SOLEDAD ESPEZÚA LLERENA

**MAPEAMENTO DE DADOS GENÔMICOS USANDO
ESCALONAMENTO MULTIDIMENSIONAL**

Dissertação apresentada à Escola de Engenharia de São Carlos da Universidade de São Paulo, como parte dos requisitos para a obtenção do título de Mestre em Engenharia Elétrica.

Área de concentração: Processamento de Sinais e Instrumentação.

Orientador:

Prof. Dr. Carlos Dias Maciel

São Carlos

2008

*Dedico esta dissertação aos meus pais Hugo e Soledad,
meu esposo Edwin e meus irmãos César e Katia.*

AGRADEÇO:

Ao meu orientador prof. Dr. Carlos Dias Maciel, pelo apoio, confiança e oportunidade, elementos essenciais para a realização deste trabalho, assim como aos professores Dra. Vilma Alves de Oliveira e Dr. José Carlos Pereira, pelo apoio e ensino valioso durante este mestrado.

A todos os amigos do LIM e de elétrica, pelo acolhimento, abertura e disponibilidade em ajudar-me e pelos gratos momentos de companheirismo importante neste período.

Aos amigos Julian, Shermila, Elmer, Madeleine, Waldo, Roxana, Renato e Anita com os quais compartilhei grandes momentos de alegria, bom humor e descontração durante este mestrado. Também agradeço a todos meus demais queridos amigos que têm me acompanhado a vida toda. Agradeço especialmente as minhas grandes amigas Miluska e Maribel que sempre torceram por mim.

A meu esposo Edwin pelo grande apoio e amor, por estar sempre nos momentos mais difíceis dando-me seu auxílio e força. E claro, sempre estive nos momentos mais inspirados e divertidos, compartilhando felicidade que continuará assim a vida toda.

Aos meus pais, por serem um exemplo de vida e meus queridos irmãos César e Katia, os quais são meus melhores amigos e serem sempre tão bonzinhos comigo, obrigada a todos.

“The best way to predict the future is to invent it.”

Alan Kay

RESUMO

ESPEZÚA, Soledad (2008). Mapeamento de dados genômicos usando Escalonamento Multidimensional. 100p. Dissertação (Mestrado). Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos 2008.

Neste trabalho são exploradas diversas técnicas de *Escalonamento Multidimensional* (MDS), com o objetivo de estudar sua aplicabilidade no mapeamento de dados genômicos resultantes da técnica RFLP-PCR, sendo esse mapeamento realizado em espaços de baixa dimensionalidade (2D ou 3D) com o fim de aproveitar a habilidade de análise e interpretação visual que possuem os seres humanos. Foi realizada uma análise comparativa de diversos algoritmos MDS, visando sua aptidão para mapear dados genômicos. Esta análise compreendeu o estudo de alguns índices de desempenho como a precisão no mapeamento, o custo computacional e a capacidade de induzir bons agrupamentos. Para a realização dessa análise foi desenvolvida a ferramenta “MDSExplorer”, a qual integra os algoritmos estudados e várias opções que permitem comparar os algoritmos e visualizar os mapeamentos. A análise realizada sobre diversos bancos de dados citados na literatura, sugerem que o algoritmo LANDMARK possui o menor tempo computacional, uma precisão de mapeamento similar aos demais algoritmos, e uma boa capacidade de manter as estruturas existentes nos dados. Finalmente, o MDSExplorer foi usado para mapear um banco de dados genômicos: o banco de estirpes de bactérias fixadoras de nitrogênio, pertencentes ao gênero *Bradyrhizobium*, com objetivo de ajudar o especialista a inferir visualmente alguma taxonomia nessas estirpes. Os resultados na redução dimensional desse banco de dados sugeriram que a informação relevante (acima dos 60% da variância acumulada) para as regiões 16S, 23S e IGS estaria nas primeiras 5, 4 e 9 dimensões respectivamente.

Palavras-Chave: Redução dimensional, Escalonamento Multidimensional, MDS, CMDS, FASTMAP, LANDMARK, dados genômicos, visualização de informação.

ABSTRACT

ESPEZÚA, Soledad (2008). *Representation of genomics data with Multidimensional Scaling*. 100p. M.Sc (Dissertation) – School of Engineering – University of São Paulo, São Carlos, 2008.

In this work were studied various *Multidimensional Scaling* (MDS) techniques intended to apply in the mapping of genomics data obtained of RFLP-PCR technique. This mapping is done in a low dimensional space (2D or 3D), and has the intention of exploiting the visual human capability on analysis and synthesis. A comparative analysis of diverse algorithms MDS was carried out in order to devise its ubiquity in representing genomics data. This analysis covers the study of some indices of performance such as: the precision in the mapping, the computational cost and the capacity to induce good groupings. The purpose of this analysis was developed a software tool called “MDSExplorer”, which integrates various MDS algorithms and some options that allow to compare the algorithms and to visualize the mappings. The analysis, carried out over diverse datasets cited in the literature, suggest that the algorithm LANDMARK has the lowest computational time, a good precision in the mapping, and a tendency to maintain the existing structures in the data. Finally, MDSExplorer was used to mapping a real genomics dataset: the RFLP-PCR images of a Brazilian collection of bacterial strains belonging to the genus *Bradyrhizobium* (known by their capability to transform the nitrogen of the atmosphere into compounds useful for the host plants), with the objective to aid the specialist to infer visually a taxonomy in these strains. The results in reduction of dimensionality in this data base, suggest that the relevant information (above 60% of variance accumulated) to the region 16S, 23S and IGS is around 5, 4 and 9 dimensions respectively.

Keywords: Dimensionality Reduction, Multidimensional Scaling, MDS, CMDS, FASTMAP, LANDMARK, genomics data, information visualization.

LISTA DE FIGURAS

Figura 1.1: Exemplo de imagens de gel-eletroforese de 4 organismos, em que as bandas são linhas de cor escuro, as quais indicam a presença de material genético. O sentido e a velocidade de migração é determinado pelo tamanho e carga das moléculas.	2
Figura 2.1: Exemplo de visualização por nuvens de pontos do banco de dados Cevada, representando as combinações entre 6 cidades e 10 variedades de cevada que cresceram em 2 anos ($6 \times 10 \times 20 = 120$ observações).....	12
Figura 2.2: Exemplo de visualização por coordenadas paralelas, adaptado de Nascimento e Ferreira (2005).	13
Figura 2.3: Exemplo de visualização utilizando a técnica de <i>Chernoff faces</i> , adaptado de Müller e Alexa, (1998).	14
Figura 3.1: Representação do problema MDS.	16
Figura 3.2: Representação do proceso de validação em MDS.....	16
Figura 3.3. Considerações em MDS: (a)Apresenta solução em MDS, (b) Não apresenta solução exata em MDS.	17
Figura 3.4: Pseudocódigo do algoritmo CMDS.	25
Figura 3.5: Exemplo de 2 iterações do método <i>Iterative Majorization</i> adaptado de (BORG; GROENEN, 2005).	26
Figura 3.6: Pseudocódigo do algoritmo SMACOF.	29
Figura 3.7: Ilustração da projeção do ponto O_i sobre a linha $O_a O_b$. Adaptado de Faloutsos, Lin (1995).	30
Figura 3.8: Projeção das distâncias sobre um hiperplano H , perpendicular à linha de pivôs $O_a O_b$. Adaptado de Faloutsos e Lin (1995).	31
Figura 3.9: Pseudocódigo do algoritmo FASTMAP.	32
Figura 3.10: Pseudocódigo do algoritmo LANDMARK.....	35
Figura 3.11: Pseudocódigo do algoritmo MaxMin.....	36
Figura 4.1: Representação do banco de dados Iris, considerando os três primeiros	

atributos e utilizando o atributo classe para dar cor. As espécies <i>setosa</i> , <i>versicolor</i> e <i>virginica</i> estão representadas pelas cores azul, vermelho e verde respectivamente.	38
Figura 4.2: Representação do banco de dados Câncer, considerando as três primeiras dimensões e utilizando o atributo classe para dar cor. Os dois tipos de células malignas e benignas estão representadas pelas cores verde e azul respectivamente.	39
Figura 4.3: Representação do banco de dados Images, considerando as três primeiras dimensões e utilizando o atributo classe para dar cor. As imagens <i>brickface</i> (1), <i>sky</i> (2), <i>foliage</i> (3), <i>cement</i> (4), <i>window</i> (5), <i>path</i> (6) e <i>grass</i> (7).	40
Figura 4.4: Representação do Banco de dados Sintético usando as 3 primeiras dimensões e utilizando o atributo classe para dar cor. Adaptado de Villanueva (2007).	41
Figura 4.5: Exemplo de imagem resultante do processo de gel eletroforese, no qual foi analisado uma única estirpe.	42
Figura 4.6: Exemplo de imagens de canaletas de 4 organismos, em que as bandas são as linhas escuras que indicam a presença de material genético. A posição das bandas na canaleta é determinado pelo tamanho dos seus fragmentos.	43
Figura 4.7: Obtenção das matrizes de distância para MDS. Cada matriz é obtida com o pré-processamento de uma determinada canaleta das 119 estirpes.	44
Figura 4.8: Processo de formação do electroferograma a partir de uma canaleta. Adaptado de Villanueva (2007).	45
Figura 4.9: Ilustração do processo de formação de uma matriz de distância.	46
Figura 4.10: Funcionamento da ferramenta MDSExplorer gerando mapeamentos de dados mediante a seleção de uma matriz de distância.	48
Figura 4.11: Interface de seleção de matrizes distância do banco de estirpes de <i>Bradyrhizobium</i>	48
Figura 4.12: Exemplo de interface do <i>MDSExplorer</i> apresentado a análise do	

banco de dados Câncer.....	49
Figura 4.13: Imagem do espaço RGB representado em um cubo.....	50
Figura 4.14: Visualização do banco <i>Bradyrhizobium</i> em VTK apresentando o resultado do mapeamento gerado no <i>MDSExplorer</i>	51
Figura 4.15: Pseudocódigo do processo de cálculo dos índices de desempenho dos algoritmos MDS nos bancos de dados de avaliação.....	55
Figura 5.1: Stress-1 para o banco Iris	58
Figura 5.2: Stress-2 para o banco Iris	58
Figura 5.3: Tempo Computacional do banco Iris (em escala logarítmica).....	59
Figura 5.4: Índice de pureza de agrupamento do banco Iris	60
Figura 5.5: Stress-1 para o banco Câncer	61
Figura 5.6: Stress-2 para o banco Câncer	61
Figura 5.7: Tempo Computacional do banco Câncer (em escala logarítmica).....	62
Figura 5.8: Índice de pureza de agrupamento do banco Câncer	63
Figura 5.9: Stress-1 para o banco Imagens	64
Figura 5.10: Stress-2 para o banco Imagens	64
Figura 5.11: Tempo Computacional do banco Imagens (em escala logarítmica).....	65
Figura 5.12: Índice de pureza de agrupamento do banco Imagens	66
Figura 5.13: Stress-1 para o banco Sintético.....	67
Figura 5.14: Stress-2 para o banco Sintético.....	67
Figura 5.15: Tempo Computacional do banco Sintético (em escala logarítmica).	68
Figura 5.16: Índice de pureza de agrupamento do banco Sintético	69
Figura 5.17: (a) Visualização das três primeiras dimensões do mapeamento da Matriz D1 (16S- Cfo), (b) Distribuição de Autovalores por dimensionalidade, (c) Critérios de <i>stress</i> para a matriz D1de	

