazes de "desdobrar" uma variedade ou *manifold* de menor dimensão, um exemplo típico aplicado pela comunidade de *Multidimensional Scaling* para verificar esse tipo de propriedade (Borg e Groenen, 2005). A Figura 2.3 apresenta esse conjunto.

Figura 2.3: O conjunto de dados **Superfície-S**. Conjunto não-linear composto por uma superfície bi-dimensional retorcida embutida em um espaço tri-dimensional.

Para gerar esse conjunto usamos a Equação (2.6), com alguns pontos cuidadosamente removidos para formar o buraco na superfície:

$$\begin{aligned} x_i &= u_i \\ y_i &= \tfrac{1}{2}\sin(2\pi v_i) \\ z_i &= v_i \end{aligned} \tag{2.6}$$

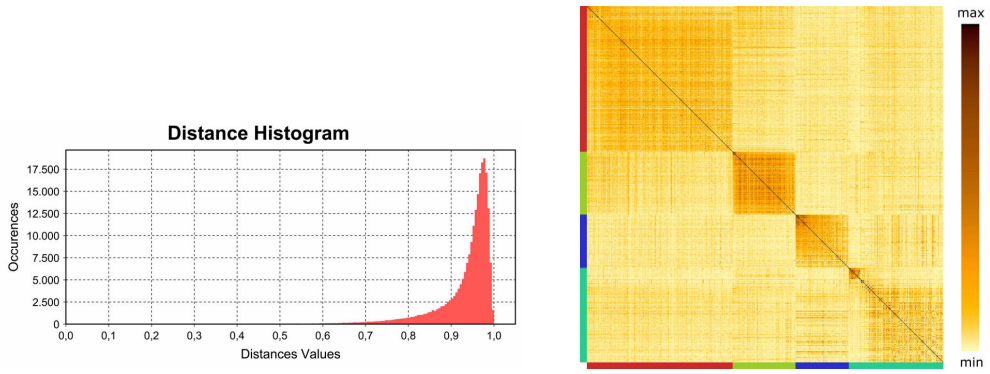
onde u e v são igualmente distribuídos entre 0 e 1.

O segundo conjunto, **CBR-ILP-IR-SON**, é uma coleção de 675 documentos composta por título, autores, afiliação, resumo e referências de artigos científicos em quatro diferentes áreas: *Case-Based Reasoning (CBR)*, *Inductive Logic Programming (ILP)*, *Information Retrieval (IR)* e *Sonification (SON)*. Os documentos sobre CBR e ILP foram retirados de periódicos nessas áreas e os demais foram obtidos por meio de uma busca na Internet. Para se compor os atributos desse conjunto, os documentos são transformados em vetores cujas coordenadas refletem a freqüência de ocorrência de certos termos (veja Seção 2.2.1 para maiores detalhes), resultando em um conjunto de 1.423 dimensões.

Para se conhecer as características desse conjunto de dados, duas análises foram aplicadas (veja, por exemplo, a Figura 2.4). A primeira consiste na verificação da distribuição das distâncias empregando-se histogramas. Idealmente, se o objetivo de uma projeção for agrupar elementos similares e separar os grupos entre si, esse histograma deveria apresentar dois picos, um referente às pequenas distâncias (objetos próximos que resultam em agrupamentos) e outro às grandes (objetos distantes que separam agrupamentos distintos). Porém, na prática, esse cenário é raro, e normalmente para conjuntos com dimensionalidade média ou alta um único pico é apresentado – a maioria dos objetos está próximo entre si, ou a maioria está distante entre si, ou os objetos estão igualmente distribuídos no espaço. Isso ocorre porque conforme a dimensionalidade cresce, a diferença entre o vizinho mais próximo e o mais distante diminui, para qualquer métrica de distância utilizada (Beyer et al., 1999). Esse efeito é conhecido por "Maldição da Alta-Dimensionalidade" (Hinneburg et al., 2000), um termo primeiramente utilizado em (Bellman, 1961) mas que vem sendo empregado como uma indicação de que alta dimensionalidade normalmente causa problemas de distinção entre items.

A segunda análise emprega a visualização da matriz de similaridades para determinar se a classificação pré-existente desse conjunto resulta em grupos separáveis utilizando-se uma determinada dissimilaridade. Para um conjunto contendo n objetos, essa representação gráfica é construída criando-se uma matriz retangular $n \times n$ de pixels onde a cor do pixel posicionado na linha i e coluna j da matriz é definida de acordo com a similaridade entre os objetos i e j. Nessa matriz, os objetos são ordenados de acordo com a classe a que pertencem, e a similaridade entre eles é dada por $s(\mathbf{x}_i, \mathbf{x}_j) = 1 - ((\delta(\mathbf{x}_i, \mathbf{x}_j) - \delta_{min})/(\delta_{max} - \delta_{min}))$. Nessa representação, se os grupos pré-definidos forem separáveis com base na métrica utilizada, a imagem resultante dessa matriz deve apresentar blocos na diagonal (Tan et al., 2005), e quanto mais destacados forem esses blocos, mais separáveis são os grupos entre si.

A Figura 2.4 apresenta os resultados de ambas análises para o conjunto **CBR-ILP-IR-SON**. Pelo histograma (Figura 2.4(a)) é possível verificar que os objetos não são muito relacionados, ou que um objeto só está relacionado a um número limitado de objetos. Isto pode ser visto como um efeito do espaço resultante da representação vetorial de documentos ser esparso e apresentar alta dimensão, sendo que em tais casos os objetos freqüentemente estão distribuídos em subespaços locais e são somente relacionados a um pequeno número de vizinhos dentro do mesmo subespaço (Martín-Merino e Muñoz, 2004). Pela matriz de similaridades (Figura 2.4(b)) é possível notar a presença de três grupos distintos, que são o CBR (em vermelho), o ILP (em amarelo) e o SON (em azul escuro) – a fina coluna colorida à esquerda e a linha na parte inferior da imagem indicam as classes dos objetos. Já o grupo IR (em verde claro) não é tão bem definido, apresentando um sub-grupo de objetos bastante relacionados e o restante sendo dissimilar aos outros grupos e também entre si. Isso pode ser visto como resultado do modo como esse grupo foi formado: uma busca na Internet usando-se palavras-chave muito genéricas ("information AND retrieval") que não são suficientes para se identificar uma única área. Como conseqüência, esse grupo pode ser visto como ruído no conjunto de dados e espera-se que seus objetos fiquem mais espalhados do que os demais na projeção final, não compondo um grupo separado, mas também não se misturando muito aos objetos das outras classes.



(a) Histograma das distâncias. (b) Matriz de similaridades.

Figura 2.4: Histograma de distâncias e matriz de similaridades para o conjunto de dados **CBR-ILP-IR-SON**. Essas análises indicam que os objetos nesse conjunto não são muito correlacionados, e que os quatro grupos de documentos que supostamente existem na verdade são três, sendo um deles ruído.

A seguir são apresentadas diferentes técnicas de projeção e os resultados da aplicação das mesmas aos dois conjuntos mencionados.

2.3.2 Force-Directed Placement (FDP)

Dentre as várias técnicas que podem ser empregadas para projeção multi-dimensional, os *Modelos baseados em Molas (MM)* (Fruchterman e Reingold, 1991; Frick et al., 1995) são os mais simples. Originalmente proposto por Eades como uma heurística para o desenho de grafos (Eades, 1984), um MM tenta levar um sistema de objetos conectados por molas a um estado de equilíbrio. Nesse

sistema os objetos são posicionados inicialmente de forma aleatória (ou por meio de alguma heurística), e então as forças geradas pelas molas são usadas para iterativamente puxar ou empurrar os objetos até se atingir uma posição de equilíbrio.

Para que um MM seja empregado para a criação de projeções, as forças no sistema são calculadas proporcionais à diferença entre as dissimilaridades $\delta(\mathbf{x}_i, \mathbf{x}_j)$ entre os objetos e a distâncias $d(\mathbf{y}_i, \mathbf{y}_j)$ entre os pontos no layout gerado. Como a analogia de força é explícita, freqüentemente esses algoritmos são denominados de *Force-Directed Placement (FDP)* (Fruchterman e Reingold, 1991).

Nas próximas seções será detalhado o funcionamento de um MM, seguido por alguns exemplos de métodos desenvolvidos que empregam esse modelo (ou uma derivação dele) especificamente para projeções multi-dimensionais e visualização de informação.

2.3.2.1 Modelo de Molas

A idéia por trás de um MM está em modelar os objetos como partículas ponto-massa[3] que estão ligadas entre si por meio de molas. Em um MM, um conjunto de partículas está sujeito às leis de *Newton*, mais especificamente a segunda lei que declara que a massa (m) de uma partícula vezes sua aceleração (a) é igual a soma das forças (f) que agem sobre a mesma ($f = m \times a$).

Como a aceleração a de uma partícula é dada como a derivada de sua velocidade v e a velocidade é a derivada de sua posição p, temos que $a = p''$. Assim, para se determinar a posição de uma partícula é necessário resolver equações diferenciais ordinárias de segunda ordem do tipo $p'' = f/m$.

Como tais equações são de segunda ordem, as mesmas não podem ser resolvidas por métodos numéricos de integração, como por exemplo os métodos de *Euler* ou *Runge-Kutta* (Press et al., 1992). Uma das formas de se contornar esse problema é dividir essa equação em duas equações de primeira ordem acopladas, $v' = a$ e $p' = v$, criando-se um sistema de equações diferenciais:

$$\begin{cases} v' = a = f/m \\ p' = v \end{cases}$$

Assim, o estado dinâmico do MM é obtido por métodos numéricos que envolvem iterações para aproximar o conjunto de equações diferenciais. Uma iteração típica pode ser: calcular as forças que são aplicadas às n partículas do sistema, usar essas forças para encontrar suas velocidades, e por fim empregar essas velocidades para definir as posições.

Como pode ser notado, as forças que agem sobre cada partícula determinam o comportamento do sistema, já que a aceleração é calculada em função dessas forças ($a = f/m$). No caso de um MM essas forças são calculadas considerando-se que partículas adjacentes estão conectadas por molas. Seja f a força agindo sobre a partícula p a partir da partícula q. Podemos calcular seu valor usando-se a lei de *Hooke*:

[3]Um partícula ponto-de-massa ideal é aquela em que o total da massa está concentrada num único ponto.

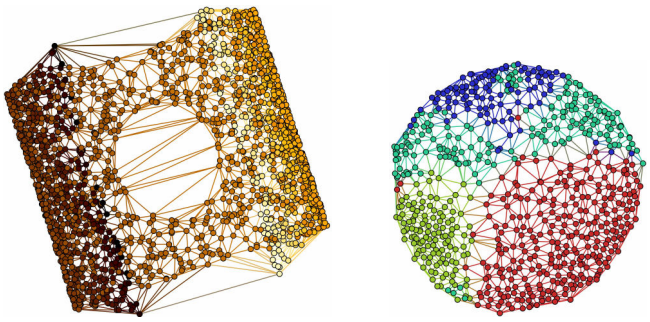
$$f = -k_s(|\overrightarrow{d}| - s)\frac{\overrightarrow{d}}{|\overrightarrow{d}|} \tag{2.7}$$

onde s é o tamanho da mola em repouso, k_s é a constante da mola e $\overrightarrow{d} = p - q$ é o vetor direção da força entre as partículas.

A Equação (2.7) mostra que quanto mais esticada estiver uma mola, maior será a força de atração entre as partículas que a mesma conecta, visando trazer a mola para um estado de repouso. Reciprocamente, quanto mais comprimida estiver a mola, maior será a força de repulsão entre as partículas.

Para que um sistema de molas possa ser usado como uma técnica de projeção multi-dimensional, o tamanho s da mola, no estado de repouso, deve ser proporcional à distância multi-dimensional entre os dois objetos representados pelas partículas p e q.

No caso geral, onde cada partícula está sujeita às forças de todas as outras partículas, o cálculo das forças é $O(n^2)$. Isto porque $n(n-1)$ cálculos de força em cada iteração são necessários. Já que para se gerar um layout estável são necessárias n iterações, o algoritmo resultante será $O(n^3)$ (Morrison et al., 2002a). Portanto, apesar dessa técnica gerar layouts que conseguem refletir com precisão as relações de distância entre os objetos, sua aplicação é limitada a pequenos conjuntos de dados. A Figura 2.5 apresenta o resultado da aplicação dessa técnica para os conjuntos de dados selecionados. Pode-se observar que o layout obtido é muito próximo do esperado para o conjunto **CBR-ILP-IR-SON**, mas a **Superfície-S** não conseguiu ser "desdobrada", indicando que essa técnica pode apresentar problemas se o espaço original a ser projetado contiver relações não-lineares entre os objetos.



(a) Conjunto de dados **Superfície-S**.

(b) Conjunto de dados **CBR-ILP-IR-SON**.

Figura 2.5: Layouts gerados usando-se o modelo baseado em molas. O resultado para o conjunto **CBR-ILP-IR-SON** está dentro do esperado, mas a **Superfície-S** não conseguiu ser "desdobrada".

2.3.2.2 Abordagem de Chalmers

Uma abordagem alternativa para tornar linear a complexidade de cada iteração de um MM foi apresentada por Chalmers (1996). Nessa abordagem, ao invés de se fazer todos os $n(n-1)$

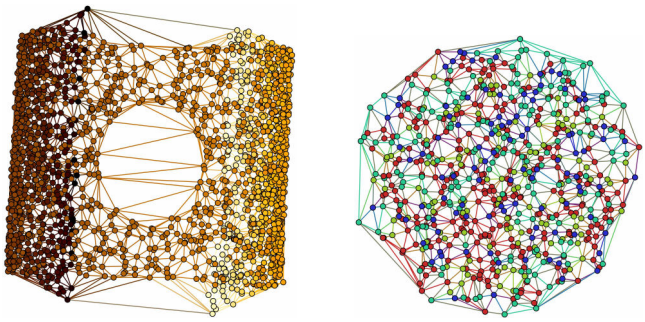
cálculos de força, são feitos cálculos entre cada objeto $\mathbf{x}_i$ e os membros de duas listas com tamanhos limitados.

A primeira lista V_i armazena as referências aos objetos vizinhos de $\mathbf{x}_i$, tendo tamanho máximo $Vmax$. Junto com V_i é mantido um valor, $maxDist_i$, que é a maior distância multi-dimensional entre $\mathbf{x}_i$ e qualquer membro de V_i. Enquanto a lista de vizinhos é mantida entre iterações, a segunda lista é construída a cada passo. Esta lista é formada por um sub-conjunto S_i de objetos escolhidos aleatoriamente que não pertençam a V_i. O tamanho da mesma é definido por uma constante $Smax$.

Em cada iteração, elementos são selecionados para serem inseridos em S_i. Toda vez que um elemento é selecionado, a dissimilaridade $\delta(\mathbf{x}_i, \mathbf{x}_j)$ é calculada. Se $\delta(\mathbf{x}_i, \mathbf{x}_j) < maxDist_i$, então $\mathbf{x}_j$ é inserido na lista V_i ao invés de S_i (observe que isso talvez mude $maxDist_i$). Esse processo é repetido até que a lista S_i tenha $Smax$ elementos. Quando isso ocorrer, as forças em $\mathbf{x}_i$ são calculadas usando um número limitado de partículas (menor ou igual à $Vmax + Smax$), reduzindo a complexidade do cálculo das forças.

Note que conforme a amostragem aleatória continua, a lista S_i evolui na direção de conter os $Smax$ elementos mais próximos de $\mathbf{x}_i$. Uma vez que adições de elementos às listas S_i tenham terminado, o processo de criação do layout geralmente se encerra entre n e $3n$ iterações.

A Figura 2.6 mostra os layouts gerados usando-se essa abordagem. Aqui novamente a **Superfície-S** não conseguiu ser "desdobrada". Além disso, apesar dessa técnica efetivamente reduzir o custo computacional do modelo de molas, quando o conjunto de dados apresenta relações de distância não tão bem definidas, o resultado final acaba sendo prejudicado, como pode ser observado pela projeção do conjunto **CBR-ILP-IR-SON** (Figura 2.6(b)).



(a) Conjunto de dados **Superfície-S**.

(b) Conjunto de dados **CBR-ILP-IR-SON**.

Figura 2.6: Layouts gerados usando-se a abordagem de Chalmers (1996). Os resultados indicam que as aproximações utilizadas para reduzir o custo computacional podem levar a criação de layouts de baixa qualidade.

2.3.2.3 Modelo Híbrido e baseado em Pivôs

Embora o algoritmo apresentado por Chalmers (1996) torne linear a complexidade das iterações, como são necessárias no mínimo n iterações, o processo como um todo ainda terá complexidade

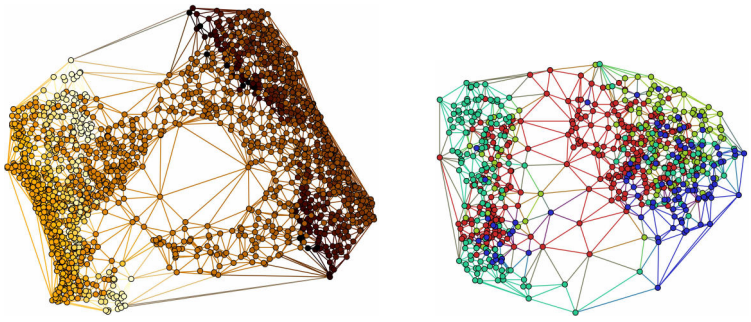
alta, O(n^2). Dessa forma, uma abordagem híbrida foi apresentada em (Morrison et al., 2002b) (depois estendida em (Morrison et al., 2003)) visando reduzir a complexidade para O($n\sqrt{n}$), mantendo a qualidade dos layouts gerados.

Nessa abordagem, uma amostragem aleatória S de $\sqrt{n}$ objetos do conjunto de dados é projetada no plano usando-se o método de Chalmers. Após isso, os objetos restantes são interpolados empregando-se uma versão modificada da estratégia proposta por Brodbeck e Girardin (1998).

Para se realizar essa interpolação é necessário definir o parente de cada um dos $n - \sqrt{n}$ objetos restantes, isto é, que não fazem parte da amostra S. Para isso, cada um desses objetos é comparado com cada um dos $\sqrt{n}$ objetos amostrados, procurando-se dentro da amostra o objeto com a menor distância multi-dimensional. Este estágio de definição de parentesco é o fator dominante dessa técnica, tornando o algoritmo O($n\sqrt{n}$) como um todo.

De forma a diminuir essa complexidade, Morrison e Chalmers (2004) apresentaram uma nova abordagem para a busca dos objetos parentes, derivada da técnica apresentada em (Burkhard e Keller, 1973), onde todos os relacionamentos de alta-dimensionalidade são tratados como uma lista de distâncias discretizadas para um número constante de ítens aleatoriamente selecionados. Com essa modificação, a complexidade do algoritmo é reduzida para O($n^{\frac{5}{4}}$) no caso médio.

A Figura 2.7 apresenta os resultados para essa técnica. Aqui, novamente, é possível notar que as aproximações que são usadas tornam essa técnica menos precisa, gerando layouts piores do que os modelos apresentados anteriormente. Assim, no caso dessa técnica e da anterior, as modificações realizadas no modelo de molas para se conseguir diminuir a complexidade computacional acabam afetando a qualidade do layout gerado de forma a torná-los inviáveis para a aplicação em conjuntos de dados mais complexos, como as coleções de documentos.



(a) Conjunto de dados **Superfície-S**.

(b) Conjunto de dados textbfCBR-ILP-IR-SON.

Figura 2.7: Layouts resultantes da aplicação do modelo híbrido. Com mais aproximações para se reduzir a complexidade computacional, os resultados se tornam ainda piores.

2.3.2.4 Force Scheme

Uma abordagem diferente das apresentadas anteriormente foi proposta por Tejada et al. (2003), chamada de *Force Scheme (FS)*. Essa abordagem, também baseada no conceito de forças de

atração e repulsão entre objetos, não trabalha empregando-se equações diferenciais, mas simplifica esse processo utilizando deslocamentos em direções determinadas.

A base para esse abordagem é a seguinte: seja $\mathbf{Y}$ um conjunto de pontos já posicionados no plano (usando-se um método de projeção qualquer ou mesmo um posicionamento aleatório), para cada ponto projetado $\mathbf{y}_i \in \mathbf{Y}$, um vetor $\overrightarrow{v_{ij}} = (\mathbf{y}_j - \mathbf{y}_i), \forall \mathbf{y}_j \neq \mathbf{y}_i$ é calculado. Então, uma perturbação em $\mathbf{y}_i$ é aplicada na direção de $\overrightarrow{v_{ij}}$. Essa perturbação depende das distâncias atuais e ideais entre os pontos projetados. Essa abordagem, também chamada de *Interactive Document Map (IDMAP)* (Minghim et al., 2006), é resumida no Algoritmo 2.1.

Algoritmo 2.1 Force Scheme.

entrada: - $\mathbf{Y}$: pontos já posicionados no plano.
- k: número de iterações.

saída: - $\mathbf{Y}$: pontos projetados com relações de distância melhoradas.

1: **para** n=1 **até** k **faça**
2: **para todo** $\mathbf{y}_i \in \mathbf{Y}$ **faça**
3: **para todo** $\mathbf{y}_j \in \mathbf{Y}$ com $\mathbf{y}_j \neq \mathbf{y}_i$ **faça**
4: Calcular $\overrightarrow{v}$ como sendo o vetor de $\mathbf{y}_i$ para $\mathbf{y}_j$.
5: Mover $\mathbf{y}_j$ em direção de $\overrightarrow{v}$ uma fração de Δ.
6: **fim para**
7: **fim para**
8: Normalizar as coordenadas da projeção na faixa $[0, 1]$ em ambas as dimensões.
9: **fim para**

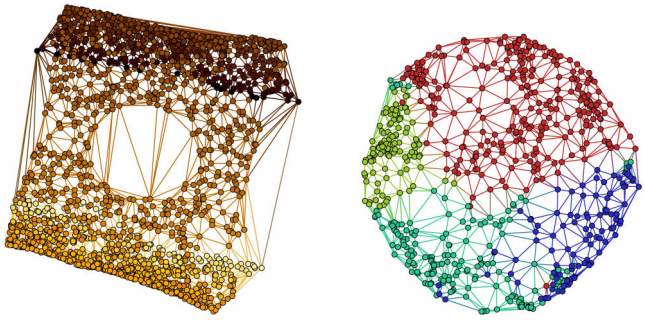
Nesse algoritmo, Δ representa a aproximação entre a distância no espaço reduzido e a distância no espaço original. Essa aproximação é dada pela Equação (2.8), onde δ_{max} e δ_{min} representam a máxima e a mínima distâncias entre objetos no espaço original, e $\mathbf{y}_i$ e $\mathbf{y}_j$ representam as projeção dos pontos $\mathbf{x}_i$ e $\mathbf{x}_j$.

$$\Delta = \frac{\delta(\mathbf{x}_i, \mathbf{x}_j) - \delta_{min}}{\delta_{max} - \delta_{min}} - d(\mathbf{y}_i, \mathbf{y}_j) \tag{2.8}$$

Essa abordagem, apesar de ser $O(n^2)$ por iteração, acaba convergindo em um número menor de iterações se comparado as abordagens tradicionais baseadas somente em molas. Isso acontece pois, em uma iteração cada um dos n objetos têm seu posicionamento alterado apenas uma vez. Já na abordagem usada pelo FS, em uma iteração cada objeto têm seu posicionamento alterado $n-1$ vezes, o que significa que uma iteração realiza muito mais trabalho sem ser mais complexa do que uma iteração do modelo original de molas.

A Figura 2.8 apresenta projeções geradas usando-se essa técnica. Comparativamente, o resultado alcançado é melhor do que o das aproximações anteriormente apresentadas para se reduzir o custo computacional do modelo de molas original. Essa melhoria pode ser vista como resultado dessa técnica usar informação local e global no processo de posicionamento dos pontos, o que difere das técnica anteriores que são baseadas em vizinhanças locais. Porém,

um efeito negativo de se considerar todas as distâncias é que as grandes distâncias acabam dominando o processo de geração do layout, distorcendo as vizinhanças locais. Isso pode ser notado na Equação (2.8) onde grandes valores de $\delta(\mathbf{x}_i, \mathbf{x}_j)$ resultarão em maiores forças de deslocamento. Como resultado, para conjuntos de dados com objetos muito dissimilares, como é o caso do **CBR-ILP-IR-SON**, existe uma tendência de ocorrer grandes concentrações de pontos nas fronteiras exteriores dos layouts gerados, com menor concentração no centro da projeção (Figura 2.8(b)). Outro efeito de se considerar informação global é que a técnica também não foi capaz de desdobrar a **Superfície-S** (Figura 2.8(a)).



(a) Conjunto de dados **Superfície-S**.

(b) Conjunto de dados **CBR-ILP-IR-SON**.

Figura 2.8: Layouts gerados usando-se a técnica *Force Scheme*. Os resultados são melhores do que as aproximações anteriores, porém vizinhanças locais podem ser distorcidas devido às grandes distâncias.

2.3.3 Multidimensional Scaling (MDS)

Originado em trabalhos da área de psicofísica, o *Multidimensional Scaling (MDS)* pode ser definido como um mapeamento injetivo entre objetos pertencentes a um espaço m-dimensional em pontos em um outro espaço p-dimensional ($p \leq m$) buscando-se preservar relações de distância. Assim, se $\delta(\mathbf{x}_i, \mathbf{x}_j)$ for a dissimilaridade entre dois objetos $\mathbf{x}_i$ e $\mathbf{x}_j$ e $d(\mathbf{y}_i, \mathbf{y}_j)$ for a distância entre os pontos referentes a tais objetos, busca-se aproximar $d(\mathbf{y}_i, \mathbf{y}_j)$ de $\delta(\mathbf{x}_i, \mathbf{x}_j)$ para todo par de objetos m-dimensionais. A forma como essa aproximação entre as distâncias é realizada é que leva a definição de diferentes técnicas de MDS, sendo normalmente divididas em duas grandes classes (Cox e Cox, 2000): *MDS métrico* e *MDS não-métrico*.

O objetivo do MDS métrico é encontrar uma configuração de pontos de tal forma que a distância entre os mesmos são relacionadas às dissimilaridades entre os objetos por alguma função de transformação g. Nesse caso, o termo "métrico" se refere ao tipo de transformação das dissimilaridades, e não ao espaço no qual a configuração de pontos é buscada (Cox e Cox, 2000). Dentre os métodos de MDS métricos mais conhecidos temos o *Classical Scaling* e o *Least Squares Scaling*. No *Classical Scaling*, se as dissimilaridades entre os objetos forem distâncias Euclideanas, a igualdade $d(\mathbf{x}_i, \mathbf{x}_j) = \delta(\mathbf{y}_i, \mathbf{y}_j)$ pode ser satisfeita fazendo-se uma decomposição espectral da matriz de dissimilaridades. Já no *Least Squares Scaling*, uma configuração dos

pontos é buscada fazendo com que $d(\mathbf{y}_i, \mathbf{y}_j) \approx g(\delta(\mathbf{x}_i, \mathbf{x}_j))$ por meio da minimização de uma função de perda.

No MDS não-métrico a idéia da preservação exata das distâncias, isto é de que $d(\mathbf{x}_i, \mathbf{x}_j) \approx \delta(\mathbf{y}_i, \mathbf{y}_j)$, é substituída pela definição de uma relação monotônica das distâncias entre os pontos com as dissimilaridades entre os objetos. Assim, a transformação g pode agora ser uma função arbitrária, somente obedecendo uma restrição de monotonicidade, isto é:

$$\delta(\mathbf{x}_i, \mathbf{x}_j) < \delta(\mathbf{x}_k, \mathbf{x}_n) \Rightarrow g(\delta(\mathbf{x}_i, \mathbf{x}_j)) < g(\delta(\mathbf{x}_k, \mathbf{x}_n)), \qquad \forall \mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k, \mathbf{x}_n \in \mathbf{X}$$

Dessa forma, somente o *rank* das dissimilaridades é preservado pela transformação g e portanto o uso do termo não-métrico.

A seguir as técnicas de *Classical Scaling*, um exemplo de *Least Squares Scaling*, definida por Sammon (1969), e uma de MDS não-métrica, desenvolvida por Kruskal (1964), são detalhadas e seus resultados para os conjuntos de dados selecionados apresentados.

2.3.3.1 Classical Scaling

A técnica *Classical Scaling* foi proposta na década de 1930 quando Young e Householder (1938) demonstraram como encontrar as coordenadas de um conjunto de pontos partindo de uma matriz contendo as distâncias entre esses em um espaço Euclideano. Posteriormente, essa ganhou popularidade quando Torgerson (1952) a sugeriu como uma técnica de MDS, trabalhando com dissimilaridades em geral, não necessariamente com distâncias Euclideanas.

Formalmente, sejam $\mathbf{x}_i$ $(i = 1, \ldots, n)$ as coordenadas de n pontos em um espaço Euclideano m-dimensional, onde $\mathbf{x}_i = (x_{i1}, \ldots, x_{im})^T$, e seja $\mathbf{B}$ a matrix do produto interno entre vetores,

$$[\mathbf{B}]_{ij} = b_{ij} = \mathbf{x}_i^T \mathbf{x}_j$$

com a distância Euclideana entre os pontos i e j dada por

$$\hat{\delta}(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^T (\mathbf{x}_i - \mathbf{x}_j) \tag{2.9}$$

A idéia do *Classical Scaling* é a partir da matriz das distâncias $\{\hat{\delta}(\mathbf{x}_i, \mathbf{x}_j)\}$, encontrar a matriz do produto interno $\mathbf{B}$, e a partir de $\mathbf{B}$ calcular as coordenadas dos pontos.

Para se encontrar a matrix $\mathbf{B}$, considerando-se que o centróide dos pontos reside na origem, isto é, que $\sum_{i=1}^{n} x_{ij} = 0$ $(j = 1, \ldots, m)$, para se evitar problemas com soluções indeterminadas devido a translações arbitrárias, é possível demonstrar que $\mathbf{B}$ pode ser reescrita como (consulte (Cox e Cox, 2000) para maiores detalhes) :

$$\mathbf{B} = \mathbf{HAH} \tag{2.10}$$

onde $\mathbf{A}$ é a matriz $[\mathbf{A}]_{ij} = a_{ij} = -\frac{1}{2}\hat{\delta}(\mathbf{x}_i, \mathbf{x}_j)$, e $\mathbf{H}$ é a matriz de centragem,

$$\mathbf{H} = \mathbf{I} - n^{-1}\mathbf{1}\mathbf{1}^T$$

com $\mathbf{1} = (1, 1, 1, \ldots, 1)^T$ um vetor com n coordenadas iguais a 1.

Considerando que $\mathbf{B}$ pode ser expressa como $\mathbf{B} = \mathbf{X}\mathbf{X}^T$, onde $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]^T$, é a matriz $n \times m$ das coordenadas. O rank de $\mathbf{B}$ será:

$$rank(\mathbf{B}) = rank(\mathbf{X}\mathbf{X}^T) = rank(\mathbf{X}) = m$$

Dessa forma, $\mathbf{B}$ é uma matriz simétrica, positiva semi-definida com rank m, e portanto apresenta m autovalores não negativos e $n - m$ autovalores nulos.

A matriz $\mathbf{B}$ pode então ser escrita em termos de sua decomposição espectral como:

$$\mathbf{B} = \mathbf{V}\Lambda\mathbf{V}^T$$

onde $\Lambda = diag(\lambda_1, \ldots, \lambda n)$ é a matriz diagonal dos autovalores de $\mathbf{B}$, e $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_n]$, é a matriz dos autovetores correspondentes, normalizados de forma que $\mathbf{v}_i^T\mathbf{v}_i = 1$. Por conveniência, os autovalores de $\mathbf{B}$ são rotulados de forma que $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n \geq 0$.

Como existem $n - m$ autovalores nulos, $\mathbf{B}$ pode ser reescrita como:

$$\mathbf{B} = \mathbf{V}_1\Lambda_1\mathbf{V}_1^T$$

onde $\Lambda_1 = diag(\lambda_1, \ldots, \lambda_p)$ e $\mathbf{V}_1 = [\mathbf{v}_1, \ldots, \mathbf{v}_p]$.

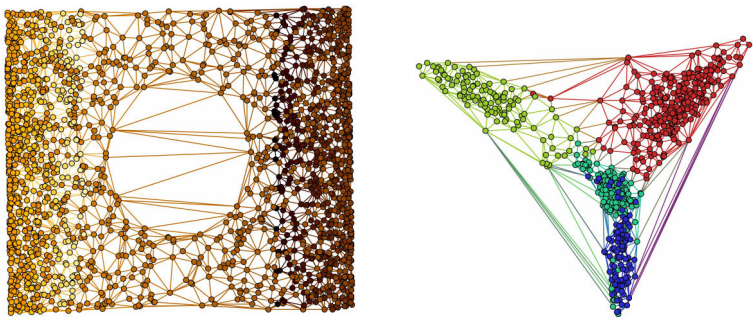
Dessa forma, como $\mathbf{B} = \mathbf{X}\mathbf{X}^T$, a matriz de coordenadas $\mathbf{X}$ é dada por:

$$\mathbf{X} = \mathbf{V}_1\Lambda_1^{-\frac{1}{2}},$$

onde $\Lambda_1^{-\frac{1}{2}} = diag(\lambda_1^{\frac{1}{2}}, \ldots, \lambda_m^{\frac{1}{2}})$. Caso a matriz de distâncias $\{\hat{\delta}(\mathbf{x}_i, \mathbf{x}_j)\}$ entre os pontos não seja Euclideana – o que normalmente é o caso das dissimilaridades $\delta(\mathbf{x}_i, \mathbf{x}_j)$ que podem nem mesmo configurar uma métrica (ver Seção 2.2) – pode acontecer de $\mathbf{B}$ não ser positiva semi-definida, apresentando autovalores negativos. Nesse caso, duas opções são possíveis: (i) descartar esses autovalores e seus respectivos autovetores na formação das matrizes Λ_1 e $\mathbf{V}_1$; ou (ii) adicionar uma constante apropriada c às dissimilaridades e repetir o processo novamente (Cox e Cox, 2000).

Da forma como aqui foi discutido, os pontos $\mathbf{X}$ residirão em um espaço m-dimensional no caso das distâncias Euclideanas, ou próximo disso para outras dissimilaridades, formando o espaço de menor dimensão que consegue representar os pontos preservando as distâncias fornecidas como entrada. Caso se queira que $\mathbf{X}$ seja um espaço p-dimensional, com $p < m$, os p autovetores que apresentarem os p maiores autovalores devem ser utilizados. Assim, no caso de uma projeção bi-dimensional, os dois autovetores com maiores autovalores devem ser empregados.

A Figura 2.9 apresenta o resultado obtido aplicando-se a técnica *Classical Scaling* para os conjuntos de dados selecionados. Para o conjunto **Superfície-S** o layout gerado (Figura 2.9(a)) não conseguiu ser "desdobrado", não resultando na superfície bi-dimensional esperada. Para o conjunto **CBR-ILP-IR-SON**, os resultados foram bastante satisfatórios (Figura 2.9(b)), separando bem as diferentes classes de documentos presentes nesse conjunto. Somente apresentando alguma mistura entre classes para os conjuntos IR e SON, um resultado já esperado.



(a) Conjunto de dados **Superfície-S**.

(b) Conjunto de dados **CBR-ILP-IR-SON**.

Figura 2.9: Layouts gerados usando-se a técnica *Classical Scaling*. Resultados satisfatórios, porém, essa não pode ser aplicada a grandes conjuntos de dados devido seu custo computacional ($O(n^2)$).

Apesar dos resultados apresentados por essa técnica serem bastante satisfatórios, a mesma apresenta uma grande restrição quanto a sua aplicabilidade a grandes conjuntos de dados, isto é, sua complexidade computacional, $O(n^2)$. Assim, apesar da grande precisão alcançada, sua aplicação não é aconselhável ou mesmo possível para conjuntos médios ou grandes, isto é, com mais de alguns milhares de objetos.

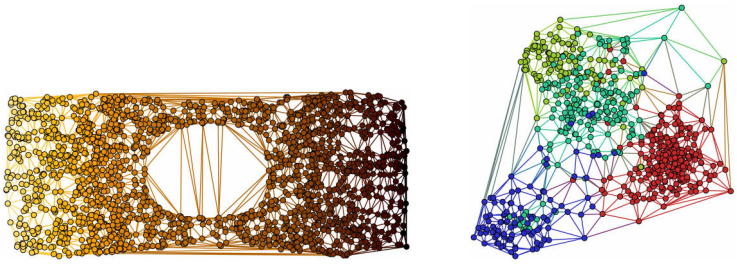
2.3.3.2 Isometric Feature Mapping (ISOMAP)

Uma variante da *Classical Scaling* que pode lidar com dados que apresentem relações não-lineares foi definida por Tenenbaum (1998); Tenenbaum et al. (2000), conhecida como *Isometric Feature Mapping (ISOMAP)*. Na verdade, a ISOMAP não é uma nova técnica de MDS, mas sim uma forma de se transformar as relações de distância entre os objetos multi-dimensionais antes que a verdadeira projeção seja criada. Dessa forma, apesar de originalmente ser definido que a *Classical Scaling* deva ser aplicada para o posicionamento dos pontos bi-dimensionais, outras técnicas de MDS podem ser empregadas nesse processo.

A idéia da ISOMAP é, ao invés de empregar distâncias Euclideanas ou outra dissimilaridade entre os objetos multi-dimensionais, utilizar distâncias geodésicas. Dado um grafo $G(A, V)$, onde A denota suas arestas e V seus vértices, a distância geodésica entre dois vértices $p, q \in V$ é dada pelo menor caminho entre p e q em G. Assim, para se calcular tais distâncias é criado um grafo onde V são os objetos multi-dimensionais e as arestas A ligam cada objeto aos seus k vizinhos mais próximos, ponderadas de acordo com as distâncias multi-dimensionais. Fazendo

uso desse grafo, as novas distâncias são calculadas empregando o algoritmo de Dijkstra (1959) para se determinar o caminho mais curto entre pares de vértices desse grafo de forma que a distância $\delta(\mathbf{x}_i, \mathbf{x}_j)$ é substituída pelo caminho mais curto entre i e j.

A Figura 2.10 apresenta resultados aplicando-se a ISOMAP. Na Figura 2.10(a) a **Superfície-S** conseguiu ser "desdobrada", revelando que essa técnica, por usar relações locais, consegue com sucesso lidar com dados que apresentam relações não-lineares. Além disso, pela Figura 2.10(b) é possível notar que a transformação das relações de distância no caso de espaços com distribuição de distâncias não muito comportadas não afeta negativamente o resultado final. O problema dessa técnica continua sendo a complexidade computacional, $O(n^2)$, sofrendo as mesmas limitações da *Classical Scaling* apresentada anteriormente.



(a) Conjunto de dados **Superfície-S**.

(b) Conjunto de dados **CBR-ILP-IR-SON**.

Figura 2.10: Resultados da aplicação da técnica ISOMAP. Por trabalhar com relações locais entre os objetos, a **Superfície-S** conseguiu ser "desdobrada", indicando que tal técnica pode lidar com sucesso com dados não-lineares.