<i>Bradyrhizobium</i>	71
Figura 5.18: (a) Visualização das três primeiras dimensões do mapeamento da Matriz D2-(16S-Dde), (b) Distribuição de Autovalores por dimensionalidade, (c) Critérios de <i>stress</i> para a matriz D2 de <i>Bradyrhizobium</i>	72
Figura 5.19: (a) Visualização das três primeiras dimensões do mapeamento da Matriz D3-(16S-Msp), (b) Distribuição de Autovalores por dimensionalidade, (c) Critérios de <i>stress</i> para a matriz D3 de <i>Bradyrhizobium</i>	73
Figura 5.20: (a) Visualização das três primeiras dimensões do mapeamento da Matriz D4-(23S-Hha), (b) Distribuição de Autovalores por dimensionalidade, (c) Critérios de <i>stress</i> para a matriz D4 de <i>Bradyrhizobium</i>	74
Figura 5.21: (a) Visualização das três primeiras dimensões do mapeamento da Matriz D5-(23S-Hha), (b) Distribuição de Autovalores por dimensionalidade, (c) Critérios de <i>stress</i> para a matriz D5 de <i>Bradyrhizobium</i>	75
Figura 5.22: (a) Visualização das três primeiras dimensões do mapeamento da Matriz D6-(23S-Hinf), (b) Distribuição de Autovalores por dimensionalidade, (c) Critérios de <i>stress</i> para a matriz D6 de <i>Bradyrhizobium</i>	76
Figura 5.23: (a) Visualização das três primeiras dimensões do mapeamento da Matriz D7-(IGS-Dde), (b) Distribuição de Autovalores por dimensionalidade, (c) Critérios de <i>stress</i> para a matriz D7 de <i>Bradyrhizobium</i>	77
Figura 5.24: (a) Visualização das três primeiras dimensões do mapeamento da Matriz D8-(IGS-Hae), (b) Distribuição de Autovalores por dimensionalidade, (c) Critérios de <i>stress</i> para a matriz D8 de <i>Bradyrhizobium</i>	78
Figura 5.25: (a) Visualização das três primeiras dimensões do mapeamento da Matriz D9-(IGS-Msp), (b) Distribuição de Autovalores por dimensionalidade, (c) Critérios de <i>stress</i> para a matriz D9 de <i>Bradyrhizobium</i>	79

Figura 5.26: Visualização em três dimensões do banco <i>Bradyrhizobium</i>	80
Figura 6.1: Stress-1 para a matriz D1 do banco <i>Bradyrhizobium</i>	96
Figura 6.2: Stress-1 para a matriz D2 do banco <i>Bradyrhizobium</i>	96
Figura 6.3: Stress-1 para a matriz D3 do banco <i>Bradyrhizobium</i>	97
Figura 6.4: Stress-1 para a matriz D4 do banco <i>Bradyrhizobium</i>	97
Figura 6.5: Stress-1 para a matriz D5 do banco <i>Bradyrhizobium</i>	98
Figura 6.6: Stress-1 para a matriz D6 do banco <i>Bradyrhizobium</i>	98
Figura 6.7: Stress-1 para a matriz D7 do banco <i>Bradyrhizobium</i>	99
Figura 6.8: Stress-1 para a matriz D8 do banco <i>Bradyrhizobium</i>	99
Figura 6.9: Stress-1 para a matriz D9 do banco <i>Bradyrhizobium</i>	100

LISTA DE TABELAS

Tabela 4.1 - Relação de enzimas de restrição utilizadas e regiões ribossomais analisadas na obtenção do banco de dados de estirpes de <i>Bradyrhizobium</i>	42
Tabela 4.2 – Tamanho de amostras apresentadas ao algoritmo LANDMARK.....	55
Tabela 5.1 - Tabela de Índices de purezas dos bancos de avaliação analisados com o algoritmo <i>K-Means</i>	70
Tabela 5.2 - Tabela resumo de porcentagem de Autovalores acumulados nas 3 primeiras dimensões para o banco <i>Bradyrhizobium</i>	80
Tabela 5.3 - Tabela resumo de mínimo valor do critério de <i>stress</i> encontrado no banco <i>Bradyrhizobium</i>	81
Tabela 6.1: Tempo computacional para o banco de dados Iris.	94
Tabela 6.2: Tempo computacional para o banco de dados Câncer.	94
Tabela 6.3: Tempo computacional para o banco de dados Imagens.	95
Tabela 6.4: Tempo computacional para o banco de dados Sintético.	95

Lista de Apêndices

APÊNDICE 1: Tabelas de tempo computacional94

APÊNDICE 2: Evolução do *stress* S1 para o banco *Bradyrhizobium*96

LISTA DE SÍMBOLOS

n	Número de objetos ou dados.
i, j	Índice de objetos, $i, j = 1, \dots, n$
m	Número de dimensões.
a	Índice de dimensões, $a = 1, \dots, m$
$\mathbf{X}$	Matriz de pontos x_{ia} de n objetos sobre m dimensões.
p_{ij}	Proximidade entre os objetos i e j ; representa medidas de similaridade ou dissimilaridade. Para similaridades um alto valor de p_{ij} indica que os pares de objetos são muito similares.
$f(p_{ij})$	Representa um mapeamento de p_{ij} . Pode-se entender também que $f(p_{ij})$ é uma transformação de p_{ij} (os termos função, transformação, e mapeamento são sinônimos neste contexto). Em lugar de escrever $f(p_{ij})$, pode-se também escrever $\hat{d}_{ij}$.
δ_{ij}	Dissimilaridade entre i e j , sempre são positivas.
Δ	Matriz simétrica de dissimilaridades $\Delta = (\delta_{ij})$ de tamanho $n \times n$ com $\delta_{ii} = 0$.
$d_{ij}(\mathbf{X})$	Distância euclidiana entre as filas i e j de $\mathbf{X}$, e $d_{ij} = \left[\sum_{a=1}^m (x_{ia} - x_{ja})^2 \right]^{1/2}$.
d_{ij}	Abreviação para a notação da distância euclidiana $d_{ij}(\mathbf{X})$.
$\hat{d}_{ij}$	Disparidade entre os objetos i e j . As disparidades são proximidades admissivelmente transformadas que aproximam as distâncias.
w_{ij}	Peso não negativo, usado para ponderar os resíduos da função de <i>stress</i> .

- W** Matriz simétrica de pesos w_{ij} com diagonal zero.
- D(X)** Matriz distância Euclidiana entre as linhas de **X**, na qual $\mathbf{D}(\mathbf{X}) = (d_{ij})$.
Abreviadamente escreve-se **D**.
- 1** Vetor coluna com os elementos iguais a 1.
- J** Matriz centrada $n \times n$, no qual $\mathbf{J} = \mathbf{I} - n^{-1}\mathbf{1}\mathbf{1}'$.
- tr(A)** Operador traço, soma os elementos diagonais da matriz **A**, por exemplo,
 $\text{tr}(\mathbf{A}) = \sum_{i=1}^n a_{ii}$.

SUMÁRIO

RESUMO.....	IX
ABSTRACT.....	XI
LISTA DE FIGURAS	XIII
LISTA DE TABELAS.....	XIX
LISTA DE APÊNDICES.....	XXI
LISTA DE SÍMBOLOS	XXIII
1. INTRODUÇÃO	1
1.1. Objetivos.....	5
1.2. Estrutura da dissertação	5
2. REVISÃO BIBLIOGRÁFICA	7
2.1. Redução dimensional.....	7
Escalonamento Multidimensional	9
2.2. Visualização de informação	11
3. FUNDAMENTOS TEÓRICOS.....	15
3.1. Conceito de Escalonamento Multidimensional.....	15
3.2. Função de mapeamento e modelos MDS	17
3.3. Medidas de distância	18
3.4. Matrizes de distância	20
3.5. Função Stress.....	22
3.6. Algoritmos MDS.....	23
Algoritmo CMDS.....	23
Algoritmo SMACOF	25
Algoritmo FASTMAP	29
Algoritmo LANDMARK	32

4. MATERIAIS E MÉTODOS.....	37
4.1. Bancos de dados de avaliação	37
Banco de dados Iris	37
Banco de dados Câncer.....	38
Banco de dados Imagens.....	39
Banco de dados Sintético	40
4.2. Banco de dados de aplicação.....	41
Pré-Processamento dos dados.....	45
4.3. Ferramenta MdsExplorer	47
Visualização em VTK.....	50
4.4. Experimentos.....	52
5. RESULTADOS E DISCUSSÕES	57
5.1. Resultados nos bancos de dados de avaliação	57
Banco de dados Iris	57
Banco de dados de Câncer.....	60
Banco de dados de Imagens	63
Banco de dados Sintético	66
Discussão	69
5.2. Resultados no banco de dados de estirpes <i>Bradyrhizobium</i>	70
Discussão	81
6. CONCLUSÕES E SUGESTÕES.....	83
Sugestões para trabalhos futuros.....	84
REFERÊNCIAS BIBLIOGRÁFICAS	85
APÊNDICES	93

Capítulo 1

INTRODUÇÃO

Nas últimas décadas vem sendo observado um aumento significativo na quantidade de dados disponíveis em diversas áreas do conhecimento humano, como consequência dos avanços na obtenção e armazenamento de dados. A Biologia Molecular é uma das áreas que tem mostrando um contínuo avanço, disponibilizando uma grande quantidade de dados genômicos. Isto traz como consequência uma crescente demanda por ferramentas e métodos computacionais, capazes de analisar um grande número de dados e de possibilitar a extração de conhecimento para facilitar e apoiar as pesquisas biológicas.

A microbiologia do solo, por exemplo, vem empregando técnicas de biologia molecular para gerar dados genômicos de microorganismos que são de utilidade para as culturas agrícolas. Um destes microorganismos é o rizóbio, o qual é uma espécie de bactéria que realiza o processo de Fixação Biológica do Nitrogênio (FBN) nas leguminosas (CHUEIRI, *et al.*, 2005). Esse processo é a principal fonte de nitrogênio para a cultura destas plantas, onde estirpes de rizóbios estabelecem uma relação simbiótica (formando nódulos) ao entrar em contato com as raízes da planta hospedeira (HUNGRIA *et al.*, 1997). A eficiência da fixação do nitrogênio depende da estirpe de bactéria utilizada, sendo atualmente um desafio a classificação e seleção de estirpes de rizóbio mais eficientes e competitivas (ARAUJO; HUNGRIA, 1999).

A identificação taxonômica das bactérias baseou-se tradicionalmente em diversas propriedades fenotípicas como: morfologia, cultivo, nutrição, bioquímica, metabolismo, patogenicidade e ecologia destes microorganismos (LIMA, *et al.*, 2005),

sendo o principal problema a dificuldade e custo na coleção desses dados. A taxonomia molecular é uma proposta alternativa para a identificação taxonômica de microorganismos, a qual permite agrupar os microorganismos com base na informação das suas propriedades genotípicas (estruturas moleculares de DNA ou RNA). Para determinar de forma exata a estrutura molecular do DNA ou RNA usa-se o *seqüenciamento*. Não obstante, esta técnica pode ter custo elevado e ser demorada, sendo em muitas situações práticas pouco viável de realizar (MILAGRE, 2003).

Uma técnica econômica bastante utilizada para realizar uma análise exploratória para a identificação de espécies e subespécies de organismos é a técnica RFLP-PCR (*Restriction Fragment Length Polymorphism - Polymerase Chain Reaction*) (ZAHA, 2000; DAVISON, 2006). Nessa técnica são usadas enzimas de restrição que cortam o DNA em um conjunto de fragmentos, os quais são ordenados em função de seus tamanhos mediante a técnica de gel-eletroforese (ZAHA, 2000). O resultado é um conjunto de *bandas* distribuídas ao longo de uma *canaleta* de gel. As bandas representam fragmentos de DNA de igual tamanho e o conjunto destas bandas pode ser vista como uma “impressão digital do DNA”, a qual permite a diferenciação entre organismos. Na Figura 1.1 é mostrado um exemplo de imagem de gel-eletroforese de 4 microorganismos, em que as bandas são as linhas de cor escura, as quais indicam a presença de material genético e a linha vertical da esquerda indica o sentido de migração dos fragmentos. Nota-se, por exemplo, que a primeira e a última canaleta são similares, o que pode implicar que sejam organismos da mesma espécie.

Figura 1.1: Exemplo de imagens de gel-eletroforese de 4 organismos, em que as bandas são linhas de cor escura, as quais indicam a presença de material genético. O sentido e a velocidade de migração é determinado pelo tamanho e carga das moléculas.

A análise dos dados genômicos resultantes da técnica RFLP-PCR tornou-se

importante para resolver as questões biológicas ao respeito da identificação taxonômica de diversos microorganismos. Muitas pesquisas realizadas recentemente tiveram como objetivo extrair conhecimento destes dados. Por exemplo, os trabalhos realizados por Milagre (2003), Lima *et al.* (2005), Germano *et al.* (2006), Araújo e Hungria (1999), Nóbrega *et al.* (2004), Vargas *et al.* (2007) entre outros, usaram algoritmos de agrupamento hierárquico como UPGMA (*unweighted pair-group method with arithmetic means*) (SNEATH; SOKAL, 1973) para representar os microorganismos mediante estruturas hierárquicas (dendrogramas), os quais ajudam a identificar relações de similaridade entre eles. Embora estas representações sejam úteis em muitos casos, estas não são sempre adequadas para representar grandes bancos de dados devido a que existem dois problemas associados: i) é difícil visualizar e interpretar dendrogramas com muitos nós, e ii) a identificação do nível apropriado onde se deve cortar o dendrograma é uma tarefa difícil, gerando diversas interpretações em quanto à formação de grupos (SCHROEDER *et al.*, 2001).

A complexidade dos dados *genômicos* obtidos por RFLP-PCR, como os apresentados na Figura 1.1, pode ser ainda maior quando são usadas varias enzimas de restrição para analisar diversas regiões ribossomais. Isto implica que várias imagens de gel eletroforese representam um mesmo organismo, sendo muito difícil extrair conhecimento a partir destas imagens. Assim, torna-se de grande importância a transformação destas imagens de eletroforese em representações visuais mais adequadas, nas quais os especialistas possam extrair conhecimento. Autores como Card *et al.*(1999) e Schroeder *et al.*(2001) enfatizam a necessidade de encontrar técnicas para mapeamento de grandes quantidades de dados que permitam incluir às pessoas no processo de classificação (em lugar de um processo automatizado) mediante a visualização interativa dos diferentes aspectos dos dados (visualização exploratória).

O Escalonamento Multidimensional (*Multidimensional Scaling* - MDS) é um conjunto de técnicas de redução dimensional, que têm por finalidade analisar as medidas de proximidade entre pares de objetos¹ (matriz de distâncias) para conseguir

¹ Um objeto é entendido neste trabalho como um elemento, entidade, amostra ou dado.

uma representação (mapeamento) dos objetos como pontos em espaços com baixa dimensão (2D, 3D). Espaços, nos quais os seres humanos possuem uma alta capacidade de análise e interpretação (BORG, GROENEN, 2005; SCHROEDER *et al.*, 2001). O sucesso das técnicas MDS deve-se a sua aplicação para uma grande variedade de problemas. Por exemplo, têm-se aplicações em psicologia (IZMAILOV *et al.*, 2005), marketing (CARROL; GREEN, 1997), mineração de dados (HUANG *et al.*, 2005; FALOUTSOS; LIN, 1995; WANG *et al.*, 2005; SILVA; TENENBAUM, 2003a), análise de microarray (BORGES, 2006), transmissões neurais (BECKMANN; GATTAZ, 2002) modelagem molecular (VENKATARAJAN; BRAUN, 2001; AGRAFIOTIS, *et al.*, 2000) e outras aplicações (SCHROEDER *et al.*, 2001, PRIEGO, 2003; BEVILACQUA, 2004; ABDI, *et al.*, 2005; ABDI, *et al.*, 2007). A popularidade de MDS deve-se a que as informações de proximidade entre objetos podem ser extraídas facilmente de diversos tipos de dados (vetores de características, seqüências temporais, juízos de similaridade, imagens, etc.) mediante o uso de alguma métrica de distância. No entanto o processo de mapear informações de proximidade como pontos num espaço de visualização, pode ser muito difícil e dependente da natureza dos dados, sendo este um problema de constante pesquisa.

Após de uma revisão bibliográfica na literatura, não foram encontrados trabalhos que utilizem técnicas MDS para representar dados genômicos resultantes da técnica RFLP-PCR (imagens de gel-eletroforese). A utilização de técnicas MDS para representar este tipo de dados pode ser justificada devido ao fato de que sempre será possível obter as informações de proximidade mediante alguma medida de similaridade ou correlação entre imagens. Desta forma, neste trabalho se estudam diversos algoritmos de Escalonamento Multidimensional, procurando avaliar sua aplicação em bancos de dados genômicos do tipo RFLP-PCR, para o qual foi realizado um estudo em duas etapas. Na primeira etapa realiza-se uma análise comparativa dos algoritmos MDS em bancos de dados de avaliação (onde se conhece suas classes) com o objetivo de determinar aquele que realize o mapeamento com a menor distorção possível, que use o menor tempo computacional e que induza agrupamentos próximos das classes existentes. Na segunda etapa é usado o algoritmo de melhor desempenho em um problema de aplicação real: o mapeamento visual de um banco de dados

genômico de rizóbios pertencentes ao gênero *Bradyrhizobium*. Uma ferramenta é desenvolvida para realizar a transformação dos dados e gerar mapeamento dos mesmos. Essa ferramenta ajuda a posicionar as proximidades entre dados RFLP-PCR, tendo apenas como entrada a informação de similaridade entre esses dados.

1.1. OBJETIVOS

O presente trabalho tem como objetivo geral estudar as diversas técnicas de redução dimensional baseadas em MDS com a finalidade de identificar a mais adequada para o mapeamento de dados genômicos representados pelas imagens de canaletas resultantes da técnica RFLP-PCR.

Os objetivos específicos foram:

- Integrar as técnicas de MDS em uma ferramenta que permita visualizar e comparar os resultados dos mapeamentos das diversas técnicas.
- Aplicar as técnicas de MDS num banco de dados genômico real, o banco de estirpes de bactérias fixadoras de nitrogênio pertencentes ao gênero *Bradyrhizobium*, visando inferir visualmente alguma taxonomia neste banco de dados.

1.2. ESTRUTURA DA DISSERTAÇÃO

Esta dissertação organiza-se da seguinte forma: o Capítulo 2 apresenta uma revisão bibliográfica das técnicas de redução dimensional; no Capítulo 3 apresentam-se alguns conceitos básicos necessários para o entendimento de MDS e os algoritmos mais relevantes para o projeto proposto; no Capítulo 4 descrevem-se os bancos de dados usados como testes para os algoritmos estudados e descreve-se a ferramenta criada neste trabalho, o MDSExplorer; no Capítulo 5 apresentam-se os resultados e discussões obtidas das comparações entre os diferentes algoritmos e suas respectivas visualizações, e finalmente no Capítulo 6 apresentam-se as conclusões e sugestões para trabalhos futuros.

Capítulo 2

REVISÃO BIBLIOGRÁFICA

2.1. REDUÇÃO DIMENSIONAL

O objetivo da redução dimensional é levar um conjunto de dados de um espaço n – dimensional a outro espaço m – dimensional ($m < n$), preservando ao máximo as relações de similaridade existentes no conjunto original dos dados. A redução da dimensão permite que os dados multidimensionais possam ser representados de forma mais eficiente na sua visualização. Desta forma são reveladas algumas características importantes como: agrupamentos, tendências ou anomalias. Em JAIN *et al.* (2000) se cita à redução dimensional como um passo essencial no processo de compressão de dados e extração de informação relevante. Por outro lado em Chizi e Maimon (2005) identificam-se quatro motivos para realizar a redução dimensional: redução de custo de aprendizado (custo computacional), incremento do desempenho de aprendizado (exatidão), redução de dimensões irrelevantes (quando os atributos possuem valores similares a níveis de ruído) e dimensões redundantes (quando os atributos são combinações lineares de outros atributos).

Enquanto os motivos para realizar redução dimensional estão claramente identificados ainda existe uma série de problemas em aberto. Entre esses problemas tem-se: i) o desconhecimento da dimensionalidade intrínseca, visto que não se tem uma forma eficiente de encontrar o número mínimo de dimensões suficientes para representar adequadamente os dados; ii) relacionamentos não lineares entre os dados, uma vez que reconhecer relações não lineares entre variáveis pode ser muito

complicado; iii) e finalmente o desconhecimento do que é informação relevante, já que existe uma grande tendência a perder certa quantidade de informação quando são aplicadas técnicas de redução de dimensionalidade (HUANG *et al.*, 2005).

Neste mesmo contexto também aparece o problema conhecido como “maldição da dimensionalidade” (*curse of dimensionality*), que se apresenta quando existe um incremento exponencial das dimensões, tornando os dados muitos esparsos e conseqüentemente, as distâncias entre eles tornam-se relativamente uniformes (SCOTT, 1992; BISHOP, 1995; MAIMON, ROKACH, 2005). Também está associada à incapacidade de gerar estruturas significativas (padrões ou modelos) e na execução de algoritmos de classificação de dados incrementa o espaço de busca em forma exponencial, aumentando a possibilidade de induzir classificações erradas (MAIMON, ROKACH, 2005).

Atualmente, o uso de técnicas de redução dimensional em diversas áreas da engenharia, ciências sociais, ciências da computação e ciências biológicas vêm mostrando um grande avanço. Em biologia molecular, por exemplo, utilizam-se as técnicas de redução dimensional para analisar *seqüências de genoma*, os quais são dados altamente complexos e extensos (TENG *et al.*, 2005). Assim estas técnicas se constituem como uma ferramenta de auxílio tanto no processo de transformação dos dados, como no processo de geração de informações visuais.

Devido à grande quantidade de técnicas de redução dimensional e à natureza multidisciplinar onde elas são aplicadas, diversas classificações delas podem ser encontradas na literatura. Assim, por exemplo, Burges (2005) apresenta uma divisão por métodos, os quais são: i) métodos geométricos para extração de características baseados em projeções (PCA, *Probabilistic PCA*, *Kernel PCA*, *Oriented PCA*) e métodos para redução dimensional que tentam modelar a variedade¹ (*manifold*) em que os dados estão imersos (MDS, *Isomap*, *Locally Linear Embedding* (LLE), *Laplacian Eigenmaps*, *Spectral Clustering*). Outros autores como Duda *et al.* (2001), Schroeder *et al.* (2001), Maimon e Rokach (2005) dividem as técnicas de redução

¹ Variedade nesta dissertação entende-se como a modelação de um espaço por uma função que consiga reduzir a dimensão.

dimensional em: i) técnicas lineares baseadas em combinações lineares dos dados para definir o espaço dimensional final (Análise por agrupamento hierárquico (HCA), PCA, *Factor Analysis*, *Projection Pursuit*) e ii) técnicas não lineares de projeção como MDS que seriam mais apropriadas para casos em que a única informação disponível é a proximidade entre os dados.

Como o interesse desta dissertação está no estudo de técnicas de redução dimensional MDS, a seguir apresenta-se uma revisão bibliográfica deste tipo de técnicas de redução dimensional.

ESCALONAMENTO MULTIDIMENSIONAL

O Escalonamento Multidimensional (ou *Multidimensional Scaling* MDS em inglês) refere-se a uma família de técnicas que tem por finalidade representar objetos num baixo espaço dimensional, tendo como entrada a informação de proximidade entre pares (matriz de distância) (BORG; GROENEN 2005). Essas proximidades podem proceder de diversas fontes, como, juízos de similaridade feitos por pessoas, tabelas de dados, agrupamentos de dados ou qualquer medida de similaridade entre pares de objetos.

Os algoritmos MDS tornaram-se populares desde a aparição do primeiro programa de computador para MDS elaborado por Kruskal em 1964 o qual apresenta uma solução analítica que não requer iterações (COX; COX, 2000). O principal ganho de MDS é que esta técnica pode ser utilizada tendo variáveis de qualquer escala enquanto que em outras técnicas são usadas unicamente variáveis de razão ou intervalares.

Entre alguns dos algoritmos de MDS que são baseados em otimização têm-se: *Simulated Annealing* e *Iterative Majorization*. O algoritmo *Simulated Annealing*, motivado pelo artigo publicado por Metropolis *et al.* 1953, apud Moins 2002², é usado para resolver problemas de otimização não-lineares. O algoritmo *Iterative Majorization* (IM) proposto por Leeuw 1977, apud Borg e Groenen 2005 apresenta

² Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller e E. Teller, (1953). *Equation of State Calculations by Fast Computing Machines*, J. of Chem. Physics, 21 pp 1087 -1092, 1953.

como característica fundamental, a geração de seqüências não crescentes de valores da função de custo que termina geralmente num mínimo local. Uma modificação do IM para o caso de minimização do erro de mapeamento ou *stress* é o algoritmo SMACOF, apresentado no Capítulo 3.

Com a intenção de reduzir o custo computacional e favorecer a escalabilidade em grandes bancos de dados, recentemente surgiram vários algoritmos MDS, entre os quais se destacam o *FastMap*, o *MetricMap* e *LandMark*. O *LandMark* (SILVA; TENENBAUM, 2003a), realiza o mapeamento a partir de um pequeno sub-conjunto de pontos conhecidos como “*landmarks*” num processo de triangulação. O *FastMap* (FALOUTSOS; LIN, 1995) está baseado em projeções sucessivas sobre linhas traçadas entre os pares de objetos mais distantes (*pivots*). O algoritmo *FastMap* pode ser interpretado como um caso particular de *LandMark* onde só são usados 2 pontos *landmark*. O algoritmo *MetricMap* (WANG *et al.*, 2005) tenta criar uma única projeção por vez, usando a generalização de *Nyström* com diferentes tamanhos de submatrizes (PLATT, 2005).