2.3.3.3 Least Squares Scaling (LSS)

A técnica *Least Squares Scaling (LSS)* busca encontrar um configuração de distâncias $d(\mathbf{y}_i, \mathbf{y}_j)$ que se aproximem de $\delta(\mathbf{x}_i, \mathbf{x}_j)$ minimizando uma função de perda S. Na verdade a LSS não é uma técnica, mas sim um conjunto de técnicas, uma vez que variando-se a função de perda S, diferentes resultados serão obtidos. Algumas referências desse conjunto de técnicas são (Sammon, 1969; Spaeth, 1969; Chang e Lee, 1973; Bloxom, 1978). Dentre essas, a técnica apresentada em (Sammon, 1969), geralmente chamada de *Sammon's Mapping (SM)*, é uma das mais conhecidas dentro da área de visualização de informação.

Na SM a seguinte função de perda é minimizada:

$$S_1 = \frac{1}{\sum_{i<j} \delta(\mathbf{x}_i, \mathbf{x}_j)} \sum_{i<j} \frac{(d(\mathbf{y}_i, \mathbf{y}_j) - \delta(\mathbf{x}_i, \mathbf{x}_j))^2}{\delta(\mathbf{x}_i, \mathbf{x}_j)} \tag{2.11}$$

Como essa função é ponderada por $\delta(\mathbf{x}_i, \mathbf{x}_j)^{-1}$, as pequenas dissimilaridades terão maior peso que as grandes, tornando o SM capaz de desdobrar dados de *manifolds* de alta dimensão. Esta opção é preferida às ponderações drásticas como $\delta(\mathbf{x}_i, \mathbf{x}_j)^{-2}$ que não auxiliam a atingir

um balanço entre preservação local e global de estruturas, particularmente quando se está trabalhando dentro de pequenas vizinhanças (Martín-Merino e Muñoz, 2004).

Na SM, as distâncias $d(\mathbf{y}_i, \mathbf{y}_j)$ e as dissimilaridades $\delta(\mathbf{x}_i, \mathbf{x}_j)$ normalmente são Euclideanas, sendo que para a minimização da função de perda é usado um método iterativo não-linear que emprega o gradiente dessa função para se encontrar um mínimo local (Pekalska et al., 1999). A m-ésima iteração desse método é definida pela Equação (2.12).

$$y_{pq}(m+1) = y_{pq}(m) - MF \times \Delta_{pq}(m) \tag{2.12}$$

onde y_{pq} denota a coordenada q do ponto p,

$$\Delta_{pq}(m) = \frac{\partial S_1(m)}{\partial y_{pq}(m)} \Big/ \left| \frac{\partial^2 S_1(m)}{\partial y_{pq}^2(m)} \right| \tag{2.13}$$

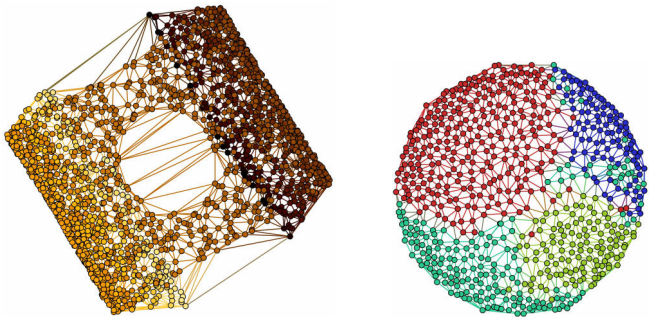
e $0 < MF \leq 1$ é um "fator mágico" que serve para otimizar a convergência do algoritmo, sendo determinado originalmente como $0.3 \leq MF \leq 0.4$. Para acelerar a convergência a um mínimo, Chang e Lee (1973) usaram um método de relaxamento heurístico para determinar o MF e Niemann e Weiss (1979) usaram um tamanho ótimo de passo calculado a cada iteração ao invés do MF.

A Figura 2.11 mostra os resultados da técnica SM para os conjuntos de dados selecionados. Pela Figura 2.11(a) é possível verificar que, apesar dessa técnica ser identificada como uma técnica que consegue lidar com dados não-lineares, a mesma não conseguiu "desdobrar" a **Superfície-S**. Para o conjunto **CBR-ILP-IR-SON**, apesar das diferentes classes não se misturarem, elas não seriam visualmente distinguíveis se não fosse a cor. Isso porque nesse processo de otimização, apesar das pequenas distâncias terem maior peso que as grandes, as grandes distâncias também são levadas em consideração, e como pode ser visto pelo histograma de distâncias desse conjunto, as instâncias de dados são relacionadas a somente uma pequeno número de vizinhos. Assim, considerar as grandes distâncias no processo de otimização acaba distorcendo o layout final.

Para tentar sobrepujar o problema dessa técnica não conseguir desdobrar *manifolds* altamente torcidos, Demartines e Herault (1997) definiram uma nova técnica, conhecida como *Curvilinear Component Analysis (CCA)*, que emprega uma nova função de perda que ignora totalmente distâncias maiores do que um limiar estabelecido, e Yang (2004) definiu uma técnica, conhecida como *GeoNLM*, que usa distâncias geodésicas ao invés de Euclideanas. Visando solucionar o problema das instâncias serem somente relacionadas localmente, Martín-Merino e Muñoz (2004) propuseram uma nova função de perda, em substituição à original, que só atua sobre uma vizinhança dos pontos. Essa nova função é apresentada na Equação (2.14).

$$S_2 = \frac{1}{\sum_i \sum_{j \in V_i} \delta(\mathbf{x}_i, \mathbf{x}_j)} \sum_i \sum_{j \in V_i} \frac{(\delta(\mathbf{x}_i, \mathbf{x}_j) - d(\mathbf{y}_i, \mathbf{y}_j))^2}{\delta(\mathbf{x}_i, \mathbf{x}_j)} \tag{2.14}$$

onde V_i é uma lista de vizinhos de $\mathbf{x}_i$.



(a) Conjunto de dados **Superfície-S**.

(b) Conjunto de dados **CBR-ILP-IR-SON**.

Figura 2.11: Layouts gerados usando-se a técnica Sammon's Mapping. Apesar de ser considerada uma técnica não-linear de projeção, a mesma não conseguiu "desdobrar" a **Superfície-S**.

Outra modificação para melhorar o processo em espaços onde as instâncias são localmente relacionadas é alterar o cálculo da distância Euclideana dando maior peso para as variáveis localmente mais discriminantes. Assumindo que os atributos com normas L_1 pequenas têm menor poder de discriminação, a nova dissimilaridade pode ser definida como (Lebart et al., 1989):

$$\delta(\mathbf{x}_i, \mathbf{x}_j) = \sum_{s,|x_s|>0} \frac{1}{\sum_{j \in V_i} |x_{js}|} (x_{is} - x_{js})^2 \tag{2.15}$$

onde a soma é feita sobre as variáveis que são sempre diferentes de zero dentro de V_i. Note que $\delta(\mathbf{x}_i, \mathbf{x}_j)$ é uma medida não simétrica, portanto a proximidade entre objetos é dada pelo componente simétrico $\delta(\mathbf{x}_i, \mathbf{x}_j)^{(s)} = (\delta(\mathbf{x}_i, \mathbf{x}_j) + \delta(\mathbf{x}_j, \mathbf{x}_i))/2$ (Martín-Merino e Muñoz, 2001). Essa dissimilaridade ignora o efeito de atributos locais ruidosos e assim ajuda a conseguir um histograma de distâncias mais suave (Aggarwal, 2001; Aggarwal e Yu, 2002).

Outro problema apresentado pela abordagem original do SM é sua complexidade, $O(n^2)$. Isso ocorre porque a função de perda é baseada em $O(n^2)$ distâncias. Assim, para melhorar essa complexidade Pekalska et al. (1999) apresentaram algumas estratégias para acelerar o processo. A idéia principal dessas estratégias se baseia em projetar somente um subconjunto de t objetos multi-dimensionais, com $t < n$; fixar os mesmos e interpolar os $n - t$ objetos restantes. Como estratégias de interpolação foram sugeridas: *Triangulação* (Biswas et al., 1981; Lee et al., 1977), *Mapeamentos de Distância* (Pekalska et al., 1999) e *Redes Neuronais Artificiais (RNA)* (de Ridder e Duin, 1997).

2.3.3.4 Otimização por Simulated Annealing

Um dos problemas não discutidos anteriormente que pode trazer problemas ao método SM é que o mesmo minimiza a função de perda até chegar a um mínimo local, e não um mínimo global. Assim, Klein e Dubes (1989) definiram uma nova função de perda:

$$S_3 = \sum \frac{1}{\sqrt{\sum \delta(\mathbf{x}_i, \mathbf{x}_j)}} \sum \frac{|d(\mathbf{y}_i, \mathbf{y}_j) - \delta(\mathbf{x}_i, \mathbf{x}_j)|}{\delta(\mathbf{x}_i, \mathbf{x}_j)} \tag{2.16}$$

e empregaram o método de *Simulated Annealing (SA)* (Kirkpatrick et al., 1983) ao invés do método de gradientes descendentes para a otimização. Diferentemente do método de gradientes descendentes que sempre visa diminuir o valor de S em cada passo, é permitido que o valor aumente de forma a superar um mínimo local. O ponto negativo é que o método de SA é computacionalmente caro, portanto não podendo ser empregado para grandes bases de dados.

2.3.3.5 MDS não-métrico

A suposição de que as dissimilaridades devem ser preservadas o máximo possível quando os objetos são projetados, definida no MDS métrico, pode ser muito restritiva em alguns casos, podendo levar a criação de layouts pouco efetivos. Um exemplo seria a projeção e exploração de julgamentos individuais de diferentes pessoas sobre algum objeto (a medida poderia ser qualidade, intensidade, etc.). Nesse caso, o problema ocorre porque a magnitude dessas medidas pode não ser confiável por serem percepções individuais de quem está julgando (Agarwal et al., 2007), podendo estar em escalas diferentes. Apesar disso, a ordem relativa desse tipo de medida normalmente é bastante consistente (Kendall e Gibbons, 1990), de forma que a preservação ordinal das dissimilaridades é preferível sobre a preservação das magnitudes. As técnicas de MDS não-métrico são mais apropriadas nesse caso.

O problema definido pelo MDS não-métrico foi inicialmente considerado por Shepard (1962a,b), mas foi Kruskal (1964) que definiu esse como um problema de otimização e introduziu um procedimento alternativo de minimização para resolvê-lo. Nesse processo de otimização busca-se encontrar uma configuração de pontos que minimize a diferença quadrática entre as dissimilaridades e as distâncias entre esses no layout final. Essa função de minimização, conhecida como *stress*, é definida como:

$$S_4 = \sqrt{\frac{\sum_{i<j} (\delta(\mathbf{x}_i, \mathbf{x}_j) - d(\mathbf{y}_i, \mathbf{y}_j))^2}{\sum_{i<j} \delta(\mathbf{x}_i, \mathbf{x}_j)^2}} \tag{2.17}$$

Para a minimização da função S_4, primeiro todas as coordenadas dos pontos em $\mathbf{X}$ são colocados em um vetor $\mathbf{x} = (x_{11}, \ldots, x_{1m}, \ldots, x_{nm})^T$, um vetor com $n \times m$ elementos. Então, a função S_4 é considerada como um função de $\mathbf{x}$, e minimizada com relação a $\mathbf{x}$ de forma iterativa usando-se o método de gradientes descendentes. Assim, se $\mathbf{x}(m)$ é o vetor de coordenadas após a m-ésima iteração, a minimização se dá resolvendo:

$$\mathbf{x}(m+1) = \mathbf{x}(m) - \frac{\partial S_4(m)}{\partial x(m)} \Big/ \left| \frac{\partial S_4(m)}{\partial x(m)} \right| * sl \tag{2.18}$$

onde sl define o tamanho do passo de otimização em cada iteração.

Apesar dessa técnica ser dita não-métrica, é interessante observar que a mesma emprega a magnitude das dissimilaridades no processo de otimização, já que a função a ser otimizada S considera essas medidas (apenas com uma normalização no denominador desse função). Além disso, comparando-se a Equação (2.18) com as Equações (2.13) e (2.12), é possível notar que o método definido por Sammon (ver Seção 2.3.3.3), dito ser um método de MDS métrico, é muito semelhante ao método definido por Kruskal. A diferença está em se otimizar um função de perda com uma normalização diferente (compare as Equações (2.11) e (2.17)). Dessa forma, as limitações presentes na algoritmo de Sammon também são esperadas nesse algoritmo, como a não capacidade de lidar com objetos que apresentem relações não-lineares entre os atributos e a dificuldade em criar layouts representativos para espaços onde as relações de distância não são tão bem definidas.

2.3.4 Técnicas de Redução de Dimensionalidade

Técnicas para redução de dimensionalidade são técnicas que, dado um conjunto de variáveis aleatórias $\mathbf{x}_i = (x_{i1}, \ldots, x_{im})$, buscam encontrar uma representação de menor dimensão $\mathbf{y}_i = (y_{i1}, \ldots, y_{ip})$, com $p < m$, que capture o conteúdo original dos dados de acordo com algum critério (Fodor, 2002). Assim, de maneira geral, fazendo-se $p = \{1, 2, 3\}$ e usando como critério de redução a preservação de relações de distância, é possível se aplicar tais tipos de técnicas no contexto de projeções multi-dimensionais.

Considerando as técnicas de redução de dimensionalidade como funções f que transformam $\mathbf{X}$ em $\mathbf{Y}$ é possível classificá-las em dois grandes grupos, das técnicas lineares e não-lineares (Fodor, 2002). Uma técnica linear pode ser definida como:

Definição 2.2 (Redução de Dimensionalidade Linear (Kirby, 2001)) *Uma técnica de redução de dimensionalidade $f : \mathbf{X} \rightarrow \mathbf{Y}$ é dita ser linear se $f(\alpha\mathbf{x}_i + \beta\mathbf{x}_j) = \alpha f(\mathbf{x}_i) + \beta f(\mathbf{x}_j)$ para todo $\mathbf{x}_i, \mathbf{x}_j \in \mathbf{X}$ e $\alpha, \beta \in \mathbb{R}$.*

Uma técnica é dita não-linear se a mesma não obedecer essa definição.

Como as técnicas lineares reduzem as dimensões por meio da combinação linear entre os diferentes atributos que definem os dados m-dimensionais, o resultado final será satisfatório, isto é, conseguirá expressar as estruturas dos dados originais, somente quando existir uma dependência linear entre tais atributos. Quando esses dados apresentarem estruturas não-lineares, como agrupamentos de formato arbitrário ou *manifolds* curvos, a melhor escolha seria a utilização de uma técnica de redução não-linear – apesar dessa constatação ser válida quando se está buscando realmente reduzir as dimensões, isto é, definir um conjunto com dimensionalidade menor ou igual à dimensionalidade intrínseca dos dados[4], se mais dimensões forem adicionadas ao conjunto reduzido, uma técnica linear conseguirá representar razoavelmente espaços não-lineares (esse é o conceito empregado pelo *kernel trick*(ver Seção 2.3.4.1)).

[4] A dimensionalidade intrínseca pode ser vista como o menor número de dimensões independentes que são necessárias para se gerar um certo conjunto de dados (Bennet, 1969; Verveer e Duin, 1995).

Nas próximas seções algumas técnicas para redução de dimensionalidade são detalhadas. Inicialmente é apresentada uma técnica linear de segunda ordem, a *Principal Component Analysis (PCA)* (Jolliffe, 1986). Técnicas de segunda ordem são as que empregam somente informação contida na matriz de covariância dos atributos para realizar a redução, sendo apropriadas para dados que apresentem distribuição Gaussiana (normal), já que em tais casos toda a distribuição dos dados pode ser capturada. Após isso, é apresentada uma técnica mais apropriada para dados não-gaussianos, por utilizar informação que não está contida somente na matriz de covariância, conhecida como *Projection Pursuit (PP)* (Friedman e Tukey, 1974). Em seqüencia, uma técnica não-linear é descrita, a *Local Linear Embedding (LLE)* (Roweis e Saul, 2000). Por fim, é apresentada uma técnica baseada em relações de distância, portanto podendo ser linear ou não-linear dependendo da métrica empregada, conhecida como FastMap (Faloutsos e Lin, 1995).

Para uma discussão mais detalhada e uma revisão mais completa sobre métodos de redução de dimensionalidade consulte (Carreira-Perpiñán, 1996, 2001; Fodor, 2002).

2.3.4.1 Principal Components Analysis (PCA)

A *Principal Component Analysis (PCA)* (Jolliffe, 1986), também conhecida como *Expansão de Karhunen-Loéve* (Fukunaga, 1990; Duda e Hart, 1973) ou *Empirical Orthogonal Functions* (Lorenz, 1956), é uma técnica utilizada para redução de dimensionalidade que visa combinar as dimensões (atributos) dos dados em um conjunto menor de dimensões. A PCA tem várias características interessantes, como a tendência de identificar os padrões mais relevantes nos dados, conseguir capturar a maior parte da variabilidade com poucas dimensões, eliminar grande parte do "ruído" existente, etc.

O processo utilizado pela PCA é baseado em se determinar combinações lineares ortogonais, os chamados *Componentes Principais (CPs)*, que melhor capturem a variabilidade dos dados. Nesse processo, o primeiro componente principal será a combinação linear com maior variância, o segundo componente será a combinação linear, ortogonal à primeira, com maior variância, e assim por diante. Existem tantos componentes principais quanto o número original de atributos, mas normalmente os primeiros componentes capturam a maior parte da variância dos dados de forma que a maioria pode ser descartada com uma pequena perda de informação (sobre a variância).

Para se aplicar a PCA, primeiro uma matriz de covariância dos atributos (colunas da matriz de dados $\mathbf{X}$) é criada. Essa é uma matriz $\mathbf{C}_{m\times m}$, onde m é o número de atributos dos dados, com seus termos $c_{ij} = cov(a_i, a_j)$, onde $cov(a_i, a_j)$ representa a covariância entre os atributos a_i e a_j.

Sejam x e y dois atributos, a covariância entre os mesmos é dada por

$$cov(x, y) = \frac{1}{n-1}\sum_{k=1}^{n}(x_k - \bar{x})(y_k - \bar{y}) \tag{2.19}$$

onde

$$\bar{x} = \frac{1}{n}\sum_{k=1}^{n} x_k, \qquad \bar{y} = \frac{1}{n}\sum_{k=1}^{n} y_k$$

são as médias de x e y, respectivamente.

Uma vez que a matriz de covariância tenha sido calculada, aplica-se uma decomposição espectral sobre a mesma de forma a encontrar seus autovetores. Para isso, $\mathbf{C}$ pode ser escrita como:

$$\mathbf{C} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{\mathbf{T}}, \tag{2.20}$$

onde $\mathbf{\Lambda} = diag(\lambda_1, ..., \lambda_m)$ é uma matriz diagonal dos autovalores ordenados de forma decrescente $\lambda_1 \geq ... \geq \lambda_m$, e $\mathbf{U}$ é uma matriz $m \times m$ ortogonal contendo os autovetores.

Empregando-se essa matriz, os componentes principais podem ser calculados por

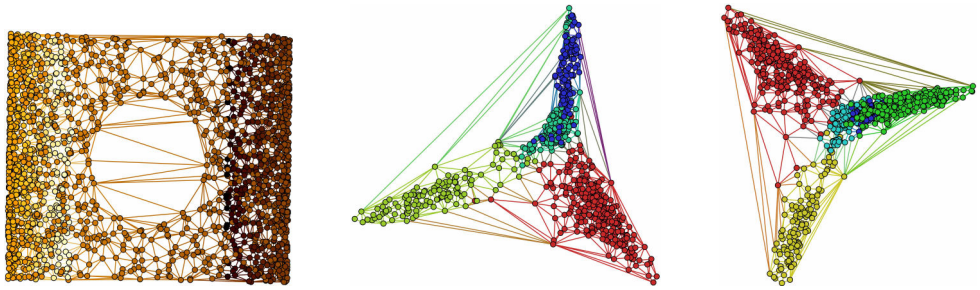
$$\mathbf{S} = \mathbf{X} * [u_1, u_2, ..., u_p]$$

onde u_i representam as colunas da matriz $\mathbf{U}$ e p é o número de dimensões requerido.

É possível mostrar (Mardia et al., 1995) que o subespaço definido pelos p primeiros autovetores tem o menor desvio médio quadrado de $\mathbf{X}$ entre todos os subespaços de dimensão p. Isto é, dentre todas as matrizes $\mathbf{A}_{n\times m}$ com *rank* no máximo p, a matriz $\mathbf{S}$ é a que minimiza $\|\mathbf{X} - \mathbf{A}\|_F^2 = \sum_{i,j}(x_{i,j} - a_{i,j})^2$, onde o subscrito F denota a norma de Frobenius (Golub e Reinsch, 1971). Portanto, PCA preserva o máximo possível as distâncias (Euclideanas) relativas entre os dados enquanto os mesmos são projetados em um espaço de menor dimensão – essa é uma característica apresentada também pela técnica de MDS *Classical Scaling* (veja Seção 2.3.3.1), sendo que os resultados obtidos aplicando-se essa técnica são idênticos aos apresentados usando-se PCA se for considerado que as distâncias no espaço original são Euclideanas (Cox e Cox, 2000).

A Figura 2.12 apresenta projeções usando-se PCA. Na Figura 2.12(a) é possível observar que essa técnica não foi capaz de "desdobrar" a **Superfície-S**, um resultado esperado já que a mesma é baseada em combinações lineares dos atributos originais e tende a ter problemas com dados que apresentem relações não-lineares. Nas Figuras 2.12(b) e 2.12(c) a principal limitação de se usar essa técnica para criar layouts bi-dimensionais para conjuntos de dados de alta dimensão pode ser notada. Como são usados dois componentes principais para se representar os dados, somente duas direções de grande variância são capturadas, de forma que se existirem diversos grupos distintos dentro do conjunto de dados, com diferentes direções de variância, essa técnica não consegue representar satisfatoriamente todos esses grupos. Apesar de PCA conseguir separar bem os quatro grandes grupos de documentos que existem no conjunto **CBR-ILP-IR-SON** (Figura 2.12(b)), quando mais um grupo é adicionado a essa coleção de documentos, a separação entre os mesmos se perde, como mostra a Figura 2.12(c). Naquela figura, para gerar a projeção, 270 artigos científicos sobre visualização de informação forem adicionados ao conjunto original. Como resultado de utilizar somente dois componentes principais, os layouts resultantes

geralmente apresentam um "cotovelo" onde os dois componentes se cruzam, e onde a maioria das instâncias de dados é posicionada, isto é, aquelas instâncias que não foram bem representadas pelos dois componentes principais utilizados.



(a) Conjunto de dados **Superfície-S**.

(b) Conjunto de dados **CBR-ILP-IR-SON**.

(c) Conjunto de dados **CBR-ILP-IR-SON** com mais 270 artigos sobre visualização de informação.

Figura 2.12: Projeções empregando PCA. Para conjuntos com alta dimensão os layouts bi-dimensionais tendem a não representar satisfatoriamente todos os grupos distintos dentro do conjunto de dados uma vez que somente dois componentes principais são utilizados.

Outras alternativas similares ao PCA foram propostas, como o PCA baseado em redes neurais (Carreira-Perpiñán, 1996; Kramer, 1991), o *Latent Semantic Index (LSI)* (Deerwester et al., 1990), que emprega decomposição espectral sobre os dados originais e não sobre a matriz de covariância, e o *Singular Value Decomposition (SVD)* (Demmel, 1997), que não retira a média no cálculo da covariância.

Para dados que apresentam relações não-lineares existe uma variante da PCA que é mais indicada, conhecida como *Kernel PCA* (Schölkopf et al., 1997, 1998, 1999). Essa técnica provê análise não-linear dos componentes principais empregando um mecanismo conhecido como *kernel trick* (Aizerman et al., 1964). Esse mecanismo permite transformar um espaço não-linear em um espaço de maior dimensão que apresente relações lineares entre as dimensões. O *kernel trick* é baseado no teorema de Mercer (Mercer, 1909), que afirma que qualquer função kernel $K(x, y)$ contínua, simétrica, positiva semi-definida pode ser expressa como o produto interno em um espaço de alta dimensão. Assim, em princípio, esse mecanismo pode transformar qualquer técnica que depende do produto interno entre vetores em uma técnica não-linear. O interessante é que não é necessário transformar os dados, criando uma nova representação de alta dimensão, mas somente usar uma função $K(x, y)$ onde o produto interno ocorra. Dessa forma, a técnica *Kernel PCA* é o emprego da PCA original, mas substituindo os produtos internos por funções $K(x, y)$.

Apesar das limitações inerentes de se aplicar PCA original para fazer projeções bi-dimensionais, o IN-SPIRE^{TM} (PNNL, 2008), que é um sistema comercial, emprega uma técnica baseada em PCA para a criação de mapas de documentos. Essa técnica, conhecida como *Anchored Least Stress (ALS)* (Wise, 1999), primeiramente projeta um pequeno sub-conjunto de instâncias de

dados (as âncoras) no plano usando PCA e depois, fazendo uso do posicionamento das âncoras, projeta as instâncias restantes usando uma estratégia de interpolação. Assim, os layouts gerados acabam herdando algumas das limitações da PCA para projeções em poucas dimensões. A Figura 2.13 apresenta um exemplo da aplicação dessa técnica para uma coleção de notícias curtas de jornal (*RSS news feeds*) coletada da Internet. Nessa figura, os pontos marcados em laranja são as âncoras (esse mapa de documentos foi gerado usando-se o IN-SPIRE^{TM}).

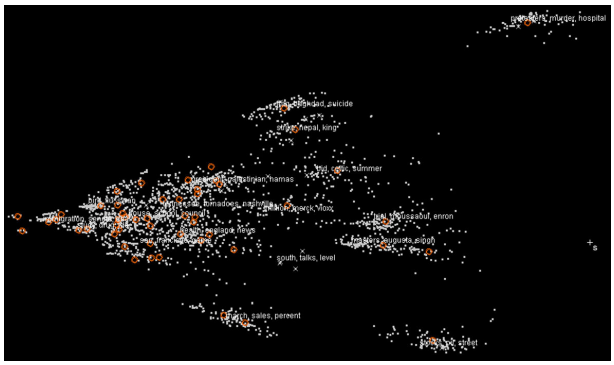


Figura 2.13: Projeção gerada usando-se a técnica implementada no IN-SPIRE^{TM}. Por ser uma interpolação baseada na projeção PCA de uma amostra dos dados originais as mesmas limitações de PCA podem ser verificadas nos resultados apresentados, como a mistura de grupos em um extremo do gráfico.

2.3.4.2 Projection Pursuit

A *Projection Pursuit (PP)* (Sun, 1993; Friedman e Tukey, 1974; Friedman, 1987; Huber, 1985; Jones e Sibson, 1987) é uma técnica desenvolvida no campo da estatística que, diferente da técnica PCA, pode incorporar mais informação do que informação de segunda ordem, portanto sendo útil para conjunto de dados não-gaussianos (Fodor, 2002). A PP visa encontrar projeções de dados multi-dimensionais que podem ser usadas para a visualização da estrutura de agrupamentos dos dados, e para propósitos como estimativa de densidade e regressão.

Na PP básica (unidimensional), busca-se encontrar uma direção $\mathbf{w}$ de tal forma que a projeção dos dados em tal direção, $\mathbf{w}^T\mathbf{x}$, apresente uma distribuição "interessante", isto é, apresente alguma estrutura. Foi sugerido por Huber (1985) e por Jones e Sibson (1987) que a distribuição Gaussiana é a menos interessante, e que as direções mais interessantes são aquelas que mais se diferenciam da distribuição Gaussiana.

Assim, a PP busca reduzir as dimensões de forma que algumas das características "interessantes" dos dados são preservadas, diferente do PCA onde o objetivo é reduzir a dimensão de forma que a representação conseguida seja a mais próxima possível dos dados originais em um sentido de mínimos quadrados.

O ponto central da PP é a definição e otimização de um *índice de projeção* que define as direções mais "interessantes". Normalmente, esse índice é alguma medida não-normal, sendo

a escolha mais natural a *entropia diferencial* (Jones e Sibson, 1987), também conhecida como *entropia negativa de Shannon* (Huber, 1985).

O problema com a entropia diferencial é que essa requer a estimativa da densidade $\mathbf{w}^T\mathbf{x}$, o que é difícil na teoria e na prática. Portanto, outras medidas não-Gaussianas têm sido propostas (Cook et al., 1993; Friedman, 1987). Estas são baseadas na distância ponderada L_2 entre a densidade de x e a densidade Gaussiana multivariada. Outra possibilidade é empregar aproximações cumulativas da entropia diferencial (Jones e Sibson, 1987).

A PP é uma grande contribuição para a análise de dados de alta dimensionalidade, embora segundo Crawford e Fall (1990) a mesma ainda apresente várias limitações. Um dos problemas mais comuns se refere à dificuldade em determinar o que realmente as soluções encontradas significam para um dado *índice de projeção*. Além disso, a PP não tem a habilidade de fazer inferências, de forma que pode retornar falsas estruturas.

2.3.4.3 Local Linear Embedding (LLE)

A *Local Linear Embedding (LLE)* (Roweis e Saul, 2000, 2001) é uma técnica para redução de dimensionalidade baseada na suposição de que localmente os dados são lineares. Isto significa que pequenos pedaços em $\mathbb{R}^m$ devem ser aproximadamente iguais (a não ser pela rotação, translação e escala) a pequenos pedaços dos dados finais em $\mathbb{R}^p$. Portanto, relações locais entre dados em $\mathbb{R}^m$ que são invariantes sobre rotação, translação e escala deveriam ser (aproximadamente) válidas em $\mathbb{R}^p$. Usando este princípio, o procedimento para encontrar as coordenadas de baixa dimensão é o apresentado no Algoritmo 2.2.

Algoritmo 2.2 Algoritmo *Local Linear Embedding (LLE).*

entrada: - X: objetos a serem reduzidos.
saída: - Y: Representação de X no espaço p-dimensional.

1: Encontrar os vizinhos mais próximos de cada objeto em X.
2: Expressar cada objeto $\mathbf{x}_i$ como uma combinação linear dos outros objetos, isto é, $\mathbf{x}_i = \sum_j w_{ij}\mathbf{x}_j$, onde $\sum_j w_{ij} = 1$ e $w_{ij} = 0$ se $\mathbf{x}_j$ não é um vizinho próximo de $\mathbf{x}_i$.
3: Encontrar as coordenadas de cada objeto no espaço p-dimensional usando os pesos encontrados no passo 2.

No passo 2, a matriz de pesos $\mathbf{W}$, cujas entradas são w_{ij}, é encontrada minimizando-se uma aproximação do quadrado do erro como medida dada pela Equação (2.21). $\mathbf{W}$ pode ser encontrada resolvendo-se um problema de mínimos quadrados.

$$erro(\mathbf{W}) = \sum_i \left(\mathbf{x}_i - \sum_j w_{ij}\mathbf{x}_j \right)^2 \tag{2.21}$$

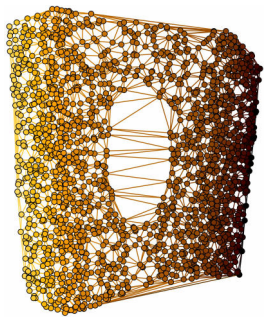
O passo 3 do procedimento realiza a verdadeira redução de dimensionalidade. Dada a matriz de pesos e o número de dimensões p, especificado pelo usuário, o algoritmo constrói

um "mapeamento preservando a vizinhança" dos dados em um espaço de menor dimensão. Se $\mathbf{y}_i$ é o vetor no espaço reduzido que corresponde a $\mathbf{x}_i$ e $\mathbf{Y}$ é a nova matriz de dados cuja i-ésima linha é $\mathbf{y}_i$, então essa projeção pode ser conseguida encontrando-se $\mathbf{Y}$ que minimize a seguinte equação:

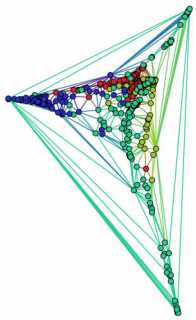
$$erro(\mathbf{Y}) = \sum_i \left(\mathbf{y}_i - \sum_j w_{ij}\mathbf{y}_j \right)^2 \tag{2.22}$$

Para a minimização dessa equação, um método baseado em autovetores é empregado. Na verdade, o que se minimiza é uma matriz $n \times n$ formada pela Equação (2.22). Assim, a LLE é uma técnica $O(n^2)$. Porém, métodos para cálculo de autovetores esparsos podem ser empregados de forma a reduzir essa complexidade.

A Figura 2.14 apresenta exemplos de projeções multi-dimensionais usando-se a LLE. Pela Figura 2.14(a) é possível notar que essa técnica conseguiu "desdobrar" a **Superfície-S**, sendo portanto uma técnica que, por considerar pequenas vizinhanças na projeção consegue com sucesso criar layouts para dados que apresentam relações não-lineares entre os atributos. Porém, pela Figura 2.14(b) nota-se que o resultado alcançado não foi satisfatório no caso de dados com alta dimensão. Isso se deve em grande parte ao fato da LLE não considerar informação global no processo de projeção, somente informação local – para se gerar essa projeção, o algoritmo foi executado com diferentes números de vizinhos mais próximos, sendo escolhido o layout que apresentou o melhor resultado visualmente (com oito vizinhos).



(a) Conjunto de dados **Superfície-S**.



(b) Conjunto de dados **CBR-ILP-IR-SON**.

Figura 2.14: Projeções utilizado-se LLE. Apesar dessa técnica conseguir efetivamente "desdobrar" a **Superfície-S**, para dados com alta-dimensão os resultados foram ruins pelo fato da mesma não considerar informação global dos dados no processo de projeção.

2.3.4.4 FastMap

Um outra técnica para redução de dimensionalidade, conhecida como *FastMap*, que também pode ser empregada para projeções bi-dimensionais foi definida por Faloutsos e Lin (1995). A FastMap visa mapear pontos de um espaço m-dimensional para pontos em um espaço p-dimensional,

projetando os mesmos em p direções mutuamente ortogonais. A idéia central dessa técnica é projetar recursivamente em p hiperplanos os objetos originais, sendo as coordenadas de cada uma das p dimensões determinadas como a projeção dos objetos sobre retas pertencentes a esses hiperplanos.

Cada uma dessas retas é definida escolhendo-se, dentro de cada hiperplano, dois objetos, O_a e O_b, denominados *pivôs*. Para se determinar a posição de um objeto O_i em relação aos dois pivôs escolhidos emprega-se a *Lei dos Cossenos*

$$\delta^2(O_b, O_i) = \delta(O_a, O_i)^2 + \delta(O_a, O_b)^2 - 2x_i\delta(O_a, O_b)$$

que, manipulada matematicamente, pode ser usada para definir a posição x_i da projeção do objeto O_i sobre a reta O_aO_b fazendo-se:

$$x_i = \frac{\delta(O_a, O_i)^2 + \delta(O_a, O_b)^2 - \delta(O_b, O_i)^2}{2\delta(O_a, O_b)} \tag{2.23}$$

Resolvido o problema para $p = 1$, o mesmo pode ser generalizado para qualquer valor de $p < m$. Para isso, considere que os objetos serão projetados em um hiperplano $(m - 1)$-dimensional perpendicular à reta O_aO_b. O problema acaba sendo o mesmo problema original, mas com m e k decrementados de um. A Figura 2.15 apresenta dois objetos O_i e O_j e suas projeções, O'_i e O'_j, sobre um hiperplano H.

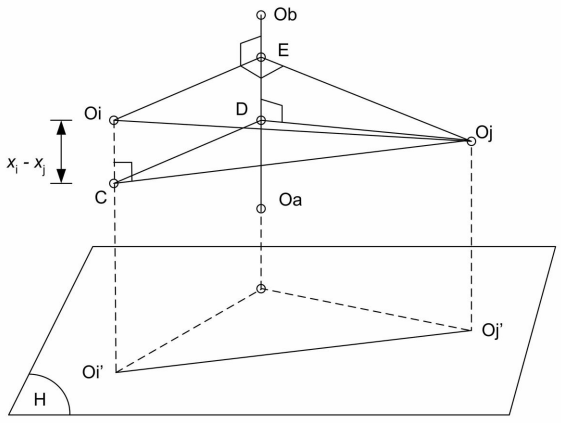


Figura 2.15: Projeção no hiperplano H, perpendicular à linha O_aO_b da figura anterior.

A última parte é determinar como as novas dissimilaridades $\delta(\mathbf{x}'_i, \mathbf{x}'_j)'$ são calculadas entre os objetos projetados nesse hiperplano. Para se determinar essas dissimilaridades, a Equação (2.24) apresenta a transformação da dissimilaridade original $\delta(\mathbf{x}_i, \mathbf{x}_j)$ na nova dissimilaridade $\delta(\mathbf{x}'_i, \mathbf{x}'_j)'$.

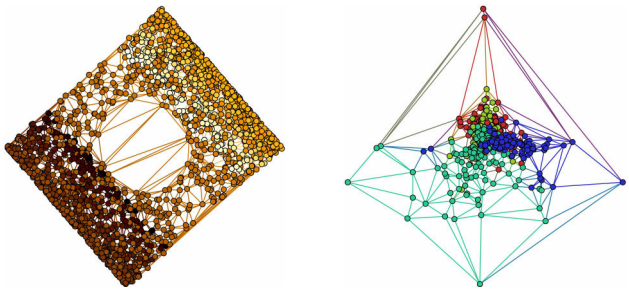
$$\delta(\mathbf{x}'_i, \mathbf{x}'_j)' = \sqrt{\delta(\mathbf{x}_i, \mathbf{x}_j)^2 - (x_i - x_j)^2} \tag{2.24}$$

A habilidade de se computar as dissimilatidades $\delta(\mathbf{x}'_i, \mathbf{x}'_j)'$ permite projetar os pontos em uma segunda linha que reside no hiperplano H, e portanto ortogonal à primeira linha O_aO_b.

Com isso os pontos podem ser projetados em um espaço bi-dimensional, ou melhor podem ser projetados em um espaço p-dimensional aplicando-se esse processo recursivamente p vezes.

Cabe ressaltar que para a FastMap obter boas projeções, a reta na qual os objetos serão projetados deve ser a maior reta que é possível formar com os objetos projetados no hiperplano; dessa forma os pivôs O_a e O_b devem ser os objetos projetados com maior distância. Embora esse tipo de busca tenha complexidade $\mathrm{O}(n^2)$, uma heurística foi proposta por Faloutsos e Lin (1995) que executa uma busca aproximada com complexidade $\mathrm{O}(n)$. Nesta heurística, um objeto qualquer do conjunto é inicialmente escolhido; em seguida procura-se o objeto mais distante a ele. Por fim, utilizando esse objeto encontrado procura-se o mais distante dele. Esses dois últimos objetos serão os pivôs.