Outras classes de algoritmos MDS estão baseadas no modelo mola-massa (*spring-mass*) (SILVA; TENENBAUM, 2003a) as quais calculam coordenadas em baixa dimensão realizando iterativas minimizações de uma função de custo ou *stress*. Aproximando a distância entre as coordenadas originais e a matriz de distância gerada. Variantes desse método combinam estratégias baseadas em forças e interpolação no algoritmo *Force Directed Placement* (FDP) (FRUCHTERMAN; REINGOLD, 1991). Neste algoritmo cada objeto é associado a algum outro por uma força, que é proporcional à distância calculada no espaço original; gerando-se no procedimento uma complexidade de $O(N^3)$. A desvantagem desses modelos baseados em molas, em geral, é que estão sujeitos a mínimos locais, e que requerem uma suposição *a priori* dos dados estão subjacentes em dimensões, por esta razão eles não são considerados no estudo desta dissertação.

Atualmente tem-se criado algoritmos MDS voltados à análise de dados difusos, entre eles têm-se o algoritmo *Fuzzy multidimensional scaling* (HEBERT *et al.*, 2006) que estende o modelo MDS para o caso onde as dissimilaridades são expressas como

intervalos ou números difusos. O algoritmo *I-Scal* (GROENEN *et al.*, 2006) é utilizado nos casos onde se tem conhecimento de intervalos de dissimilaridade e não se conhece uma medida exata de dissimilaridade, modelando assim um range entre distâncias. Esses algoritmos não são estudados neste trabalho porque os dados que são objetos de estudo deste trabalho não apresentam estas características.

2.2. VISUALIZAÇÃO DE INFORMAÇÃO

As técnicas de redução dimensional como MDS obtêm como saída um conjunto de vetores de coordenadas definidos em um espaço, no qual se tenta preservar ao máximo as proximidades das distâncias originais. Estes vetores podem ser representados graficamente usando formas estruturadas, adequadas para sua visualização em computador. Essas estruturas devem existir nos dados originais ou devem ser derivados dos mesmos, de tal forma que se possa inferir conhecimento a partir dessas estruturas gráficas. Segundo Card *et al.* (1999) “o propósito da visualização é a percepção e não as figuras”; sendo que os principais objetivos dessa percepção são a descoberta de informações, a tomada de decisões e o entendimento do que representam os dados.

A visualização de dados utiliza um espaço de trabalho onde o usuário está familiarizado, e apesar das saídas gráficas comuns serem bidimensionais, é possível reproduzir espaços tridimensionais ou multidimensionais recorrendo a técnicas de visualização de informações. Por exemplo, para visualizar acima de 3 dimensões, são usados elementos como cores, formas, orientações, superfícies de textura, codificação de movimento e codificação de cintilação entre outras. Em Schroeder *et al.* (2001) destaca-se o fato de oito dimensões serem provavelmente o máximo número de dimensões que pode ser representado diretamente.

Entre as técnicas de visualização mais conhecidas e que são comumente utilizadas para representar os dados multidimensionais a partir de coordenadas, destacam-se: a visualização por nuvem de pontos, coordenadas paralelas e visualização por Ícones.

A nuvem de pontos (*Scatterplot*) é a técnica de representação mais usada tanto

para a visualização de ocorrências de dados com dois ou três atributos quanto para mostrar os valores de uma variável conhecida em determinadas posições num espaço 2D ou 3D. No entanto existem duas limitações associadas a esta técnica. A primeira refere-se a que apenas é possível visualizar tuplas de 2 ou 3 dimensões e não todas as dimensões simultaneamente. A segunda limitação aparece quando há um grande aumento no número de dados, o que torna as representações menos efetivas porque estas ficam com uma densidade muito alta de pontos (INSELBERG, 1985; INSELBERG; DIMSDALE, 1990; CARMO, 2003).

A Figura 2.1 exemplifica a visualização mediante a técnica de nuvem de pontos do banco de dados Cevada (*Data Barley*) descrita em Becker *et al.* (1996), a qual apresenta os resultados de um experimento de crescimento de cevada.

Figura 2.1: Exemplo de visualização por nuvens de pontos do banco de dados Cevada³, representando as combinações entre 6 cidades e 10 variedades de cevada que cresceram em 2 anos ($6 \times 10 \times 20 = 120$ observações).

A visualização por Coordenadas Paralelas está baseada no mapeamento de um espaço n -dimensional, em uma estrutura bidimensional que utiliza n eixos

³ Figura obtida de Causeway Graphical Systems

URL: <http://www.causeway.co.uk/tutorial/rainpro/tutor/apl2000/chcloud.htm>.

eqüidistantes denominados coordenadas. Os eixos verticais representam as dimensões ou atributos de dados. Uma linha representa cada item de dado conectado aos eixos com os seus respectivos valores, permitindo a visão de padrões. Uma vantagem desta visualização é a representação de todos os atributos numa mesma visualização, permitindo fazer interpretações visuais entre os atributos. No entanto esta mesma vantagem associa-se a uma limitação referida ao aumento do número de atributos que impossibilitam o entendimento dos mesmos (NASCIMENTO; FERREIRA, 2005). A Figura 2.2 ilustra um exemplo de visualização mediante esta técnica.

Figura 2.2: Exemplo de visualização por coordenadas paralelas, adaptado de Nascimento e Ferreira (2005).

A visualização mediante a técnica de Ícones (*Glyphs*) cria um objeto gráfico que contém dados de múltiplos valores. Com esta técnica consegue-se representar combinações de tamanho, forma ou cor, entre outras; para identificar um dado (SCHROEDER *et al.*, 2001). Exemplos populares dessa técnica são: *Chernoff faces* (CHERNOFF, 1973), *Data Jacks* (COX, 1990), *Autoglyph* (BEDDOW, 1990), *Stick figures* (PICKETT, 1970; PICKETT; GRINSTEIN, 1988) e *Color Icons* (LEVKOWITZ, 1991). A Figura 2.3 ilustra um exemplo de visualização para classificação de situações econômicas entre cidades.

Figura 2.3: Exemplo de visualização utilizando a técnica de *Chernoff faces*, adaptado de Müller e Alexa, (1998).

A visualização por nuvens de pontos pode ser considerada como um caso particular da técnica *Glyph*, já que um ponto pode ser visto como um elemento gráfico cuja forma, cor ou orientação traduzem os valores a representar. Em razão disto é que esta visualização foi utilizada pela ferramenta criada nesta dissertação, a qual é descrita no Capítulo 4, porque permite a inserção de propriedades visuais (cor, tamanho, forma, orientação, entre outros) que aumentam o número de dimensões que podem ser representados.

Capítulo 3

FUNDAMENTOS TEÓRICOS

Neste capítulo são apresentados os fundamentos teóricos necessários ao entendimento dos algoritmos de MDS.

3.1. CONCEITO DE ESCALONAMENTO MULTIDIMENSIONAL

O Escalonamento Multidimensional do inglês “*MultiDimensional Scaling*” (MDS) é comumente usado em duas abordagens com diferente significado (LEEUEW; HEISER, 1982 apud NAUD, 2001, p. 40). O primeiro significado (sentido estreito) refere-se ao MDS como uma família de técnicas dirigidas a representar medidas de proximidade¹ entre objetos em um espaço com baixa dimensão (COX; COX, 2000; BORG; GROENEN, 2005; GROENEN; VELDEN, 2004). O segundo significado de MDS (sentido amplo) refere-se às técnicas que geram uma representação gráfica de objetos. Esta definição inclui várias formas de análise de agrupamentos (*clustering*) ou métodos de análise estatística como PCA ou *Correspondence Analysis* (CA) (RIPLEY, 1996; NAUD, 2001).

A abordagem em MDS pode ser definida como: “Dada uma matriz de dissimilaridade $\Delta = (\delta_{ij}) \in \Re^{n,n}$, onde δ_{ij} representa uma medida de proximidade entre os objetos i e j . Um algoritmo de redução dimensional MDS deve obter uma configuração de pontos (vetores de coordenadas) $\mathbf{X} = (x_1, \dots, x_n) \in \Re^{n \times k}$ em uma

¹ Para indicar medidas de similaridade ou dissimilaridade entre dados utiliza-se indistintamente o termo genérico “proximidade”.

dimensão menor² ($k < n$), nos quais se deve verificar que a matriz de distância euclidiana $\mathbf{D} = (d_{ij}) \in \mathfrak{R}^{n \times n} / d_{ij} = \|x_i - x_j\|$, obtida a partir desse conjunto de pontos, se aproxime ao máximo à matriz de dissimilaridade original, isto é, $\mathbf{D} \approx \mathbf{\Delta}$.

Uma representação descritiva do problema em MDS é mostrada na Figura 3.1, onde para uma matriz de dissimilaridade $\mathbf{\Delta}$ em $\mathfrak{R}^{n \times n}$, através de um algoritmo MDS, obtém-se uma matriz de coordenadas $\mathbf{X} \in \mathfrak{R}^{n \times k}$, $k < n$. A linha i -ésima da matriz $\mathbf{X}$ representa o vetor de coordenadas do objeto i . A distância euclidiana d_{ij} calculada entre as linhas i e j aproxima-se da dissimilaridade entre os objetos i e j , ou seja, $d_{ij} \approx \delta_{ij}$.

Figura 3.1: Representação do problema MDS.

Na Figura 3.2 apresenta-se a validação feita em MDS, na qual, a partir da matriz de coordenadas $\mathbf{X}$, pode-se obter uma matriz de distância $\mathbf{D} \in \mathfrak{R}^{n \times n}$. Usa-se uma função de *stress* para medir a diferença entre a matriz de dissimilaridade original e a matriz de distância calculada.

Figura 3.2: Representação do proceso de validação em MDS.

² Encontrar o valor mais adequado para k tem sido definido como o problema associado a redução de dimensão.

É importante destacar que nem toda matriz distância pode ser visualizada em um espaço euclidiano. Por exemplo, na Figura 3.3, os pontos A, B, e C têm as mesmas distâncias e o ponto D está no centro do triângulo formado (ver Figura 3.3a). Considerando uma matriz distância onde os pontos A, B e C também são todos eqüidistantes, enquanto que as distâncias de A, B, e C para D são levemente menores que no desenho anterior (ver Figura 3.3b) então a matriz de distância satisfaz todos os requisitos de uma métrica de distância, mas é impossível desenhar essas distâncias. Isto é encontrar a localização do ponto D, em qualquer espaço euclidiano (SCHROEDER *et al.*, 2001).

Figura 3.3. Considerações em MDS: (a) Apresenta solução em MDS, (b) Não apresenta solução exata em MDS.

É importante destacar que independentemente da escolha do algoritmo de MDS, é a determinação do número de dimensões, tarefa determinante para seu bom desempenho. Caso seja escolhido um grande número de dimensões o erro decresceria, mas o resultado não poderia ser facilmente interpretado. No entanto, escolhendo-se um número pequeno de dimensões pode ser que não seja suficiente para revelar as estruturas latentes do conjunto de dados.

3.2. FUNÇÃO DE MAPEAMENTO E MODELOS MDS

Um mapeamento MDS é definido por uma função $f : p_{ij} \rightarrow d_{ij}(\mathbf{X})$ que especifica como as proximidades p_{ij} aproximam-se das distâncias $d_{ij}(\mathbf{X})$ de um espaço $\mathbf{X}$. O termo $f(p_{ij})$ denota um mapeamento ou transformação de p_{ij} . Assim, um modelo MDS é uma proposição na qual a partir de medidas de proximidades e depois de alguma transformação de f obtêm-se distâncias entre pontos em $\mathbf{X}$:

$$f(p_{ij}) = d_{ij}(\mathbf{X}) \quad (3.1)$$

Usualmente, os modelos MDS necessitam que cada valor de proximidade seja mapeado exatamente em suas correspondentes distâncias (BORG; GROENEN, 2005). Na prática, usualmente não se tenta estritamente satisfazer f , mas procura-se satisfazer uma configuração (em uma dimensão de baixa ordem), onde as distâncias obtidas aproximam-se de f tanto quanto seja possível. A condição “tão próximo quanto possível” é quantificada por uma medida de ajuste entre as distâncias $\{d_{ij}\}$ e as dissimilaridades $\{\delta_{ij}\}$. Existem diferentes definições para nomear esta medida, comumente chamada de *stress*.

Segundo Borg e Groenen (2005) os modelos de MDS podem genericamente ser classificados em dois tipos dependendo da natureza dos objetos observados: MDS métrico e MDS não-métrico. O modelo MDS métrico é usado quando a transformação aplicada às similaridades (ou dissimilaridades) preserva as propriedades métricas das distâncias porque pertence a escalas de razão ou intervalo (NAUD, 2001). O modelo MDS não-métrico é usado quando unicamente tem-se a informação de dados avaliados segundo uma escala ordinal. Aqui, os algoritmos MDS não têm que tratar de reproduzir as dissimilaridades, unicamente sua ordem. Este modelo não-métrico não pressupõe uma relação linear entre as proximidades e as distâncias, mas estabelece uma relação monotônica entre ambas.

3.3. MEDIDAS DE DISTÂNCIA

Os dados geralmente são de diversas fontes como imagens, textos ou conjuntos de impressões digitais. As informações contidas nesses dados, muitas vezes, não possuem ordem ou grandeza explícita e para tanto é necessário utilizar uma forma de mensurá-los de acordo com sua natureza.

A proximidade p_{ij} entre um par de objetos, numa coleção de dados, pode ser expressa como sua medida de similaridade, consenso, dissimilaridade ou correlação de suas distâncias (BASALAJ, 2000). Os objetos, normalmente, são descritos por vetores de características $\mathbf{X} = (x_1, x_2, \dots, x_n)$, ou podem ser definidos por alguma medida de

proximidade entre objetos.

As medidas de distância consideram os atributos dos objetos como dimensões de um espaço multidimensional e cada objeto como sendo um ponto no espaço multidimensional. Essas medidas avaliam a dissimilaridade (δ_{ij}) entre os objetos pois quanto maior o valor calculado (maior distância), menor o grau de semelhança entre os objetos; e quanto menor a distância, maior a similaridade (s_{ij}) entre os objetos (DAMIANCE, 2006).

Entre algumas das medidas de distância mais usadas em MDS estão:

$$\text{Distância City Block:} \quad d(\mathbf{x}_i, \mathbf{x}_j) = \sum_{a=1}^M |x_{ia} - x_{ja}| \quad (3.2)$$

$$\text{Distância Mahalanobis:} \quad d(\mathbf{x}_i, \mathbf{x}_j) = \sum_{a=1}^M \left((x_{ia} - x_{ja})^T P^{-1} (x_{ia} - x_{ja}) \right)^{\frac{1}{2}} \quad (3.3)$$

$$\text{Distância Chebyshev:} \quad d(\mathbf{x}_i, \mathbf{x}_j) = \max_{a=1}^M |x_{ia} - x_{ja}| \quad (3.4)$$

$$\text{Distância Euclidiana:} \quad d(\mathbf{x}_i, \mathbf{x}_j) = \left(\sum_{a=1}^M (x_{ia} - x_{ja})^2 \right)^{\frac{1}{2}} \quad (3.5)$$

onde $d(\mathbf{x}_i, \mathbf{x}_j)$ indica a distância entre os vetores $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{iM})$ e $\mathbf{x}_j = (x_{j1}, x_{j2}, \dots, x_{jM})$ que são descritos pelos respectivos valores dos M atributos. O termo P representa uma matriz de covariância para um vetor $\mathbf{x} = (x_1, x_2, \dots, x_M)^T$ (EVERITT *et al.*, 2001).

A distância Euclidiana é a medida de distância usualmente selecionada, não sendo a única possível. Por exemplo, as distâncias *City-block*, *Chebyshev* e *Malahanobis* podem ser mais apropriadas em alguns casos, onde os atributos dos dados não são numéricos ou contínuos, correspondendo a dados simbólicos ou binários.

Em situações onde os dados são descritos por funções (contínuas ou discretas) pode ser usada uma medida de correlação. Uma das medidas de correlação mais usada é o *Coeficiente de Correlação de Pearson*, o qual representa uma medida de intensidade da associação entre duas variáveis quantitativas. Por exemplo, O_i e O_j , as

quais não precisam ter uma dependência definida. Os valores nesse coeficiente variam entre -1 e 1, sendo que o valor 0 (zero) significa que não há relação linear, o valor 1 indica uma relação linear perfeita e o valor -1 também indica uma relação linear perfeita, mas inversa. Quanto mais próximo estiver de 1 ou -1 mais forte é a associação linear entre as duas variáveis (BORG; GROENEN, 2005).

O coeficiente de correlação de *Pearson* (eq. 3.6) (DUDA *et al.*, 2001) é normalmente representado pela letra “ r ” e a sua equação de cálculo é:

$$r_{x,y} = \frac{\sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y})}{\left(\sum_{i=1}^N (x_i - \bar{x})^2\right)^{1/2} \left(\sum_{i=1}^N (y_i - \bar{y})^2\right)^{1/2}} \quad (3.6)$$

onde $\bar{x}$ e $\bar{y}$ são as respectivas médias de x e y , calculadas sobre N observações.

3.4. MATRIZES DE DISTÂNCIA

Em MDS existem 3 tipos de matrizes comumente utilizadas nos modelos MDS: matriz de distância euclidiana, matriz de similaridade e matriz de dissimilaridade. Essas matrizes são apresentadas a seguir.

Uma matriz de dissimilaridade $\Delta = [\delta_{ij}]$ de $n \times n$ é uma matriz quadrada e simétrica onde seus elementos representam medidas de proximidade. Nessa matriz os valores de cada entrada δ_{ij} são positivos e indicam a dissimilaridade entre os objetos (i, j) . Os valores da diagonal principal são iguais a zero.

Se os dados são apresentados em forma de vetores de características é preciso definir uma função de dissimilaridade entre pares de objetos. Assim, uma matriz de dissimilaridade entre dois vetores é definida como a soma das dissimilaridades dos seus atributos: $\Delta(x_i, x_j) = \sum_p \Delta(x_{ip}, x_{jp})$. Podem-se usar pesos w , quando os atributos têm diferenças importantes, assim $\Delta(x_i, x_j) = \sum_p w \Delta(x_{ip}, x_{jp})$ (BORG; GROENEN, 2005).

Em uma matriz de similaridade $S = [s_{ij}]$ de $n \times n$ os seus elementos representam medidas de proximidade que indicam quão similares são dois objetos. Esta matriz

apresenta as mesmas propriedades que a matriz de dissimilaridade.

Geralmente, em lugar de se determinar as dissimilaridades entre objetos calculam-se as suas similaridades. As medidas de correlações, por exemplo, podem ser interpretadas como similaridades entre objetos. Os modelos MDS aceitam medidas de similaridade através de uma conveniente transformação para medidas de dissimilaridade. A seguinte equação é usada geralmente para conseguir esta transformação:

$$\delta_{ij} = 1 - s_{ij} \quad (3.7)$$

ambos os símbolos δ_{ij} e s_{ij} também são chamados índices de proximidade (GROENEN; VELDEN, 2004).

Uma matriz de distância Euclidiana $\mathbf{D} = [d_{ij}] \in \mathbb{R}^n$ é uma matriz de dissimilaridade em que seus elementos d_{ij} representam a norma euclidiana entre os elementos i e j (BORG; GROENEN, 2005). Esta matriz é formada a partir de um conjunto de dados, representados por uma matriz $\mathbf{X} = [\mathbf{x}_{ij}] \in \mathbb{R}^{n \times m}$, na qual os índices i, j indicam o número de observações e dimensões de cada dado respectivamente. Para calcular as distâncias entre os elementos de $\mathbf{X}$, utiliza-se a equação (3.8) genérica:

$$d_{ij}^2(\mathbf{X}) \equiv \mathbf{D}^{(2)}(\mathbf{X}) = \sum_{a=1}^m (x_{ia} - x_{ja})^2 \quad (3.8)$$

Um desenvolvimento da equação (3.8) de três dados (x_1, x_2, x_3) definidos em duas dimensões é:

$$\mathbf{D}^{(2)}(\mathbf{X}) = \sum_{a=1}^m \begin{bmatrix} x_{1a}^2 & x_{1a}^2 & x_{1a}^2 \\ x_{2a}^2 & x_{2a}^2 & x_{2a}^2 \\ x_{3a}^2 & x_{3a}^2 & x_{3a}^2 \end{bmatrix} + \sum_{a=1}^m \begin{bmatrix} x_{1a}^2 & x_{2a}^2 & x_{3a}^2 \\ x_{1a}^2 & x_{2a}^2 & x_{3a}^2 \\ x_{1a}^2 & x_{2a}^2 & x_{3a}^2 \end{bmatrix} - 2 \sum_{a=1}^m \begin{bmatrix} x_{1a}^2 & x_{1a}x_{2a} & x_{1a}x_{3a} \\ x_{2a}x_{1a} & x_{2a}^2 & x_{2a}x_{3a} \\ x_{3a}x_{1a} & x_{3a}x_{2a} & x_{3a}^2 \end{bmatrix}$$

Assim, uma matriz de distância Euclidiana $\mathbf{D}^{(2)}(\mathbf{X})$ pode ser generalizada na seguinte equação (BORG; GROENEN, 2005):

$$\mathbf{D}^{(2)}(\mathbf{X}) = \mathbf{c}\mathbf{1}^T + \mathbf{1}\mathbf{c}^T - 2\sum_{a=1}^m \mathbf{x}_a \mathbf{x}_a^T = \mathbf{c}\mathbf{1}^T + \mathbf{1}\mathbf{c}^T - 2\mathbf{X}\mathbf{X}^T \quad (3.9)$$

onde $\mathbf{x}_a$ representa uma coluna a da matriz $\mathbf{X}$, $\mathbf{1}$ representa um vetor de elementos uns, e $\mathbf{c}$ representa um vetor com elementos igual a $\sum_{a=1}^m \mathbf{x}_{ia}^2$ da diagonal $\mathbf{X}\mathbf{X}^T$ (BORG; GROENEN, 2005). A generalização da equação (3.9) é apresentada na matriz seguinte:

$$\mathbf{D}(\mathbf{X}) = \begin{bmatrix} 0 & d_{12}^2 & \dots & d_{1n}^2 \\ d_{12}^2 & 0 & & d_{2n}^2 \\ \vdots & \vdots & \ddots & \vdots \\ d_{n1}^2 & d_{n2}^2 & \dots & 0 \end{bmatrix}_{n \times n} \quad (3.10)$$

3.5. FUNÇÃO STRESS

Os modelos de MDS requerem que cada valor de proximidade seja mapeado exatamente em suas correspondentes distâncias. As proximidades empiricamente sempre contêm ruído devido à imprecisão na medição, erros de amostragem, etc. Usualmente, não se consegue satisfazer $f(p_{ij}) = d_{ij}(\mathbf{X})$ mas a melhor aproximação possível ($f(p_{ij}) \approx d_{ij}(\mathbf{X})$) pode ser quantificada por uma medida de ajuste entre as distâncias $\{d_{ij}\}$ e $f(p_{ij})$ denotada como função de *stress*. Quanto menor for o valor do *stress* menor será a distorção causada pelo mapeamento (BORG; GROENEN, 2005).

A função *stress* (equação 3.11) é uma expressão que representa a soma de todos os erros sobre os pares (i, j) . Essa informação pode perder exatidão em uma representação MDS por causa da dependência da normalização de suas dissimilaridades.

$$e_{ij}^2 = \sigma_r(\mathbf{X}) = \sum_{(i,j)} [f(p_{ij}) - d_{ij}(\mathbf{X})]^2 \quad (3.11)$$

Entre as diferentes medidas de *stress* propostas para normalizar a dependência da escala dos valores usados nesta dissertação têm-se:

- *Stress I*, introduzido por Kruskal (1964a), apud Borg e Groenen (2005), mede a diferença entre a raiz quadrada das medidas de distâncias e das medidas das

medidas de dissimilaridades.

$$Stress - 1 = \sqrt{\frac{\sum_{i < j} [\delta_{ij} - d_{ij}(\mathbf{X})]^2}{\sum_{i < j} d_{ij}(\mathbf{X})}} \quad (3.12)$$

- *Stress 2* também proposto por Kruskal (1964a), apud Borg e Groenen (2005), agrega uma forma diferente de normalização implícita aumentado no denominador o quadrado da diferença entre a medida de distância e a média das distâncias representada por $\bar{d}$.

$$Stress - 2 = \sqrt{\frac{\sum_{i < j} [\delta_{ij} - d_{ij}(\mathbf{X})]^2}{\sum_{i < j} [d_{ij}(\mathbf{X}) - \bar{d}]^2}} \quad (3.13)$$

- *S-Stress*, introduzida por Takane *et al.* (1977), apud Borg e Groenen (2005) mede a diferença entre a raiz quadrada dos quadrados das medidas de distâncias e das medidas de dissimilaridades.

$$S - Stress = \sqrt{\sum_{i < j} [d_{ij}^2(\mathbf{X}) - \delta_{ij}^2]^2} \quad (3.14)$$

3.6. ALGORITMOS MDS

Nesta dissertação são estudados os algoritmos: CMDS como algoritmo básico na criação do MDS, FASTMAP, LANDMARK como algoritmos rápidos baseados em *Nyström*, e SMACOF como um algoritmo de otimização. Existem diversos outros algoritmos que não são estudados aqui, por não serem relevantes à abordagem deste trabalho ou por terem um alto custo computacional segundo o pesquisado na literatura.