Apesar da FastMap apresentar a menor complexidade, $\mathrm{O}(n)$, dentre as técnicas aqui apresentadas, a mesma leva a uma grande perda da informação quando usada para criar layouts bi-dimensionais. A Figura 2.16 apresenta duas projeções usando-se essa técnica. Pela Figura 2.16(a) é possível notar que a mesma não conseguiu "desdobrar" a **Superfície-S**, portanto não sendo efetiva para conjunto de dados não-lineares. E pela Figura 2.16(b) observa-se que com dados que apresentem distribuição de distâncias não muito comportadas, os resultados são bastante ruins, o que acaba limitando sua aplicação.



(a) Conjunto de dados **Superfície-S**.

(b) Conjunto de dados **CBR-ILP-IR-SON**.

Figura 2.16: Apesar da FastMap ser uma das técnicas com menor complexidade computacional, os resultados apresentados para dados com alta dimensionalidade não são satisfatórios, limitando a sua aplicação.

2.4 Avaliação das Projeções

Como pode ser observado pelas seções anteriores, existem diversas técnicas que podem ser empregadas para a criação de projeções multi-dimensionais, cada uma apresentando diferentes características que levam a diferentes resultados. Assim, uma forma de se avaliar objetivamente as projeções resultantes, que não somente uma inspeção visual baseada em critérios subjetivos, se torna necessária para que seja possível determinar comparativamente a qualidade dos layouts gerados.

Normalmente esse processo de avaliação é conduzido empregando uma função de *stress*, ou seja, uma das funções minimizadas pelas técnicas de MDS (veja Equações (2.11) e (2.17)), sendo considerado o melhor layout o que apresentar o menor valor dessa função. Apesar desse método ser bastante difundido, esse tipo de análise não é adequado por dois fatores principais. Primeiramente, layouts totalmente diferentes podem ter o mesmo valor de *stress*, e pequenas variações no *stress* podem levar a significativas modificações no layout gerado (Chalmers, 1996). Segundo, diferentes técnicas de projeção normalmente funcionam minimizando diferentes funções de *stress* (quando são baseadas em minimização, o que nem sempre é o caso), de forma que é esperado que uma técnica avaliada usando sua própria função seja melhor que outras que empregam funções diferentes. Assim, o resultado desse tipo de análise é sempre influenciado pela função empregada (Paulovich e Minghim, 2008).

Além desses problemas, é comum encontrar exemplos de incompatibilidade entre os resultados de *stress* e a interpretação visual dos layouts gerados. A Figura 2.17 apresenta um conjunto de dados bi-dimensional projetado sobre espaços uni-dimensionais. Quando os pontos são projetados sobre a direção horizontal (projeção **(2)**), a separação entre os dois agrupamentos antes visíveis no espaço original é perdida. Por outro lado, quando os dados são projetados sobre a direção vertical (projeção **(1)**), os dois agrupamentos são identificáveis no espaço reduzido. Apesar da projeção **(2)** ser visualmente inferior no que tange a representação das estruturas intrínsecas dos dados originais (os agrupamentos), o valor de *stress* apresentado, 0.26522127, é muito inferior ao apresentado pela projeção **(1)**, 0.87028027. Assim, é possível argumentar que nem sempre o menor valor de *stress* identifica a melhor projeção, ou a projeção mais útil para análise visual. Nessa figura, as projeções foram geradas simplesmente zerando as coordenadas-x para a projeção **(1)** e as coordenadas-y para a **(2)**. Os valores de *stress* foram calculados usando-se a função de Kruskal (Equação (2.17)) empregando-se distâncias Euclideanas, normalizadas entre $[0, 1]$ para evitar distorções.

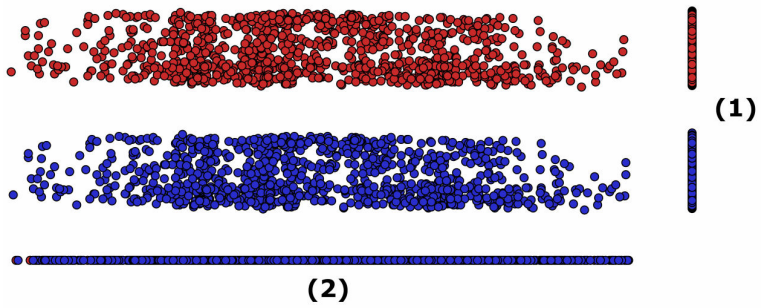


Figura 2.17: Avaliando diferentes projeções usando *stress*. Apesar da projeção **(1)** apresentar melhor resultado visual, seu *stress* é muito maior do que o calculado para projeção **(2)**, indicando que nem sempre o *stress* é uma medida confiável para a avaliação de projeções.

Isso mostra que o *stress*, apesar de ser extensivamente empregado para a avaliação da perda de informação em projeções multi-dimensionais, não é a medida ideal para se determinar comparativamente a qualidade de projeções multi-dimensionais no que tange a identificação

visual de estruturas relevantes nos dados sendo analisados. Em substituição ao emprego do *stress* definimos duas técnicas diferentes para a avaliação analítica de projeções multi-dimensionais.

A primeira, conhecida como *Neighborhood Hit* (Paulovich et al., 2008a), visa analisar se dada uma pré-classificação é possível identificar na projeção gerada a separação entre as diferentes classes existentes nos dados. A idéia é calcular os k vizinhos mais próximos de um ponto e verificar a proporção desses que pertencem à mesma classe desse ponto (a precisão final é uma média das precisões para cada ponto). Quanto mais separados e agrupados os pontos estiverem no layout final, de acordo com essas classes, maior será a precisão. Dessa forma, é possível avaliar numericamente quão (visivelmente) destacadas estão as classes pré-existentes na projeção final e a facilidade de encontrar fronteiras bem definidas entre as mesmas. A Figura 2.18 apresenta os resultados de precisão para as projeções do conjunto de dados **CBR-ILP-IR-SON** apresentadas nas seções anteriores. Pelos resultados é possível verificar que as três projeções que visivelmente foram os piores são identificados como as menos precisas: projeções usando-se a técnica de *Chalmers* (Seção 2.3.2.2), o *Modelo Híbrido* (Seção 2.3.2.3), e a técnica *FastMap* (Seção 2.3.4.4). As demais projeções praticamente apresentam a mesma precisão, sendo que os melhores resultados foram alcançados usando-se *PCA* (Seção 2.3.4.1) e *Classical Scaling* (Seção 2.3.3.1), o que coincide com a inspeção visual dessas projeções.

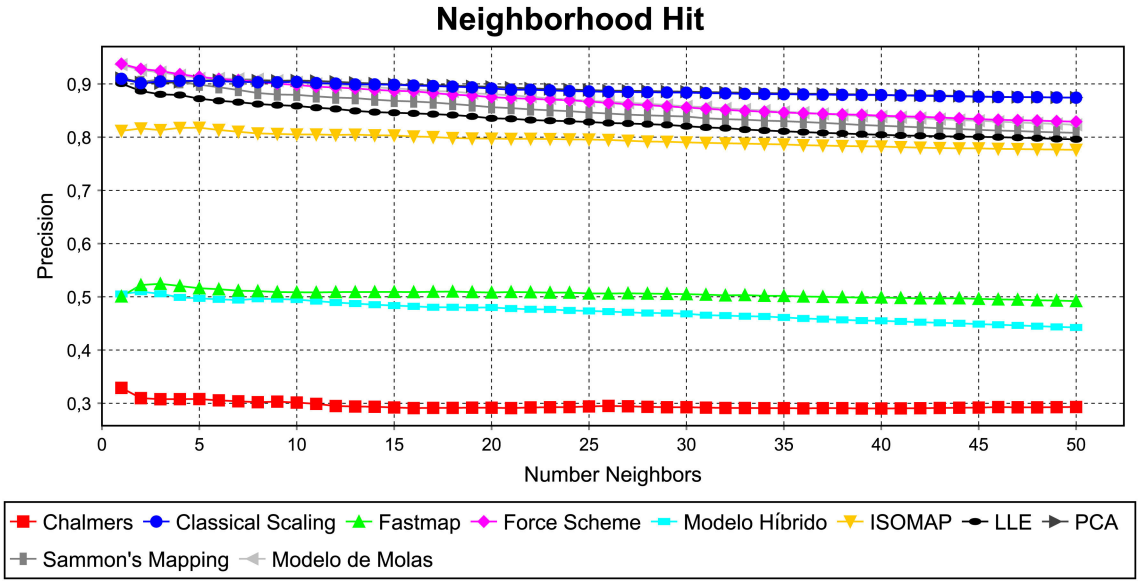


Figura 2.18: Avaliação das projeções usando-se a técnica *Neighborhood Hit*. As projeções que apresentaram os piores resultados visualmente foram as piores avaliadas.

Apesar desse tipo de avaliação poder identificar a separação entre as classes pré-existentes, a mesma não consegue verificar a qualidade dos grupos formados no layout final, nem o quanto as relações de distância são preservadas. Assim, uma segunda técnica, complementar a essa, chamada de *Neighborhood Preservation* (Paulovich e Minghim, 2008), foi definida. Essa visa avaliar a preservação da vizinhança dos objetos multi-dimensionais no layout final. A *Neighborhood Preservation* é calculada tomando os k vizinhos mais próximos de um objeto multi-dimen-

sional e os k vizinhos mais próximos da sua projeção, e verificando-se que proporção da vizinhança é preservada no layout. A precisão final é uma média das precisões para cada ponto.

A Figura 2.19 emprega essa avaliação para comparar os vários layouts produzidos nas seções anteriores para o conjunto de dados **CBR-ILP-IR-SON**. Aqui, novamente as técnicas com piores resultados visuais foram as que receberam os menores valores de precisão: a técnica de *Chalmers* (Seção 2.3.2.2), o *Modelo Híbrido* (Seção 2.3.2.3), e a técnica *FastMap* (Seção 2.3.4.4). Porém, as técnicas *PCA* (Seção 2.3.4.1) e *Classical Scaling* (Seção 2.3.3.1), que tinham sido interpretadas como as melhores na análise anterior, falham em preservar relações de distância e vizinhança entre os pontos projetados, uma característica que não é possível ser verificada somente pela inspeção visual. Considerando essas duas análises, é possível verificar que a técnica que apresentou um melhor compromisso entre separação entre classes e preservação da vizinhança, foi a *Force Scheme* (Seção 2.3.2.4), porém a um custo computacional muito grande.

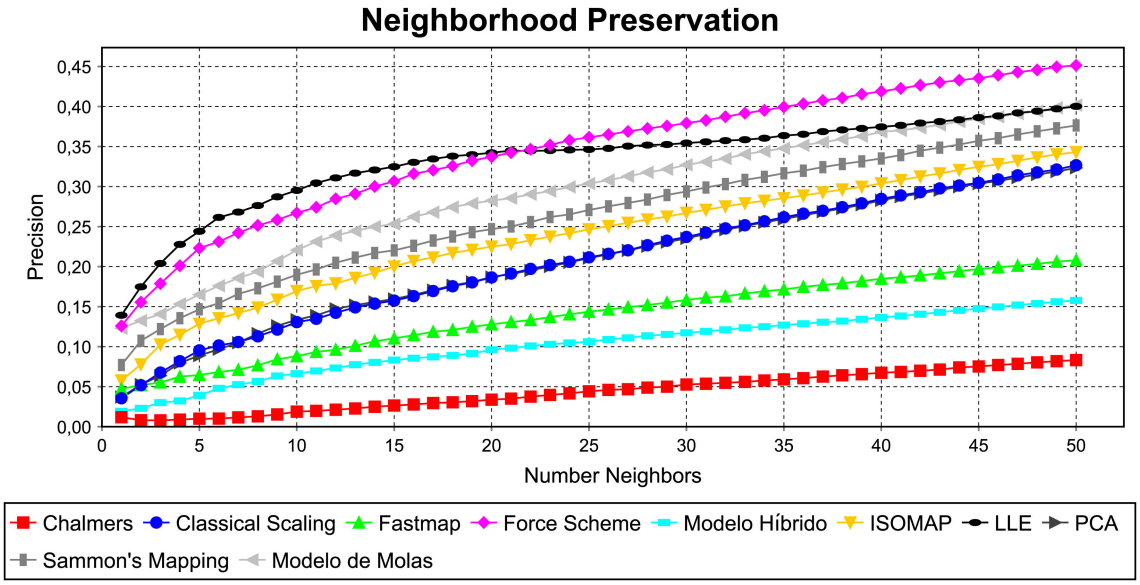


Figura 2.19: Avaliação das projeções criadas nas seções anteriores usando-se a técnica *Neighborhood Preservation*. Usando-se essa análise e a anterior é possível definir a técnica que melhor consegue separar as classes existentes nos dados e preservar a vizinhança dos objetos nos pontos projetados.

2.5 Considerações Finais

Nesse capítulo foi apresentado um estudo sobre técnicas de projeção multi-dimensional. Resumidamente, tais técnicas visam mapear injetivamente objetos ditos em um espaço m-dimensional em pontos em outro espaço p-dimensional, com $p < m$ e $p = \{1, 2, 3\}$, preservando o máximo possível as relações de distância existentes entre os objetos nessa projeção. Dessa forma, dados multi-dimensionais podem ser visualmente analisados através de estruturas que revelem características "interessantes", como agrupamentos, tendências e *outliers*.

Dentre as técnicas apresentadas aqui, as mesmas foram divididas em três grandes grupos: (1) *Force-Directed Placement (FDP)*; (2) *Multidimensional Scaling (MDS)*; e (3) técnicas de redução de dimensionalidade. Cada um desses grupos apresenta suas características positivas e negativas, mas pelo estudo realizado nesse capítulo algumas observações importantes podem ser feitas:

Complexidade computacional e aproximações: as técnicas que apresentaram os melhores resultados foram as de maior complexidade computacional. Porém, a aplicação das mesmas é limitada a conjunto de dados de tamanho menor. Por outro lado, as técnicas que utilizam aproximações para reduzir essa complexidade geraram resultados menos satisfatórios, o que inviabiliza sua utilização em conjuntos mais complexos, como as coleções de documentos. Dessa forma, o que se deve buscar é *reduzir a complexidade mantendo um compromisso com a qualidade do layout gerado.*

Informação local e global: no processo de projeção é possível verificar dois componentes diferentes; um que leva à aproximação das instâncias muito relacionadas (informação local) e outro que leva à separação dos grupos de elementos similares (informação global). Assim, para geração de layouts mais úteis, ambos os componentes devem ser levados em consideração. Porém, o grau de preservação de cada componente deve estar relacionado com a distribuição de distâncias do conjunto de dados. Por exemplo, para coleções de documentos a distribuição normalmente indicará que somente poucos documentos estão relacionados entre si (um efeito esperado para espaços esparsos e de alta-dimensão) de forma que a preservação de toda informação global irá afetar negativamente a projeção, uma vez que as grandes distâncias irão dominar o layout gerado, distorcendo as pequenas vizinhanças. Assim, deve-se buscar *preservar as relações globais, porém minimizando o efeito nas vizinhanças locais.*

Relações não-lineares: em conjuntos de dados reais, os atributos que descrevem os dados normalmente apresentam relações não-lineares entre si. Porém, considerando-se vizinhanças locais, geralmente essas relações são lineares e os objetos normalmente residem em um sub-espaço de menor dimensão. Com isso, para que uma técnica possa lidar com dados não-lineares com sucesso, a mesma deve ser baseada em relações de vizinhança, preservando as relações globais somente entre os grupos de dados. Isso leva a uma conclusão relacionada à anterior de que deve-se buscar *preservar vizinhanças locais para dados que apresentem relações não-lineares.*

Os próximos capítulos apresentam técnicas de projeção desenvolvidas no contexto desta tese, que visam reduzir os problemas apresentados pelas técnicas pré-existentes aqui apresentadas.

Capítulo

3

Projection by Clustering (ProjClus)

3.1 Considerações Iniciais

Como pode ser verificado no capítulo anterior, o problema de projeções multi-dimensionais tem sido objeto de pesquisa de diversos pesquisadores devido à variedade de aplicações que podem se beneficiar do uso de representações visuais que consigam revelar relações de similaridade entre objetos multi-dimensionais. Dentre os tipos de dados multi-dimensionais que têm recebido grande atenção estão as coleções de documentos. Acelerar a análise de grandes coleções é tarefa estratégica para instituições de ensino, empresas e órgãos governamentais.

Para a projeção de coleções de documentos vários fatores devem ser observados, como a esparsidade do espaço gerado, a distribuição das distâncias, etc., configurando normalmente um tipo de dado bastante difícil de se gerar representações visuais satisfatórias. A primeira técnica desenvolvida no contexto desse projeto de doutorado com o objetivo de projetar coleções de documentos, conhecida como *Interactive Document Map (IDMAP)* (Minghim et al., 2006), emprega a técnica FastMap (Seção 2.3.4.4) para se criar um posicionamento inicial dos pontos, seguida da técnica Force Scheme (FS) (Seção 2.3.2.4) para melhorar esse posicionamento. Na IDMAP, o emprego da FastMap para se criar um layout inicial tem como objetivo diminuir o número necessário de iterações da FS para se conseguir um layout onde as relações de distância são preservadas satisfatoriamente, uma vez que a projeção inicial já apresenta alguma preservação dessas relações.

Na verdade a IDMAP não configura uma nova técnica, mas sim uma combinação de técnicas existentes para a criação de projeções de coleções de documentos, herdando os problemas e

limitações das técnicas nas quais a mesma é baseada. Assim, embora essa técnica resulte em layouts que consigam refletir bem certas relações de similaridade entre documentos, a mesma não pode ser aplicada a grandes coleções devido à complexidade da FS, $O(n^2)$.

Buscando manter ou melhorar a qualidade dos layouts criados usando-se a IDMAP, mas reduzindo sua complexidade computacional, desenvolvemos uma nova técnica conhecida como *Projection by Clustering (ProjClus)* (Paulovich e Minghim, 2006). A seguir a ProjClus é detalhada, os resultados obtidos com sua aplicação são apresentados e é descrita a avaliação comparativa desses resultados.

3.2 Descrição da Técnica

O processo empregado pela ProjClus para projeção multi-dimensional pode ser dividido em três grandes passos: (1) primeiro os objetos multi-dimensionais são separados em agrupamentos de objetos similares; (2) esses agrupamentos são projetados individualmente no plano; e (3) essas projeções são unidas criando-se o layout final. Deste ponto em diante iremos nos referir a objetos multi-dimensionais, não somente a documentos, uma vez que essa e as demais técnicas desenvolvidas podem ser empregadas em qualquer conjunto de dados multi-dimensionais. Assim, busca-se fazer $|\delta(\mathbf{x}_i, \mathbf{x}_j) - d(\mathbf{y}_i, \mathbf{y}_j)|$ o mais próximo possível de zero $\forall \mathbf{x}_i \in \mathbf{X}$, porém com $\mathbf{x}_j \in S_i$, onde S_i é um conjunto contendo um número limitado de objetos multi-dimensionais relacionados com $\mathbf{x}_i$.

Uma vez que o objetivo é melhor representar a similaridade entre objetos multi-dimensionais, posicionando os objetos mais similares à $\mathbf{x}_i$ perto de $\mathbf{y}_i$ quando esses forem projetados, o conjunto S_i deve ser composto de objetos que pertençam a uma vizinhança de $\mathbf{x}_i$, isto é, $S_i = \{\mathbf{x} : \delta(\mathbf{x}_i, \mathbf{x}) < \varepsilon, \mathbf{x} \in \mathbf{X}\}$, com ε definido de forma a limitar o número de elementos em S_i. Embora a escolha direta seja definir o conjunto S_i contendo os vizinhos mais próximos de x_i, essa não é a melhor escolha devido à complexidade computacional desse tipo de busca ($O(n^2)$), mas principalmente porque esse processo dificilmente gera grupos de objetos disjuntos, normalmente resultando em conjuntos que compartilham objetos. Isso traria problemas, já que um mesmo objeto seria projetado no plano mais de uma vez. Assim, empregamos uma técnica de agrupamento baseada em similaridade para definir os conjuntos S_i, gerando grupos disjuntos e não comprometendo a complexidade final da técnica.

De forma a criar os agrupamentos, é empregado um algoritmo de agrupamento por particionamento, chamado *bisecting k-means* (Steinbach et al., 2000).

Embora seja pensamento corrente de que algoritmos de agrupamento aglomerativos, tais como o *Agrupamento Hierárquico* (Johnson, 1967) ou o *Unweighted Pair Group Method with Arithmetic Mean (UPGMA)* (Sneath e Sokal, 1973), sejam melhores do que os de particionamento, Zhao et al. (2005) apresentaram uma avaliação experimental mostrando que algoritmos de particionamento sempre levam a melhores soluções, argumentando que essa impressão se deve principalmente ao fato dos experimentos serem executados em conjuntos de dados com baixa

dimensionalidade, o que não é o caso de grande parte dos conjuntos de dados. Além disso, abordagens de particionamento são bastante convenientes para o agrupamento de grandes conjuntos de dados devido ao baixo custo computacional normalmente requerido por tais técnicas (Zhao e Karypis, 2002). No caso específico do *bisecting k-means* essa complexidade é $O(k \times n)$, onde k é o número de agrupamentos. Outra característica interessante dessa técnica é que ela se mostra menos sensível à escolha inicial das sementes para a geração dos agrupamentos se comparada com outras técnicas de particionamento, como por exemplo o algoritmo original *k-means* (Tan et al., 2005).

O algoritmo *bisecting k-means* funciona dividindo as instâncias de dados em hiper-esferas de forma que cada instância de dados esteja mais próxima do centro da hiper-esfera a que pertence do que de todos centros das outras hiper-esferas – isso para a medida de dissimilaridade (baseada no cosseno) empregada aqui para definir a similaridade entre documentos; se outra medida for utilizada, agrupamentos de outros formatos podem ser obtidos. Por exemplo, a distância *City Block* (veja Seção 2.2), levaria a uma divisão em hiper-cubos (Tan et al., 2005). Os agrupamentos são definidos por particionar os dados em pares de agrupamentos sucessivamente. Em cada particionamento, um agrupamento é escolhido e dividido em dois novos agrupamentos, até que o número de agrupamentos solicitado seja criado.

Existem várias formas de se escolher qual agrupamento será dividido, mas uma vez que a diferença entre esses métodos é pequena (Steinbach et al., 2000), optamos por usar uma abordagem simples: escolher o maior agrupamento (o que contém mais instâncias). Esta abordagem é rápida e tende a produzir agrupamentos de tamanhos semelhantes (Zhao e Karypis, 2002), uma característica importante para a ProjClus, como será discutido mais adiante – também testamos outras funções objetivo em tal tarefa, como as medidas de *coesão* e *separação* (Zhao e Karypis, 2004), o *coeficiente de silhueta* (Kaufman e Rousseeuw, 1990) e o *coeficiente de silhueta simplificada* (Hruschka et al., 2006), mas de fato os resultados finais não foram melhores. O Algoritmo 3.1 descreve o processo completo do *bisecting k-means*.

Uma vez que $k = \sqrt{n}$ (veja justificativa para esse número na Seção 6.2) agrupamentos tenham sido encontrados, o centróide[1] de cada um é calculado, resultando em um conjunto de pontos multi-dimensionais $C = \{\mathbf{c}_1, \ldots, \mathbf{c}_k\} \in \mathbb{R}^m$. Em seguida, cada agrupamento S_i é individualmente projetado usando a IDMAP, e as coordenadas dessas projeções são normalizadas em um intervalo proporcional à maior distância entre o centróide $\mathbf{c}_i$ de cada agrupamento e os elementos pertencentes aos mesmos. Por sua vez, o valor máximo de cada intervalo de normalização é dividido por uma constante F, que é um fator de densidade dos agrupamentos no $\mathbb{R}^2$ (conforme F cresce, os agrupamentos se tornam mais densos). Nos testes realizados o intervalo $4 \leq F \leq 10$ se mostrou apropriado para definir a densidade das projeções dos agrupamentos.

Finalmente, os centróides C são projetados usando-se a IDMAP, e a projeção de cada agrupamento é determinada no layout final de acordo com a projeção de seu centróide. Este

[1] O centróide de um agrupamento é, geralmente, a média aritmética das instâncias que pertencem a ele.

Algoritmo 3.1 Bisecting k-means.

entrada: - **X**: objetos multi-dimensionais a serem agrupados.
- k: número de agrupamentos a serem formados.
- $r = 10$: número de iterações (uma constante do algoritmo).

saída: - $S = \{S_1, ..., S_k\}$: lista de agrupamentos formados.

procedimento $Particionar(\mathbf{X}, k)$

1: $S_1 \leftarrow \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ *Formar um agrupamento contendo todos os objetos.*
2: Adicionar S_1 à lista de agrupamentos S.
3: **para** $i = 1$ **até** $k - 1$ **faça**
4: Selecionar um agrupamento S_p de S para ser particionado em dois. *Selecionar o agrupamento com maior número de elementos.*
5: Remover S_p da lista de agrupamentos S.
6: Criar dois novos agrupamentos S_{p1} e S_{p2}.
7: Selecionar dois objetos representativos $\mathbf{x}_{p1}, \mathbf{x}_{p2}$ de S_p. *Selecionar dois objetos distantes.*
8: $S_{p1} \leftarrow \{\mathbf{x}_{p1}\}$.
9: $S_{p2} \leftarrow \{\mathbf{x}_{p2}\}$.
10: **para** $j = 1$ **até** r **faça**
11: Calcular o centróide $\mathbf{x}_{c1}$ do agrupamento S_{p1}.
12: Calcular o centróide $\mathbf{x}_{c2}$ do agrupamento S_{p2}.
13: Remover todos elementos de S_{p1} e S_{p2}.
14: **para todo** objeto $\mathbf{x}_s \in S_p$ **faça**
15: **se** $\delta(\mathbf{x}_s, \mathbf{x}_{c1}) < \delta(\mathbf{x}_s, \mathbf{x}_{c2})$ **então**
16: Adicionar $\mathbf{x}_s$ à S_{p1}.
17: **senão**
18: Adicionar $\mathbf{x}_s$ à S_{p2}.
19: **fim se**
20: **fim para**
21: **fim para**
22: Adicionar S_{p1} e S_{p2} à lista dos agrupamentos S.
23: **fim para**

posicionamento final garante que agrupamentos similares sejam projetados próximos, e que os dissimilares sejam projetados distantes. Assim, objetos multi-dimensionais similares serão agrupados, e os dissimilares serão separados. Essa técnica é descrita no Algoritmo 3.2.

3.3 Complexidade Computacional

A complexidade computacional da ProjClus pode ser calculada como $O(H + P_k + P_c + J)$, onde H é a complexidade de se criar os agrupamentos, P_k é a complexidade de projetar esses agrupamentos, P_c é a complexidade para se projetar os centróides e J é a complexidade de se unir todas as projeções dos agrupamentos no layout final. A complexidade de se criar $\sqrt{n}$ agrupamentos é $O(n\sqrt{n})$. A complexidade de se posicionar $\sqrt{n}$ centróides usando-se a IDMAP é $O(n)$. A complexidade de se criar o layout final é $O(n)$. Finalmente, se tivermos em cada

Algoritmo 3.2 Projection by Clustering (ProjCLus).

entrada: - $\mathbf{X}$: pontos a serem projetados no plano.
- F: fator de densidade dos agrupamentos.

saída: - $\mathbf{Y}$: pontos projetados.

procedimento *ProjClus*($\mathbf{X}$)

1: $S = Particionar(\mathbf{X}, \sqrt{n})$ *Dividir os objetos multi-dimensionais em agrupamentos.*
2: **para todo** agrupamento $S_p \in S$ **faça** *Calcular os centróides dos agrupamentos.*
3: $c_{pj} = \frac{1}{|S_p|} \sum_{\mathbf{x}_i \in S_i} x_{ij}$ para toda coordenada $j = 1, \ldots, m$.
4: Adicionar $\mathbf{c}_p$ à lista de centróides C.
5: **fim para**
6: Projetar no plano os centróides C usando a IDMAP.
7: $\lambda_{max} = 0$ *Armazena o maior tamanho de agrupamento no plano.*
8: **para todo** agrupamento $S_p \in S$ **faça**
9: Calcular $\lambda_p = \max\{\delta(\mathbf{x}, \mathbf{c}_p), \forall \mathbf{x} \in S_p\}/F$.
10: **se** $\lambda_p > \lambda_{max}$ **então**
11: $\lambda_{max} = \lambda_p$.
12: **fim se**
13: **fim para**
14: **para todo** agrupamento $S_p \in S$ **faça** *Projetar cada agrupamento individualmente.*
15: Projetar os objetos multi-dimensionais de S_p no plano usando a IDMAP.
16: Calcular $\lambda_p = \max\{\delta(\mathbf{x}, \mathbf{c}_p), \forall \mathbf{x} \in S_p\}/F$.
17: Normalizar as coordenadas da projeção do agrupamento S_p entre $[0, \frac{\lambda_p}{\lambda_{max}}]$.
18: **fim para**
19: **para todo** agrupamento $S_p \in S$ **faça** *Compor a projeção final.*
20: Posicionar a projeção do agrupamento S_p no layout final de forma que o centróide da mesma coincida com a projeção do centróide c_p do agrupamento S_p.
21: **fim para**

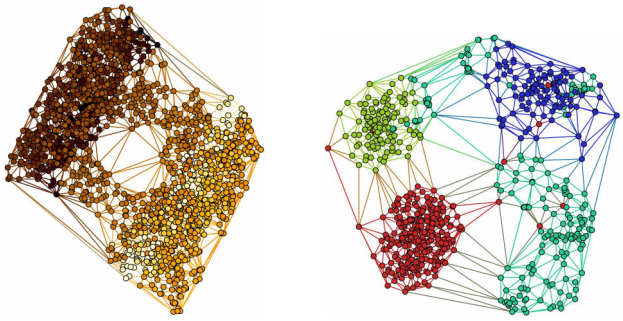
agrupamento $r \times \sqrt{n}$ objetos multi-dimensionais, com $r = 1$, o que indicaria agrupamentos com o mesmo tamanho, a complexidade de se projetar todos os agrupamentos seria $O(n\sqrt{n})$. É fato que nem todos os agrupamentos terão o mesmo tamanho, porém nenhum agrupamento terá $r \gg 2$, de forma que a complexidade P_k será $O(r^2 \times n\sqrt{n}) = O(n\sqrt{n})$ uma vez que r é uma constante pequena. Assim, a complexidade final dessa técnica é $O(n\sqrt{n} + n\sqrt{n} + n + n) = O(n\sqrt{n})$.

3.4 Resultados e Avaliação da Técnica

A seguir são apresentados alguns exemplos de projeções empregando a ProjClus de forma a comparar os resultados produzidos com os das técnicas apresentadas no capítulo anterior.

A Figura 3.1 apresenta diferentes projeções usando a ProjClus para os mesmos conjuntos de dados empregados no capítulo anterior. É possível notar que os resultados são semelhantes aos alcançados pela FS, que foi uma das melhores técnicas avaliadas na capítulo anterior,

mas a um custo computacional menor. Porém, a **Superfície-S** não conseguiu ser desdobrada (Figura 3.1(a)), um resultado esperado já que a ProjClus é baseada na FS e espera-se encontrar as mesmas limitações. A projeção do conjunto **CBR-ILP-IR-SON** é tão boa quanto o que se alcança com a FS, sendo possível separar visualmente os diferentes grupos de artigos desse conjunto (Figura 3.1(b)), usando o valor de $F = 4.5$.



(a) Conjunto de dados **Superfície-S**.

(b) Conjunto de dados **CBR-ILP-IR-SON**.

Figura 3.1: Layouts gerados usando-se a ProjClus. O resultado para o conjunto **CBR-ILPIR-SON** conseguiu separar bem os quatro grupos de artigos, mas a **Superfície-S** não conseguiu ser "desdobrada".

Outros exemplos da aplicação da ProjCLus são apresentados na Figura 3.2. Na Figura 3.2(a) um exemplo da projeção de outra coleção de artigos científicos, denominada **KDViz**, é apresentada. Essa coleção foi obtida em um repositório na Internet[2] e é composta por 1,624 artigos (título, resumo e referências bibliográficas) de quatro diferente áreas: *Bibliographic Coupling (BC)*, *Co-citation Analysis (CA)*, *Milgrams (MG)* e *Information Visualization (IV)*. Diferente da coleção **CBR-ILP-IR-SON**, que apresenta um número equilibrado de artigos em cada área, na **KDViz** existe a predominância de uma determinada área, com mais de 75% dos artigos. Nessa figura é possível notar que apesar desse desequilíbrio, a ProjClus foi capaz de separar bem os documentos com base em seus conteúdos. Os documentos em amarelo, que compõem a área predominante, são de IV, os documentos em verde claro são de MG, os azuis são de BC e os vermelhos são de CA. Nesse caso, existe uma forte mistura entre os documentos de BC e CA uma vez que ambas áreas são bastante relacionadas, praticamente lidando com o mesmo assunto.

Como pode ser observado pelos exemplos anteriores, os resultados da ProjClus para as coleções de documentos testadas são bastante satisfatórios. De forma a verificar se a qualidade dos layouts gerados é mantida quando a ProjClus é aplicada em um tipo de coleção de documentos com textos que apresentem uma linguagem mais livre do que artigos científicos, foi criada uma projeção de uma coleção de documentos contendo 400 mensagens de 4 diferentes grupos de discussão da Usenet. Esse conjunto foi obtido de um repositório da Internet (Hettich e Bay, 1999). A projeção para esse conjunto é apresentada na Figura 3.2(b). Embora esse

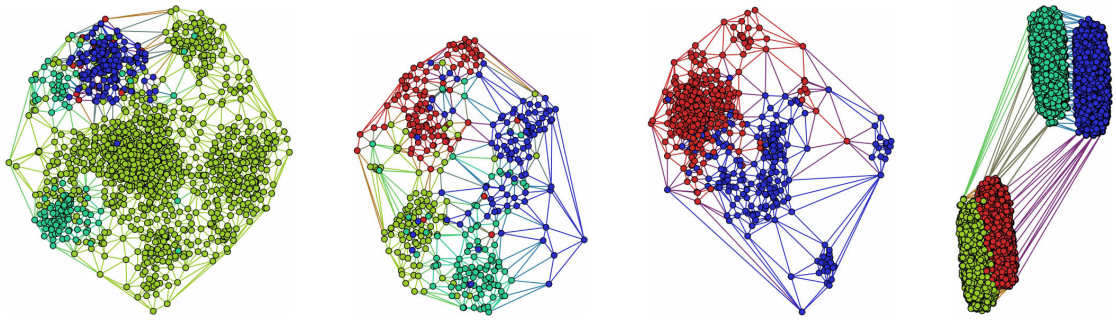
[2]http://ella.slis.indiana.edu/~katy/outgoing/hitcite/{fbc,sc,mb,ivg}.txt

conjunto não seja tão separável quanto as coleções de documentos de artigos científicos, ainda é possível verificar os quatro grupos de documentos que a compõem bem distintos na representação visual gerada, indicando que a ProjClus pode gerar resultados satisfatórios para conjuntos de documentos de diferentes características.

Apesar da ProjClus ter sido desenvolvida com objetivo de criar projeções multi-dimensionais de coleções de documentos, a mesma pode ser, a priori, empregada para qualquer conjunto de dados para os quais se possa definir alguma medida de dissimilaridade. A Figura 3.2(c) apresenta o resultado da ProjClus para o conjunto de dados **Wisconsin Diagnostic Breast Cancer (WDBC)** (UCI-MLR, 2008). O **WDBC** é composto de dados sobre imagens digitalizadas contendo punções da massa da mama de pacientes que apresentaram ou não câncer maligno – 357 imagens apresentando células de câncer benigno e 212 apresentando células de câncer maligno. Para se compor os atributos desse conjunto, características dos núcleos das células presentes nas imagens foram extraídas, como raio, textura, perímetros, área, etc., formando uma coleção de 30 diferentes atributos. Imagens desse conjunto de dados podem ser obtidas em (UW-MLCDP, 2008). Pela Figura 3.2(c) é possível notar que a ProjClus foi capaz de separar satisfatoriamente as imagens que apresentavam células cancerosas (em vermelho) das que não (em azul). Nesse caso foi empregada a distância Euclideana para definir a dissimilaridade entre os diferentes atributos extraídos das imagens.

Um outro exemplo da aplicação da ProjClus para a projeção multi-dimensional para conjuntos não textuais é apresentado na Figura 3.2(d). Nessa figura a projeção para o conjunto de dados **Quadruped Mammals** (Gennari et al., 1989) é apresentada. Este é um conjunto de dados sinteticamente gerado composto por 100.000 instâncias divididas em quatro classes (cachorros, gatos, cavalos e girafas). Nesse conjunto, cada instância é composta por 72 atributos com medidas relacionadas com o pescoço, número de patas, cabeça, etc., de cada mamífero. Aqui, novamente o resultado visual é bastante satisfatório, separando bem as classes. As duas classes que aparecem mais próximas são cachorro e gato, um resultado recorrente para esse conjunto de dados. O tempo gasto para a criação dessa projeção foi de aproximadamente 164 segundos, mostrando que a mesma pode ser aplicada para a projeção de grandes conjuntos de dados, sendo pelo menos uma ordem de grandeza mais rápida se comparada com apenas uma iteração da *Force Scheme*. Esses tempos foram medidos para uma implementação em linguagem de programação Java em um computador Intel(R) Core(TM) 2 Duo, 1.80GHz com 2G de memória RAM.

Como observado anteriormente, conforme o fator de densidade F aumenta, os agrupamentos tendem a ficar mais densos no layout final. A Figura 3.3 apresenta quatro diferentes projeções do conjunto **CBR-ILP-IR-SON** variando-se o valor de F. Na Figura 3.3(a), com $F = 1$, os quatro grupos de documentos não são distinguíveis porque a normalização dos agrupamentos nesse caso não conseguiu evitar sobreposições dos agrupamentos gerados pelo *bisecting k-means*. Na verdade, se o fator de densidade F não for empregado, isto é, se $F = 1$, não existem garantias de que não haja forte sobreposição dos agrupamentos, e é esperado que isso ocorra se a dimensionalidade intrínseca do conjunto de dados sendo projetado for maior do que dois,



(a) Coleção de documentos com uma área predominante.

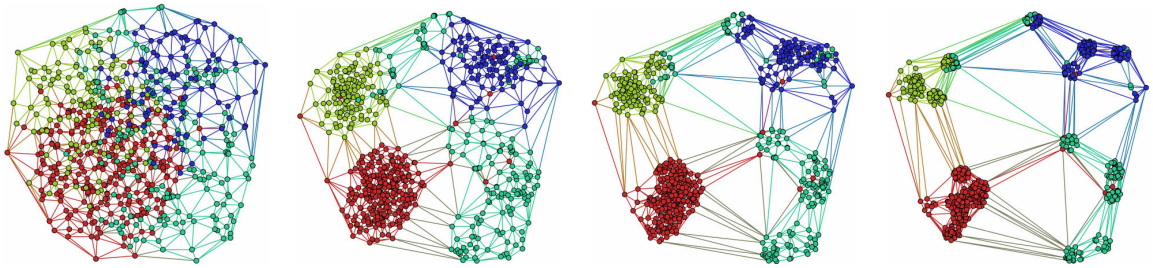
(b) Coleção de documentos de grupos de discussão.