ALGORITMO CMDS

O algoritmo *Classical Multidimensional Scaling* (CMDS) proposto por Torgeson em 1952 (BORG; GROENEN, 2005), é um algoritmo clássico de MDS. Esse algoritmo consiste em assumir que como entrada existe uma única matriz de dissimilaridade e produz como saída uma matriz $\mathbf{X}$ de dimensões que explicam essas distâncias (GROENEN, 1997).

Nesse algoritmo realiza-se um processo conhecido como “*double centering*”, que

consiste em multiplicar a matriz de dissimilaridade Δ , à esquerda e à direita por uma matriz de centralização $\mathbf{J} = \mathbf{I} - \mathbf{n}^{-1}\mathbf{1}\mathbf{1}^T$, e pelo fator $-1/2$, obtendo-se $-\frac{1}{2}\mathbf{J}\Delta^2\mathbf{J}$.

Desenvolvendo a matriz de dissimilaridade como matriz de distância euclidiana (ver eq. 3.9) obtém-se:

$$-\frac{1}{2}\mathbf{J}\Delta^2\mathbf{J} = -\frac{1}{2}\mathbf{J}(\mathbf{c}\mathbf{1}^T + \mathbf{1}\mathbf{c}^T - 2\mathbf{X}\mathbf{X}^T)\mathbf{J} \quad (3.15)$$

$$= -\frac{1}{2}\mathbf{J}\mathbf{c}\mathbf{0}^T - \frac{1}{2}\mathbf{0}\mathbf{c}^T\mathbf{J} + \mathbf{J}\mathbf{X}\mathbf{X}^T\mathbf{J} \quad (3.16)$$

os dois primeiros termos podem ser removidos devido a que um vetor de uns duplamente centrado aproxima-se a um vetor de zeros $\mathbf{1}^T\mathbf{J} = \mathbf{0}$.

$$-\frac{1}{2}\mathbf{J}\Delta^2\mathbf{J} = \mathbf{J}\mathbf{X}\mathbf{X}^T\mathbf{J} \quad (3.17)$$

O *double centering* de $\mathbf{X}\mathbf{X}'$ também pode ser removido porque se assume que $\mathbf{X}$ é uma matriz que tem colunas com média igual a zero (matriz centrada) (BORG; GROENEN, 2005) e assim:

$$\mathbf{B} = \mathbf{X}\mathbf{X}^T \quad (3.18)$$

finalmente realiza-se a auto-decomposição da matriz $\mathbf{B}$ para obter a matriz solução dos pontos.

Esse algoritmo é popular porque apresenta uma solução analítica e não requer iterações, resultando em um custo computacional de $O(N^3)$, onde N é o número de dados. Devido à combinação do seu procedimento também é conhecido como *Principal Coordinate Analysis* (PCO) (BASALAJ, 2000). A Figura 3.4 apresenta o pseudocódigo do algoritmo CMDS.

CMDS (Δ^2)**Dados:** Δ^2 matriz de dissimilaridade**Saída:** matriz $\mathbf{X}$ de coordenadas**Início**

1. Calcular a matriz de centralização $\mathbf{J}$, onde n denota o número de linhas e $\mathbf{1}$ é um vetor de uns de tamanho n .

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

2. Realizar o processo de “*double centering*” à matriz de dissimilaridade

$$\mathbf{B} = -\frac{1}{2} \mathbf{J} \Delta^2 \mathbf{J}$$

3. Realizar auto-decomposição de $\mathbf{B}$, onde Λ representa os autovalores e $\mathbf{Q}$ denota os autovetores.

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

4. Considerando m a dimensionalidade da solução. Denota-se Λ_+ os autovalores com valores maiores a zero e $\mathbf{Q}_+$ representando as primeiras m colunas de $\mathbf{Q}$, obtendo-se assim, a partir de $\mathbf{B}$ uma matriz definida positiva. No CMDS os autovalores e autovetores negativos são ignorados como erro.

5. Obter a matriz de coordenadas $\mathbf{X}$

$$\mathbf{X} = \mathbf{Q}_+ \Lambda_+^{1/2}$$

fim

Figura 3.4: Pseudocódigo do algoritmo CMDS.

ALGORITMO SMACOF

Devido a que os algoritmos existentes para resolver o problema de MDS não garantem a obtenção de mínimos globais, nos últimos anos vêm-se aplicando métodos de Otimização Estocástica que conseguem uma convergência ao ótimo global, entre os mais conhecidos têm-se o *Iterative Majorization* (IM). A idéia geral do IM consiste em substituir iterativamente uma função a minimizar $f(x)$ por uma função auxiliar $g(x, z)$ onde z é uma constante. A Figura 3.5 exemplifica duas iterações da função $f(x)$ com o algoritmo IM. A primeira iteração inicia encontrando uma função auxiliar $g(x, x_0)$ localizada acima da função $f(x)$ e que toca o ponto de suporte x_0 . O ponto mínimo da função $g(x, x_0)$ é alcançado por x_1 , onde $f(x_1)$ nunca pode ser maior do que $g(x_1, x_0)$,

este passo completa a primeira iteração. A segunda iteração segue os mesmos passos que a primeira.

Figura 3.5: Exemplo de 2 iterações do método *Iterative Majorization* adaptado de (BORG; GROENEN, 2005).

O algoritmo SMACOF (*Scaling by MAjorizing a COnvex Function*) também conhecido como “*Transformação de Guttman*” está baseado no algoritmo IM para reduzir uma função de *stress* (*Majorization stress*) utilizando a transformada de Guttman (BORG; GROENEN, 2005). Algoritmos anteriores usaram o método do gradiente descendente (KRUSKAL, 1964 e GUTTMAN, 1968 apud BORG; GROENEN, 2005), no entanto a teoria de SMACOF é mais simples e poderosa porque garante convergência monótona do *stress* (BORG; GROENEN, 2005).

Segundo Borg e Groenen (2005) a regra de atualização de SMACOF segue os seguintes passos:

Dada uma função de *stress* $\sigma_r(\mathbf{X})$ a minimizar

$$\begin{aligned}
 \sigma_r(\mathbf{X}) &= \sum_{i < j} w_{ij} (\delta_{ij} - d_{ij}(\mathbf{X}))^2 \\
 &= \sum_{i < j} w_{ij} \delta_{ij}^2 + \sum_{i < j} w_{ij} d_{ij}^2(\mathbf{X}) - 2 \sum_{i < j} w_{ij} \delta_{ij} d_{ij}(\mathbf{X}) \\
 &= \eta_\delta^2 + \eta^2(\mathbf{X}) - 2\rho(\mathbf{X})
 \end{aligned} \tag{3.19}$$

onde $d_{ij}(\mathbf{X})$ é a distância euclidiana. A partir da equação 3.19 pode-se observar que o

stress pode ser decomposto em três partes. A primeira parte η_δ^2 depende unicamente dos valores constantes de w_{ij} e das dissimilaridades δ_{ij} e não dependem de $\mathbf{X}$, então η_δ^2 se converte em uma constante. A segunda parte $\eta^2(\mathbf{X})$ corresponde à soma dos pesos das distâncias quadradas de $d_{ij}^2(\mathbf{X})$. A parte final, $-2\rho(\mathbf{X})$ corresponde à soma das distâncias $d_{ij}(\mathbf{X})$. Assume-se que os pesos da matriz de pesos $\mathbf{W}$ são irredutíveis.

Os elementos $\eta^2(\mathbf{X})$ podem ser expressos como:

$$\eta^2(\mathbf{X}) = \text{tr}(\mathbf{X}^T \mathbf{V} \mathbf{X}) \quad (3.20)$$

$$\mathbf{V} = \sum_{i < j} w_{ij} \mathbf{A}_{ij} \quad (3.21)$$

onde $\mathbf{A}_{ij}$ é uma matriz centrada com elementos $a_{ii} = a_{jj} = 1$, $a_{ij} = a_{ji} = -1$ e os outros elementos iguais a zero; $\mathbf{V}$ corresponde à soma dos pesos das linhas e colunas da matriz centrada $\mathbf{A}_{ij}$, conseqüentemente $\mathbf{V}$ também é uma matriz centrada. Devido ao que foi assumido, onde os pesos w_{ij} são irredutíveis, então, o posto de $\mathbf{V}$ é $n-1$ e os autovalores zero correspondem aos autovetores $n^{-1/2} \mathbf{1}$.

Os elementos $-\rho(\mathbf{X})$ podem ser expressos como:

$$-\rho(\mathbf{X}) = -\text{tr}(\mathbf{X}^T \mathbf{B}(\mathbf{Z}) \mathbf{Z}) \quad (3.22)$$

onde a matriz $\mathbf{B}(\mathbf{Z})$ apresenta os elementos:

$$b_{ij} = \begin{cases} -\frac{w_{ij} \delta_{ij}}{d_{ij}(\mathbf{Z})} & \text{Para } i = j \text{ e } d_{ij}(\mathbf{Z}) \neq 0 \\ & \text{Para } i \neq j \text{ e } d_{ij}(\mathbf{Z}) = 0 \end{cases}$$

$$b_{ij} = - \sum_{j=1, j \neq i}^n b_{ij} \quad (3.23)$$

Combinando (3.20) e (3.22) na equação (3.19) obtêm-se em a função de *stress* para o algoritmo SMACOF:

$$\begin{aligned} \sigma_r(\mathbf{X}) &= \eta_\delta^2 + \text{tr}(\mathbf{X}^T \mathbf{V} \mathbf{X}) - 2\text{tr}(\mathbf{X}^T \mathbf{B}(\mathbf{X}) \mathbf{Z}) \\ &\leq \eta_\delta^2 + \text{tr}(\mathbf{X}^T \mathbf{V} \mathbf{X}) - 2\text{tr}(\mathbf{X}^T \mathbf{B}(\mathbf{X}) \mathbf{Z}) = \tau(\mathbf{X}, \mathbf{Z}) \end{aligned} \quad (3.24)$$

onde $\tau(\mathbf{X}, \mathbf{Z})$ é uma função de *stress* que é quadrática em $\mathbf{X}$, seu mínimo pode ser

alcançado analiticamente igualando a zero a derivada de $\tau(\mathbf{X}, \mathbf{Z})$ com respeito a $\mathbf{X}$:

$$\frac{\partial \tau(\mathbf{X}, \mathbf{Z})}{\partial \mathbf{X}} = 2\mathbf{V}\mathbf{X} - 2\mathbf{B}(\mathbf{Z})\mathbf{Z} = 0 \quad (3.25)$$

de tal forma que $\mathbf{V}\mathbf{X} = \mathbf{B}(\mathbf{Z})\mathbf{Z}$. O sistema de equações lineares para $\mathbf{X}$ pode ser alcançado pré-multiplicando ambos os lados por $\mathbf{V}^{-1}$. Entretanto, a inversa de $\mathbf{V}^{-1}$ não existe devido a que $\mathbf{V}$ é singular, pelo que, para resolver o sistema tem-se que utilizar a inversa generalizada.

$$\mathbf{V}^+ = (\mathbf{V} + \mathbf{1}\mathbf{1}^T)^{-1} - n^{-2}\mathbf{1}\mathbf{1}^T \quad (3.26)$$

O último termo ($n^{-2}\mathbf{1}\mathbf{1}^T$) não é importante em SMACOF devido a que $\mathbf{V}^+$ é subsequêntemente multiplicado por uma matriz ortogonal de $\mathbf{1}$, e como $\mathbf{B}(\mathbf{Z})\mathbf{Z}$ tem um vetor próprio $\mathbf{1}$ com valor próprio 0. Isto permite atualizar a fórmula do algoritmo SMACOF.

$$\mathbf{X}^u = \mathbf{V}^+\mathbf{B}(\mathbf{Z})\mathbf{Z} \quad (3.27)$$

Quando todos os pesos $w_{ij} = 1$, os valores de $\mathbf{V}^+ = n^{-1}\mathbf{J}$, onde $\mathbf{J}$ é uma matriz de centralização ($\mathbf{J} = \mathbf{I} - n^{-1}\mathbf{1}\mathbf{1}^T$), tal que a regra de atualização simplifica-se em:

$$\mathbf{X}^u = n^{-1}\mathbf{B}(\mathbf{Z})\mathbf{Z} \quad (3.28)$$

A equação (3.27) é conhecida como a transformada de *Guttman* (BORG; GROENEN, 2005). O algoritmo IM garante uma serie de valores de *stress* não incrementais. Quando o algoritmo pára, a condição de estacionaridade $\mathbf{X} = \mathbf{V}^+\mathbf{B}(\mathbf{X})\mathbf{X}$ continua. O pseudocódigo do algoritmo SMACOF é apresentado na Figura 3.6.