(c) Conjunto de dados sobre imagens de punções da mama.

(d) Conjuntos de dados sintético sobre mamíferos.

Figura 3.2: Outros exemplos de projeção empregando a ProjClus. A qualidade dos layouts gerados pode ser verificada para coleções de documentos com diferentes características e para outros tipos de conjuntos de dados multi-dimensionais.

o que acontece na maioria dos casos. Por outro lado, se F aumenta muito, o resultado é uma projeção composta de sub-projeções referentes a cada agrupamento, sendo difícil a análise entre agrupamentos com certa similaridade e a análise das fronteiras entre eles. O efeito de se aumentar mais do que um limite razoável o fator de densidade pode ser observado na Figura 3.3(d), onde $F = 16$.



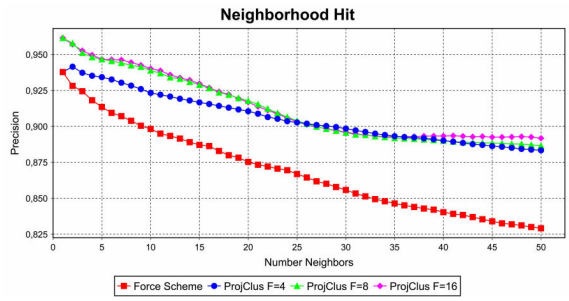
(a) $F = 1$ (b) $F = 4$ (c) $F = 8$ (d) $F = 16$

Figura 3.3: Projeções para o conjunto **CBR-ILP-IR-SON** variando-se o valor de F. Se F for muito pequeno, não há uma clara separação entre os agrupamentos, ocorrendo sobreposição entre os mesmos. Por outro lado, se F for muito grande os agrupamentos tendem a ficar muito densos, dificultando a análise dos relacionamentos entre diferentes agrupamentos e das fronteiras entre eles.

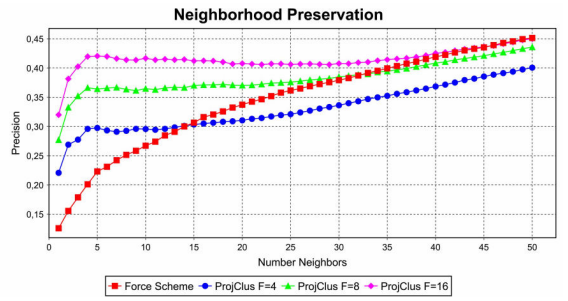
Para se entender melhor o que acontece com relação a preservação dos agrupamentos e das relações de distância quando o fator de densidade é alterado, as análises *Neighborhood Hit* e *Neighborhood Preservation* foram executadas nas projeções com fatores $F = 4, 8, 16$ apresentadas anteriormente. A projeção com $F = 1$ foi descartada uma vez que o resultado é consideravelmente pior o que atrapalharia a análise das outras projeções. Os resultados dessas análises são apresentados na Figura 3.4, comparando essas com a projeção criada usando-se a FS. Pela Figura 3.4(a), a análise *Neighborhood Hit*, é possível observar que quanto maior o fator de

densidade F, melhor será essa precisão, um resultado esperado devido a natureza desse tipo de análise – verificar a porcentagem dos vizinhos dos pontos que são da mesma classe (ver Seção 2.4) – inclusive os resultados obtidos sempre foram melhores do que o alcançado empregando-se a FS. A Figura 3.4(b) apresenta a análise *Neighborhood Preservation* para as mesmas projeções. Nesse caso, também quanto maior o fator F, maior será a precisão até um certo número de vizinhos, sendo que a projeção produzida pela FS só começa a apresentar qualidade compatível com as projeções criadas com maiores valores de F a partir dos 25 vizinhos.

A ProjClus divide o espaço em $\sqrt{n}$ agrupamentos e busca fazer com que todos os agrupamentos tenham um número similar de elementos, isto é, $\sqrt{n}$. Como o conjunto **CBR-ILP-IR-SON** é composto por 675 documentos, temos $\sqrt{675} \approx 26$ elementos por agrupamento, que é aproximadamente o número de vizinhos para o qual a ProjClus é melhor do que a FS. Isso pode ser visto como resultado da ProjClus levar em consideração somente informação de vizinhança local quando está projetando um agrupamento, ignorando as grandes distâncias que distorcem o layout produzido pela FS. Assim, a ProjClus não somente reduz a complexidade computacional da FS, mas também melhora as relações locais de vizinhança quando uma projeção é criada, uma característica interessante para a projeção de espaços de alta-dimensão e esparsos, como as coleções de documentos.



(a) Análise *Neighborhood Hit*.



(b) Análise *Neighborhood Preservation*.

Figura 3.4: Análises comparativas de projeções geradas variando-se o fator de densidade F. Quanto maior esse fator, maior a precisão da projeção criada, inclusive se comparada com uma projeção empregando-se a Force Scheme (Figura 2.8(b)).

3.5 Considerações Finais

Nesse capítulo foi apresentada a *Projection by Clustering (ProjClus)*, uma técnica para projeção de dados multi-dimensionais. A ProjClus é uma extensão de uma abordagem previamente desenvolvida, chamada de *Force Scheme (FS)* (veja Seção 2.3.2.4), com objetivo de reduzir sua complexidade computacional quadrática para $O(n\sqrt{n})$. Para tal, primeiro os objetos multi-dimensionais são divididos em grupos de objetos similares, empregando-se uma técnica de agrupamento. Então, esses grupos são separadamente projetados no plano usando-se a FS, e essa projeções são unidas criando-se o layout final.

Apesar da abordagem empregada pela ProjClus ser uma aproximação de uma técnica precisa, os resultados alcançados mantiveram a qualidade das projeções geradas com melhoria significativa da complexidade computacional. Isso representa um avanço em relações a outras técnicas baseadas em aproximações, como a técnica desenvolvida por Chalmers (ver Seção 2.3.2.2) ou o *Modelo Híbrido* de projeção (ver Seção 2.3.2.3). Nessas, aproximações também são empregadas visando-se reduzir a complexidade computacional de técnicas previamente definidas, mas resultam em projeções de baixa qualidade conforme apresentado no Capítulo 2.

De fato, para espaços de alta-dimensão e esparsos, os resultados da ProjClus foram melhores do que os da FS, tanto na facilidade de identificação dos agrupamentos pré-existentes quanto na preservação de vizinhanças locais. A melhoria em revelar os grupos se deve ao fato da existência de um parâmetro na ProjClus, o fator de densidade (F), que funciona como um meio para aumentar a densidade dos agrupamentos dos objetos multi-dimensionais nas projeções geradas, o que facilita a identificação desses agrupamentos. Já a melhoria na preservação de vizinhanças locais se deve ao fato do processo definido na ProjClus projetar individualmente agrupamentos de objetos altamente relacionados. Assim, as grandes distâncias que distorcem as vizinhanças locais no FS são evitadas, o que melhora a preservação de vizinhanças locais.

Embora os resultados da ProjClus sejam promissores por conseguirem diminuir a complexidade computacional de uma das técnicas anteriores bem avaliada, mantendo a qualidade do layout gerado, como nenhuma informação sobre os outros agrupamentos é levada em consideração quando um agrupamento é projetado, problemas de posicionamento podem ocorrer. Isso porque, como uma projeção gerada usando-se a FS é invariante à rotação, não há garantias de que a fronteira compartilhada por projeções de diferentes agrupamentos apresentem os elementos mais similares entre si. Apesar desse fato ser controlado na ProjClus criando-se e projetando-se uma grande quantidade de agrupamentos, de forma que um agrupamento contenha somente instâncias muito similares entre si, para pequenas vizinhanças nas fronteiras das projeções dos agrupamentos o problema ainda persiste e precisa ser tratado.

Na busca por uma técnica que tratasse melhor este problema de fronteira, sem piorar a qualidade e complexidade da ProjClus, foi desenvolvida a *Least Square Projection (LSP)*, descrita a seguir.

Capítulo

4

Least Square Projection (LSP)

4.1 Considerações Iniciais

Normalmente quando o espaço a ser projetado apresenta alta dimensionalidade, os objetos que o compõe acabam distribuídos em subespaços locais, estando somente relacionados a um pequeno número de vizinhos dentro do mesmo subespaço. Assim, quando relacionamentos entre objetos pertencentes a diferentes subespaços são levados em consideração no processo de projeção, o layout final pode ser distorcido (Martín-Merino e Muñoz, 2004).

De forma a lidar com esse tipo de dado e evitar os problemas inerentes da ProjClus, foi desenvolvida uma técnica de projeção multi-dimensional chamada *Least Square Projection (LSP)* (Paulovich et al., 2008a). A LSP adota uma estratégia diferente das projeções convencionais. Ela busca preservar relações de vizinhança entre os objetos m-dimensionais no espaço projetado, ao invés de tentar preservar relações de distância. Assim, quando um conjunto de objetos multi-dimensionais é projetado o que se busca é garantir que os objetos vizinhos no espaço multi-dimensional sejam projetados dentro de uma mesma vizinhança no plano.

Dois passos principais são executados nesse processo de projeção. Primeiramente, um subconjunto de objetos multi-dimensionais, chamados de "pontos de controle", é cuidadosamente escolhido e projetado no $\mathbb{R}^p$ usando-se uma técnica que preserve as relações de distância com precisão. Depois, fazendo-se uso das relações de vizinhança dos objetos no $\mathbb{R}^m$, e das respectivas coordenadas cartesianas dos pontos de controle no $\mathbb{R}^p$, é construído um sistema linear cuja solução visa projetar os objetos restantes de forma que os mesmos residam no fecho convexo de seus k vizinhos mais próximos, considerando-se uma vizinhança no $\mathbb{R}^m$.

Técnicas baseadas no conceito de preservação de vizinhança foram desenvolvidas anteriormente, principalmente para a redução de dimensionalidade. Um exemplo seria a *Local Linear Embedding (LLE)* (ver Seção 2.3.4.3). A diferença nesse caso é que no processo empregado pela LLE somente informação local é preservada (vizinhanças locais), não levando em consideração informação global sobre os objetos, como, por exemplo, relações de similaridade e distância entre grupos de objetos multi-dimensionais. Já na LSP, além da informação local, informação global também é empregada no processo de projeção por meio do posicionamento dos pontos de controle, conforme será visto no decorrer do Capítulo.

A idéia de gerar as coordenadas de um conjunto de pontos a partir de pontos de controle já vem sendo explorada no contexto de recuperação e edição de malhas por Sorkine e Cohen-Or (2004); Sorkine et al. (2004). De fato, a LSP generaliza as idéias de Sorkine e Cohen-Or de forma a lidar com espaços de alta dimensionalidade enquanto evita a necessidade de uma malha para definir o sistema linear.

A seguir os detalhes da LSP são apresentados, enfatizando a construção do sistema linear envolvido na estratégia de projeção e como os pontos de controle são definidos.

4.2 Construindo o Sistema Linear

Seja $V_i = \{p_{i_1}, \ldots, p_{i_{k_i}}\}$ um conjunto k_i pontos em uma vizinhança de um ponto p_i e $\tilde{p}_i$ sejam as coordenadas de p_i no $\mathbb{R}^p$. Suponha que $\tilde{p}_i$ sejam dadas pela seguinte equação:

$$\begin{aligned} &\tilde{p}_i - \textstyle\sum_{p_j \in V_i} \alpha_{ij} \tilde{p}_j = 0 \\ &0 \leq \alpha_{ij} \leq 1; \textstyle\sum \alpha_{ij} = 1 \end{aligned} \tag{4.1}$$

Se a Equação (4.1) for resolvida para os pontos em S então cada p_i será posicionado no fecho convexo dos pontos em V_i. Particularmente, quando $\alpha_{ij} = \frac{1}{k_i}$ teremos p_i no centróide dos pontos em V_i (Floater, 1997; Tutte, 1963).

A Equação (4.1) resulta em um conjunto de sistemas lineares com os quais é possível calcular as coordenadas dos $\tilde{p}_i$, isto é:

$$L\mathbf{x}_1 = 0,\ L\mathbf{x}_2 = 0,\ \cdots\ L\mathbf{x}_p = 0 \tag{4.2}$$

onde $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_p$ são os vetores contendo as coordenadas cartesianas $(x_1, \ldots, x_n)$ dos pontos e L é a matriz $n \times n$ cuja as entradas são dadas por:

$$l_{ij} = \begin{cases} 1 & i = j \\ -\alpha_{ij} & p_j \in V_i \\ 0 & \text{caso contrário} \end{cases}$$

A matriz L é normalmente chamada de matriz Laplaciana e seu *rank* depende da relação de vizinhança entre os pontos. Quando uma malha é dada, a vizinhança dos pontos pode ser

obtida a partir das relações de incidência na malha. Neste caso, o *rank* de L é $n - r$, onde r é o número de componentes conexos da malha (Sorkine e Cohen-Or, 2004). No caso da LSP, onde uma malha não existe, é importante definir a vizinhança dos pontos de forma a assegurar a condição de sobreposição dada na Definição (4.1).

Definição 4.1 (Condição de Sobreposição) *Seja $S = \{p_1, \ldots, p_n\}$ um conjunto de pontos e $V = \{V_1, \ldots, V_n\}$ o conjunto de relações de vizinhança dos pontos em S. Diz-se que a lista V satisfaz a condição de sobreposição se para cada dois pontos p_i e p_j existir uma seqüência de vizinhos $V_1^{ij}, \ldots, V_q^{ij}$ tal que $V_1^{ij} = V_i$, $V_q^{ij} = V_j$ e $V_k^{ij} \cap V_{k+1}^{ij} \neq \emptyset$, $k = 1, \ldots, q-1$.*

A condição de sobreposição assegura propriedades sobre L como se a mesma fosse uma malha com um único componente conexo, isto é, L terá um *rank* igual a $n - 1$, levando a uma solução não-trivial. O problema é que nenhuma informação geométrica está contida em L, então as soluções do sistema linear podem não ser úteis. De forma a tornar tais soluções mais atrativas, é necessário adicionar alguma informação geométrica ao sistema. Isto é feito por meio dos pontos de controle que podem ser obtidos pela projeção de alguns pontos de X em $\mathbb{R}^p$.

Os pontos de controle são inseridos no sistema linear como novas linhas na matriz. As coordenadas cartesianas dos pontos de controle são adicionadas do lado direito do sistema, levando a um vetor não-nulo. Dessa forma, dado um conjunto de pontos de controle $S_c = \{p_{c_1}, \ldots, p_{c_{nc}}\}$, é possível re-escrever a Equação (4.2) na forma:

$$A\mathbf{x} = \mathbf{b} \tag{4.3}$$

onde A é uma matriz retangular $(n + nc) \times n$ dada por:

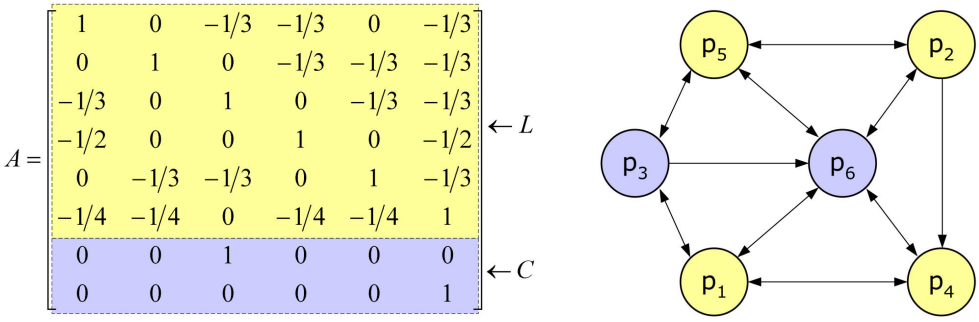
$$A = \begin{pmatrix} L \\ C \end{pmatrix}, \quad c_{ij} = \begin{cases} 1 & x_j \text{ é um ponto de controle} \\ 0 & \text{caso contrário} \end{cases}$$

e $\mathbf{b}$ é o vetor:

$$b_i = \begin{cases} 0 & i \leq n \\ x_{i_c} & n < i \leq n + nc \end{cases}$$

onde $x_{p_{c_i}}$ é uma das coordenadas cartesianas do ponto de controle p_{c_i}. A Figura 4.1(a) apresenta um exemplo de matriz A para um conjunto S com seis pontos. Os vizinhos de cada ponto são dados pelas relações de incidência do grafo direcionado da Figura 4.1(b) e os nós em azul são os pontos de controle (neste exemplo L é uma matriz Laplaciana).

O sistema linear com os pontos de controle apresenta *rank completo* e pode ser resolvido aplicando-se mínimos quadrados. Isso significa que devemos encontrar $\mathbf{x}$ que minimize $||Ax - b||^2$, isto é, $\mathbf{x} = (A^T A)^{-1} A^T \mathbf{b}$. O sistema $A^T A\mathbf{x} = A^T\mathbf{b}$ que deve ser resolvido é simétrico e esparso (Sorkine e Cohen-Or, 2004), o que permite que métodos eficientes de resolução possam ser empregados, como a decomposição de Cholesky (Davis, 2006) (método direto) ou o de gradientes conjugados (Shewchuk, 1994; Saad, 2003) (método iterativo).



(a) Matriz A (com pontos de controle p_3 e p_6). (b) Relações de vizinhanças.

Figura 4.1: Relações de vizinhança e matriz A resultante.

4.3 Pontos de Controle

De forma a determinar o conjunto de pontos de controle, uma amostra de nc objetos deve ser cuidadosamente escolhida do conjunto X. Idealmente, objetos devem ser selecionados de forma a representar o melhor possível a distribuição dos dados em $\mathbb{R}^m$ e os possíveis grupos de objetos existentes no espaço multi-dimensional. De forma a realizar essa seleção, inicialmente um algoritmo de agrupamento é executado, criando nc agrupamentos. Depois, um objeto representativo de cada agrupamento é escolhido como ponto de controle. Aqui, nós definimos que esses pontos devem ser os medóides dos agrupamentos, ou seja, os objetos mais próximos dos centróides.

A priori, qualquer método de agrupamento pode ser utilizado nesse processo. Nesta tese nós empregamos o *bisecting k-means* (ver Seção 3.2) quando os dados têm uma representação vetorial e o método de *k-medoids* (Berkhin, 2002) quando essa representação não existir, só existindo uma forma de se calcular a distância entre os objetos multi-dimensionais. O algoritmo *k-medoids* é bastante parecido com o *k-means* original, com a diferença de que ao invés de se empregar o centróide de cada agrupamento no processo de divisão dos elementos em grupos, emprega-se o médoide (Berkhin, 2002).

Apesar desse tipo de processo ser mais caro ($O(n \times nc)$) que uma amostragem aleatória simples, nos experimentos realizados ele trouxe melhores resultados, sendo necessário um menor número de pontos de controle para atingir projeções de boa qualidade. Uma melhor representação dos possíveis grupos existentes nos dados (ou da distribuição presente), é um fator importante na LSP, já que essa é a informação global que será preservada. Além disso, esses agrupamentos podem ser usados para definir as relações de vizinhança entre os objetos, como será explicado na próxima seção.

Um vez que os pontos de controle tenham sido definidos, os mesmos devem ser projetados em $\mathbb{R}^p$ por meio de um método de projeção multi-dimensional convencional. É importante notar que o sucesso da LSP está intimamente ligado ao posicionamento desses pontos de controle uma vez que os objetos restantes serão interpolados no layout final de acordo com esse posicionamento

inicial. Assim, a técnica utilizada na projeção desses pontos deve ser a melhor possível. Observe que nesse caso a preservação de todas as relações de distâncias não é tão prejudicial quanto nos casos da técnica *Sammon's Mapping* (ver Seção 2.3.3.3) ou do *Modelo de Molas* (ver Seção 2.3.2.1). Isso se deve ao fato de que os pontos de controle representam grupos de objetos. Então, a distorção causada pelas grandes distâncias em espaços esparsos de alta-dimensão, que induzem os pontos projetados a estarem uniformemente distribuídos no plano, não resultará em problemas para a diferenciação dos grupos de objetos.

De forma a manter a complexidade computacional da LSP dentro de um limite aceitável, o número de pontos de controle nc deve ser escolhido de acordo com a complexidade da técnica de projeção envolvida. Se uma técnica de projeção $O(n^2)$ for usada, $nc = \sqrt{n}$ fará com que a complexidade da escolha e projeção dos pontos de controle seja $O(n\sqrt{n})$ devido ao método de agrupamento utilizado. Observe que quando nc é próximo do número de objetos em X, a projeção final será muito semelhante a projeção obtida utilizando-se a técnica de projeção sobre todos os objetos em X (e, é claro, será tão lenta quanto ela).

4.4 Definindo as Relações de Vizinhança

Juntamente com as coordenadas cartesianas dos pontos de controle também é necessário definir uma lista de pontos $V_i \in X$ para cada objeto $p_i \in X$. Uma vez que p_i será posicionado no fecho convexo de V_i, esta lista deve refletir uma vizinhança de p_i, tornando o layout final baseado em relações locais no $\mathbb{R}^m$. Esta é uma característica importante se $\mathbb{R}^m$ é um espaço esparso e de alta dimensão.

Normalmente o procedimento para encontrar os vizinhos mais próximos de cada ponto é proibitivo, $O(n^2)$. Mas existem algumas formas de se reduzir essa complexidade. Chávez et al. (2001) apresenta um número de técnicas para a busca de vizinhos mais próximos, dividindo essa técnicas em dois grandes grupos: algoritmos baseados em pivôs e técnicas de agrupamento. No primeiro, alguns objetos são escolhidos para agir como pivôs, tornando possível evitar cálculos de distância. No segundo, o espaço é dividido em agrupamentos não sobrepostos de forma que alguns agrupamentos e seus objetos possam ser descartados durante uma busca.

Aqui é empregada uma técnica simples baseada em agrupamentos para encontrar a vizinhança dos pontos. Esta foi a escolhida uma vez que o espaço já foi dividido em agrupamentos pelo processo de definição dos pontos de controle (veja Seção 4.3). Nesta técnica, primeiramente uma busca pelos vizinhos mais próximos dos medóides dos agrupamentos é realizada, definindo os k agrupamentos mais próximos de cada agrupamento. Assim, quando uma busca pelos vizinhos mais próximos de um ponto p_i for realizada, somente o agrupamento a que p_i pertence e os agrupamentos mais próximos desse serão examinados. Esta é uma aproximação da busca por vizinhos mais próximos, mas normalmente leva a bons resultados – mesmo quando os vizinhos mais próximos retornados não são os reais vizinhos mais próximos, eles ainda estarão bem próximos de p_i. A complexidade de tal técnica é determinada pelo número de agrupamentos; com $\sqrt{n}$ agrupamentos ela será $O(n\sqrt{n})$.

O resultado desse processo pode ser visto como um grafo que liga os objetos multi-dimensionais a seus vizinhos mais próximos. Porém, para a composição do sistema linear isso não é suficiente, sendo necessário gerar um grafo conexo que respeite a **Condição de Sobreposição** (Definição (4.1)). Para assegurar essa condição de sobreposição desenvolvemos um processo simples que sucessivamente adiciona ligações entre os nós (objetos multi-dimensionais) até que se obtenha um grafo conexo. A idéia é, a partir de um nó, visitar todos os nós ligados a esse. Depois, visitar os nós ligados a esses que ainda não tenham sido visitados, e assim por diante até que não seja possível alcançar mais nenhum nó. Caso se tenha visitado todos os nós, o grafo é conexo. Caso contrário, o último nó visitado é conectado ao nó mais próximo ainda não visitado, e o processo de visitar os nós ligados continua. Esse procedimento é apresentado em detalhes no Algoritmo 4.1.

Algoritmo 4.1 Definindo um Grafo Conexo

input: - N: conjunto de n listas contendo os índices dos k vizinhos mais próximos de cada elemento do conjunto de dados. O operador $N[z]$ retorna a lista contendo os índices dos vizinhos mais próximos do elemento z.
- $\mathbf{X}$: matriz dos pontos m-dimensionais.

output: - N: conjunto N com elementos adicionados de forma a se definir um grafo conexo.

1: $VISITADOS = \emptyset$ *lista que contém os índices dos nós já visitados.*
2: $VISITAR = \{0\}$ *lista que contém os índices dos nós a serem visitados.*
3: $NVISITADOS = \{1, \ldots, (n-1)\}$ *lista que contém os índices dos nós não visitados.*
4: **enquanto** $NVISITADOS \neq \emptyset$ **faça**
5: **se** $VISITAR \neq \emptyset$ **então**
6: $p = VISITAR$.primeiro() *pegar o primeiro elemento da lista.*
7: $VISITADOS$.adicionar(p) *adicionar um elemento no final da lista.*
8: $NVISITADOS$.remover(p) *remover um elemento da lista.*
9: $VISITAR$.remover(p)
10: **para todo** $n \in N[p]$ **faça** *adicionar os nós ligados a p na lista de nós a visitar.*
11: **se** $n \notin VISITADOS$ **e** $n \notin VISITAR$ **então**
12: $VISITAR$.adicionar(k)
13: **fim se**
14: **fim para**
15: **senão** *existe uma desconexão no grafo.*
16: $p = NVISITADOS$.primeiro()
17: $NVISITADOS$.remover(p)
18: $VISITAR$.adicionar(p)
19: *ligar o nó p ao nó mais próximo presente na lista de nós já visitados.*
20: $v = \min\{\delta(\mathbf{x}_p, \mathbf{x}_v)\}\ \forall v \in VISITADOS$
21: $N[p]$.adicionar(v) *adicionar v como vizinho de p.*
22: $N[v]$.adicionar(p) *adicionar p como vizinho de v.*
23: **fim se**
24: **fim enquanto**

4.5 Complexidade Computacional

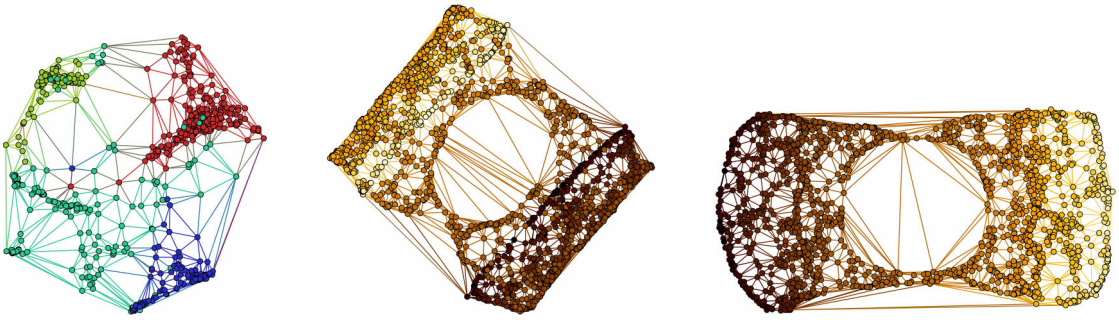
A complexidade global da LSP pode ser calculada como $O(C + N + S)$, onde C é a complexidade de escolher os pontos de controle, N é a complexidade de definir o grafo de vizinhança e S é a complexidade para se resolver o sistema linear. Como já discutido anteriormente na Seção 4.3, $C = N = O(n\sqrt{n})$ se $\sqrt{n}$ pontos de controle forem utilizados. Uma vez que a matriz do sistema linear gerado é simétrica definida positiva (e esparsa), um método iterativo de solução pode ser empregado, tal como o de gradientes conjugados (Shewchuk, 1994; Saad, 2003). Nesse caso, a complexidade para resolver tal sistema é $O(n\sqrt{k})$, onde k é o número de condição da matriz $A^T A$ (Shewchuk, 1994). Portanto, a complexidade final da LSP será $O(\max\{n\sqrt{n}, n\sqrt{k}\})$.

4.6 Resultados e Avaliação da Técnica

A seguir, alguns exemplos de projeções empregando a LSP são apresentados para mostrar sua eficácia na criação de projeções multi-dimensionais, entender os efeitos resultantes da parametrização existente e comparar os resultados produzidos por essa técnica com os das demais técnicas apresentadas no Capítulo 2.

A Figura 4.2 apresenta projeções usando-se a LSP para os conjuntos de dados **CBR-ILP-IR-SON** e **Superfície-S**. Para o primeiro conjunto, o resultado é bastante satisfatório, sendo uma das técnicas analisadas nessa tese que melhor consegue separar os quatro grupos de artigos científicos (Figura 4.2(a)). Já para o segundo conjuntos de dados, a superfície é "desdobrada" dependendo da técnica empregada para o posicionamento dos pontos de controle. Caso uma técnica que não consiga "desdobrar" a superfície seja aplicada, como por exemplo a Force Scheme, o resultado final da LSP também não obterá êxito (Figura 4.2(b)). Porém, se uma técnica que consegue "desdobrar" for utilizada, como por exemplo a ISOMAP, o resultado final também será uma superfície "desdobrada" (Figura 4.2(c)). Na verdade esse é um efeito esperado uma vez que o processo empregado pela LSP pode ser entendido como uma interpolação (com restrições de vizinhança) dos objetos de um conjunto dado o posicionamento inicial de uma amostra desse conjunto. Assim, a projeção dos pontos de controle acaba ditando em grande parte a qualidade final das projeções geradas pela LSP.

A Figura 4.3 apresenta outros exemplos usando a LSP. A Figura 4.3(a) apresenta o resultado para a projeção do conjunto **KDViz** (ver Seção 3.4). Assim como na ProjClus, o resultado apresentado pela LSP conseguiu separar bem os grupos de artigos científicos apesar de existir uma área dominante com mais de 75% dos documentos. A Figura 4.3(b) mostra a projeção para o conjunto de dados contendo mensagens de discussão de Usenet. Aqui, o resultado foi tão bom quanto o alcançado pela ProjClus, separando bem os quatro grupos de mensagens. Na Figura 4.3(c) é apresentada a projeção para o conjunto **WDBC**. Comparando esse com o apresentado pela ProjClus, é possível notar que as duas classes de imagens, as que apresentam células cancerosas e as que não, são melhor separadas, o que facilitaria uma possível análise desse conjunto de imagens se a tarefa fosse definir a qual classe uma determinada imagem pertence.

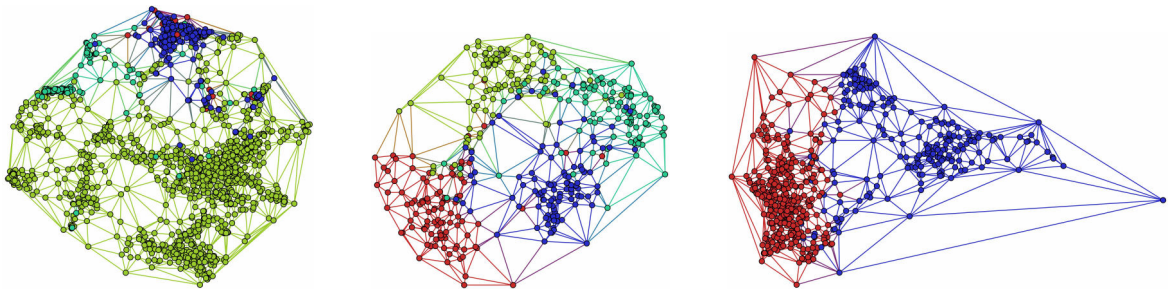


(a) Conjunto de dados **CBR-ILP-IR-SON**.

(b) Conjunto de dados **Superfície-S** (pontos de controle projetados usando-se a IDMAP).

(c) Conjunto de dados **Superfície-S** (pontos de controle projetados usando-se a ISOMAP).

Figura 4.2: Layouts gerados usando-se a LSP. O resultado para o conjunto **CBR-ILP-IR-SON** conseguiu separar bem os quatro grupos de artigos. Porém, a **Superfície-S** só foi "desdobrada" quando uma técnica (ISOMAP), que consegue de fato lidar com dados não-lineares, foi aplicada no posicionamento dos pontos de controle.



(a) Coleção de documentos com uma área predominante.

(b) Coleção de documentos de grupos de discussão.

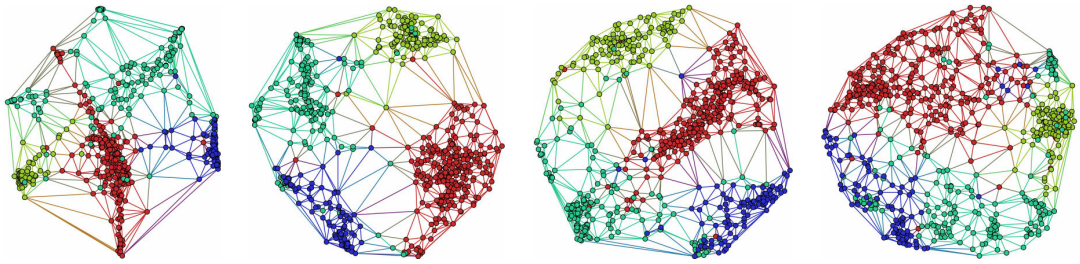
(c) Conjunto de dados sobre imagens de punções da mama.

Figura 4.3: Outros exemplos de projeção empregando a LSP. A qualidade dos layouts gerados pode ser verificada para coleções de documentos com diferentes características e para outros tipos de conjuntos de dados multi-dimensionais.

Como discutido na Seção 4.3, a escolha dos pontos de controle tem um papel importante no processo de projeção da LSP. Nas Figuras 4.6 e 4.4 projeções são apresentadas de forma a exemplificar o impacto dos pontos de controle na qualidade final das projeções. Na prática, pequenas mudanças na escolha e posicionamento dos pontos de controle não afetam a qualidade do layout gerado, mas o número de pontos de controle e a ausência de representativos de um grupo entre os pontos de controle influenciam a qualidade.

A Figura 4.4 apresenta o que acontece quando o número de pontos de controle se aproxima do número de objetos multi-dimensionais a serem projetados. Conforme o número de pontos de controle aumenta, a projeção final gerada pela LSP se torna cada vez mais parecida com a projeção gerada pela técnica de MDS empregada para posicionar os pontos de controle. As projeções nessa figura são do conjunto de dados **CBR-ILP-IR-SON**. A Figura 4.4(a) apresenta o resultado se forem usados 5% de pontos de controle. A Figura 4.4(b) apresenta o resultado se

forem usados 25% de pontos de controle. A Figura 4.4(c) apresenta o resultado se forem usados 50% de pontos de controle. E por fim, a Figura 4.4(d) apresenta o resultado se forem usados 75% de pontos de controle – compare esses resultados com o apresentado na Figura 2.8(b), que é o gerado quando se aplica a *Force Scheme*, a técnica de MDS empregada para posicionar os pontos de controle, sobre todos os objetos multi-dimensionais.

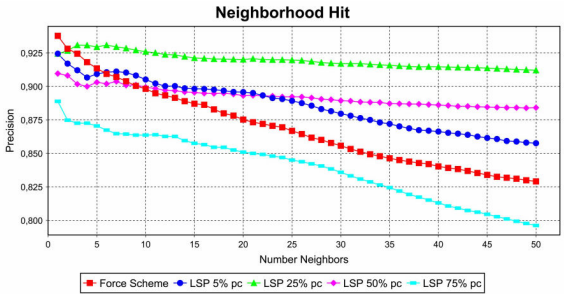


(a) 5% de pontos de controle. (b) 25% de pontos de controle. (c) 50% de pontos de controle. (d) 75% de pontos de controle.

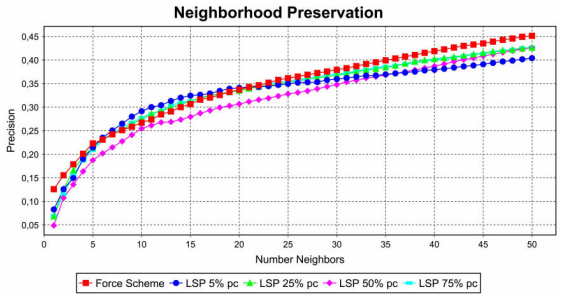
Figura 4.4: Efeito da mudança do número de pontos de controle sobre a projeção gerada usando a LSP. Quanto maior o número de pontos de controle, mais o layout produzido se assemelha àquele produzido quando a técnica empregada para posicionar os mesmos é executada sobre todos os objetos multi-dimensionais do conjunto de dados.

Na Figura 4.5 as análises *Neighborhood Hit* e *Neighborhood Preservation* são apresentadas para as projeções da Figura 4.4 e para a projeção empregando a *Force Scheme (FS)* (Figura 2.8(b)) para o conjunto **CBR-ILP-IR-SON**. Pela primeira análise é possível notar que quanto maior o número de pontos de controle, mais o resultado da LSP se aproxima do resultado da FS (algo esperado para essa técnica), sendo que o melhor resultado é alcançado quando o número de pontos de controle escolhido não é tão pequeno que possa deixar de representar bem alguns grupos de objetos multi-dimensionais, nem tão grande de forma a herdar da FS o problema relativo às distorções provocadas por se levar em consideração no processo de projeção as grandes distâncias. Esse efeito pode ser observado também na segunda análise, quando o pior resultado de preservação de vizinhança é alcançado empregando muito pontos de controle. Porém, observe pela segunda análise que a preservação da vizinhança não é tão afetada pelo número de pontos de controle escolhido. Mais à frente (Figura 4.8) será mostrado que o número de vizinhos aos quais um objeto é ligado para definir o sistema linear influencia muito mais essas relações de vizinhança.

A qualidade das projeções geradas pela LSP se mantém consistente desde que a quantidade e distribuição dos pontos de controle seja razoável. Nos resultados dos nossos testes, normalmente a baixa qualidade os layouts produzidos estava relacionada à escolha de um conjunto de pontos de controle que não refletia bem a distribuição do conjunto de dados. Duas situações são as mais relevantes neste cenário: (1) selecionar um conjunto de pontos de controle que não contenha ao menos um representante de um importante grupo de objetos multi-dimensionais nos dados originais; e (2) selecionar um conjunto de pontos de controle que apresentem uma determinada tendência, por exemplo, todos os pontos residindo aproximadamente em uma linha.



(a) Análise *Neighborhood Hit*.



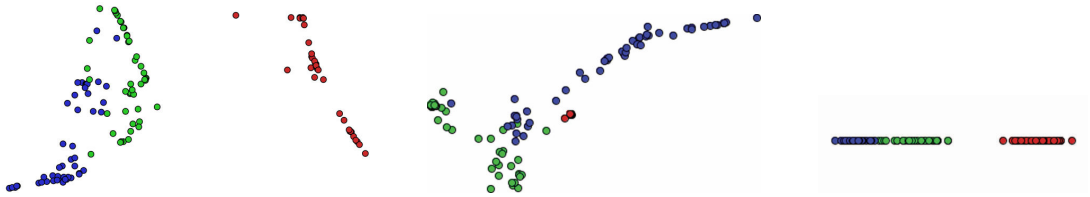
(b) Análise *Neighborhood Preservation*.

Figura 4.5: Análises para diferentes escolhas no número de pontos de controle. Se poucos pontos forem empregados, algum grupo pode deixar de ser representado, o que resulta em layouts pouco precisos. Por outro lado, se muitos pontos forem usados, o layout gerado herda os problemas inerentes à Force Scheme relacionados às distorções causadas pelas grandes distâncias.