Figura 3.6: Pseudocódigo do algoritmo SMACOF.

ALGORITMO FASTMAP

O algoritmo FASTMAP proposto por Faloutsos e Lin, (1995) visa otimizar o CMDS respeito ao cálculo da auto-decomposição, apresentando uma solução recursiva que em cada iteração encontra um vetor de coordenadas. O FASTMAP realiza projeções ortogonais iterativas dos pontos sobre linhas denominadas pivôs, as quais são encontradas selecionando os 2 pontos mais distantes de um hiper-plano de

projeção.

A Figura 3.7 ilustra exemplifica o procedimento deste algoritmo para encontra a primeira iteração. Primeiramente são selecionados dois pontos pivôs O_a e O_b , logo se traça uma linha entre estes pontos, denominada de linha pivô e finalmente para encontrar o valor da projeção x_i de um ponto O_i sobre a linha pivô (ponto E), deve-se calcular a distância entre o ponto pivô e o ponto E, seguindo a lei dos co-senos definida pela equação 3.29:

$$d_{b,i}^2 = d_{a,i}^2 + d_{a,b}^2 - 2x_i d_{ab} \quad (3.29)$$

Isolando x_i o valor da primeira coordenada do ponto O_i é calculado diretamente pela equação 3.30:

$$x_i = \frac{d_{a,i}^2 + d_{a,b}^2 - d_{b,i}^2}{2d_{ab}} \quad (3.30)$$

Figura 3.7: Ilustração da projeção do ponto O_i sobre a linha $O_a O_b$. Adaptado de Faloutsos, Lin (1995).

A Figura 3.8 exemplifica o procedimento seguido para projetar as distâncias entre os pontos, sobre um hiperplano perpendicular à linha pivô. Por exemplo, nessa figura é projetada no hiperplano H , a distância entre 2 pontos O_i e O_j , que por sua vez têm projeções sobre a linha pivô, ponto E e ponto D , respectivamente. Note-se que da subtração de estas duas projeções encontra-se um dos catetos do triângulo reto formado pelos pontos O_i , O_j e C . Pelo teorema de Pitágoras encontra-se o outro cateto (linha O_j e C) a qual estaria representando a distância euclidiana D' entre os pontos O_i' e O_j' projetados no plano H . A equação 3.31 generaliza a equação do

teorema de Pitágoras para encontrara a projeção de todos os pontos sobre o hiperplano.

$$(\mathbf{D}'(\mathbf{O}'_i, \mathbf{O}'_j))^2 = (\mathbf{D}(\mathbf{O}_i, \mathbf{O}_j))^2 - (x_i - x_j)^2 \quad i, j = 1, \dots, N \quad (3.31)$$

Figura 3.8: Projeção das distâncias sobre um hiperplano H , perpendicular à linha de pivôs $\mathbf{O}_a \mathbf{O}_b$. Adaptado de Faloutsos e Lin (1995).

Uma vez que todas as distâncias são projetadas no hiperplano pode-se volver ao primeiro procedimento para encontrar a projeção dos pontos sobre uma linha pivô encontrando assim um novo vetor de coordenadas. Aplicando recursivamente os procedimentos anteriores pode-se alcançar um número de dimensões desejadas. O custo computacional deste algoritmo é aproximadamente $O(Nk)$, onde N é o número de dados e k é a dimensionalidade desejada. A Figura 3.39 apresenta o pseudocódigo deste algoritmo.

FASTMAP ($\mathbf{X_out}, \mathbf{P_out}$) = ($k, \mathbf{D}, \mathbf{X}, \mathbf{P}$)

Dados: número k de dimensionalidade desejada, matriz de dissimilaridades $\mathbf{D}$ ingressada como matriz distância, matriz de pontos $\mathbf{X}$ e matriz de pivôs $\mathbf{P}$.

Saída: matriz de pontos $\mathbf{X}$

Início

1. Iniciar n = número de filas da matriz de $\mathbf{D}$

Se ($k \leq 0$) **então**

$\mathbf{X_out} = \mathbf{X}$

$\mathbf{P_out} = \mathbf{P}$

fim

2. Selecionar dois objetos pivôs O_a e O_b (os mais separados)

3. Armazenar os índices dos pivôs.

$\text{pivos} = [O_a; O_b]$

$\mathbf{P_out} = [\mathbf{P} \quad \text{pivos}]$

Se ($\mathbf{D}(O_a, O_b) = 0$) **fazer**

$\mathbf{X_out} = [\mathbf{X} \quad \text{zeros}(n,1)]$ //zeros(n,1) é uma matriz de elementos zeros

fim

4. Projetar todos os objetos sobre a linha O_a e O_b

Para ($O_i = 1 \quad a \quad n$) **fazer**

$$x(O_i) = \frac{\mathbf{D}(O_a, O_i)^2 + \mathbf{D}(O_a, O_b)^2 - \mathbf{D}(O_b, O_i)^2}{2 \times \mathbf{D}(O_a, O_b)}$$

fim

$\mathbf{X_out} = [\mathbf{X} \quad \mathbf{x}']$

5. Obter a projeção dos objetos sobre o plano perpendicular

$$\mathbf{D}'(O_i', O_j')^2 = \mathbf{D}(O_i, O_j)^2 - (x_i - x_j)^2$$

6. Chamar recursivamente a FastMap ate $k = 1$

$[\mathbf{X_out}, \mathbf{P_out}] = \text{FASTMAP}(k-1, \mathbf{D}', \mathbf{X_out}, \mathbf{P_out})$

fim

Figura 3.9: Pseudocódigo do algoritmo FASTMAP.

ALGORITMO LANDMARK

O algoritmo LANDMARK proposto por Silva e Tenenbaum (2003a) visa

otimizar o CMDS com respeito ao cálculo dos autovalores e os autovetores mais altos de uma matriz alto dimensional Δ , obtendo uma sub-matriz Δ_n de $(n \times n)$ sobre a qual é calculada o procedimento CMDS, encontrando assim os n pontos representativos (marcas) denominados “*landmarks*” (SILVA; TENENBAUM, 2003a). A equação 3.32 apresenta o processo aplicado sobre a matriz Δ_n , onde a matriz $\mathbf{H}$ é a média centrada definida por $\mathbf{H} = \delta_{ij} - \mathbf{n}^{-1}$.

$$\mathbf{B} = -\frac{1}{2} \mathbf{H} \Delta_n^2 \mathbf{H} \quad (3.32)$$

Realizando a auto-decomposição sobre a matriz $\mathbf{B}$ (equação 3.33) obtêm-se os k maiores autovalores positivos $(\lambda_1, \dots, \lambda_k)$, denominados Λ_+ , junto com os correspondentes autovetores ortonormais $(v_1, \dots, v_k)$ denominados $\mathbf{Q}_+$.

$$\mathbf{B} = \mathbf{Q} \Lambda \mathbf{Q}^T \quad (3.33)$$

A partir de esses autovalores e autovetores pode-se encontrar a sub-matriz de pontos *landmark* ($\mathbf{L}_k$) de $n \times k \in \mathbb{R}^k$:

$$\mathbf{L}_k = \Lambda_+^{\frac{1}{2}} \mathbf{Q}_+ \quad (3.34)$$

Em um segundo passo realiza-se um procedimento de triangulação baseado em distância, para encontrar uma matriz $\mathbf{X}$ de $k \times N \in \mathbb{R}^k$ que represente os N pontos originais da matriz Δ . Assim, as coordenadas de cada ponto restante (que não é *landmark*) são calculadas usando as distâncias já conhecidas com respeito aos pontos *landmarks*, obtendo-se primeiro o vetor Δ_x (média da matriz Δ_n), isto é, $\Delta_x = (\delta_1 + \delta_2 + \dots + \delta_n) / n$.

Calculando a matriz transposta pseudoinversa $\mathbf{L}_k^\#$ de $\mathbf{L}_k$, a qual pode ser construída diretamente a partir de Λ_+ e $\mathbf{Q}_+$, da seguinte forma:

$$\mathbf{L}_k^\# = \mathbf{Q}_+ \Lambda_+^{-\frac{1}{2}} \quad (3.35)$$

Finalmente, a matriz resultante $\mathbf{X}$ é dada pela seguinte equação:

$$\mathbf{X} = -\frac{1}{2} \mathbf{L}_k^\# (\bar{\Delta}_n - \Delta_x) \quad (3.36)$$

onde $\bar{\Delta}_n$ denota o vetor coluna igual à raiz quadrada das distâncias entre a Δ e a matriz de pontos *landmarks* $\mathbf{L}_k$.

A principal característica de LANDMARK é que permite a introdução de novos pontos em forma contínua; e unicamente realiza um cálculo global se for exigido que as coordenadas encontradas estejam alinhadas com respeito aos eixos principais. O custo computacional deste algoritmo é aproximadamente $O(Nnk + n^3)$, onde k é a dimensão desejada, N é o número de dados e n é o número de pontos *landmarks*. A Figura 3.10 apresenta o pseudocódigo deste algoritmo.

LANDMARK (Δ, n, k)

Dados: matriz de dissimilaridade Δ , número n de pontos *landmarks*, dimensionalidade desejada k .

Saída: matriz de pontos $\mathbf{X}$, número de autovalores positivos $\mathbf{A}$,

Início

1. Iniciar N = número de filas de Δ
2. Calcular uma submatriz de pontos *landmarks* $\mathbf{L}$ de $n \times k$ que represente os n pontos *landmarks* em $\mathcal{R}^k$. Cada novo ponto *landmark* é obtido escolhendo o máximo das mínimas distâncias dos pontos não usados (pontos não *landmark*).

$\mathbf{L} = \text{MaxMin}(N, \Delta, n, 1)$ // vetor de índices de pontos *landmark*

3. Chamar ao algoritmo CMDS, como entrada, usar a sub-matriz Δ_n de $(n \times n)$ que contém as distâncias entre os pontos *landmarks*.

$\Delta_n = \mathbf{D}(\mathbf{L}, \mathbf{L})$

$[\mathbf{L}, \mathbf{A}] = \text{CMDS}(\Delta_n)$

$ki = \min(k, \text{size}(\mathbf{A}))$ // número de dimensões de saída

4. Aplicar a triangulação baseada em distâncias para encontrar uma matriz $\mathbf{X}$ de $(k \times N)$ que represente os N pontos que encaixam em $\mathcal{R}^k$. Como entrada, usar uma matriz $\Delta_{n, N-n}$ de $(n \times N - n)$ distâncias entre os pontos *landmarks* e os pontos restantes. As novas coordenadas são derivadas a partir da raiz quadrada das distâncias por uma transformação linear.

$\mathbf{un} = \text{mean}(\Delta_n)$

$\text{sqrA} = \sqrt{\mathbf{A}(1:ki)}$

$\mathbf{Li} = \frac{\mathbf{V}(:, 1:ki)}{\mathbf{ones}(n, 1) * \text{sqrA}^T}$

$\mathbf{F} = \frac{\mathbf{Li}' * (\Delta(\mathbf{L}, :) - \mathbf{un}(:) * \mathbf{ones}(1, N))}{2}$

5. Centrar os dados com respeito a sua média e usar PCA para alinhar os eixos principais dos novos dados com as coordenadas dos eixos, em ordem decrescente.

$\mathbf{u} = \text{mean}(\mathbf{F}^T)$

$\mathbf{X} = \mathbf{F} - \mathbf{u}(:) * \mathbf{ones}(1, N)$

$[\mathbf{U}, \mathbf{Au}] = \text{eigen}(\mathbf{X} \times \mathbf{X}^T)$ // onde $\mathbf{U}$ (autovetores), $\mathbf{Au}$ (autovalores)

$\mathbf{F} = \mathbf{U}^T \times \mathbf{X}$

fim

Figura 3.10: Pseudocódigo do algoritmo LANDMARK.