A primeira situação é exemplificada na Figura 4.6(b). Essa figura mostra uma projeção para o conjunto de dados **Iris**. Esse conjunto consiste de informação sobre 150 flores iris de 3 diferentes espécies, com 50 exemplares de cada espécie (Fisher, 1936; UCI-MLR, 2008). Nessa projeção, as cores indicam as classes das flores. De forma a se gerar tal projeção, todos os pontos de controle foram intencionalmente escolhidos das classes verde e azul. A classe vermelha não foi representada por nenhum ponto de controle. Isso induz a um posicionamento ruim dos objetos pertencentes à classe vermelha, prejudicando a distinção dos objetos dessa classe devido à excessiva sobreposição que ocorre; compare esse resultado com o gerado empregando-se o processo completo definido na LSP (Figura 4.6(a)). Embora esse cenário seja possível, o emprego de um algoritmo básico de agrupamento na escolha dos pontos de controle, como sugerido aqui (veja Seção 4.3), torna bastante improvável não selecionarmos um representante de importantes grupos do conjunto de dados. Este cenário é mais possível quando poucos pontos de controle são empregados. Nos nossos testes, um número de pontos de controle próximo de 10% do conjunto de dados se mostrou suficiente.

Desenvolvemos um segundo exemplo para ilustrar o segundo problema com a LSP referente à má escolha dos pontos de controle. Uma vez que a estratégia de interpolação empregada pela LSP é baseada na idéia de posicionar os pontos no fecho convexo de seus vizinhos, se os pontos de controle forem projetados em uma linha, o layout final também será aproximadamente uma linha. Na Figura 4.6(c) primeiro projetamos os pontos de controle em uma linha para simular um caso real (porém difícil de acontecer), então realizamos a interpolação dos pontos de controle restantes. Novamente, este cenário somente acontece se os pontos de controle selecionados residem em uma linha reta no espaço original, ou se o posicionamento inicial no plano é mal feito De fato, estes cenários são mais passíveis de ocorrer se o número de objetos multi-dimensionais a serem projetados for pequeno (conjuntos com menos de 200 objetos). Nessas situações outras técnicas de projeção são preferíveis, tais como *Sammon's Mapping* (veja Seção 2.3.3.3) ou a *Force*

Scheme (veja Seção 2.3.2.4). Ambas são rápidas suficiente para definirem layouts de pequenos conjuntos de dados em tempo real.



(a) Projeção normal usando o processo completo.

(b) Nenhum ponto de controle é escolhido entre os objetos da classe vermelha.

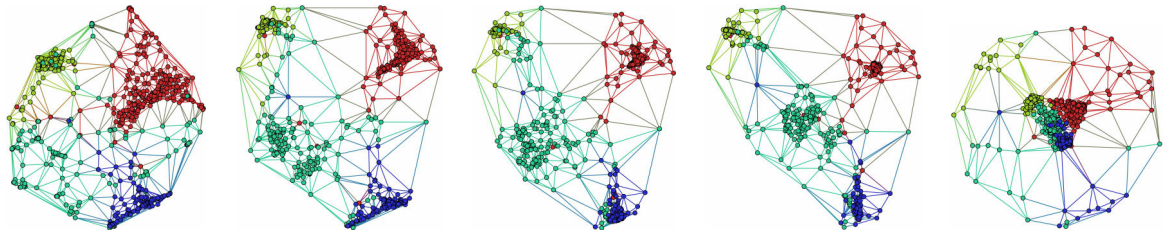
(c) Todos os pontos de controle residem em uma linha.

Figura 4.6: Problemas que podem ocorrer na LSP quando existe uma forte tendência na escolha ou posicionamento dos pontos de controle.

Além do número e posicionamento dos pontos de controle, um parâmetro que afeta o resultado final da projeção gerada pela LSP é o número de vizinhos aos quais cada objeto é ligado para se criar o grafo de vizinhança. A Figura 4.7 mostra o resultado de se variar esse parâmetro para projeções do conjunto de dados **CBR-ILP-IR-SON**. Conforme o número de vizinhos cresce, mais densos serão os grupos de objetos multi-dimensionais no layout final gerado. Apesar de parecer interessante aumentar deliberadamente o número de vizinhos para criar layouts mais compactos, facilitando a identificação dos grupos de objetos multi-dimensionais, alguns problemas devem ser observados. Com o aumento do número de vizinhos, aumenta a sobreposição dos pontos no plano o que dificulta a extração de sub-grupos dentro desses grupos, a diferenciação entre os objetos na projeção gerada e a análise da densidade (número de objetos) desses grupos.

Contudo, o maior problema é criar ligações que unem grupos de objetos não muito relacionados, aqui denominadas ligações em "curto circuito". Esse efeito pode ser observado na Figura 4.7(e) onde cada objeto é ligado a outros 200 vizinhos. Como os grupos ILP, IR e SON tem menos de 200 objetos, o número de ligações requeridas acaba unindo esses grupos como um único grupo, resultando em um layout pouco preciso. Nessa figura, os pontos que não foram agrupados no meio da projeção são os pontos de controle, revelando que quando o número de ligações aos vizinhos mais próximos é muito grande, esses não conseguem distribuir o restantes dos objetos pelo plano. De fato esse cenário deve ser evitado definindo grafos de vizinhança dentro de limites razoáveis. Nos testes realizados, e na maioria dos exemplos apresentados nessa tese, o número de vizinhos é mantido constante em 10. Essa tem se mostrado uma escolha que consegue definir bem os grupos e evitar as ligações em "curto circuito". Essas ligações ainda podem ocorrer se existirem grupos com menos de 10 objetos, mas considerando um cenário com milhares de objetos multi-dimensionais, esse nível de detalhe pode ser ignorado.

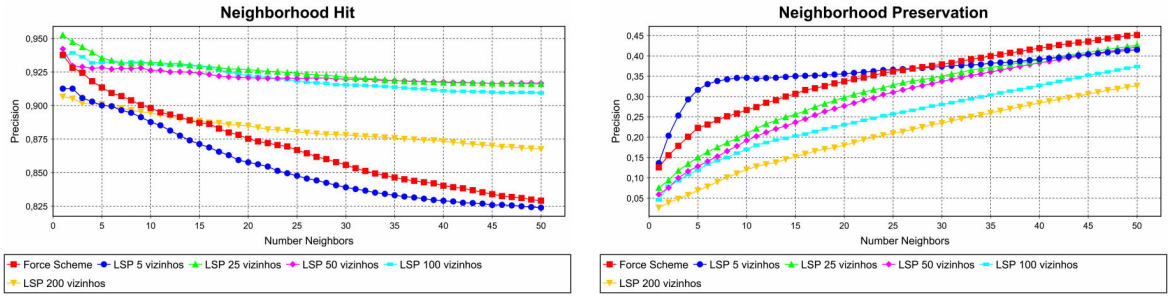
A Figura 4.8 apresenta as análises *Neighborhood Hit* e *Neighborhood Preservation* para as projeções apresentadas na Figura 4.7 e a projeção empregando a *Force Scheme (FS)* (Figura 2.8(b)) para o conjunto **CBR-ILP-IR-SON**. Pela primeira análise é possível notar que se poucos



(a) 5 vizinhos. (b) 25 vizinhos. (c) 50 vizinhos. (d) 100 vizinhos. (e) 200 vizinhos.

Figura 4.7: Efeito da mudança no número de vizinhos quando é gerada uma projeção usando a LSP. Quanto maior o número de vizinhos, mais densos serão os grupos de objetos no layout final gerado.

vizinhos (5 vizinhos) são empregados na criação do grafo de vizinhança, o resultado final será pior do que o apresentado pela FS. O mesmo ocorrendo se muitos vizinhos (200 vizinhos) forem utilizados. Porém, para números razoáveis de vizinhos, o resultado de identificação dos grupos de objetos multi-dimensionais é sempre melhor do que o da FS. Pela segunda análise é possível notar que conforme o número de vizinhos cresce, a preservação de relações de vizinhança diminui, mas que para poucos vizinhos, o resultado é melhor do que o atingido pela FS. Assim, com esses resultados, fica claro que para criar layouts mais precisos, o número de vizinhos empregados para a criação do grafo de vizinhança também tem grande influência na projeção final.



(a) Análise *Neighborhood Hit*. (b) Análise *Neighborhood Preservation*.

Figura 4.8: Análises comparando o efeito de se mudar o número de vizinhos quando o grafo de vizinhança é definido. Por esses resultados fica claro que definir poucos ou muitos vizinhos têm efeito negativo no layout final gerado.

4.7 Considerações Finais

Nesse capítulo foi apresentada a *Least Square Projection (LSP)*, uma técnica de projeção multi-dimensional que é capaz de projetar grandes conjuntos de dados de alta dimensionalidade em um tempo computacional satisfatório, reconstruindo com precisão as relações de similaridade existentes entre objetos multi-dimensionais. Uma vez que a LSP busca preservar relações de vizinhança, a mesma é indicada para espaços não lineares esparsos, tais como os criados na

representação vetorial de coleções de documentos. Porém, nos testes realizados bons resultados também foram conseguidos para outros tipos de dados, como por exemplo, conjuntos de imagens.

O primeiro passo da LSP é a escolha cuidadosa de uma pequena amostra de objetos, chamados pontos de controle, que são projetados empregando uma técnica de MDS. Em seguida, a LSP interpola os objetos restantes resolvendo um sistema linear esparso de equações que visam posicionar cada ponto no fecho convexo de seus vizinhos mais próximos. De fato, a LSP generaliza um método criado para reconstrução e edição de malhas de forma a lidar com espaços de alta dimensão enquanto evita a necessidade de uma malha para definir o sistema linear.

Uma vez que somente uma pequena porção dos objetos multi-dimensionais é escolhida como pontos de controle, a técnica de MDS empregada para projetá-los pode ser computacionalmente cara, tal como a *Sammon's Mapping* (veja Seção 2.3.3.3). Isso demonstra uma importante característica da LSP: ela pode ser utilizada como uma técnica de interpolação de alta precisão de outras técnicas de projeção mais caras. Assim, métodos que não podem ser aplicados a grandes conjuntos de dados podem ser usados para projetar os pontos de controle, tendo, potencialmente seus resultados refletidos para o conjunto inteiro de dados aproximado pela LSP. Embora não tenha sido discutido aqui, a LSP pode também ser empregada como um método de interpolação para qualquer tipo de método de redução de dimensionalidade, não somente como técnica de projeção. Para isso é somente necessário trocar a técnica de MDS empregada para a projeção dos pontos de controle por um método de redução de dimensionalidade.

A idéia da preservação de vizinhanças já foi empregada em outras técnicas, como por exemplo na técnica de redução de dimensionalidade *Local Linear Embedding (LLE)* (veja Seção 2.3.4.3). Porém, diferentemente da LLE, que somente preserva informação local no processo de redução de dimensionalidade, a LSP busca preservar também informação global por meio dos pontos de controle, isto é, informação sobre os possíveis grupos de objetos dentro do conjunto de dados. Assim, para reduções drásticas de dimensionalidade, como é o caso das projeções multi-dimensionais, que reduzem o conjunto para duas dimensões, os resultados apresentados pela LSP foram muito superiores aos apresentados pela LLE, conseguindo separar e agrupar coerentemente os objetos multi-dimensionais, o que não ocorre em muitos casos com a LLE.

Embora o resultados apresentados pela LSP tenham sido satisfatórios no sentido de separação e agrupamento de objetos no layout e de preservação das relações de vizinhança e similaridade, observações sobre a aplicação da mesma devem ser feitas. Um fator da LSP que pode representar problema dependendo do tipo de aplicação é a tendência desta de concentrar bem os objetos de um determinado grupo em uma pequena área do layout final. Apesar dessa ser uma característica importante para a localização de grupos de elementos relacionados, uma forte sobreposição entre os objetos é comum ocorrer, o que pode atrapalhar a distinção visual de objetos individuais, ou mesmo a avaliação da densidade dos grupos formados. Apesar disso representar um problema real, a coloração dos pontos no layout final pode ser empregada para ajudar na identificação de densidade, e mecanismos de interação podem ser utilizados para ajudar na exploração da projeção, evitando os problemas de sobreposição. Por fim, como a LSP é um processo de interpolação de uma pequena amostra do conjunto de dados, se esse for pequeno, com menos de

200 objetos, sua aplicação não é indicada, sendo melhor empregar uma técnica de MDS sobre todos os objetos multi-dimensionais.

Uma preocupação especial durante este projeto de doutorado foi a exploração de conjuntos de documentos. Neste caso, a análise visual do conjunto de dados pode ser profundamente beneficiada por uma forma eficiente de auxílio à detecção de tópicos sendo abordados pelos documentos sob análise. Nesta tese uma técnica para este fim foi desenvolvida, e ela é descrita no próximo Capítulo.

Capítulo

5

Extração de Tópicos por Covariância

5.1 Considerações Iniciais

Embora uma projeção multi-dimensional seja útil para revelar relações de similaridade entre objetos e grupos de objetos, nenhuma informação é dada sobre o motivo de certos grupos terem se formado no layout final. No caso específico da exploração de coleções de documentos, um mecanismo que pode auxiliar no processo de análise de uma projeção é a extração automática e semi-automática de tópicos. Aqui, nós definimos tópicos como conjuntos de termos relacionados que buscam identificar o assunto comum a um determinado grupo de documentos (Lopes et al., 2007).

A idéia de extração de tópicos de coleções de documentos tem sido o objetivo de pesquisa de diferentes áreas. Um exemplo é na ajuda à interpretação de resultados de busca na Web. Atualmente existem duas diferentes abordagens para se realizar essa tarefa (Toda e Kataoka, 2005): a abordagem baseada em documentos; e a abordagem baseada em descritores. Na primeira, agrupamentos são criados e termo(s) ou sentença(s) são extraídos como tópicos a partir de cada agrupamento (por exemplo, (Hearst e Pedersen, 1996; Leuski, 2001)). Na segunda, termos informativos, na forma de palavras ou frases, são extraídos dos resultados de busca como descritores, e agrupamentos são definidos pelos documentos que incluem certos termos (Mooter., 2008; Vivísimo., 2008; Toda e Kataoka, 2005; Zeng et al., 2004; Kummamuru et al., 2004; Ohta et al., 2004).

Um uso típico de tópicos mais relacionado ao trabalho aqui apresentado é para a criação dos *mapas de documentos*. Um exemplo dessa abordagem são os mapas "cartográficos" de documentos de Skupin (2002). Esse assinala tópicos a agrupamentos hierárquicos sobre documentos

em um mapa criado usando-se a técnica *Self-Organizing Map (SOM)* (Kohonen, 1990). Para agrupamentos de alto nível, os termos com maior contagem são escolhidos, para os de baixo nível um tópico é construído a partir dos três termos com maior valor de acordo com uma variação da ponderação *term frequency-inverse document frequency (tf-idf)* (veja Seção 2.2.1).

Outro exemplo de mapa de documentos é o criado empregando-se a técnica *ThemeScape* (Wise, 1999). A *ThemeScape* é uma visualização orientada a tópicos onde uma visão de *landscape* é construída sucessivamente sobrepondo as contribuições (pesos) de termos temáticos sobre um plano onde documentos são distribuídos de acordo com suas similaridades. Se um dado documento tem um termo, a altura de sua região é incrementada proporcionalmente a contribuição desse termo. A Figura 5.1 apresenta um exemplo de mapa de tópicos usando-se a *ThemeScape* para o conjunto **CBR-ILP-IR-SON**. Esse mapa foi gerado usando-se uma versão de avaliação da ferramenta IN-SPIRE^{TM} (PNNL, 2008).

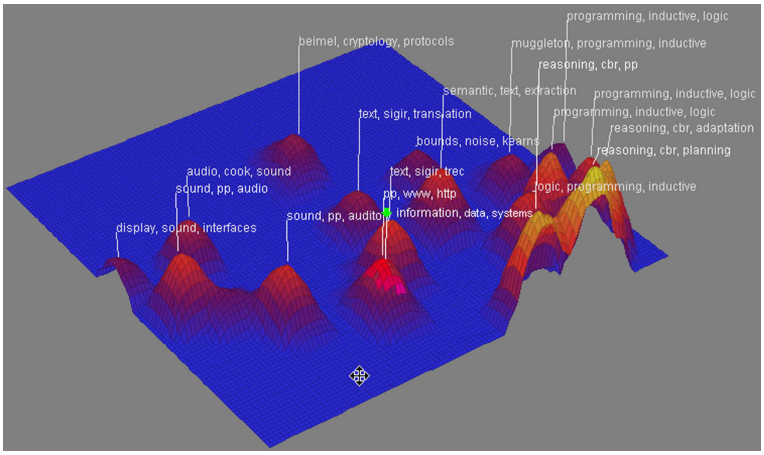


Figura 5.1: Mapa de tópicos do conjunto **CBR-ILP-IR-SON** criado usando-se abordagem *ThemeScape*.

A seguir é apresentada uma abordagem simples mas efetiva que desenvolvemos para detecção e extração de tópicos a partir de uma seleção de documentos em uma projeção. Essa abordagem é baseada na covariância entre diferentes termos da matriz de "termos x documentos". Diferentemente da maioria das técnicas discutidas anteriormente, com essa técnica espera-se que alguma semântica seja levada em consideração na escolha dos termos que formam um tópico, mais do que o apresentado a partir de simples contagens de freqüências.

5.2 Processo de Extração de Tópicos por Covariância

Na abordagem aqui definida, o processo de extração de um tópico se inicia com o usuário selecionando uma determinada área de uma projeção. Com base nessa seleção, uma matriz de "termos x documentos" é criada considerando-se somente os documentos escolhidos. Para isso, o mesmo processo apresentado na Seção 2.2.1 é utilizado, mas sem o emprego da lematização e do *inverse document frequency (idf)*. A lematização não é empregada porque deseja-se usar os

termos originais e não suas contrações em radicais. O *idf* não é aplicado uma vez que se busca capturar os termos mais comuns entre os documentos selecionados, sem ponderações. Além disso, para acelerar o processo do cálculo das covariâncias, ao invés de se usar todos os termos, somente os $k = 200$ primeiros termos, com maior freqüência de ocorrência, são empregados. O valor de k pode variar, mas 200 tem se mostrado o suficiente para capturar os termos mais importantes.

Uma vez que a matriz de "termos x documentos" tenha sido criada, os dois termos que apresentarem a maior covariância são inicialmente escolhidos e adicionados ao tópico. A covariância entre dois termos é uma medida do grau do quanto ambos variam juntos, sendo calculada como:

$$cov(t_i, t_j) = \frac{1}{n-1} \sum_{k=1}^{n} (t_{ki} - \bar{t_i})(t_{kj} - \bar{t_j}) \tag{5.1}$$

onde $\bar{t_i}$ é a média de ocorrência do i-ésimo termo t_i, e t_{ki} e t_{kj} são os valores do i-ésimo e j-ésimo termos para o k-ésimo documento.

Uma vez que os dois termos com maior covariância tenham sido encontrados, a covariância média entre esses e os demais é calculada, isto é,

$$cov_m(t_i, t_j, t_k) = \frac{cov(t_i, t_k) + cov(t_j, t_k)}{2} \tag{5.2}$$

onde t_i e t_j são os dois termos com maior covariância calculados anteriormente, e t_k é o termo para o qual se deseja calcular a covariância média com relação a t_i e t_j.

Se a razão do valor da covariância média pela maior covariância for igual ou ultrapassar um limiar pré-estabelecido (α), isto é, se $cov_m(t_i, t_j, t_k)/cov(t_i, t_j) \geq \alpha$, onde t_i e t_j são os dois termos com a maior covariância, o termo é adicionado ao tópico. Assim, somente os termos que apresentam alta dependência linear são adicionados, algo preferível a adicionar os mais freqüentes, uma vez que a freqüência não revela a existência de relacionamento entre termos, podendo resultar em tópicos com palavras não relacionadas.

O resultado final desse processo é uma lista de termos que apresentam forte relação linear. Dois fatores diferentes cooperam para o tamanho dessa lista: (1) o número de documentos selecionados; e (2) o valor de α escolhido. Quanto menor o número de documentos selecionados, isto é, quanto menor a vizinhança desses documentos na projeção, maior será a lista de termos. Isso ocorre uma vez que se espera que tais documentos sejam altamente similares, a característica principal de uma projeção, apresentando mais termos relacionados. Em relação a α, quanto maior seu valor, menor será de lista de termos, porém mais linearmente relacionados serão. Empiricamente definimos $\alpha = 0.5$, mas esse é um parâmetro que pode ser alterado de acordo com as variações da aplicação.

Analisando-se o processo de criação de um tópico é possível notar que o resultado final está fortemente ligado à escolha inicial dos dois termos com maior covariância porque os demais termos são derivados desses. Assim, criar apenas um tópico por conjunto de documentos

selecionados pode ocultar outros tópicos importantes, já que outros pares de termos com alta covariância podem não estar relacionados aos dois termos inicialmente escolhidos. De forma a evitar que isso ocorra, permitimos que múltiplos tópicos sejam criados para o grupo de documentos selecionados. Para tal, qualquer par de palavras cuja razão de sua covariância pela maior covariância ultrapassar ou for igual a um limiar pré-estabelecido (β), acaba gerando um novo tópico. O valor desse limiar pode variar conforme a necessidade de quem analisa uma projeção, mas empregando esse algoritmo em diferentes conjuntos de dados, com diferentes características, notamos que um valor $\beta = 0.75$ leva à criação de um número razoável e consistente de tópicos.

O processo de extração de tópicos por covariância é delineado no Algoritmo 5.1.

Algoritmo 5.1 Extração de Tópicos por Covariância.

entrada: - **M**: matriz $n \times m$ de "termos x documentos" processada para os n documentos selecionados contendo m termos.
- α : limiar para a adição de um novo termo a um tópico.
- β : limiar para a adição de um novo tópico à lista de tópicos.

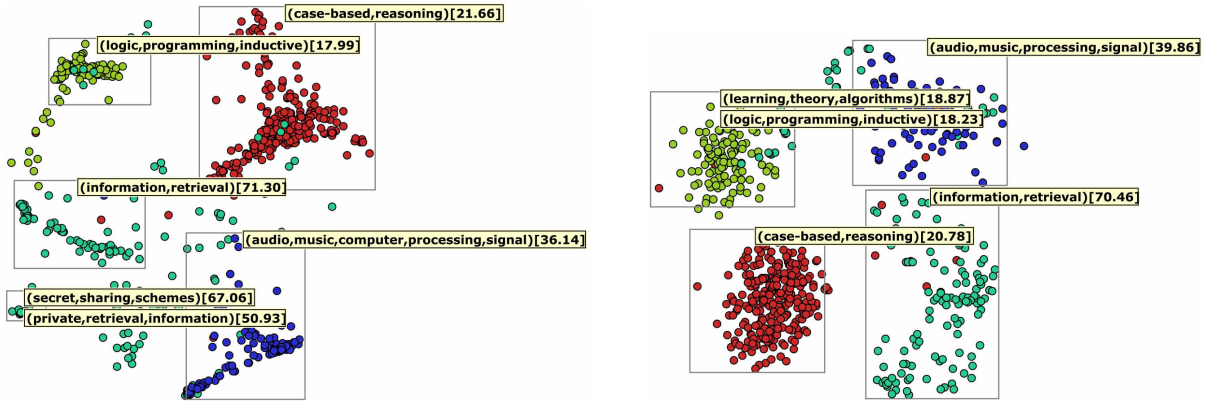
saída: - L: lista de tópicos extraídos.

1: $t_{i_{max}}, t_{j_{max}} = \max\{cov(t_i, t_j)\}\ \forall t_i, t_j \in \mathbf{M}$
2: $cov_{max} = cov(t_{i_{max}}, t_{j_{max}})$
3: **repita**
4: $t_i, t_j = \max\{cov(t_i, t_j)\}\ \forall t_i, t_j \in \mathbf{M}$ **e** ($t_i \notin l_b$ **ou** $t_j \notin l_b$, $\forall l_b \in L$)
5: $cov = cov(t_i, t_j)$
6: **se** $cov/cov_{max} \geq \beta$ **então**
7: criar um novo tópico $T = \emptyset$
8: adicionar os termos t_i e t_j ao tópico T.
9: **para todo** $t_k \in M$ e $t_k \neq t_i$ e $t_k \neq t_j$ **faça**
10: **se** $cov_m(t_i, t_j, t_k)/cov \geq \alpha$ **então**
11: adicionar o termo t_k ao tópico T.
12: **fim se**
13: **fim para**
14: adicionar o tópico T a lista de tópicos L.
15: **fim se**
16: **até** $cov/cov_{max} > \beta$

5.3 Resultados

A Figura 5.2 apresenta tópicos extraídos usando-se essa abordagem por covariância para projeções criadas usando-se a LSP (mesma projeção da Figura 4.2(a)) e ProjClus (mesma projeção da Figura 3.1(b)). Os dois primeiros termos desses tópicos são os que apresentaram a maior covariância, a qual é indicada pelo número entre colchetes. Os demais termos são os que excederam o limite α. Para a criação dos tópicos na projeção LSP (Figura 5.2(a)), as cinco áreas que apresentam conjuntos de documentos claramente identificáveis foram selecionadas. Já no

caso da projeção gerada pela ProjClus (Figura 5.2(b)), as quatro áreas facilmente identificáveis foram selecionadas para criar os tópicos. Os retângulos delimitam essas seleções. É possível notar que apesar de ser um processo simples de detecção e extração de tópicos, as quatro grandes áreas dentro desse conjunto de documentos foram identificadas em ambas as projeções. Além disso, no caso da projeção LSP, os tópicos indicam que a separação da grande área de *Information Retrieval* em duas ocorre porque existe um sub-grupo de documentos bem específico dentro dessa grande área, identificado principalmente como "(secret, sharing, schemes)" – documentos relacionados com recuperação de informação de dados sigilosos –, mostrando-se um mecanismo eficiente para a descoberta da causa da formação dos agrupamentos de documentos nas projeções.



(a) Projeção LSP, $\alpha = 0.5$ e $\beta = 0.75$

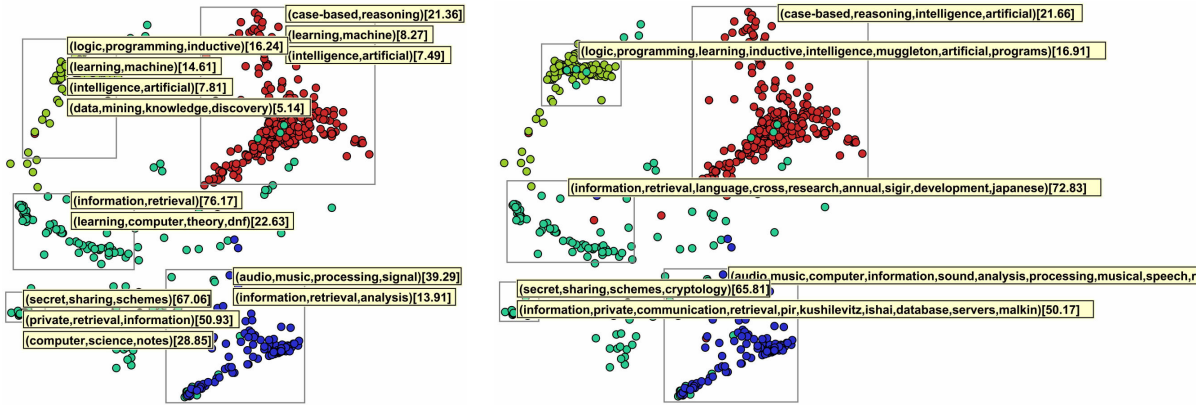
(b) Projeção ProjClus, $\alpha = 0.5$ e $\beta = 0.75$.

Figura 5.2: Tópicos extraídos para as projeções LSP e ProjClus do conjunto de dados **CBRILP-IR-SON**. Os tópicos conseguem identificar com precisão os grupos de documentos selecionados.

O processo de extração de tópicos apresentado aqui envolve dois diferentes parâmetros: (1) β que limita a quantidade de tópicos extraídos para um dado grupo de documentos; e (2) α que limita a quantidade de termos adicionados a um tópico. A Figura 5.3 apresenta o resultado de variar esses dois parâmetros para a projeção LSP do conjunto **CBR-ILP-IR-SON**. Na Figura 5.3(a) fazendo $\beta = 0.25$, mais tópicos para um mesmo grupo de documentos são extraídos se comparado com o valor padrão de $\beta = 0.75$ (Figura 5.2(a)). Essa possibilidade de extrair mais tópicos para uma mesma área pode ser útil no cenário em que dentro dos documentos selecionados existem sub-áreas, levando assim à definição de sub-tópicos. Isso pode ser observado para os tópicos extraídos para os documentos de ILP. O tópico de maior valor é o que identifica a área, "(logic, programming, inductive)", mas também existem sub-tópicos relacionados a essa área, como "(data, mining, knowledge, discovery)", definindo sub-grupos específicos de documentos.

Além de β, α também pode variar. Na Figura 5.3(b) toma-se $\alpha = 0.25$, definindo dessa forma tópicos com mais termos – compare essa com a produzida com $\alpha = 0.5$ (Figura 5.2(a)). Apesar da adição de termos poder ajudar o usuário na identificação das áreas, já que mais informação é fornecida sobre um determinado grupo de documentos, tornar α muito pequeno pode resultar na definição de tópicos com palavras não muito relacionadas. Assim, deve-se ter

cuidado na definição tanto de α quanto β para que não se escolha valores muito altos, que limitem a identificação correta do conteúdo presente na coleção de documentos, nem valores muito pequenos que tragam mais informação que o necessário para o usuário interpretar ou mesmo que defina tópicos não muito coerentes.

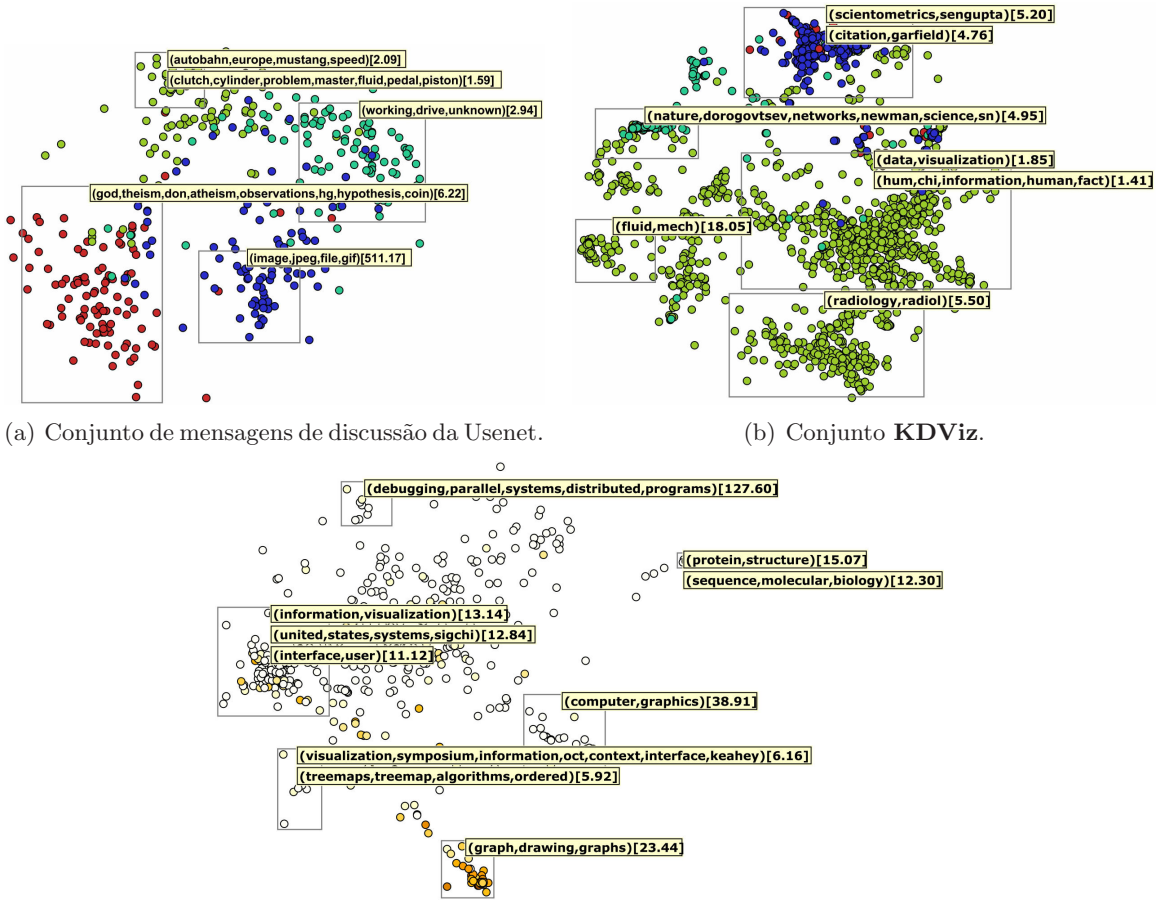


(a) $\alpha = 0.5$ e $\beta = 0.25$. (b) $\alpha = 0.25$ e $\beta = 0.75$.

Figura 5.3: Exemplos de tópicos extraídos para o conjunto **CBR-ILP-IR-SON** variando-se os parâmetros α e β. Quanto menor o α, mais termos são adicionados ao tópico. Quanto menor o β, mais tópicos são extraídos para um mesmo grupo de documentos.

A Figura 5.4 apresenta tópicos extraídos para outras coleções de documentos. Todas essas projeções foram criadas usando a LSP. Na Figura 5.4(a) são mostrados os tópicos extraídos para o conjunto de mensagens de quatro grupos de discussão da Usenet. Dos quatro grupos, três são facilmente identificáveis pelos tópicos extraídos, que são "atheism", "autos" e "computer graphics". O quarto é um grupo definido como "miscellaneous/forsale", portanto não definindo uma única área. Dessa forma, o tópico extraído não identifica o conteúdo desses documentos por não tratarem de um assunto realmente em comum. Na Figura 5.4(b) os tópicos para os grupos de documentos mais destacados para a projeção do conjunto **KDViz** são apresentados. Nesse caso, o grande grupo inicialmente identificado apenas como "information visualization" é na verdade um grupo composto por diferentes sub-grupos, com documentos relacionados a mecânica dos fluidos, visualização de dados, radiologia, etc.

A Figura 5.4(c) apresenta a projeção e os tópicos extraídos para o conjunto de dados **INFOVIS04**. Esse foi disponibilizado no *2004 IEEE Information Visualization Contest* (Fekete et al., 2004) e é composto por documentos publicados na mesma conferência e alguns freqüentemente citados nessa. Assim, o conteúdo desse conjunto é bastante homogêneo, o que sugere maior dificuldade para se definir agrupamentos por conteúdo. Mesmo nesse caso, documentos com assuntos similares foram agrupados e separados bem na projeção, e os tópicos extraídos conseguiram identificar bem esses grupos. Nessa projeção, alguns sub-tópicos dentro do campo de visualização de informação são identificados, e os pontos são coloridos de acordo com a freqüência da palavra "graphs".



(a) Conjunto de mensagens de discussão da Usenet.

(b) Conjunto **KDViz**.

(c) Conjunto **INFOVIS04**.

Figura 5.4: Tópicos extraídos para projeções de diferentes coleções de documentos. Os resultados indicam que a extração de tópicos consegue identificar bem grupos de documentos presentes em coleções compostas de documentos com diferentes características.

Os exemplos apresentados até agora foram para coleções de documentos compostas por documentos com um tamanho razoável, próximo do tamanho do resumo de um artigo científico. No exemplo a seguir mostramos o resultado para uma coleção composta de documentos pequenos. Esses documentos são notícias curtas (*RSS feeds*) disponíveis em páginas Web de agências de notícias, normalmente incluindo um título e uma pequena frase sintetizando a notícia. Esse conjunto, aqui chamado de **NOTÍCIAS**, é composto de notícias coletadas dos sites da *Associated Press* (`www.ap.org`), *Reuters* (`www.reuters.com`), *BBC* (`www.bbc.com`), e *CNN* (`www.cnn.com`) durante dois dias em Abril de 2006, formando uma coleção com 2.625 documentos. A Figura 5.5 apresenta a projeção LSP para esse conjunto e os tópicos mais relevantes extraídos.

Apesar desse ser um conjunto difícil por ser composto de documentos com pouca informação (Cuadros et al., 2007), é possível notar que a projeção agrupa bem os documentos que tratam do mesmo assunto, e os tópicos extraídos conseguem refletir as principais notícias divulgadas pelas diferentes agências citadas. Dessa forma é possível analisar, por exemplo, os assuntos em comum que essas agências deram maior importância nesses dois dias de Abril. Nessa

Figura 5.5: Projeção LSP e tópicos extraídos para uma coleção de documentos pequenos. Apesar do pouco conteúdo de cada documento, o resultado final é satisfatório, agrupando bem documentos sobre o mesmo assunto.

figura, os pontos estão coloridos de acordo com a freqüência de ocorrência de certos termos que aparecem nos tópicos extraídos, revelando a coerência dos agrupamentos formados e dos tópicos extraídos.

Pelos tópicos extraídos podemos destacar algumas das principais notícias durante esses dois dias (6 e 7 de Abril de 2006):

- um cisne foi encontrado morto na Escócia, gerando preocupações sobre a febre aviária ("bird flu"). Em seguida, o pássaro foi testado e ficou confirmado a presença do vírus H5N1, o responsável pela febre. A população foi avisada sobre os problemas relacionados à mesma, gerando certa tensão;
- o projeto de lei sobre imigração e imigrantes ilegais nos Estados Unidos foi apresentado no senado pela primeira vez;
- o caso sobre vazamento de informação na "Casa Branca" sobre a guerra no Iraque estava sendo investigado;
- a execução de um atentado suicida em Baghdad e outro em Najaf, Iraque, deixaram várias pessoas mortas;
- no oriente médio, o Hamas assumiu o governo da Palestina; ataques aéreos israelenses aconteceram na faixa de Gaza;

- o prefeito de Nova Iorque Rudolph Giuliani testemunha no julgamento de Zacarias Moussaoui sobre os atentados de 11 de Setembro de 2001;
- acontecem protestos contra o rei do Nepal após a declaração de toque de recolher devido a uma passeata anti-monarquia;
- o torneio de golfe em Augusta está acontecendo.

5.4 Considerações Finais

Nesse capítulo apresentamos uma abordagem simples para a extração de tópicos de projeções multi-dimensionais de coleções de documentos. Apesar de simples, essa abordagem se mostra bastante eficiente na identificação dos assuntos comuns tratados por certos grupos de documentos. Diferentemente da maioria das abordagens para extração de tópicos, a apresentada aqui não se baseia na freqüência de ocorrência de termos, mas sim na dependência linear entre a ocorrência deles. Assim, espera-se que os termos empregados para definir um tópico sejam mais coerentes, já que a freqüência de ocorrência pode não revelar a existência de relação entre termos.

O processo de extração de um tópico de acordo com essa abordagem inicia-se com o usuário selecionando um grupo de documentos em uma determinada projeção multi-dimensional. A partir dessa seleção, uma matriz de "termos x documentos" é criada levando-se em consideração somente esses documentos. Em seguida, com base nessa matriz, a covariância entre os termos é calculada e os termos que apresentarem maior relação linear são escolhidos para se formar o tópico.

Nesse processo dois parâmetros são definidos, um que limita o número de termos presentes em um tópico (α) e um que limita a quantidade de tópicos e sub-tópicos extraídos para um certo grupo de documentos (β). Assim, controlando esses parâmetros o usuário pode extrair tópicos em diferentes níveis de detalhamento, isto é, tópicos com mais ou menos informação (número de termos), ou limitar a quantidade de sub-tópicos extraídos para os documentos selecionados. Apesar dessa flexibilidade ser interessante, deve-se ter cuidado na escolha desses parâmetros para que não se defina tópicos com termos não muito relacionados, ou que se extraia muito sub-tópicos, forçando o usuário a interpretar mais informação do que o necessário. O inverso também sendo verdade, deve-se evitar que se defina tópicos com pouca informação, ou que se deixe de extrair sub-tópicos importantes. De qualquer forma, esses são parâmetros do algoritmo apresentado e o usuário pode modificá-los e analisar os resultados, escolhendo os melhores valores para uma determinada tarefa.

Uma observação importante sobre essa técnica é a de que a covariância não é uma relação de causalidade entre termos, isto é, essa não indica que um termo ocorre devido a outro. Essa é somente uma medida do grau no qual dois termos variam juntos. Por exemplo, na Figura 5.2(a), os termos "audio" e "music" apresentam um alto grau de covariância, mas não é possível estabelecer que o termo "audio" causa o termo "music" ou o inverso. A única indicação que temos é que a ocorrência do termo "audio" normalmente está associada à ocorrência do

termo “music”, mais do que a ocorrência associada de quaisquer outros termos dentro dos documentos selecionados. Uma técnica, também utilizada para extração de tópicos de projeções multi-dimensionais, desenvolvida no nosso grupo de pesquisa[1] e que pode apresentar relações de causalidade foi definida em (Lopes et al., 2007). Nela, a partir da matriz de “termos x documentos”, criada considerando documentos selecionados, regras de associação são geradas e filtradas, identificando os termos mais relacionados dentro dessa seleção.

Apesar da técnica de extração de tópicos definida aqui ter apresentado resultados satisfatórios para os testes realizados, ela só funciona quando a projeção multi-dimensional gerada consegue agrupar bem documentos similares. Se os documentos selecionados não forem similares, pouca informação em comum deve existir e a covariância não revela conteúdo. Nesse caso, o tópico refletirá os termos de maior freqüência que podem ocorrer, por exemplo, em um único ou poucos documentos. Assim, o sucesso da extração dos tópicos está intimamente relacionado à qualidade dos layouts gerados.

[1]Grupo de Computação Gráfica e Processamento de Imagens (CG&PI) do Instituto de Ciências Matemáticas e de Computação (ICMC), Universidade de São Paulo (USP), São Carlos/SP.

Capítulo

6

Hierarchical Point Placement Strategy (HiPP)

6.1 Considerações Iniciais

Pela literatura atual sobre projeções multi-dimensionais, inclusive pelas técnicas desenvolvidas no contexto desse projeto de doutorado (ProjClus e LSP), é possível notar uma crescente preocupação com a capacidade dessas de lidar com grandes conjuntos de dados, ou seja, com a escalabilidade. Nesse sentido, muito do quem sido desenvolvido se concentra na redução da complexidade computacional, possibilitando o processamento de volumes de dados cada vez maiores. Apesar dessa ser uma preocupação legítima, outro fator de escalabilidade também deve ser considerado: a *escalabilidade visual*. Por escalabilidade visual entende-se a capacidade da representação visual, e das ferramentas de visualização, de efetivamente apresentar grandes conjuntos de dados, tanto em termos do número de dimensões quanto de elementos individuais (Eick e Karr, 2002). Os fatores que afetam a escalabilidade visual incluem a qualidade visual do mecanismo empregado para apresentar a visualização (normalmente o monitor de um computador), as metáforas visuais empregadas na representação da informação, as técnicas usadas para interagir com a representação visual, e a capacidade de percepção do sistema cognitivo humano (Thomas e Cook, 2005).

A Figura 6.1 apresenta um exemplo de projeção multi-dimensional onde a escalabilidade visual começa a ser afetada, apesar da complexidade computacional e o tempo empregado na sua geração serem aceitáveis. Essa é uma projeção LSP para uma coleção de 30.000 documentos extraídos do conjunto de notícias da Reuters (Lewis et al., 2004). Devido ao tamanho do espaço visual disponível e à metáfora visual empregada (um círculo para cada documento), a interpretação dessa projeção é prejudicada, sendo difícil a identificação dos grupos de documentos, e a

interpretação das relações de similaridade entre grupos e entre as instâncias individuais de dados. Além disso, a apresentação de muita informação simultânea para o usuário pode significar uma sobrecarga na tarefa de interpretação de grandes coleções de documentos, dificultando a extração de informação relevante.

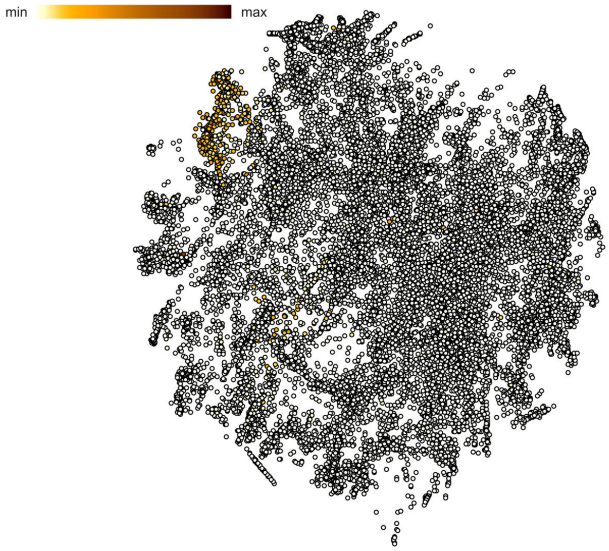


Figura 6.1: Problemas de escalabilidade visual associados à representação normalmente empregada para para visualizar o resultado de uma projeção multi-dimensional.

De forma a apontar uma possível solução para esse problema, e partindo-se da observação de que uma projeção multi-dimensional normalmente é empregada para extrair grupos de objetos multi-dimensionais similares, relações entre esses, e entre objetos individuais, desenvolvemos uma técnica chamada *Hierarchical Point Placement (HiPP)* (Paulovich e Minghim, 2008). Na HiPP, os elementos visuais podem representar tanto instâncias individuais de objetos multi-dimensionais quanto grupos de instâncias. Ela define uma árvore de agrupamentos hierárquicos de forma que cada nível representa uma visão diferente dos possíveis agrupamentos que podem existir no conjunto de dados. O primeiro nível representa os agrupamentos mais abstratos, e os níveis subseqüentes representam sub-grupos mais detalhados até instâncias de dados individuais. Isso possibilita que um processo exploratório baseado em refinamentos sucessivos seja aplicado, começando com uma visão geral do conjunto de dados, e concentrando-se nos agrupamentos de interesse, finalmente alcançar as instâncias individuais de dados. Assim, menos informação é imposta ao usuário para a interpretação de uma primeira visão de um conjunto de dados, e o usuário pode mais facilmente se concentrar somente na informação de interesse. Esta estratégia está de acordo com a filosofia corrente na comunidade de análise visual de dados de que a interpretação global seguida de foco em áreas de interesse é ideal (Shneiderman, 1996; Card et al., 1999).

A idéia da aplicação de uma organização hierárquica e projeções multi-dimensionais já foi sugerida em uma outra técnica para a construção de mapas de documentos, chamada *Infosky* (Andrews et al., 2002). A Infosky é uma abordagem onde o usuário pode ampliar ou reduzir

certas áreas da projeção multi-dimensional, analogamente à manipulação de um telescópio, de forma a explorar coleções de documento em diferentes níveis de detalhe. Porém, diferentemente da técnica apresentada nesse capítulo, aquela assume que os documentos já estão organizados em uma hierarquia de coleções e sub-coleções, portanto não podendo ser aplicada para explorar conjuntos não-estruturados de documentos, limitando sua aplicação.

Em geral, técnicas hierárquicas de visualização têm sido amplamente empregadas na análise de dados. Elas são normalmente classificadas em dois diferentes grupos (Ward, 2002): técnicas de preenchimento do espaço visual (tais como *Treemap* (Johnson e Shneiderman, 1991), *Voronoi Treemap* (Balzer et al., 2005), ou *Circle Packing* (Wang et al., 2006)), e as de ligação-de-nós (tais como *Cone Tree* (Robertson et al., 1991) ou *Hierarchical Clustering Explorer* (Seo e Shneiderman, 2002)). Embora as técnicas de preenchimento do espaço visual façam um melhor uso do espaço se comparado à HiPP, e os dois grupos de técnicas sejam bastante efetivos em revelar a estrutura hierárquica dos dados, as representações visuais resultantes não contém um elemento importante, que é a capacidade de revelar a similaridade entre os elementos (nós ou folhas) no layout final. A técnica de *Coordenadas Paralelas Hierárquicas* (Fua et al., 1999) é uma técnica hierárquica que representa alguma informação sobre similaridade baseada na coloração dos elementos visuais, porém impõe uma ordenação linear das dissimilaridades entre os agrupamentos, o que nem sempre é possível, sendo portanto bastante restrita para a investigação de similaridade em diferentes níveis de detalhamento. O objetivo da HiPP é a preservação de relações de similaridade nos vários níveis de detalhe, agrupando instâncias similares enquanto também separa visualmente grupos dissimilares, assim impondo uma restrição adicional ao processo de criação da representação visual.

Por fim, outro trabalho relacionado, porém desenvolvido para resolver um problema em uma área diferente, foi apresentado em (Walshaw, 2001). Nele, uma estratégia de *Force Direct Placement (FDP)* (veja Seção 2.3.2) é empregada para acelerar o processo de planarizar um grafo. Embora estratégias de FDP possam ser usadas para projeções multi-dimensionas, normalmente essas não podem lidar com nós de tamanhos variados, o que é uma parte importante da hierarquia de similaridades definida aqui. Apesar de estratégias de FDP poderem ser adaptadas para considerar nós de tamanho diferentes, normalmente com essa adaptação elas tendem a convergir muito lentamente (Harel e Koren, 2002), limitando a aplicação a grandes conjuntos de dados.

As idéias, conceitos e técnicas que formam a HiPP são detalhadas nas próximas seções. Em síntese, podemos dividir essa técnica em dois grandes passos: (1) a árvore de agrupamentos hierárquicos é criada; e (2) os elementos dessa árvore são mapeados para o espaço bi-dimensional para criar a representação visual.

6.2 Criando a Árvore Hierárquica de Agrupamentos

A árvore hierárquica de agrupamentos é construída usando um processo recursivo de particionamento onde os nós internos são agrupamentos e as folhas são instâncias individuais de dados.

Este processo começa com um nó, $RAIZ$, contendo todas as instâncias de dados. Então, essas instâncias são divididas em $k = \sqrt{|RAIZ|}$ nós, criando os nós filhos de $RAIZ$, $C_1, C_2, \ldots, C_k$ ($|\cdot|$ denota o número de instâncias de dados em um nó). Em seguida, cada nó filho C_i é dividido em $\sqrt{|C_i|}$ nós, e os nós resultantes são ligados como seus filhos. Este processo de particionamento é aplicado recursivamente para cada novo nó até que um número mínimo de instâncias de dados nos nós seja atingida (min). Quando um nó atinge esse mínimo, suas instâncias de dados são convertidas em folhas da árvore resultante. Esse processo é descrito no Algoritmo 6.1.

Apesar de parecer arbitrária, a divisão de cada agrupamento em $\sqrt{\cdot}$ sub-agrupamentos se baseia em uma heurística freqüentemente empregada em algoritmos de agrupamento para definir o limite superior de agrupamentos que podem existir em um conjunto de dados (Pal e Bezdek, 1995). Como a primeira visão que um usuário tem do conjunto é o primeiro nível de agrupamentos, se menos agrupamentos forem criados do que o que realmente existe, a visão geral apresentada irá ocultar informação, sem que o usuário tenha conhecimento disso. Dessa forma, optamos por empregar um limite superior para evitar esse problema e também porque, como será mostrado nas próximas seções, agrupamentos com conteúdo similar serão posicionados próximos no plano, e a técnica apresentada dá suporte ao usuário para unir agrupamentos conforme a representação visual é explorada.

Algoritmo 6.1 Construção de Árvore Hierárquica de Agrupamentos

entrada: - N: nó a ser dividido.
- min: número mínimo de elementos que um nó deve conter de forma a ser dividido.

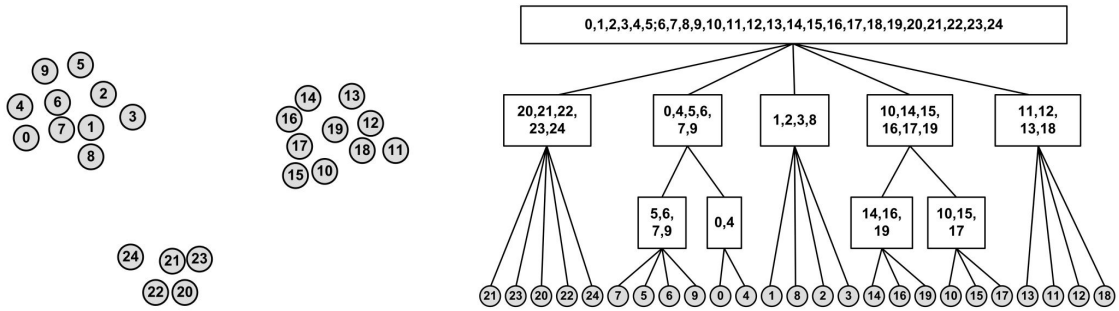
saída: - uma hierarquia de agrupamentos.

procedimento $Hierarquia(N)$
1: **se** $|N| > min$ **então**
2: $CH \leftarrow Particionar(N, \sqrt{|N|})$
3: **para todo** $C \in CH$ **faça**
4: adicionar C como filho de N
5: $Hierarquia(C)$
6: **fim para**
7: **senão**
8: **para todo** instância de dados d em N **faça**
9: adicionar d como filho de N
10: **fim para**
11: **fim se**

O número mínimo de instâncias que um nó deve conter de forma a ser dividido influencia a interação com o usuário e a quantidade de área usada na representação visual final. Se $min \ll n$, onde n é o número de instâncias no conjunto de dados, o usuário precisará executar vários cliques no mouse para expandir os agrupamentos até atingir as instâncias reais de dados. Além disso, uma vez que a representação visual desenha um conjunto de círculos dentro da área de um círculo pai, a área que é usada em cada novo nível é sempre menor que a área do nível superior.

Assim, uma árvore muito profunda leva a pouca utilização da área visual disponível para a apresentação dos últimos níveis da hierarquia de agrupamentos. Aqui, nós usamos $min = \sqrt{n}$ e $min > 2$. Quando a árvore for balanceada, essa não terá mais de 3 níveis. Em nossos testes, com diferentes conjuntos de dados, a profundidade nunca excedeu 5 níveis, e na média ficou em 4. Portanto, o usuário normalmente tem que executar 2 cliques nos agrupamentos de forma a atingir as instâncias de dados.

A Figura 6.2 mostra um exemplo simples de 25 pontos (círculos) no plano, representando instâncias de dados, e uma árvore hierárquica de agrupamentos criada para tais pontos. Os nós da árvore são representados por retângulos com números indicando as instâncias de dados que cada nó contém. Inicialmente o nó raiz contém todas as instâncias de dados. Em cada nível essas instâncias são divididas em novos nós até que nenhum nó tenha mais do que $min = 5$ elementos. Quando um nó não pode ser dividido, cada uma de suas instâncias de dados se torna uma folha (os círculos cinza).



(a) Pontos no plano representando um grupo de instâncias de dados bi-dimensionais.

(b) Uma árvore hierárquica de agrupamentos para o conjunto bi-dimensional de pontos.

Figura 6.2: Um exemplo de um conjunto bi-dimensional de instâncias de dados e uma possível árvore hierárquica de agrupamentos para esse conjunto.

6.2.1 Estratégia de Particionamento

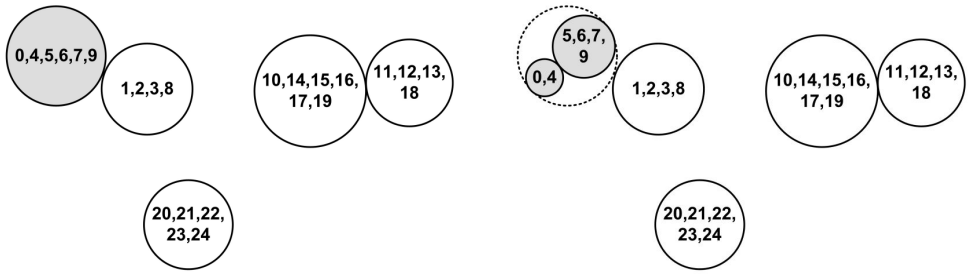
De forma a dividir as instâncias de dados e criar os nós filhos, um algoritmo de agrupamento por particionamento, chamado *bisecting k-means* (Steinbach et al., 2000), é empregado. Esse algoritmo foi escolhido por alcançar bons resultados para coleções de documentos e por apresentar baixo custo computacional, sendo portanto bastante conveniente para grandes coleções. Além disso, leva a definição de agrupamentos de tamanho semelhante (conforme discutido na Seção 3.2), uma característica importante para a técnica aqui apresentada.

6.3 Projetando a Hierarquia

Uma vez que a árvore hierárquica de agrupamentos tenha sido criada, seus nós são projetados de forma a compor a representação visual. Projetar um nó significa que seus nós filhos são

posicionados no plano dentro da área definida pelo mesmo, preservando o máximo possível as relações de vizinhança e similaridade. Inicialmente o nó *RAIZ* é projetado no plano. Quando requisitado pelo usuário, seus nós filhos podem ser projetados, e assim por diante, até atingir os nós folha, isto é, as instâncias de dados.

A Figura 6.3 mostra um exemplo de uma representação visual gerada para a hierarquia apresentada na Figura 6.2(b). Na Figura 6.3(a) a projeção do nó *RAIZ* é apresentada. O tamanho de cada círculo é proporcional ao número de instâncias de dados que pertencem a ele, e os números dentro de cada círculo indicam as instâncias de dados. A Figura 6.3(b) mostra o resultado se um usuário selecionar o nó *(0,4,5,6,7,9)* para ser projetado – este nó é substituído por seus filhos no layout final.



(a) Projetando o nó *ROOT*. (b) Projetando o nó *(0,4,5,6,7,9)*.

Figura 6.3: Exemplo de projeção da hierarquia definida na Figura 6.2(b).

Embora a escolha direta para mapear objetos multi-dimensionais no plano preservando relações de similaridade sejam as estratégias de projeção multi-dimensional (veja Capítulo 2), uma vez que no nosso caso temos círculos (que ocupam uma certa área) ao invés de pontos, estas técnicas não podem ser diretamente aplicadas. Além disso, a sobreposição dos nós deve ser evitada e as técnicas de projeção multi-dimensional normalmente não resolvem isso. A sobreposição não deve ocorrer uma vez que nós filhos de um determinado nó são posicionados dentro da sua área, e na presença de sobreposições, instâncias de dados que não são relacionadas podem ser projetadas próximas no layout final.

A abordagem proposta aqui divide o processo de projeção em dois passos. Primeiro, os nós filhos são projetados no plano usando uma estratégia de projeção multi-dimensional sem considerar seus tamanhos, isto é, eles são representados por seus centros. Então, estes nós são espalhados no plano de forma a resolver as sobreposições de área.

Para a projeção empregamos a técnica *Least Square Projection (LSP)* (ver Capítulo 4) ligeiramente modificada para levar em consideração o posicionamento dos nós vizinhos de um nó quando ele é projetado. Essa modificação será discutida na Seção 6.3.2.

O algoritmo para a distribuição dos nós foi projetado como segue: para cada nó filho C_i é verificado se uma sobreposição ocorre com todos os outro nós C_j. Para cada nó com o qual existe sobreposição, ambos os nós são movidos em direções opostas entre si, com C_j dando um passo maior do que C_i (sem sair da área de seu nó pai). Esse passo é proporcional ao tamanho

dos dois nós, mas ao invés de dar o passo inteiro para se evitar a sobreposição – que seria a soma dos raios dos nós – uma fração desse passo é empregada ($frac$). Executando esse algoritmo em diferentes conjuntos de dados e observando os resultados, nós verificamos que um valor entre $\frac{1}{4} \leq frac \leq \frac{1}{8}$ previne uma mudança na posição dos nós que é muito grande em cada iteração do algoritmo, o que poderia afetar severamente as relações de vizinhança e similaridade entre esses. Usando essa fração, os nós são suavemente espalhados em cada iteração. De qualquer forma, esse é um parâmetro do algoritmo que pode ser mudado e os resultados finais avaliados pelo usuário.

Se na projeção original não existir sobreposição entre dois nós, então uma constante (*const*) é adicionada à separação que deve existir entre eles após o algoritmo de espalhamento ser aplicado. Esta constante é proporcional à área do nó pai e visa preservar no layout final a separação entre grupos definidos na projeção original. A ordem de execução deste algoritmo é a partir do nó com maior número de instâncias para o nó com o menor número de instâncias. Assim, os maiores nós ficarão mais "fixos" do que os menores, assegurando uma convergência mais rápida do algoritmo. Este processo de espalhamento é executado até que os nós não se movam mais (ausência de sobreposição) ou até que um máximo número de iterações seja alcançado (max). Este processo é esboçado no Algoritmo 6.2.

No Algoritmo 6.2 $dist(C_i, C_j)$ significa a distância no plano entre os centros dos círculos C_i e C_j, e $tamanho(C_i)$ se refere ao tamanho (raio) de C_i.

6.3.1 Definindo o tamanho dos nós

Toda vez que um nó é projetado, os raios dos nós filhos devem ser definidos. Isto é feito de forma que a área dos nós filhos seja proporcional ao número de instâncias de dados que pertençam aos mesmos, e que a área do nó pai (o nó sendo projetado) seja preenchida, até um limite, sem que haja sobreposição. Uma vez que esse é um problema de otimização não-linear difícil de ser resolvido, nós empregamos uma solução simples.

Ignorando a condição de não-sobreposição, este problema pode ser resolvido assinalando um fator f_i a cada nó filho C_i, com $0 \leq f_i \leq 1$ e $\sum f_i = 1$, calculando k de acordo com a seguinte equação:

$$\pi(k * f_1)^2 + \pi(k * f_2)^2 + \ldots + \pi(k * f_n)^2 = \pi r^2$$

isto é,

$$k = \sqrt{\frac{r^2}{f_1^2 + f_1^2 + \ldots + f_n^2}} \tag{6.1}$$

e calculando o raio do nó C_i como $k * f_i$.

Algoritmo 6.2 Distribuindo os nós para evitar sobreposições.

entrada: - N: nó ser apresentado.
- max: número máximo de iterações.
- $frac$: fração de movimento para se evitar sobreposição.
- $const$: constante de separação entre dois agrupamentos se não existir sobreposição na projeção original.

saída: - posicionamento dos filhos do nó N no plano.

procedimento $Espalhador(N)$

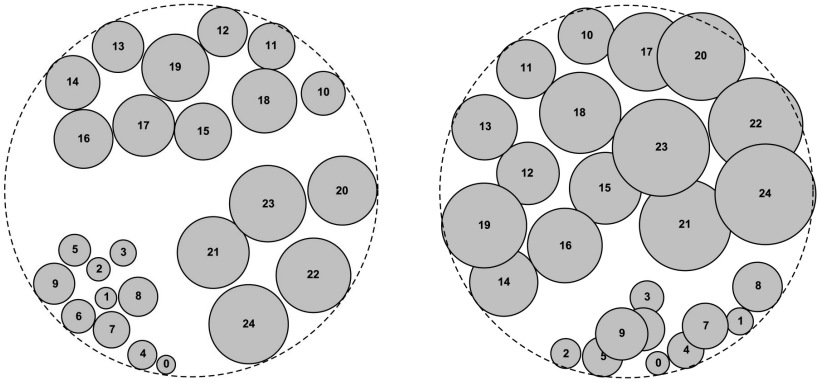
1: $iteracao \leftarrow 1$
2: $CH \leftarrow$ pegar of filhos de N.
3: **repita**
4: $mudou \leftarrow$ **falso**
5: **para todo** $C_i \in CH$ **faça**
6: **para todo** $C_j \in CH$ **e** $C_i \neq C_j$ **faça**
7: $d \leftarrow dist(C_i, C_j)$
8: $s \leftarrow tamanho(C_i) + tamanho(C_j)$
9: **se** C_i **não** sobrepõe originalmente C_j **então**
10: $s \leftarrow s + const$
11: **fim se**
12: **se** $s > d$ **então**
13: $\Delta \leftarrow (s - d) * frac$
14: Calcular o vetor $\overrightarrow{v}$ de C_i para C_j.
15: Mover C_j na direção de $\overrightarrow{v}$ um passo do tamanho $\frac{3\Delta}{4}$.
16: Mover C_i na direção oposta de $\overrightarrow{v}$ um passo do tamanho $\frac{\Delta}{4}$.
17: $mudou \leftarrow$ **verdadeiro**
18: **fim se**
19: **fim para**
20: **fim para**
21: $iteracao \leftarrow iteracao + 1$
22: **até** $\neg mudou$ **ou** $iteracao > max$.

Nessa equação, r é o raio do nó pai e os fatores f_i são proporcionais ao número de instâncias de dados que pertence ao nó filho C_i. Esses fatores são calculados como $f_i = |C_i|/|P|$, onde P denota o nó pai.

Se a abordagem é empregada dessa forma, a soma da área dos nós filhos vai ser igual a área do nó pai, o que leva a sobreposições e a uma separação não clara entre os grupos de nós relacionados. A solução adotada então é usar uma fração da área do nó pai. Nos resultados apresentados nessa tese usamos 50% da área do nó pai, mas isto pode ser mudado de forma a se usar mais ou menos área do espaço visual – somente observa-se que se muita área for requerida, as condições de não-sobreposição e a separação entre os grupos de nós relacionados podem não ser completamente respeitadas.

A Figura 6.4 apresenta o posicionamento dos nós da Figura 6.2(a) (agora usando nós de diferentes tamanhos), variando a fração da área usada. Na Figura 6.4(a) 50% da área foi utilizada

e na Figura 6.4(b) 80%. No segundo caso a sobreposição não é evitada, e é difícil distinguir os três possíveis grupos de nós que originalmente existem. No primeiro caso, é possível distinguir visualmente os três grupos de nós e o posicionamento final produz vizinhanças bem próximas às vizinhanças originais dos pontos.



(a) Usando 50% da area. (b) Usando 80% da area.

Figura 6.4: Diferentes frações de áreas ocupadas pelos nós filhos.

6.3.2 Projetando com o uso de âncoras

Quando um nó é projetado seus nós filhos são posicionados no plano de forma que filhos similares sejam colocados próximos. Quando os filhos são instâncias de dados, a dissimilaridade entre eles é diretamente calculada uma vez que esses são vetores multi-dimensionais. No caso dos agrupamentos, a dissimilaridade é calculada considerando seus centróides. Centróides são vetores pertencentes ao mesmo espaço das instâncias de dados, assim o mesmo processo usado para calcular a dissimilaridade entre essas pode ser aplicado para calcular a dissimilaridade entre agrupamentos, e entre agrupamentos e instâncias de dados.

De forma a realizar tal projeção empregamos a técnica *Least Square Projection (LSP)* devido às suas características favoráveis para coleções de documentos (veja Capítulo 4).

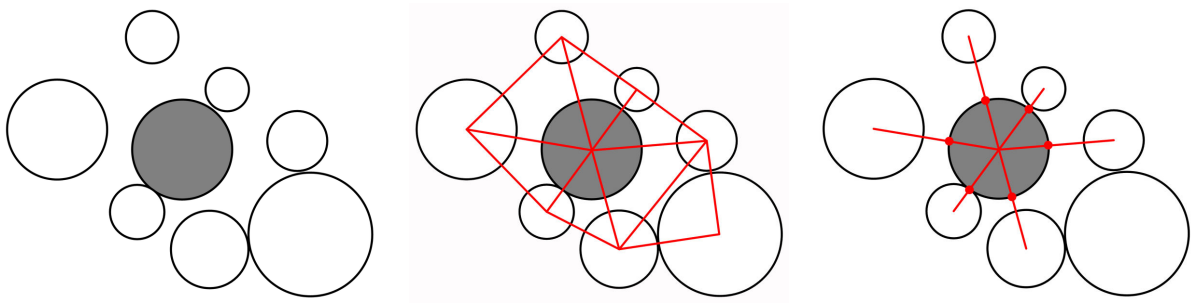
A LSP normalmente resulta em layouts precisos em termos de agrupar as instâncias similares e separar as não correlacionadas – embora exista uma tendência de se criar layouts com grande sobreposição. Isto, entretanto, é resolvido pelo algoritmo de espalhamento (Algoritmo 6.2). Porém, se a abordagem original da LSP for empregada para projetar a hierarquia de agrupamentos, um problema de posicionamento pode ocorrer. Suponha que existam dois nós irmãos bem próximos no layout (possivelmente compartilhando uma fronteira) e que ambos sejam projetados. Uma vez que nesse processo de projeção nenhuma informação sobre os nós irmãos é levada em consideração, é possível que as projeções resultantes posicionem, como vizinhos, filhos não tão similares, isto é, filhos de nós irmãos posicionados perto da fronteira entre eles.

Para evitar esse problema, quando um nó C_n é projetado, primeiro uma triangulação Delaunay envolvendo C_n e seus irmãos é gerada – considerando o seus centros. Usando essa triangulação determina-se quais nós compõem o fecho convexo de C_n, e seus centros são mapeados na

fronteira de C_n. Então, estes pontos são usados no processo da LSP como pontos de controle adicionais, e os centróides dos agrupamentos que esses representam são empregados quando o grafo de vizinhança é definido. Vale ressaltar que esses serão fixos, agindo como âncoras quando os outros pontos de controle, selecionados pelo processo usual da LSP, forem projetados no plano.

Usando essa abordagem, a informação sobre os irmãos que compõem o fecho convexo do nó C_n é levada em consideração quando ele é projetado, reduzindo assim o problema de um layout final apresentar instâncias não relacionadas como vizinhas.

A Figura 6.5 exemplifica esse processo. Na Figura 6.5(a) é apresentado um exemplo de configuração inicial de nós, onde o nó C_n, a ser projetado, é identificado em cinza. Depois, na Figura 6.5(b), a triangulação Delaunay é apresentada considerando somente o centro dos nós (círculos), definindo assim os nós que serão usados para criar os pontos de controle âncoras (os nós que compartilham arestas com C_n). Por fim, na Figura 6.5(c) é mostrado o mapeamento do centro desses nós na fronteira do nó C_n, definindo-se as coordenadas cartesianas dos pontos de controle âncoras.



(a) Configuração inicial dos nós.

(b) Triangulação Delaunay considerando os centros dos nós.

(c) Mapeamento do pontos de controle âncoras.

Figura 6.5: Exemplificação do processo de definição e mapeamento dos pontos de controle âncoras.

O processo de mapeamento do centro de um nó C_i sobre a fronteira do nó C_n é bastante simples. Nele um segmento de reta é traçado entre os centros de C_i e C_n, e a intersecção desse segmento com a borda do círculo C_n define as coordenadas cartesianas do mapeamento. A coordenada x desse mapeamento pode ser definida aplicando-se a Equação (6.2), onde $\overrightarrow{v}$ representa o vetor de C_n até C_i, e $tamanho(C_n)$ indica o tamanho do raio do nó C_n. A coordenada y é calculada de forma similar.

$$x = (x_i - x_n)/||\overrightarrow{v}|| * tamanho(C_n) \tag{6.2}$$

6.4 Re-Arranjando a Hierarquia

Técnicas de agrupamento visam extrair possíveis grupos de instâncias de dados relacionadas de um conjunto de dados. Para realizar tal tarefa, uma heurística é aplicada, e dependendo do que é empregado e o que é entendido como um agrupamento, diferentes resultados podem ser alcançados (Tan et al., 2005). Portanto, é possível que os agrupamentos resultantes sejam diferentes do esperado pelo usuário, possivelmente apresentando agrupamentos com instâncias que não são adequadamente relacionadas, ou colocando instâncias altamente relacionadas em diferentes agrupamentos.

De forma a superar esse problema, uma estratégia de unir agrupamentos baseada na inspeção visual do layout é sugerida aqui. Nessa estratégia, após o usuário ter selecionado S agrupamentos para serem unidos, um novo nó N é criado contendo as instâncias de dados desses nós e ligado ao ancestral comum desses. Em seguida, o nó N é projetado no plano, seu tópico é extraído, e todos os nós que tiveram instâncias removidas são reduzidos e seus tópicos renovados. Desta forma, um usuário pode reconstruir a hierarquia de agrupamentos usando sua própria interpretação, reorganizando o conjunto de dados baseado em seu próprio conhecimento e necessidade. De forma a dividir um agrupamento é somente necessário descer na hierarquia, ou criar um agrupamento selecionando instâncias individuais de dados. Esta estratégia é resumida no Algoritmo 6.3.

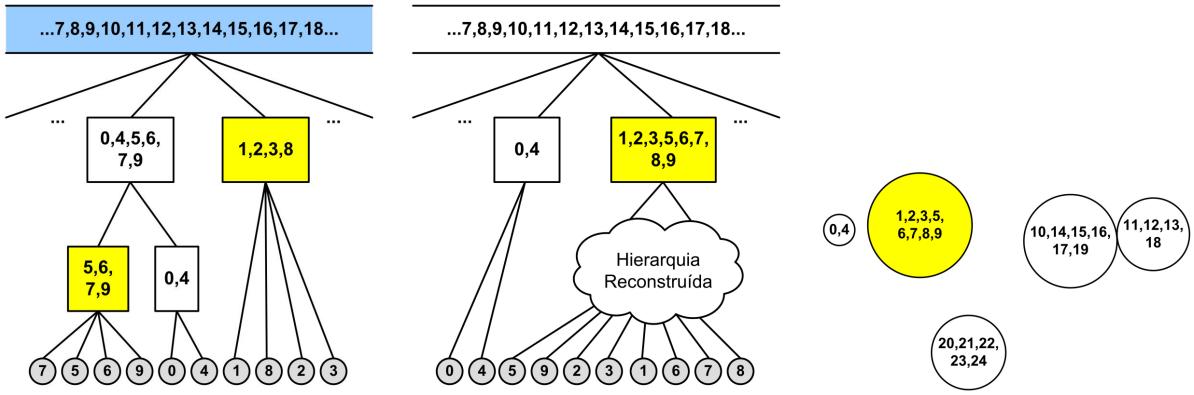
Algoritmo 6.3 Unindo nós na árvore hierárquica de agrupamentos.

entrada: - S: os nós selecionados para se unir.
saída: - uma árvore modificada de agrupamentos.

procedimento $Unir(S)$

1: Criar um novo nó N contendo todas as instâncias de dados dos nós em S.
2: Projetar N de acordo com os nós em S (usando a Equação 6.3).
3: $C \leftarrow$ pegar o ancestral comum dos nós em S.
4: **para todo** $S_i \in S$ **faça**
5: **se** S_i é um filho de C **então**
6: Remover S_i de C.
7: **senão**
8: $A \leftarrow$ pegar o ancestral do nó S_i que é um filho de C.
9: Remover as instâncias de A que são comuns as instâncias de S_i.
10: Diminuir o tamanho de A para ser proporcional ao número de instâncias que pertencem a esse.
11: Recriar o tópico de A.
12: **fim se**
13: **fim para**
14: Definir o tamanho de N proporcional ao número de instâncias que pertencem a esse.
15: Adicionar N como filho de C.
16: $Espalhar(C)$

A Figura 6.6 exemplifica o processo de reestruturação da árvore hierárquica de agrupamentos apresentada na Figura 6.3. Partindo de uma seleção de nós a serem unidos (Figura 6.6(a)), nesse exemplo os nós *(1,2,3,8)* e *(5,6,7,9)* (em amarelo), o nó ancestral comum é encontrado, nesse exemplo o nó raiz (em azul). Uma vez feito isso, um novo nó, *(1,2,3,5,6,7,8,9)* (em amarelo), contendo todas as instâncias de dados dos nós selecionados, é criado e ligado ao nó ancestral comum (Figura 6.6(b)). A partir disso, a hierarquia é reconstruída, diminuindo os nós que perderam instâncias de dados e redefinindo a hierarquia abaixo no novo nó, empregando o mesmo processo de particionamento apresentado anteriormente. Por fim, o novo nó e os afetados por essa reestruturação são projetados no plano, definindo-se o layout final (Figura 6.6(c)).



(a) Seleção dos nós a serem unidos e definição do ancestral comum.

(b) Criação de um novo nó com todas instâncias de dados selecionadas.

(c) Posicionamento final no plano.

Figura 6.6: Ilustração do processo de reconstrução da árvore hierárquica de agrupamentos para uma possível seleção do usuário.

O novo nó é posicionado de forma a não alterar muito o layout que já está construído, somente fazendo modificações locais (na vizinhança dos agrupamentos selecionados). Essa é uma característica importante, caso contrário grande parte da informação que o usuário já extraiu de uma determinada representação visual, baseada no posicionamento dos agrupamentos, pode ser severamente prejudicada. O novo agrupamento N é posicionado no fecho convexo dos agrupamentos selecionados, próximo aos agrupamentos que apresentem o maior número de instâncias de dados. Para calcular a coordenada x do centro de N, a seguinte equação é aplicada (a coordenada y é calculada de forma similar):

$$x = \sum_{i=1}^{k} \left(\frac{|S_i|}{\sum_{n=1}^{k} |S_n|} x_i \right) \tag{6.3}$$

onde $S = \{S_1, S_2, \ldots, S_k\}$ denota os nós selecionados, e x_i é a coordenada x do nó S_i.

6.5 Extraindo Tópicos Multi-Níveis

Na abordagem definida pela HiPP, como não somente documentos individuais são representados por elementos gráficos, mas também grupos de documentos relacionados, é necessário que exista um mecanismo que possa identificar o conteúdo dos documentos pertencentes a esses agrupamentos.

De forma a ajudar no entendimento do conteúdo dos agrupamentos de documentos definidos na representação visual gerada, um tópico é atribuído a cada agrupamento, descrevendo os assuntos mais relevantes tratados pelos documentos pertencentes a esses. O processo de criação desses tópicos é parecido com o apresentado no Capítulo 5, ligeiramente modificado para apoiar a extração de tópicos em diferentes níveis de detalhamento.

Quando um tópico é criado escolhemos as duas palavras iniciais, os termos com maior covariância, diferentes dos dois primeiros termos do tópico de seus nós ancestrais. Desta forma, um termo pode aparecer em tópicos subseqüentes, mas o tópico final nunca será o mesmo. Por conveniência, o tópico do nó $RAIZ$ é vazio.

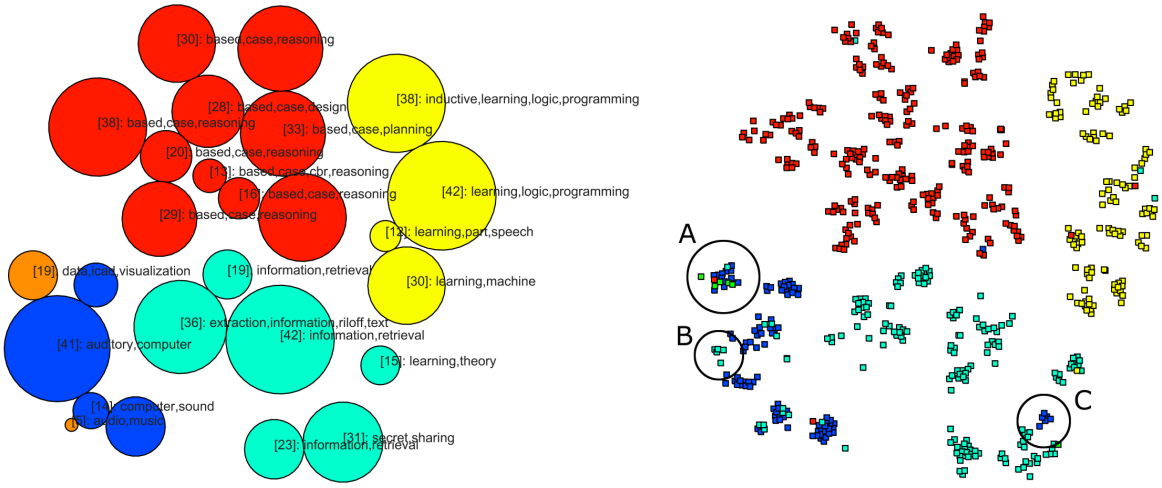
6.6 Resultados e Avaliação da Técnica

Nessa seção apresentamos alguns resultados da aplicação da HiPP para a visualização de coleções de documentos. Com esses exemplos busca-se mostrar sua eficácia no que tange a representação desse tipo de conjunto de dados em diferentes níveis de detalhamento, bem como na precisão dos layouts gerados em relação à preservação das relações de similaridade.

A Figura 6.7 apresenta o mapa de documentos para o conjunto de dados **CBR-ILP-IR-SON**. Nesse mapa, as cores indicam as áreas dos documentos, em vermelho são os documentos de CBR, em amarelo os de ILP, em verde os de IR, e em azul escuro os de SON. Na Figura 6.7(a) é apresentada a visão de mais alto nível do mapa (a projeção do nó $RAIZ$), onde os círculos representam os agrupamentos de documentos. Nela, para colorir um agrupamento foi empregada a área mais comum entre os documentos que esse contém. Se a área mais comum ocorre em menos de 70% dos documentos, o agrupamento é colorido usando uma cor neutra (bege). A lista de palavras colocadas sobre cada nó são os tópicos extraídos deles. É possível notar que para cada uma das quatro áreas de artigos científicos existem nós representando-as, e que os grupos de artigos com tópicos similares são posicionados próximos entre si na representação visual.

De forma a verificar se a precisão global em termos de separação e agrupamento dos documentos baseado em suas similaridades de conteúdo se mantém em pequenas porções do mapa, cinco documentos pertencentes a área SON, que são conhecidos por compartilharem um conteúdo bem relacionado, foram intencionalmente adicionados a coleção. A Figura 6.7(b) apresenta o resultado se todos os agrupamentos e sub-agrupamentos da Figura 6.7(a) forem expandidos mostrando os documentos. Nesta figura, estes cinco documentos são coloridos em verde, circulados e identificados como "**A**", indicando que documentos altamente correlacionados

são colocados próximos no layout final. Os documentos circulados e identificados como "**B**", que a priori parecem mal posicionados, são mais relacionados com "audio analysis", portanto melhor classificados como artigos de SON. Os documentos em "**C**" são os únicos documentos que lidam com o assunto "parallel programming", lidando com análise de performance. Embora a partir da identificação baseada em cor possa ser pensado que eles estão mal posicionados, se seus vizinhos forem mais detalhadamente investigados, pode ser notado que esses são também centrados em análise de performance, mas em uma área diferente, de forma que os documentos em "**C**" são posicionados próximos de documentos que compartilham um assunto em comum. Cabe notar que um vez que os artigos de IR e SON foram coletados a partir de buscas na internet, usando somente termos pertencentes a essas áreas, a classificação original não é equivalente a uma classificação de um especialista, portanto justificando a aparente má classificação visual de acordo com a cor.



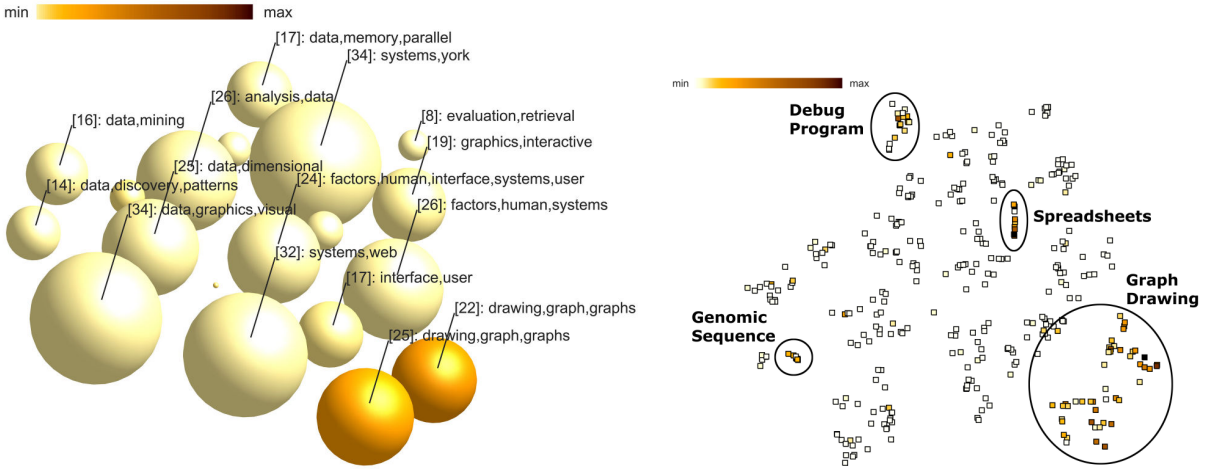
(a) Representação visual dos agrupamentos de mais alto nível.

(b) Representação visual mostrando só instâncias de dados.

Figura 6.7: Exemplo de mapa de documento do conjunto **CBR-ILP-IR-SON**. As quatro áreas de artigos científicos são satisfatóriamente representadas e identificadas, e agrupamentos com documentos similares são posicionados proximamente na representação visual.

De forma a verificar se a técnica consegue também produzir bons resultados em uma coleção de documentos mais homogênea, um mapa de documentos do conjunto de dados **INFOVIS04** (ver Seção 5.3) foi criado. A projeção resultante é apresentada na Figura 6.8. Na Figura 6.8(a) é apresentada a visão de mais alto nível dos agrupamentos gerados. Nela, os agrupamentos estão coloridos de acordo com a porcentagem de seus documentos que apresentam as palavras "graph AND drawing", empregando esferas ao invés de círculos para representar os agrupamentos. A Figura 6.8(b) apresenta o resultado se todos os agrupamentos forem expandidos. Nessa figura os documentos são coloridos de acordo com a freqüência de ocorrência de algumas palavras-chave – "debug program", "spreadsheets","graph drawing", and "genomic sequence" – e as áreas onde esses assuntos ocorrem são circuladas e nomeadas. Por essas figuras é possível notar que os

resultados anteriores em termos de separação e agrupamento baseados em conteúdo também são verificados para coleções mais homogêneas de documentos.



(a) Representação visual dos agrupamentos de mais alto nível.

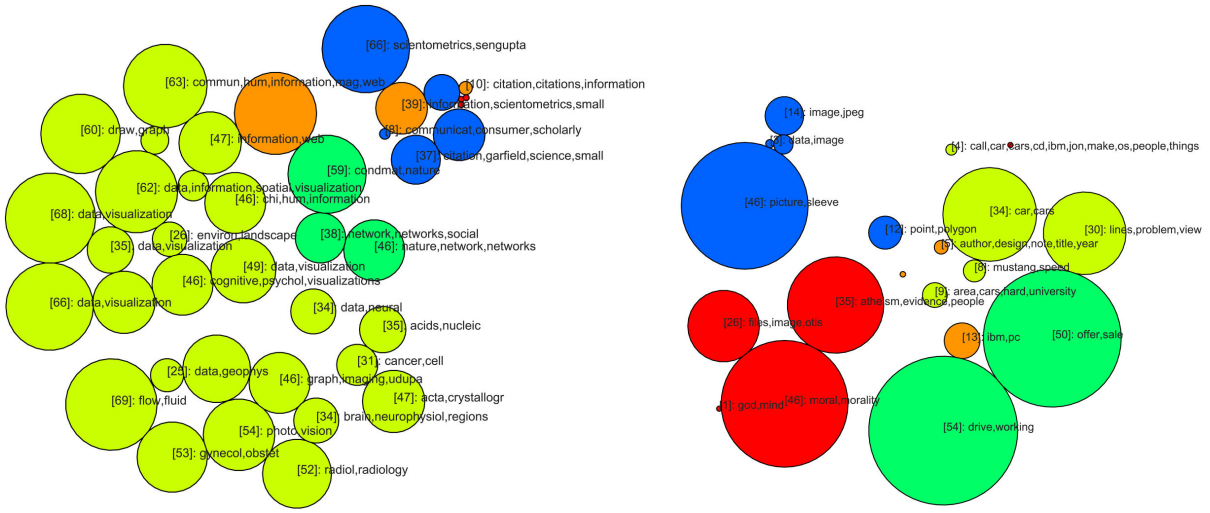
(b) Representação visual mostrando só instâncias de dados.

Figura 6.8: Um mapa de documento de uma coleção mais homogênea de artigos científicos.

A Figura 6.9 apresenta as visões de mais alto nível produzidas para outras duas coleções de documentos. Na Figura 6.9(a) é apresentado o resultado para o conjunto **KDViz**. Nessa pode ser visto que o layout produzido consegue identificar três das quatro diferentes áreas desse conjunto (IV, CA e MG). Isso ocorre porque a quarta área de artigos científicos (BC) é muito correlacionada a outra área (CA), havendo uma mistura entre seus documentos, resultando nos agrupamentos coloridos em bege (a cor neutra quando não existe prevalência de uma determinada área de documentos). Esse é um efeito esperado, que também pode ser observado nos outros layouts apresentados nessa tese produzidos para esse conjunto de dados. A Figura 6.9(b) apresenta o resultado para o conjunto **MENSAGENS**, também de boa qualidade, apesar de ser composto de documentos com uma linguagem mais livre se comparados aos artigos científicos.

A Figura 6.10 apresenta o mapa de documentos para o conjunto de notícias curtas de jornal **NOTÍCIAS**. Na Figura 6.10(a) a visão mais abstrata dessa coleção de documentos é apresentada, e os tópicos dos grandes agrupamentos são mostrados, identificando os assuntos comuns divulgados pelas diferentes agências de notícias: “bird AND flu”, “immigration AND senate”, “game AND microsoft”, e assim por diante. Os agrupamentos são coloridos de acordo com o percentual de seus documentos que apresentam as palavras presentes nos tópicos.

A Figura 6.10(b) apresenta o mapa resultante quando os agrupamentos identificados pelo tópico “bird AND flu” foram unidos em um único agrupamento. A Figura 6.10(c) apresenta esse mapa de documento após todos os agrupamentos serem expandidos, mostrando que os documentos unidos compartilham uma mesma vizinhança. Nesta figura os documentos são coloridos de acordo com a freqüência de ocorrência dos termos “bird AND flu”.



(a) Conjunto de artigos científicos (**KDViz**) com uma área contendo a maior parte dos documentos.

(b) Conjunto de mensagens de discussão (**MENSAGENS**) com documentos com um conteúdo menos formal que os artigos científicos.

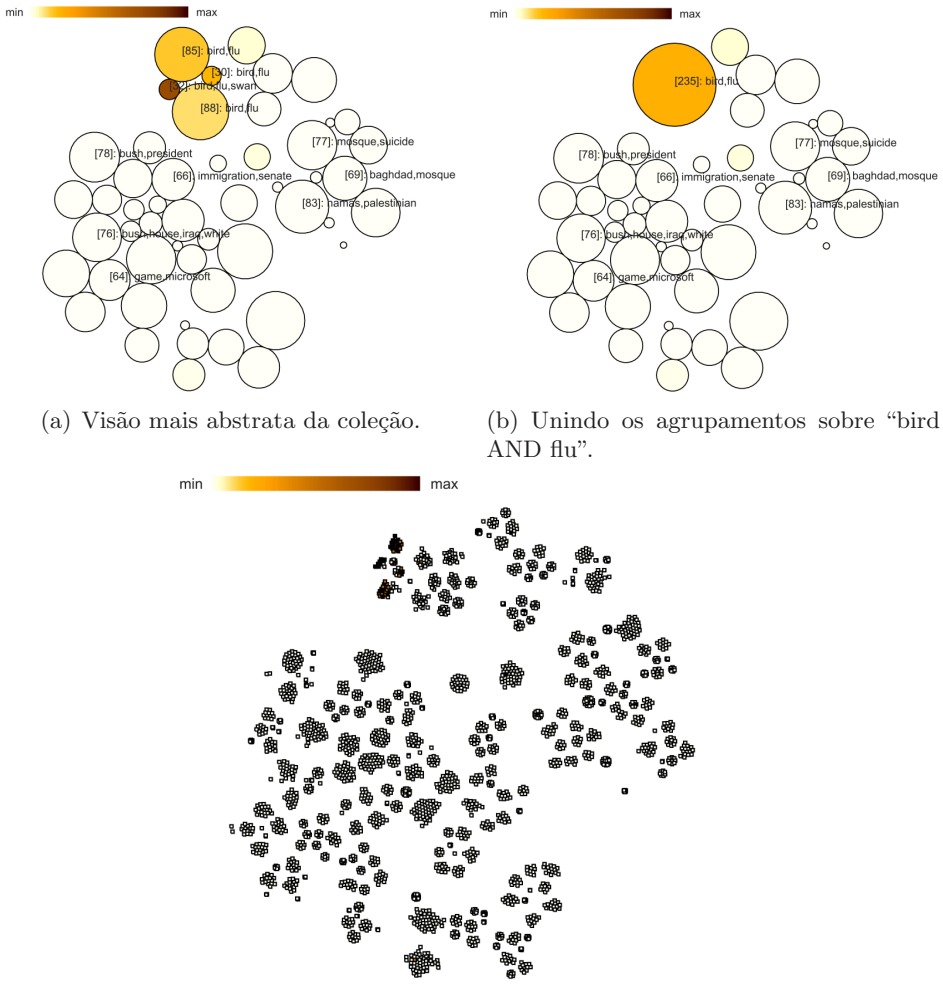
Figura 6.9: Resultados da aplicação da HiPP para conjuntos de características diferentes. Em diferentes situações o resultado alcançado consegue separar e agrupar bem os documentos com base em seus conteúdos.

O número de instâncias que podem ser desenhadas é limitado pela área visual disponível, e se não existir espaço suficiente para desenhar todos elementos, sobreposições irão ocorrer, prejudicando a interpretação do usuário e dificultando a identificação de áreas densas.

Esse problema é ilustrado na Figura 6.1, onde uma projeção LSP para um conjunto com 30.000 documentos é apresentada. Naquela figura, os documentos estão coloridos de acordo com a freqüência de ocorrência da palavra "Clinton". Neste caso, existe uma severa sobreposição, e é difícil determinar os agrupamentos de documentos; quando esses agrupamentos são identificados é difícil avaliar suas densidades. Com a representação visual da HiPP (Figura 6.11), os grupos de documentos são explicitamente representados, e suas densidades são mapeadas para tamanho (e possivelmente cor). Além disso, a possibilidade de unir e dividir agrupamentos pode ser empregada para organizar os dados conforme esses são analisados, apoiando melhor uma análise geral da coleção de documentos.

6.6.1 Avaliação e Discussão

A Figura 6.12 emprega as abordagens *Neighborhood Preservation* e *Neighborhood Hit* para comparar diferentes layouts produzidos pelas técnicas *Force Scheme*, LSP e HiPP. Também apresentamos o resultado se a HiPP fosse aplicada sem empregar âncoras na projeção (veja Seção 6.3.2). Esses layouts foram gerados para o conjunto **CBR-ILP-IR-SON** com adição de cinco novos documentos ao sub-conjunto SON (os citados no exemplo da Figura 6.7(b)). Portanto, os resultados dessa análise são ligeiramente diferentes dos apresentados nos capítulos anteriores uma vez que a matriz "termos x documentos" produzida é diferente.



(a) Visão mais abstrata da coleção.

(b) Unindo os agrupamentos sobre "bird AND flu".

(c) Visão com todos os agrupamentos expandidos.

Figura 6.10: Mapa de documento para o conjunto **NOTÍCIAS**.

Na análise *Neighborhood Preservation*, a HiPP é a técnica mais precisa até cerca de 30 vizinhos. Para mais vizinhos, a *Force Scheme* se torna melhor. A HiPP é melhor do que a *Force Scheme* para menos vizinhos porque os pontos são agrupados antes da projeção, resultando em layouts com relações de similaridade locais mais precisas uma vez que as grandes distâncias são ignoradas. Para mais vizinhos, a *Force Scheme* é melhor porque todos os pontos são considerados juntos no processo de projeção, portanto relações globais são melhor preservadas. Se âncoras não são empregadas na HiPP, os resultados são tão precisos quanto os resultados com âncoras para um pequeno número de vizinhos. Neste caso, uma vez que a vizinhança de um ponto é normalmente definida dentro do agrupamento ao qual essa pertence, ignorar informação sobre os outros agrupamentos não afeta a precisão. Porém, conforme o número de vizinhos cresce, a vizinhança começa a ser definida sobre diferentes agrupamentos. Assim ignorar informação sobre os outros agrupamentos quando um agrupamento é projetado diminui a precisão da preservação de vizinhança no layout final.

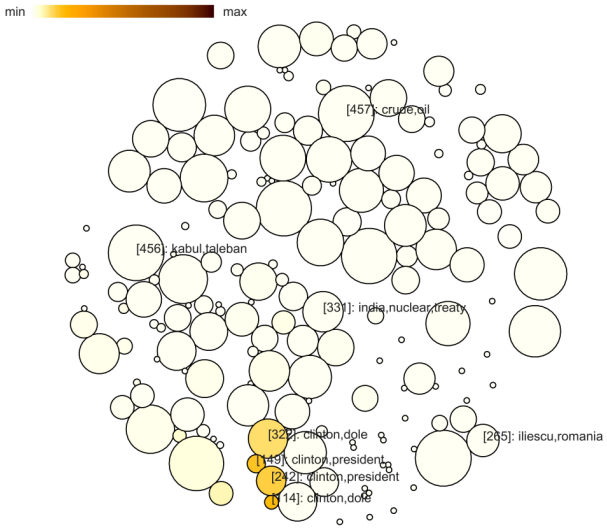
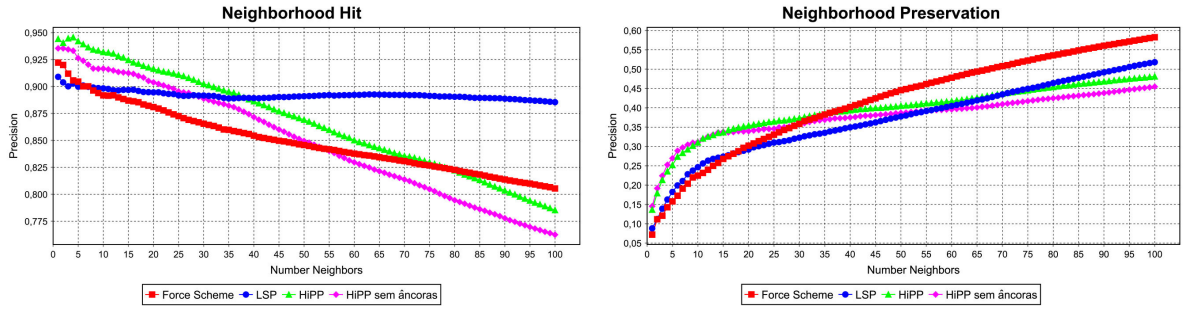


Figura 6.11: Resultado visual produzido pela HiPP para uma coleção de 30.000 documentos.

Um efeito semelhante pode ser notado nos resultados da análise *Neighborhood Hit*. Nessa, novamente a HiPP é melhor para pequenas vizinhanças, até cerca de 35 vizinhos, e a partir desse ponto a LSP se torna melhor. Aqui, novamente o emprego das âncoras se revela como um mecanismo eficiente para aumentar a precisão do layout gerado. Assim como na análise *Neighborhood Preservation*, nessa pode ser observado que como a HiPP agrupa os pontos antes de projetar, para pequenas vizinhanças essa é mais precisa por ignorar as grandes distâncias, porém para vizinhanças maiores, a LSP e a *Force Scheme* se tornam melhores por considerarem informação global no processo.



(a) *Neighborhood Hit.* (b) *Neighborhood Preservation.*

Figura 6.12: Análises comparativas entre as projeções definidas pela HiPP e outras técnicas de projeção. Com o emprego de âncoras, o resultado produzido pela HiPP é superior às técnicas anteriormente analisadas, principalmente para pequenas vizinhanças.

6.7 Complexidade Computacional

Em termos da complexidade computacional, a complexidade da HiPP pode ser calculada como $O(C + P + S)$, onde C é a complexidade para a criação da árvore hierárquica de agrupamentos, P é a complexidade para projetá-la, e S é a complexidade para espalhar essas projeções. Considerando o pior cenário onde a árvore é completamente desbalanceada – cada nível é composto de$\sqrt{n} - 1$ agrupamentos únicos (agrupamentos com um elemento) e um grande agrupamento –, essa terá $\sqrt{n}$ níveis. Nesse caso, o valor de C, que é a complexidade do algoritmo de agrupamento empregado, será $O(n\sqrt{n})$ (Tan et al., 2005). A complexidade para projetar os pontos usando a LSP é $O(n^2)$, mas uma vez que essa é aplicada a no máximo $\sqrt{n}$ instâncias quando um nível é projetado, essa é linear considerando o número de instâncias no conjunto de dados. Uma vez que é necessário projetar $\sqrt{n}$ níveis, a complexidade total será $O(n\sqrt{n})$. Essa mesma análise pode ser aplicada para a complexidade do algoritmo de espalhamento. Portanto, a complexidade total da HiPP é $O(n\sqrt{n} + n\sqrt{n} + n\sqrt{n}) = O(n\sqrt{n})$.

6.8 Considerações Finais

Nesse capítulo apresentamos a HiPP, uma técnica para a visualização, interação e organização de conjuntos de dados multi-dimensionais, com especial atenção para as coleções de documentos. A HiPP define uma estrutura hierárquica que possibilita a análise de conjuntos de dados multi-dimensionais em diferentes níveis de detalhamento. A representação visual adotada é composta não somente de instâncias individuais de dados, mas também de círculos representando agrupamentos de instâncias. Além disso a HiPP oferece uma estratégia de reorganização dos dados com base na representação visual gerada. Nela, conforme a projeção é explorada, agrupamentos podem ser unidos ou separados, reorganizando os dados de acordo com alguma necessidade específica ou conhecimento adquirido.

Os resultados obtidos com a HiPP, considerando a preservação das relações de vizinhança, são tão bons ou melhores do que outras técnicas de projeção existentes, e normalmente muito mais precisos para pequenas vizinhanças, assegurando que as propriedades apresentadas em uma visão geral da representação visual são preservadas quando menores grupos de documentos são explorados. A complexidade computacional da HiPP é similar ou inferior às técnicas estudadas aqui, e tempo computacional é salvo uma vez que agrupamentos são posicionados por demanda (quando solicitado pelo usuário), e processamento é somente gasto para projetar (posicionar) a parte do conjunto de dados que está sendo explorada em um determinado momento.

Desde o princípio o foco da HiPP foi a representação visual de coleções de documentos. Assim, todas as escolhas de projeto sempre foram voltadas a esse fim. Contudo, essa pode ser adaptada a outros domínios onde representações hierárquicas possam ser mecanismos eficientes para análise de dados, ou onde estruturas hierárquicas representem o domínio natural da aplicação. Para tal, é necessário que exista uma forma de se calcular a similaridade entre agrupamentos e entre agrupamentos e instâncias individuais de dados, além de um mecanismo que resuma o conteúdo

de um agrupamento, para que o usuário não tenha que navegar sempre até às instâncias de dados para entender o conteúdo de um agrupamento.

Embora tenhamos empregado o algoritmo *bisecting k-means* para a criação da árvore de agrupamentos por esse apresentar bons resultados para coleções de documentos (Steinbach et al., 2000), outros algoritmos podem ser empregados. A árvore hierárquica de agrupamentos também pode ser construída de diferentes formas, ou pode ser uma hierarquia pré-definida para o conjunto de dados. Porém, hierarquias profundas devem ser evitadas.

Dentre as escolhas de projeto, o emprego de círculos para a representação dos agrupamentos, apesar de não ser tão eficiente quanto o emprego de retângulos na utilização do espaço visual, se justifica por duas principais razões: (1) uma vez que o algoritmo de agrupamento empregado divide o espaço multi-dimensional em agrupamentos de formatos hiper-esféricos (veja Seção 6.2.1), é mais natural mapear estas hiper-esferas para círculos no plano; e (2) o algoritmo que espalha os agrupamentos no plano (Algoritmo 6.2) muda as posições desses em direções arbitrárias a partir de um ponto inicial, não somente nas direções horizontal e vertical. Portanto, a condição de não-sobreposição, que é um ponto crucial na técnica proposta, pode somente ser assegurada se círculos forem empregados.

O próximo Capítulo coloca em perspectiva as contribuições individuais das técnicas aqui desenvolvidas e descreve as demais conclusões desta tese.

Capítulo

7

Conclusões

7.1 Contribuições

Nesta tese foram apresentadas diferentes técnicas desenvolvidas visando dar apoio ao processo de mineração visual de dados, com especial atenção à análise e exploração de coleções de documentos. Elas seguem uma abordagem conhecida com projeção multi-dimensional de dados, que consiste no mapeamento de instâncias de dados multi-dimensionais em um espaço uni-, bi- ou tri-dimensional de forma a ser possível gerar uma representação visual com a qual o usuário pode interagir. Nessa transformação, cada instância de dados é mapeada em um marcador gráfico, normalmente um ponto ou círculo, de forma que a proximidade entre eles reflita as relações de similaridade entre as instâncias de dados. Esta representação é capaz de revelar relações de similaridade entre instâncias individuais de dados e grupos dessas instâncias.

Para a definição de um arcabouço teórico para o desenvolvimento desse trabalho de doutorado, um estudo comparativo entre as técnicas de projeção mais relevantes ao trabalho aqui proposto foi realizado. Como maior contribuição desse estudo podemos destacar a identificação de alguns problemas que precisam ser tratados de forma a tornar as projeções multi-dimensionais ferramentas efetivas para a mineração visual de dados. Entre esses, os mais relevantes são:

Compromisso entre complexidade computacional e precisão: apesar de ser legítima a preocupação com a redução de complexidade computacional de forma a ser possível tratar grandes conjuntos de dados, o desenvolvimento de uma nova técnica de projeção deve buscar um compromisso entre a complexidade computacional e a precisão dos layouts

resultantes, já que há pouco sentido em acelerar a criação de projeções se o resultado final não for satisfatório;

Preservação de informação global e local: na preservação das relações de distância em um layout, dois componentes diferentes podem ser observados, um que leva à aproximação das instâncias muito relacionadas (informação local) e outro que leva a separação dos grupos de instâncias (informação global). Para que uma projeção seja coerente, ambos os componentes devem ser preservados o máximo possível e em equilíbrio, sendo que a preservação de um componente mais que o outro pode levar a distorções no layout final;

Relações de vizinhança e similaridade: normalmente, para dados reais os atributos apresentam relações não-lineares entre si, de forma que é preferível buscar preservar relações de vizinhança localmente entre as instâncias de dados, e de similaridade entre os grupos de instâncias. Um exemplo seria a projeção de coleções de documentos, sendo interessante pautar o desenvolvimento de técnicas específicas para esse domínio no fato de que normalmente os documentos estão relacionados a pequenas vizinhanças, sendo bastante dissimilares à maioria dos documentos dentro de uma coleção;

Escalabilidade visual: por fim, é necessário considerar que tão importante quanto diminuir a complexidade computacional de um técnica deve ser a preocupação em se definir um representação gráfica que leve em consideração a escalabilidade visual, isto é, sua capacidade de conseguir efetivamente sintetizar grandes conjuntos de dados no espaço permitido da tela.

Com base nessas observações, três técnicas diferentes de projeção foram desenvolvidas. A primeira, *Projection by Clustering (ProjClus)* (veja Capítulo 3), é uma técnica de propósito geral com objetivo de reduzir a complexidade de uma das técnicas bem avaliadas no estudo comparativo, denominada *Force Scheme (FS)* (veja Seção 2.3.2.4). Nela, os dados são inicialmente agrupados; posteriormente os grupos são projetados um a um e por fim unidos para a criação do layout final. Apesar da ProjClus ser de fato uma aproximação da FS, os resultados alcançados conseguiram manter um compromisso razoável entre a redução de complexidade e qualidade, resultando em layouts que preservam muito mais as relações de vizinhança e similaridade se comparados aos produzidos por outras técnicas que também são aproximações de modelos mais precisos, como por exemplo a abordagem definida por Chalmers (veja Seção 2.3.2.2) ou os modelos Híbrido e baseado em pivôs (veja Seção 2.3.2.3).

A segunda técnica, a *Least Square Projection (LSP)* (veja Capítulo 4), embora possa ser aplicada a diferentes domínios, foi desenvolvida visando a projeção de coleções de documentos. Nessa, construindo-se um sistema de equações lineares, busca-se mapear cada instância de dados no fecho convexo de seus vizinhos mais próximos quando essas são projetadas. Esse conceito de preservação de relações de vizinhança ao invés de puramente similaridades é o que torna a LSP especialmente indicada para coleções de documentos e outros espaços esparsos, onde normalmente as instâncias de dados são somente relacionadas a pequenas vizinhanças. A idéia

de preservar vizinhanças já foi objetivo de outras técnicas, porém a LSP contribui com a idéia de não somente preservar relações locais de vizinhança, mas também relações globais de similaridade por meio do emprego de pontos de controle – instâncias de dados cuidadosamente escolhidas como representativos de grupos de instâncias –, o que torna os layouts produzidos mais precisos e úteis para a exploração de coleções de documentos. Com a LSP foi possível melhorar o layout por projeção, de forma que grupos ficam visualmente bem definidos, ao contrário das técnicas convencionais onde a mistura entre grupos é mais freqüente.

A terceira e última técnica de projeção desenvolvida, denominada *Hierarchical Point Placement (HiPP)* (veja Capítulo 6), define uma nova abordagem onde não somente instâncias individuais de dados são representadas por elementos gráficos, mas também grupos de instâncias altamente correlacionadas. Na HiPP, uma estrutura hierárquica de agrupamentos é extraída, possibilitando ao usuário navegar em diferentes níveis de detalhamento, partindo de uma visão mais geral dos dados, e concentrado em grupos de interesse, até alcançar as instâncias de dados mais relevantes para uma determinada busca. Essa abordagem hierárquica não somente acelera o processo de criação de uma projeção, como também define uma representação mais escalável visualmente, já que menos elementos gráficos são apresentados simultaneamente ao usuário. Além disso, essa técnica permite que a estrutura hierárquica de agrupamentos seja modificada conforme a representação visual é explorada, não só diminuindo a carga cognitiva imposta ao usuário numa primeira visão dos dados, mas também possibilitando ao mesmo aplicar seu conhecimento para alterar a hierarquia definida previamente por alguma heurística (ou técnica de agrupamento). Assim, a HiPP configura uma abordagem que efetivamente une técnicas de mineração e visualização em um só processo para a mineração visual de dados, principalmente coleções de documentos.

Embora as técnicas de projeção multi-dimensional desenvolvidas consigam revelar mais estruturas em conjuntos de dados, um problema recorrente a esse tipo de abordagem é a dificuldade de identificar os motivos que levaram à formação dos grupos no layout final ou as razões que tornam as instâncias dentro de um determinado grupo similares entre si. Para superar esse problema no caso específico de coleções de documentos, foi apresentada uma abordagem que serve para a extração de tópicos em subconjuntos de documentos selecionados pelo usuário. Como resultado, o emprego de tópicos facilita a exploração de projeções de coleções de documentos, simplificando e acelerando o processo de exploração desse tipo de dado. Assim, essa técnica em combinação com projeções contribui para a redução dos problemas relacionados à sobrecarga de informação no processo de interpretação de dados na forma textual.

Para o teste das técnicas aqui desenvolvidas e as do estudo comparativo, bem como para dar maior suporte ao processo de mineração visual de dados baseado em projeções multi-dimensionais, três diferentes ferramentas foram criadas. A primeira, denominada *Projection Explorer (PEx)* (Paulovich et al., 2007), é uma evolução da *Text Map Explorer (TME)* (Paulovich e Minghim, 2006) e tem por objetivo realizar o layout por projeção e posicionamento de pontos de dados multi-dimensionais em geral. A funcionalidade disponível nessa ferramenta inclui a possibilidade de gerar projeções com diferentes técnicas, coordenar essas projeções de diferentes formas, avaliar

os resultados obtidos usando-se as abordagens de *Neighborhood Hit* e *Neighborhood Preservation*, analisar os dados mapeados executando diversas tarefas exploratórias, processar conjuntos de documentos para criar representações vetoriais, etc. A PEx foi projetada e implementada de forma modular podendo ser estendida para a adição de novas técnicas, medidas de similaridade, ou outros requisitos, configurando um arcabouço genérico para teste e avaliação de técnicas para projeção de dados multi-dimensionais e para mineração visual de coleções de documentos.

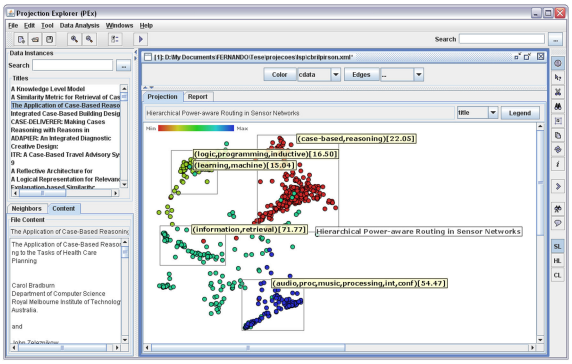
A segunda ferramenta desenvolvida, chamada de *Hierarchical Projection Explorer (H-PEx)* (Paulovich e Minghim, 2008), foi desenvolvida para dar suporte a criação e exploração das projeção multi-dimensionais hierárquicas. Nela, um usuário pode, a partir de uma coleção de documentos, criar uma projeção hierárquica utilizando a *Hierarchical Point Placement (HiPP)* (veja Capítulo 6), aplicando todos os mecanismos dessa técnica para a exploração de coleções de documentos.

A última ferramenta, chamada *Projection Explorer Web (PEx-Web)* (Paulovich et al., 2008b), é uma adaptação da PEx para visualizar resultados textuais de buscas na Internet. Ela apresenta mecanismos específicos que permitem a comparação entre diferentes buscas, a união de diferentes projeções em uma única projeção, etc. Além disso, ela oferece funcionalidade para mineração de dados, como a classificação de dados empregando *Support Vector Machines (SVM)* (Cristianini e Taylor, 2000), acoplando os resultados desse algoritmo às representações visuais resultantes. Com isso, o usuário pode, com ajuda de projeções multi-dimensionais, entender os efeitos dos parâmetros escolhidos no processo de classificação, modificá-los e verificar os resultados dessa modificação, configurando assim uma ferramenta efetiva de apoio à mineração visual de dados.

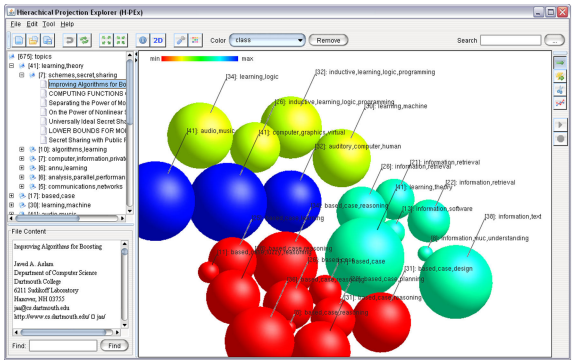
A Figura 7.1 apresenta as telas principais dessas ferramentas.

As ferramentas citadas, bem como outras derivações delas, estão disponíveis em `http://infoserver.lcad.icmc.usp.br/`.

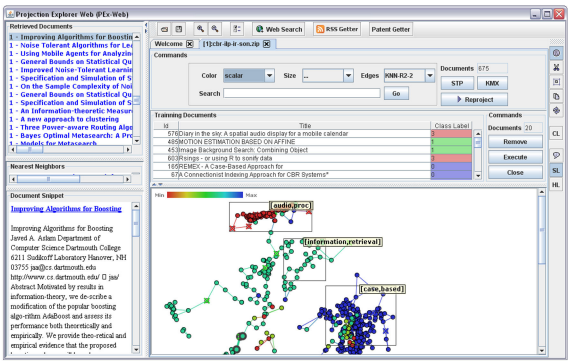
De forma geral, além das técnicas e ferramentas desenvolvidas, este trabalho de doutorado contribui para um melhor entendimento das características desejáveis de uma técnica de projeção multi-dimensional, principalmente as específicas para a projeção de coleções de documentos, de forma a possibilitar suas aplicações ao processo de mineração visual de dados. Nesse caso específico, o que se pode concluir é que embora muitas das técnicas vigentes para a projeção de coleções de documentos busquem preservar as relações de similaridade entre todos os documentos, a preservação de vizinhanças locais leva a resultados mais precisos. Além disso, outra contribuição foi mostrar que, tão importante quanto buscar a redução da complexidade computacional das técnicas de projeção de forma a ser possível tratar grandes conjuntos de dados, deve ser a preocupação com a escalabilidade visual das representações gráficas produzidas. Isso porque, além das limitações físicas de espaço visual para o desenho dessas representações (normalmente o monitor de um computador) existe também o fato de que as habilidades e capacidades humanas não são tão escaláveis (Thomas e Cook, 2005). Do contrário a metáfora, normalmente empregada nas representações visuais de projeções, que mapeia cada instância de dados em um elemento gráfico, poderia prejudicar o processo de exploração visual dos dados já que muita informação é simultaneamente apresentada ao usuário.



(a) Projection Explorer (PEx).



(b) Hierarchial Projection Explorer (H-PEx).



(c) Projection Explorer Web (PEx-Web).

Figura 7.1: Telas principais das ferramentas desenvolvidas dentro desse projeto de doutorado.

Por fim, uma grande contribuição das técnicas aqui apresentadas foi a capacidade de, no layout final, revelar mais grupos que as demais técnicas, separando-os satisfatoriamente, sem que, com isso, se perdesse a identificação de fronteiras flexíveis entre grupos, isto é, regiões intermediárias que podem revelar dados comuns a mais de um grupo, como os vários exemplos ilustraram.

7.2 Limitações

Embora os resultados apresentados nessa tese sejam promissores, tornando as projeções multi-dimensionais ferramentas efetivas para a mineração visual de dados, algumas limitações devem ser observadas.

Um dos problemas das técnicas desenvolvidas aqui, e das técnicas de projeção multi-dimensional em geral, é a sua falta de capacidade em revelar relacionamentos entre os atributos dos dados. Apesar disso ocorrer devido ao tipo de abordagem de visualização empregada, que busca revelar as relações entre as instâncias de dados, em um ambiente real para a mineração visual de dados isso pode representar uma grande limitação. Para o caso específico das coleções de documentos aqui foi sugerida uma técnica capaz de extrair os tópicos principais abordados dentro de seleções de sub-grupos de documentos. Porém, para outros tipos de dados, técnicas específicas de resumo

de atributos em sub-grupos devem ser desenvolvidas, isto é, a estratégia deve ser adaptada a cada aplicação.

Outro problema persistente é aquele da escalabilidade visual. Embora aqui tenha sido sugerido minimizar esse efeito com o uso de uma nova abordagem baseada na hierarquização tanto do processo de criação de uma projeção quanto do de sua exploração, para bases de dados muito grandes esse problema persiste uma vez que possivelmente muito grupos serão apresentados simultaneamente ao usuário. Nesse caso, a única solução seria a definição de metáforas visuais ainda mais compactas do que a aqui sugerida.

7.3 Desenvolvimentos Futuros

Ficam como desenvolvimentos futuros a definição de uma nova metáfora visual para projeções multi-dimensionais que suporte a análise visual de conjuntos de dados cada vez maiores, e uma forma de acoplar às projeções às capacidades de outras técnicas de visualização de informação e de mineração de dados, definindo uma representação que não somente revele os relacionamentos entre as instâncias de dados, mas que também possa ser empregada para analisar as dependências entre seus atributos. Adicionalmente, é preciso tratar algoritmicamente a extração, para cada aplicação, dos motivos que levaram à criação de certos grupos no layout final.

Outro ponto a ser investigado é o emprego de múltiplas projeções coordenadas de forma a superar problemas específicos das técnicas usadas. Por exemplo, uma técnica mais precisa em expressar relações locais de vizinhança poderia ser utilizada em conjunto com outra mais precisa na preservação de similaridades globais. Dessa forma, o usuário poderia investigar, de forma coordenada, a visão geral entre os grupos de instâncias de dados em uma, e os relacionamentos de similaridades locais em outra.

O aspecto de avaliação de projeções e de estratégias multi-níveis em termos de espaço vetorial ainda devem ser explorados dentro do grupo de pesquisa em que este trabalho se insere.

Em várias etapas do processo, é possível adaptar os processos desenvolvidos a aplicações específicas, modificando, por exemplo, a geração do espaço vetorial, a medida de distância, e os parâmetros das técnicas. Essas adaptações podem levar a sistemas para aplicações específicas a partir das técnicas desenvolvidas, a exemplo daqueles já desenvolvidos para extração de corpus, visualização de coleções de imagens, visualização de séries temporais e visualização de dados de sensores eletrônicos.

Referências Bibliográficas

Agarwal, S.; Wills, J.; Cayton, L.; Lanckriet, G.; Kriegman, D.; Belongie, S. Generalized non-metric multidimensional scaling. In: *Proceedings of 11th International Conference on Artificial Intelligence and Statistics (AISTATS 2007)*, San Juan, Puerto Rico, 2007, p. 11–18.

Aggarwal, C. C. Re-designing distance functions and distance-based applications for high dimensional data. *SIGMOD Record*, v. 30, n. 1, p. 13–18, 2001.

Aggarwal, C. C.; Yu, P. S. Redefining clustering for high-dimensional applications. *IEEE Transactions on Knowledge and Data Engineering*, v. 14, n. 2, p. 210–225, 2002.

Aizerman, A.; Braverman, E. M.; Rozoner, L. I. Theoretical foundations of the potential function method in pattern recognition learning. *Automation and Remote Control*, v. 25, p. 821–837, 1964.

Alencar, A. B.; Paulovich, F. V.; Minghim, R.; Andrade Filho, M. G.; Oliveira, M. C. F. Similarity-based visualization of time series collections: An application to analysis of streamflows. In: *Proceedings of the 12th International Conference on Information Visualization (IV'08)*, Los Alamitos, CA, USA: IEEE Computer Society, 2008, p. 280–286.

Andrews, D. F. Plots of high-dimensional data. *Biometrics*, v. 29, p. 125–136, 1972.

Andrews, K.; Kienreich, W.; Sabol, V.; Becker, J.; Droschl, G.; Kappe, F.; Granitzer, M.; Auer, P.; Tochtermann, K. The InfoSky visual explorer: exploiting hierarchical structure and document similarities. *Information Visualization*, v. 1, n. 3/4, p. 166–181, 2002.

Ankerst, M. Visual data mining with pixel-oriented visualization techniques. In: Keim, D.; Eick, S., eds. *SIGKDD Workshop on Visual Data Mining*, San Francisco, USA: ACM, 2001, p. 85–88.

ANKERST, M.; KEIM, D.; KRIEGEL, H.-P. Circle segments: A technique for visually exploring large multidimensional data sets. In: *Proceedings of the IEEE Visualization 1996 (Vis'96)*, 1996.

ANSI/IEEE STD 100–1984 *IEEE Standard Dictionary of Electrical and Electronics Terms (STD 100–1984)*. New York: American National Standards Institute/Institute of Electrical and Electronics Engineering, Inc., 1984.

ASIMOV, D. The grand tour: a tool for viewing multidimentional data. *SIAM Journal of Science & Statistical Computing*, v. 6, p. 128–143, 1985.

BALZER, M.; DEUSSEN, O.; LEWERENTZ, C. Voronoi treemaps for the visualization of software metrics. In: *Proceedings of the 2005 ACM Symposium on Software Visualization (SoftVis'05)*, New York, NY, USA: ACM, 2005, p. 165–172.

BECKS, A.; SEELING, C.; MINKENBERG, R. Benefits of document maps for text access in knowledge management: a comparative study. In: *Proceedings of the 2002 ACM Symposium on Applied Computing (SAC'02)*, New York, NY, USA: ACM, 2002, p. 621–626.

BEDDOW, J. Shape coding of multidimensional data on a microcomputer display. In: *Proceedings of the IEEE Visualization 1990 (Vis'90)*, 1990, p. 238–246.

BELLMAN *Adaptive control processes*. Princeton University Press, 1961.

BENNET, R. The intrinsic dimensionality of signal collections. *IEEE Transactions on Information Theory*, v. 15, n. 5, p. 517–525, 1969.

BERKHIN, P. *Survey of clustering data mining techniques*. Relatório Técnico, Accrue Software, San Jose, CA, `http://citeseer.ist.psu.edu/berkhin02survey.html` (acessado em 14/09/2008), 2002.

BEYER, K. S.; GOLDSTEIN, J.; RAMAKRISHNAN, R.; SHAFT, U. When is "nearest neighbor" meaningful? In: *Proceeding of the 7th International Conference on Database Theory (ICDT'99)*, London, UK: Springer-Verlag, 1999, p. 217–235.

BISWAS, G.; JAIN, A. K.; DUBES, R. C. Evaluation of projection algorithms. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1981, p. 701–708.

BLOXOM, B. Constrained multidimensional scaling in N spaces. *Psychometrika*, v. 43, n. 3, p. 397–408, 1978.

BORG, I.; GROENEN, P. J. F. *Modern multidimensional scaling: Theory and applications*. second ed. Springer, 2005.

BRODBECK, D.; GIRARDIN, L. Combining topological clustering and multidimensional scaling for visualising large data sets. unpublished paper (accepted for, but not published). In: *Proceedings of the IEEE Symposium on Information Visualization (InfoVis'98)*, 1998.

Buja, A.; Swayne, D. F.; Littman, M.; Dean, N. XGvis: Interactive data visualization with multidimensional scaling. *Journal of Computational and Graphical Statistics*, 1998.

Burkhard, W. A.; Keller, R. M. Some approaches to best-match file searching. *Communications of ACM*, v. 16, n. 4, p. 230–236, 1973.

Card, S. K.; Mackinlay, J. D.; Shneiderman, B., eds. *Readings in information visualization: using vision to think*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1999.

Carreira-Perpiñán, M. Á. *A review of dimension reduction techniques*. Relatório Técnico CS–96–09, Department of Computer Science, University of Sheffield, `http://www.dcs.shef.ac.uk/~miguel/papers/cs-96-09.html` (acessado em 14/09/2008), 1996.

Carreira-Perpiñán, M. Á. *Continuous latent variable models for dimensionality reduction and sequential data reconstruction*. Tese de Doutoramento, Department of Computer Science, University of Sheffield, Inglaterra, 2001.

Chalmers, M. A linear iteration time layout algorithm for visualising high-dimensional data. In: *Proceedings of the IEEE Visualization 1996 (VIS'96)*, Los Alamitos, CA, USA: IEEE Computer Society Press, 1996, p. 127–ff.

Chang, C. L.; Lee, R. C. T. A heuristic relaxation method for nonlinear mapping in cluster analysis. *IEEE Transactions on Systems, Man, and Cybernetics*, v. 3, p. 197–200, 1973.

Chávez, E.; Navarro, G.; Baeza-Yates, R.; Marroquín, J. L. Searching in metric spaces. *ACM Computing Surveys*, v. 33, n. 3, p. 273–321, 2001.

Chen, C. *Information visualization: Beyond the horizon*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

Chernoff, H. The use of faces to represent points in k-dimensional space grafically. *Journal of American Statistical Association*, v. 68, p. 361–368, 1973.

Christian, G. R.; Blumenthal, C.; Patterson, M. The information explosion and the adult learner: Implication for reference librarians. *The Reference Librarian*, v. 33, n. 60–70, p. 19–30, `http://www.library.ubc.ca` (acessado em 14/09/2008), 2000.

Cilibrasi, R.; Vitányi, P. Clustering by compression. *IEEE Transactions on Information Theory*, v. 51, n. 4, p. 1546–1555, 2005.

Cleveland, W. S. *Visualizing data*. Hobart Press, 1993.

Cook, D.; Buja, A.; Cabrera., J. Projection pursuit indexes based on orthonormal function expansions. *Journal of Computational and Graphical Statistics*, v. 2, n. 3, p. 225–250, 1993.

Cox, T. F.; Cox, M. A. A. *Multidimensional scaling.* Second ed. Chapman & Hall/CRC, 2000.

Crawford, S. L.; Fall, T. C. Projection pursuit techniques for visualizing high-dimensional data sets. In: Nielson, G. M.; Shriver, B., eds. *Visualization in Scientific Computing*, IEEE Computer Society Press, 1990, p. 94–108.

Cristianini, N.; Taylor, J. S. *An introduction to support vector machines and other kernel-based learning methods.* Cambridge University Press, 2000.

Cuadros, A. M.; Paulovich, F. V.; Minghim, R.; Telles, G. P. Point placement by phylogenetic trees and its application for visual analysis of document collections. In: *IEEE Symposium on Visual Analytics Science and Technology (VAST 2007)*, 2007, p. 99–106.

Davis, T. A. *Direct methods for sparse linear systems (fundamentals of algorithms 2).* Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2006.

Deerwester, S.; Dumais, S. T.; Landauer, T. K.; Furnas, G. W.; Harshman, R. A. Indexing by latent semantic analysis. *Journal of the Society for Information Science*, v. 41, n. 6, p. 391–407, 1990.

Demartines, P.; Herault, J. Curvilinear component analysis: a self-organizing neural network for nonlinear mapping of data sets. In: *IEEE Transactions in Neural Networks*, 1997, p. 148–154.

Demmel, J. W. *Applied numerical linear algebra.* SIAM Press, 1997.

Dijkstra, E. W. A note on two problems in connexion with graphs. *Numerische Mathematik*, v. 1, n. 1, p. 269–271, 1959.

Duda, R. O.; Hart, P. E. *Pattern classification and scene analysis.* New York: Wiley, 1973.

Eades, P. A. A heuristic for graph drawing. *Congressus Numerantium*, v. 42, p. 149–160, 1984.

Edelsbrunner, H. *Geometry and topology for mesh generation.* Cambridge Monographs on Applied and Computational Mathematics. Cambridge, 2001.

Eick, S. G.; Karr, A. F. Visual scalability. *Journal of Computational & Graphical Statistics*, v. 11, n. 1, p. 22–43, 2002.

Eler, D.; Paulovich, F.; Oliveira, M.; Minghim, R. Coordinated and multiple views for visualizing text collections. In: *Proceedings of the 12th International Conference on Information Visualization (IV'08)*, 2008a, p. 246–251.

Eler, D. M.; Nakazaki, M.; Paulovich, F. V.; Santos, D. P.; Oliveira, M. C. F.; Neto, J. E. S. B.; Minghim, R. Multidimensional visualization to support analysis of image collections. In: *Proceedings of the XXI Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI 2008) (to appear)*, Washington, DC, USA: IEEE Computer Society, 2008b.

Faloutsos, C.; Lin, K.-I. FastMap: a fast algorithm for indexing, data-mining and visualization of traditional and multimedia datasets. In: *Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data (SIGMOD'95)*, New York, NY, USA: ACM, 1995, p. 163–174.

Fayyad, U. M. Mining databases: Towards algorithms for knowledge discovery. *Data Engineering Bulletin*, v. 21, n. 1, p. 39–48, 1998.

Fayyad, U. M.; Piatetsky-Shapiro, G.; Smyth, P. *From data mining to knowledge discovery: an overview* Menlo Park, CA, USA: American Association for Artificial Intelligence, p. 1–34, 1996.

Fekete, J.-D.; Grinstein, G.; Plaisant, C. IEEE InfoVis 2004 Contest, the history of InfoVis. `http://www.cs.umd.edu/hcil/iv04contest` (acessado em 14/09/2008), 2004.

Fisher, R. A. The use of multiple measurements in taxonomic problems. *Annals Eugenics*, v. 7, p. 179–188, 1936.

Floater, M. S. Parametrization and smooth approximation of surface triangulations. *Computer Aided Geometric Design*, v. 14, n. 3, p. 231–250, 1997.

Fodor, I. K. *A survey of dimension reduction techniques.* Relatório Técnico UCRL-ID-148494, Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, 2002.

Frick, A.; Ludwig, A.; Mehldau, H. A fast adaptive layout algorithm for undirected graphs. In: *Proceedings of the DIMACS International Workshop on Graph Drawing (GD'94)*, London, UK: Springer-Verlag, 1995, p. 388–403.

Friedman, J. H. Exploratory projection pursuit. *Journal of the American Statistical Association*, v. 82, n. 397, p. 249–266, 1987.

Friedman, J. H.; Tukey, J. W. A projection pursuit algorithm for exploratory data analysis. *IEEE Transactions on Computers*, v. 23, n. 9, p. 881–890, 1974.

Fruchterman, T. M. J.; Reingold, E. M. Graph drawing by force-directed placement. *Software - Practice and Experience*, v. 21, n. 11, p. 1129–1164, 1991.

Fua, Y.-H.; Ward, M. O.; Rundensteiner, E. A. Hierarchical parallel coordinates for exploration of large datasets. In: *Proceedings of the IEEE Visualization 1999 (VIS'99)*, Los Alamitos, CA, USA: IEEE Computer Society Press, 1999, p. 43–50.

FUKUNAGA, K. *Introuction to statistical pattern recognition*. second ed. Academic Press, 1990.

GENNARI, J. H.; LANGLEY, P.; FISHER, D. Models of incremental concept formation. *Artificial Intelligence*, v. 40, n. 1-3, p. 11–61, 1989.

GOLUB, G.; REINSCH, C. *Handbook for matrix computation II, linear algebra*. New York: Springer-Verlag, 1971.

GRINSTEIN, G.; TRUTSCHL, M.; CVEK, U. High-dimensional visualizations. In: *Proceedings of the 7th Data Mining Conference KDD Workshop*, San Francisco, CA, 2001, p. 7–19.

HAREL, D.; KOREN, Y. Drawing graphs with non-uniform vertices. In: *Proceeding of Working Conference on Advanced Visual Interfaces (AVI'02)*, ACM Press, 2002, p. 157–166.

HEARST, M. A.; PEDERSEN, J. O. Reexamining the cluster hypothesis: scatter/gather on retrieval results. In: *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'96)*, New York, NY, USA: ACM Press, 1996, p. 76–84.

HETTICH, S.; BAY, S. D. The UCI KDD archive. `http://kdd.ics.uci.edu` (acessado em 14/09/2008), 1999.

HINNEBURG, A.; AGGARWAL, C. C.; KEIM, D. A. What is the nearest neighbor in high dimensional spaces? In: *Proceedings of the 26th International Conference on Very Large Data Bases (VLDB '00)*, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2000, p. 506–515.

HOFFMAN, P. E. *Table visualizations: A formal model and its applications*. Tese de Doutoramento, Computer Science Department, University of Massachusetts Lowell, 1999.

HRUSCHKA, E. R.; CAMPELLO, R. J.; CASTRO, L. N. Evolving clusters in gene-expression data. *Information Sciences, Elsevier*, v. 176, n. 13, p. 1898–1927, 2006.

HUBER, P. J. Projection pursuit. In: *Annals of Statistics*, 1985, p. 435–475.

INSELBERG, A. The plane with parallel coordinates. *The Visual Computer - Special Issue on Computational Geometry*, v. 1, p. 69–91, 1985.

INSELBERG, A. Multidimensional detective. In: *Proceedings of IEEE Symposium on Information Visualization 1997 (InfoVis'97)*, 1997, p. 100–107.

INSELBERG, A.; DIMSDALE, B. Parallel coordinates: a tool for visualizing multidimensional geometry. In: *Proceedings of the IEEE Visualization 1990 (Vis'90)*, 1990, p. 361–375.

JOHNSON, B.; SHNEIDERMAN, B. Tree-maps: a space-filling approach to the visualization of hierarchical information structures. In: *Proceedings of the IEEE Visualization 1991 (Vis'90)*, Los Alamitos, CA, USA: IEEE Computer Society Press, 1991, p. 284–291.

JOHNSON, S. C. Hierarchical clustering schemes. *Psychometrika*, v. 32, n. 3, p. 241–254, 1967.

JOLLIFFE, I. T. *Principal component analysis*. Springer-Verlag, 1986.

JONES, M. C.; SIBSON, R. What is projection pursuit? *Journal of the Royal Statistical Society*, v. A, n. 150, p. 1–36, 1987.

KAUFMAN, L.; ROUSSEEUW, P. *Finding groups in data: an introduction to cluster analysis*. Wiley Series in Probability and Mathematical Statistics, 1990.

KEIM, D. A. Designing pixel-oriented visualization techniques: Theory and applications. *IEEE Transactionson Visualization and Computer Graphics*, v. 6, n. 1, p. 59–78, 2000.

KEIM, D. A.; KREIGEL, H. P. Visdb: Database exploration using multidimensional visualization. *IEEE Computer Graphics and Applications*, v. 14, n. 5, p. 40–49, 1994.

KEIM, D. A.; KRIEGEL, H. P. Visualizations techniques for mining large databases: a comparison. *IEEE Transactions on Knowledge and Data Engineering*, v. 8, n. 6, p. 923–936, 1996.

KENDALL, M.; GIBBONS, K. D. *Rank correlation methods*. Oxford University Press, 1990.

KIRBY, M. *Geometric data analysis: An empirical approach to dimension reduction and the study of patterns*. John Wiley & Sons, 2001.

KIRKPATRICK, S.; GELATT, C. D.; VECCHI, M. P. Optimization by simulatted annealing. *Science*, v. 220, n. 4598, p. 671–680, 1983.

KLEIN, R. W.; DUBES, R. C. Experiments in projection and clustering by simulated annealing. In: *Pattern Recognition*, 1989, p. 213–220.

KOHONEN, T. The self-organizing map. *Proceedings of the IEEE*, v. 78, n. 9, p. 1464–1480, 1990.

KRAMER, M. A. Non-linear principal component analysis using autoassociative neural networks. *Journal of American Institute of Chemical Engineers (AIChE)*, v. 37, p. 233–243, 1991.

KRUSKAL, J. B. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, v. 1, n. 29, p. 115–129, 1964.

KUMMAMURU, K.; LOTLIKAR, R.; ROY, S.; SINGAL, K.; KRISHNAPURAM, R. A hierarchical monothetic document clustering algorithm for summarization and browsing search results. In: *Proceedings of the 13th International Conference on World Wide Web (WWW'04)*, New York, NY, USA: ACM Press, 2004, p. 658–665.

LEBART, L.; MORINEAU, A.; WARWICK, J. F. Multivariate descriptive statistical analysis (correspondence analysis and related techniques for large matrices). *John Wiley & Sons*, v. 5, n. 2, 1989.

LEBLANC, J.; WARD, M. O.; WITTELS, N. Exploring n-dimensional databases. In: *Proceedings of the IEEE Visualization 1990 (Vis'90)*, 1990, p. 230–237.

LEE, R. C. T.; SLAGLE, J. R.; BLUM, H. A triangulation method for the sequential mapping of points from n-space to two-space. *IEEE Transactions on Computers*, v. 26, n. 3, p. 288–292, 1977.

LEUSKI, A. Evaluating document clustering for interactive information retrieval. In: *Proceedings of the 10th ACM Conference on Information and Knowledge Management (CIKM'01)*, 2001, p. 33–40.

LEWIS, D.; YANG, Y.; ROSE, T.; LI, F. RCV1: A New Benchmark Collection for Text Categorization Research. *Journal of Machine Learning Research*, v. 5, p. 361–397, 2004.

LOPES, A. A.; PINHO, R.; PAULOVICH, F. V.; MINGHIM, R. Visual text mining using association rules. *Computer & Graphics Journal, Special Issue on Visual Analytics*, v. 31, n. 3, p. 316–326, 2007.

LORENZ, E. *Empirical orthogonal eigenfunctions and statistical weather prediction.* Relatório Técnico, M.I.T., Cambridge, MA, 1956.

LUHN, H. P. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, v. 2, n. 2, p. 159–165, 1968.

MARDIA, K. V.; KENT, J. T.; BIBBY, J. M. *Multivariate analysis (probability and mathematical statistics).* Academic Press, 1995.

MARTÍN-MERINO, M.; MUÑOZ, A. Self organizing map and sammon mapping for asymmetric proximities. In: *Proceedings of the International Conference on Artificial Neural Networks (ICANN'01)*, London, UK: Springer-Verlag, 2001, p. 429–435.

MARTÍN-MERINO, M.; MUÑOZ, A. A new sammon algorithm for sparse data visualization. In: *Proceedings of the 17th International Conference on Pattern Recognition (ICPR'04)*, Washington, DC, USA: IEEE Computer Society, 2004, p. 477–481.

MERCER, J. Functions of positive and negative type, and their connection with the theory of integral equations. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, v. 83, n. 559, p. 69–70, 1909.

MINGHIM, R.; PAULOVICH, F. V.; ANDRADE LOPES, A. Content-based text mapping using multi-dimensional projections for exploration of document collections. In: ERBACHER, R. F.; ROBERTS, J. C.; GRÖHN, M. T.; BÖRNER, K., eds. *Proceedings of IS&T/SPIE Conference - Visualization and Data Analysis 2006*, SPIE, 2006, p. 60600S.

MOOTER. The power of relevance. `http://www.mooter.com/` (acessado em 14/09/2008), 2008.

MORRISON, A.; CHALMERS, M. A pivot-based routine for improved parent-finding in hybrid MDS. *Information Visualization*, v. 3, n. 2, p. 109–122, 2004.

MORRISON, A.; ROSS, G.; CHALMERS, M. *Combining and comparing clustering and layout algorithms*. Relatório Técnico 148, Department of Computer Science, University of Glasgow, 2002a.

MORRISON, A.; ROSS, G.; CHALMERS, M. A hybrid layout algorithm for sub-quadratic multidimensional scaling. In: *Proceedings of the IEEE Symposium on Information Visualization (InfoVis'02)*, Washington, DC, USA: IEEE Computer Society, 2002b, p. 152.

MORRISON, A.; ROSS, G.; CHALMERS, M. Fast multidimensional scaling through sampling, springs and interpolation. *Information Visualization*, v. 2, n. 1, p. 68–77, 2003.

MUNIZ, M.; PAULOVICH, F. V.; MINGHIM, R.; INFANTE, K.; ALUÍSIO, S. Taming the tiger topic: a XCES compliant corpus portal to generate subcorpus based on automatic text topic identification. In: *Corpus Linguistics 2007 Online Publication*, `http://www.corpus.bham.ac.uk/conference2007/` (acessado em 14/09/2008), 2007.

NIEMANN, H.; WEISS, J. A fast-converging algorithm for nonlinear mapping of high-dimensional data to a plane. *IEEE Transactions on Computers*, v. 28, n. 2, p. 142–147, 1979.

OHTA, M.; NARITA, H.; OHNO, S. Overlapping clustering method using local and global importance of feature terms at NTCIR-4 web task. In: *Working Notes of NTCIR (NII-NACSIS Test Collection for IR Systems)-4*, 2004, p. 37–44.

OLIVEIRA, M.; LEVKOWITZ, H. From visual data exploration to visual data mining: a survey. *IEEE Transactions on Visualization and Computer Graphics*, v. 9, n. 3, p. 378–394, 2003.

PAL, N. R.; BEZDEK, J. C. On cluster validity for the fuzzy c-means model. *IEEE Transactions on Fuzzy Systems*, v. 3, n. 3, p. 370–379, 1995.

PAULOVICH, F. V.; MINGHIM, R. Text map explorer: a tool to create and explore document maps. In: *Proceedings of the 10th International Conference on Information Visualization (IV'06)*, Washington, DC, USA: IEEE Computer Society, 2006, p. 245–251.

PAULOVICH, F. V.; MINGHIM, R. HiPP: A novel hierarchical point placement strategy and its application to the exploration of document collections. *IEEE Transactions on Visualization and Computer Graphics (Proceedings of Information Visualization 2008)*, v. 14, n. 6, p. 1229–1236, 2008.

PAULOVICH, F. V.; NONATO, L. G.; MINGHIM, R.; LEVKOWITZ, H. Least square projection: a fast high precision multidimensional projection technique and its application to document mapping. *IEEE Transactions on Visualization and Computer Graphics*, v. 14, n. 3, p. 564–575, 2008a.

PAULOVICH, F. V.; OLIVEIRA, M. C. F.; MINGHIM, R. The projection explorer: A flexible tool for projection-based multidimensional visualization. In: *Proceedings of the XX Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI 2007)*, Washington, DC, USA: IEEE Computer Society, 2007, p. 27–36.

PAULOVICH, F. V.; PINHO, R.; BOTHA, C. P.; HEIJS, A.; MINGHIM, R. Pex-web: Content-based visualization of web search results. In: *Proceedings of the 12th International Conference on Information Visualization (IV'08)*, Los Alamitos, CA, USA: IEEE Computer Society, 2008b, p. 208–214.

PEKALSKA, E.; RIDDER R. P. W. DUIN, D.; KRAAIJVELD, M. A. A new method of generalizing Sammon mapping with application to algorithm speed-up. In: BOASSON, M.; KAANDORP, J. A.; TONINO, J. F. M.; VOSSELMAN, M. G., eds. *Proceedings of the 5th Annual Conference of the Advanced School for Computing and Imaging (ASCI'99)*, Delft, Netherlands, 1999, p. 221–228.

PICKETT, R. M.; GRISTEIN, G. G. Iconographic displays fos visualizing multidimensional data. In: *Proceedings of IEEE Conference on Systems, Man, and Cybernetics*, 1988, p. 514–519.

PNNL IN-SPIRE^{TM}. Pacific Northwest National Laboratory (PNNL). `http://in-spire.pnl.gov/` (acessado em 14/09/2008), 2008.

PORTER, M. F. An algorithm for suffix stripping. *Readings in information retrieval*, p. 313–316, 1997.

PRESS, W. H.; FLANNERY, B. P.; TEUKOLSKY, S. A.; VETTERLING, W. T. *Numerical recipes: The art of scientific computing*. 2nd ed. Cambridge (UK) and New York: Cambridge University Press, 1992.

RAO, R.; CARD, S. K. The table lens: Merging graphical and symbolic representation in an interactive focus+context visualization of tabular information. In: *Proceedings of Human Factors in Computing Systems (CHI'94)*, 1994, p. 318–322.

Ridder, D.; Duin, R. P. W. Sammon's mapping using neural networks: a comparison. *Pattern Recognition Letters*, v. 18, n. 11-13, p. 1307–1316, 1997.

Rijsbergen, C. J. *Information retrieval*. second edition ed. Butterworths, 1979.

Robertson, G. G.; Mackinlay, J. D.; Card, S. K. Cone trees: animated 3D visualizations of hierarchical information. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'91)*, New York, NY, USA: ACM, 1991, p. 189–194.

Rose, S.; Wong, P. C. Driftweed - a visual metaphor for interactive analysis of multivariate data. In: *Proceedings of IS&T/SPIE Conference - Visual Data Exploration and Analysis*, 2000.

Roweis, S. T.; Saul, L. K. Nonlinear dimensionality reduction by locally linear embedding. *Science*, v. 290, n. 5500, p. 2323–2326, 2000.

Roweis, S. T.; Saul, L. K. *An introduction to locally linear embedding*. Relatório Técnico, AT&T Labs – Research, 2001.

Saad, Y. *Iterative methods for sparse linear systems*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2003.

Salton, G.; Buckley, C. Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, v. 24, n. 5, p. 513–523, 1988.

Sammon, J. W. A nonlinear mapping for data structure analysis. *IEEE Transactions on Computers*, v. 18, n. 5, p. 401–409, 1969.

Schölkopf, B.; Smola, A.; Müller, K.-R. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, v. 10, n. 5, p. 1299–1319, 1998.

Schölkopf, B.; Smola, A. J.; Müller, K.-R. Kernel principal component analysis. In: *Proceedings of the 7th International Conference on Artificial Neural Networks (ICANN '97)*, London, UK: Springer-Verlag, 1997, p. 583–588.

Schölkopf, B.; Smola, A. J.; Müller, K.-R. Kernel principal component analysis. *Advances in Kernel Methods: Support Vector Learning*, p. 327–352, 1999.

Seo, J.; Shneiderman, B. Interactively exploring hierarchical clustering results. *IEEE Computer*, v. 35, n. 7, p. 80–86, 2002.

Shepard, R. N. Analysis of proximities: Multidimensional scaling with an unknown distance function I. *Psychometrika*, v. 27, n. 2, p. 125–140, 1962a.

Shepard, R. N. Analysis of proximities: Multidimensional scaling with an unknown distance function II. *Psychometrika*, v. 27, n. 3, p. 219–246, 1962b.

SHEWCHUK, J. R. *An introduction to the conjugate gradient method without the agonizing pain.* Relatório Técnico, University of California at Berkeley, `http://www.cs.cmu.edu/~quake-papers/painless-conjugate-gradient.pdf` (acessado em 14/09/2008), 1994.

SHNEIDERMAN, B. The eyes have it: A task by data type taxonomy for information visualizations. In: *IEEE Symposium on Visual Languages*, Los Alamitos, CA, USA: IEEE Computer Society, 1996, p. 336–343.

SIEDLECKI, W.; SIEDLECKA, K.; SKLANSKY, J. An overview of mapping techniques for exploratory pattern analysis. *Pattern Recognition*, v. 21, n. 5, p. 411–429, 1988.

SKUPIN, A. A cartographic approach to visualizing conference abstracts. *IEEE Computer Graphics and Applications*, v. 22, n. 1, p. 50–58, 2002.

SNEATH, P. H. A.; SOKAL, R. R. *Numerical taxonomy: the principles and practice of numerical classification.* San Francisco, USA: W.H. Freeman and Company, 1973.

SORKINE, O.; COHEN-OR, D. Least-squares meshes. In: *Proceedings of the Shape Modeling International 2004 (SMI'04)*, Washington, DC, USA: IEEE Computer Society, 2004, p. 191–199.

SORKINE, O.; COHEN-OR, D.; LIPMAN, Y.; ALEXA, M.; RÖSSL, C.; SEIDEL, H.-P. Laplacian surface editing. In: *Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing (SGP'04)*, New York, NY, USA: ACM, 2004, p. 175–184.

SPAETH, H. J. The use and utility of the monotone criterion in multidimensional scaling. *Multivariate Behavioral Research*, v. 4, n. 4, p. 501–515, 1969.

STEINBACH, M.; KARYPIS, G.; KUMAR, V. A comparison of document clustering techniques. In: *Workshop on Text Mining, 6th ACM SIGKDD International Conference on Data Mining (KDD'00)*, Boston, Massachusetts, USA: ACM, 2000, p. 109–110.

SUN, J. Some practical aspects of exploratory projection pursuit. *SIAM Journal of Science Computer*, v. 14, p. 68–80, 1993.

SWEENEY, L. *Confidentiality, disclosure, and data access: Theory and practical applications for statistical agencies*, cáp. Information Explosion Washington, DC: Urban Institute, 2001.

TAN, P.-N.; STEINBACH, M.; KUMAR, V. *Introduction to data mining, (first edition).* Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2005.

TEJADA, E.; MINGHIM, R.; NONATO, L. G. On improved projection techniques to support visual exploration of multidimensional data sets. *Information Visualization*, v. 2, n. 4, p. 218–231, 2003.

Telles, G. P.; Minghim, R.; Paulovich, F. V. *Visual mapping of text collections using an approximation of kolmogorov complexity.* Technical report, Instituto de Ciências Matemáticas e de Computação, University of São Paulo, 2005.

Telles, G. P.; Minghim, R.; Paulovich, F. V. Normalized compression distance for visual analysis of document collections. *Computer & Graphics*, v. 31, n. 3, p. 327–337, 2007.

Tenenbaum, J. B. Mapping a manifold of perceptual observations. In: *Proceedings of the 1997 Conference on Advances in Neural Information Processing Systems (NIPS'97)*, Cambridge, MA, USA: MIT Press, 1998, p. 682–688.

Tenenbaum, J. B.; Silva, V.; Langford, J. C. A global geometric framework for nonlinear dimensionality reduction. *Science*, v. 290, n. 5500, p. 2319–2323, 2000.

Thomas, J. J.; Cook, K. A., eds. *Illuminating the path: The research and development agenda for visual analytics.* Los Alamitos: IEEE Computer Society Press, 2005.

Toda, H.; Kataoka, R. A search result clustering method using informatively named entities. In: *Proceedings of the 7th Annual ACM International Workshop on Web Information and Data Management (WIDM'05)*, New York, NY, USA: ACM Press, 2005, p. 81–86.

Torgerson, W. Multidimensional scaling: I. theory and method. *Psychometrika*, v. 17, n. 4, p. 401–419, 1952.

Tutte, W. T. How to draw a graph. *Lodon Mathematical Society*, v. 13, p. 743–768, 1963.

UCI-MLR University of California Irvine, Machine Learning Repository:. `http://archive.ics.uci.edu/ml/datasets.html` (acessado em 14/09/2008), 2008.

UW-MLCDP University of Wisconsin, Machine Learning for Cancer Diagnosis and Prognosis:. `http://pages.cs.wisc.edu/~olvi/uwmp/cancer.html` (acessado em 14/09/2008), 2008.

Verveer, P. J.; Duin, R. P. W. An evaluation of intrinsic dimensionality estimators. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 17, n. 1, p. 81–86, 1995.

Vivísimo. Search done right. `http://www.vivisimo.com/` (acessado em 14/09/2008), 2008.

Walshaw, C. A multilevel algorithm for force-directed graph drawing. In: *Proceedings of the 8th International Symposium on Graph Drawing (GD'00)*, London, UK: Springer-Verlag, 2001, p. 171–182.

Wang, W.; Wang, H.; Dai, G.; Dai, G.; Wang, H. Visualization of large hierarchical data by circle packing. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'06)*, New York, NY, USA: ACM, 2006, p. 517–520.

Ward, M. O. A taxonomy of glyph placement strategies for multidimensional data visualization. *Information Visualization*, v. 1, n. 3/4, p. 194–210, 2002.

WISE, J. A. The ecological approach to text visualization. *Journal of the American Society for Information Science*, v. 50, n. 13, p. 1224–1233, 1999.

WONG, P. C. Guest editor's introduction: Visual data mining. *IEEE Computer Graphics and Applications*, v. 19, n. 5, p. 20–21, 1999.

WURMAN, R. S. *Information anxiety.* New York & Toronto: Doubleday, 1989.

YANG, M.-L. Distance-preserving projection of high-dimensional data for nonlinear dimensionality reduction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 26, n. 9, p. 1243–1246, senior Member-Li Yang, 2004.

YOUNG, G.; HOUSEHOLDER, A. Discussion of a set of points in terms of their mutual distances. *Psychometrika*, v. 3, n. 1, p. 19–22, 1938.

ZENG, H.-J.; HE, Q.-C.; CHEN, Z.; MA, W.-Y.; MA, J. Learning to cluster web search results. In: *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'04)*, New York, NY, USA: ACM Press, 2004, p. 210–217.

ZEZULA, P.; AMATO, G.; DOHNAL, V.; BATKO, M. *Similarity search: The metric space approach (advances in database systems).* Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2005.

ZHAO, Y.; KARYPIS, G. Evaluation of hierarchical clustering algorithms for document datasets. In: *Proceedings of the 11th International Conference on Information and Knowledge Management (CIKM'02)*, New York, NY, USA: ACM, 2002, p. 515–524.

ZHAO, Y.; KARYPIS, G. Empirical and theoretical comparisons of selected criterion functions for document clustering. *Machine Learning*, v. 55, n. 3, p. 311–331, 2004.

ZHAO, Y.; KARYPIS, G.; FAYYAD, U. Hierarchical clustering algorithms for document datasets. *Data Mining and Knowledge Discovery*, v. 10, n. 2, p. 141–168, 2005.

Livros Grátis

(http://www.livrosgratis.com.br)

Milhares de Livros para Download:

Baixar livros de Literatura
Baixar livros de Literatura de Cordel
Baixar livros de Literatura Infantil
Baixar livros de Matemática
Baixar livros de Medicina
Baixar livros de Medicina Veterinária
Baixar livros de Meio Ambiente
Baixar livros de Meteorologia
Baixar Monografias e TCC
Baixar livros Multidisciplinar
Baixar livros de Música
Baixar livros de Psicologia
Baixar livros de Química
Baixar livros de Saúde Coletiva
Baixar livros de Serviço Social
Baixar livros de Sociologia
Baixar livros de Teologia
Baixar livros de Trabalho
Baixar livros de Turismo