Para encontrar o conjunto inicial de pontos *landmarks* podem ser usadas duas alternativas: i) seleção aleatória de pontos *landmarks* ou ii) seleção de pontos usando o algoritmo MaxMin. Esse algoritmo seleciona os pontos (um por vez) escolhendo o máximo das mínimas distâncias entre os pontos que não são *landmark*. Nesse processo o primeiro ponto é escolhido aleatoriamente (SILVA, TENENBAUM, 2004b). A desvantagem de usar este algoritmo é o aumento do custo computacional em $O(nN)$. A Figura 3.11 apresenta o pseudocódigo deste algoritmo.

MaxMin(N, Δ, n, s)

Saída: lista de índices de pontos *landmark* : l_out .

Início

- Seleciona pontos aleatoriamente
 $\mathbf{P} = \text{randperm}(N)$
- Obter um vetor de índices de pontos *landmark*

$\mathbf{L} = \mathbf{P}(1:s)$

Se ($s = 1$) **então**

$\mathbf{m} = \Delta(\mathbf{L}, :)$

senão

$\mathbf{m} = \min(\Delta(\mathbf{L}, :))$

fim

$l_out = \mathbf{L}$

Para ($i = s + 1 : n$) **fazer**

$[\text{val}, \mathbf{L}] = \max(\mathbf{m})$

$\mathbf{m} = \min(\mathbf{m}, \Delta(\mathbf{L}, :))$

$l_out(i) = \mathbf{L}$

fim

fim

Figura 3.11: Pseudocódigo do algoritmo MaxMin.

Capítulo 4

MATERIAIS E MÉTODOS

Neste capítulo são descritos os bancos de dados usados e os experimentos realizados para avaliar os algoritmos MDS estudados. Os bancos de dados são divididos em: bancos de dados de avaliação e um banco de dados de aplicação. Na Seção 4.1 descrevem-se os bancos de dados de avaliação, os quais consistem em 4 bancos de dados (3 bancos reais e 1 banco artificial) com diferente tamanho e dimensionalidade. Estes bancos de avaliação são usados para analisar o desempenho dos distintos algoritmos MDS implementados. O banco de dados de aplicação é descrito na Seção 4.2, o qual consiste em dados genômicos extraídos de uma coleção brasileira de estirpes de bactérias fixadoras de nitrogênio no solo, pertencentes ao gênero *Bradyrhizobium*. O objetivo deste banco de dados é aplicar os algoritmos MDS implementados para obter um mapeamento visual da informação de proximidade entre as distintas estirpes. Na Seção 4.3 descreve-se a ferramenta *MDSExplorer* criada para realizar os testes e visualizar os resultados. Finalmente, na Seção 4.4 é descrita a metodologia dos experimentos realizados nesta dissertação.

4.1. BANCOS DE DADOS DE AVALIAÇÃO

BANCO DE DADOS IRIS

O banco de dados de plantas Iris ou “*Iris Plant*” é bem conhecido na literatura (DASARATHY, 1980) e encontra-se disponível na *website* da Universidade de Califórnia (ASUNCION; NEWMAN, 2007). O conjunto de dados que conformam este banco, contém informação sobre as flores do gênero Iris, as quais estão

igualmente divididas em três classes: *Iris setosa*, *Iris versicolor* e *Iris virginica*. Cada flor é descrita por 4 atributos numéricos contínuos: comprimento da sépala, largura da sépala, comprimento da pétala, e largura da pétala. Na Figura 4.1 são visualizados os dados tomando os 3 primeiros atributos. Note-se que a classe *Iris setosa* (azul) está bem separada das outras, no entanto as classes *Iris versicolor* (vermelha) e *Iris virginica* (verde) não apresentam uma separação bem definida.

Figura 4.1: Representação do banco de dados Iris, considerando os três primeiros atributos e utilizando o atributo classe para dar cor. As espécies *setosa*, *versicolor* e *virginica* estão representadas pelas cores azul, vermelho e verde respectivamente.

BANCO DE DADOS CÂNCER

O banco de dados Câncer utilizado neste trabalho pertence à *Wisconsin Breast Cancer Database*, obtido da *website* da Universidade de Califórnia (ASUNCION; NEWMAN, 2007). Este banco de dados contém 699 objetos com 9 atributos de dados (o atributo 10 identifica a pertinência da classe que indica se é maligno ou benigno).

Na Figura 4.2 são visualizados os dados tomando os 3 primeiros atributos, onde

cada objeto representa uma célula de um paciente com câncer. Uma célula é descrita por 9 atributos numéricos na faixa de 1 a 10 (código identificador, espessura, tamanho da célula, forma da célula, adesão marginal, tamanho epitelial da célula, núcleos desencapados, cromatina suave, nucléolos normais, mitoses, classe). Para gerar a visualização deste banco de dados não foram considerados os 16 elementos que não têm valores no atributo núcleos desencapados, porém só se consideraram 683 pontos.

Figura 4.2: Representação do banco de dados Câncer, considerando as três primeiras dimensões e utilizando o atributo classe para dar cor. Os dois tipos de células malignas e benignas estão representadas pelas cores verde e azul respectivamente.

BANCO DE DADOS IMAGENS

O banco de dados *Imagens Segmentation*, que aqui será referenciado como Banco Imagens, foi obtido da *website* da Universidade de Califórnia (ASUNCION; NEWMAN, 2007). Neste banco, as instâncias (cada instância é uma região de 3x3) foram aleatoriamente extraídas de um banco de dados de 7 imagens segmentadas para criar uma classificação para cada pixel. Este banco possui 19 atributos reais que foram

extraídos das regiões das imagens. No total, existem 7 diferentes classes de regiões: *brickface*, *sky*, *foliage*, *cement*, *window*, *path* e *grass*. Na Figura 4.3 são visualizados os dados deste banco de dados tomando os 3 primeiros atributos.

Figura 4.3: Representação do banco de dados Images, considerando as três primeiras dimensões e utilizando o atributo classe para dar cor. As imagens *brickface*(1), *sky*(2), *foliage*(3), *cement*(4), *window*(5), *path*(6) e *grass*(7).

BANCO DE DADOS SINTÉTICO

Este banco de dados foi construído artificialmente a partir da mistura de 5 gaussianas (distribuições normais), consta de 1000 pontos e apresenta 5 classes. A Figura 4.4 apresenta a visualização em 3 dimensões gerada por Villanueva (2007). Como pode ser observado 4 grupos apresentam sobreposição.

Figura 4.4: Representação do Banco de dados Sintético usando as 3 primeiras dimensões e utilizando o atributo classe para dar cor. Adaptado de Villanueva (2007).

4.2. BANCO DE DADOS DE APLICAÇÃO

O banco de dados de aplicação escolhido neste trabalho consiste de dados genômicos extraídos de uma coleção brasileira de estirpes de bactérias pertencentes ao gênero *Bradyrhizobium*. Essas estirpes exibem características fenotípicas das espécies *Bradyrhizobium japonicum* e *Bradyrhizobium elkanii* isoladas de 33 espécies de leguminosas tropicais. Estas bactérias são importantes na agricultura por sua conhecida habilidade de nodular a soja e transformar o nitrogênio atmosférico (N_2) em compostos assimiláveis pela planta. Maiores detalhes podem ser encontrados em (GERMANO *et al.*, 2006).

O banco de dados *Bradyrhizobium* usado neste trabalho foi obtido pelo laboratório de Biotecnologia do Solo, da Empresa Brasileira de Pesquisa Agropecuária – Centro Nacional de Pesquisa de Soja (Embrapa Soja), em Warta, distrito de Londrina, Paraná. Este Banco de dados consiste de 119 imagens onde cada imagem corresponde a uma estirpe. Cada estirpe é descrita por 9 canaletas resultantes da análise de RFLP-PCR como é apresentado na Figura 4.5. Cada canaleta corresponde à análise de uma das regiões ribossomais 16S, 23S ou IGS, e para cada uma dessas

regiões utilizaram-se três enzimas de restrição diferentes como é descrito na Tabela 4.1.

Figura 4.5: Exemplo de imagem resultante do processo de gel eletroforese, no qual foi analisado uma única estirpe.

Tabela 4.1 - Relação de enzimas de restrição utilizadas e regiões ribossomais analisadas na obtenção do banco de dados de estirpes de *Bradyrhizobium*.

<i>Canaleta</i>	<i>Região Ribossomal</i>	<i>Enzima de restrição</i>
I	16S	Cfo I
II	16S	Dde I
III	16S	Msp I
IV	23S	Hae III
V	23S	Hha I
VI	23S	Hinf I
VII	IGS	Dde I
VIII	IGS	Hae III
IX	IGS	Msp I

O método de obtenção dos dados genômicos foi por meio da análise de genes ribossomais usando a técnica RFLP-PCR (*Restriction Fragment Length Polymorphism - Polymerase Chain Reaction*) (DAVISON, 2006). Esta técnica é popular devido ao seu baixo custo econômico e é usada comumente como método inicial para avaliar biodiversidade e posição taxonômica. RFLP-PCR consiste em várias etapas, primeiro é extraído o DNA (ou RNA) mediante processos de rompimento de células, centrifugação e substâncias que são capazes de desnaturar e retirar as proteínas que

estão acopladas ao DNA. Logo, realiza-se um processo de crescimento do DNA mediante a técnica PCR, na qual uma região específica do DNA é amplificada em ciclos repetidos de desnaturação, hibridação e extensão, sendo que em cada ciclo a quantidade do DNA é dobrada. Posteriormente, realiza-se um processo de polimorfismo de tamanho de fragmentos de DNA (RFLP), o qual consiste em um tratamento do DNA com enzimas de restrição. Estas enzimas são proteínas que reconhecem uma sequência de nucleotídeos específica (sítios de restrição) e a digerem, cortando o DNA em diversos fragmentos. O número e tamanho dos fragmentos dependem do número de sítios de restrição reconhecidos pela enzima no DNA. Finalmente os fragmentos obtidos são separados por seu tamanho usando eletroforese. A eletroforese é realizada colocando os fragmentos em uma lâmina de gel de agarose ou poliacrilamida, sendo a lâmina submetida a um campo elétrico. Devido à carga elétrica negativa dos fragmentos (pelo radical fosfato), eles vão se movimentando através do gel na direção do eletrodo positivo. Os fragmentos menores movimentam-se mais rápido do que os maiores. No final do processo os fragmentos de igual tamanho agrupam-se em *bandas*¹ ao longo do gel. O conjunto de bandas é chamado de *canaleta*. Na Figura 4.6 é mostrado um exemplo de imagens de canaletas de 4 organismos, nas quais as bandas são as linhas de cor escura que indicam a presença de material genético e a linha vertical da esquerda indica o sentido de migração dos fragmentos. Note-se, por exemplo, que a primeira e a última canaleta são similares, podendo implicar que sejam organismos da mesma espécie.

Figura 4.6: Exemplo de imagens de canaletas de 4 organismos, em que as bandas são as linhas escuras que indicam a presença de material genético. A posição das bandas na canaleta é determinado pelo tamanho dos seus fragmentos.

¹ As bandas no gel de eletroforese indicam a concentração de material genético com igual peso molecular e são identificadas como as linhas de cor escura no gel.

A grande quantidade de dados (estirpes) e a alta dimensionalidade (várias canaletas por estirpe) presentes no banco de dados *Bradyrhizobium* torna muito difícil a realização de uma análise visual usando diretamente as imagens de canaletas. Com a finalidade de se obter uma melhor representação das estirpes, e assim ganhar uma maior compreensão dos mesmos é que neste trabalho são usados os algoritmos MDS. Como esses algoritmos trabalham a partir de matrizes de distância, é necessário realizar um pré-processamento dos dados para obter as matrizes de distância.

No pré-processamento são calculadas 9 matrizes de distância, uma para cada tipo de canaleta, as quais representam a informação de uma determinada região ribossomal analisada por uma enzima de restrição. A Figura 4.7 ilustra de forma geral como são arranjados os dados para obter as matrizes de distância. Por exemplo, para formar a primeira matriz (Δ_I) são selecionadas as canaletas I (cor azul) de cada uma das 119 estirpes e colocadas como entrada no pré-processamento.

Figura 4.7: Obtenção das matrizes de distância para MDS. Cada matriz é obtida com o pré-processamento de uma determinada canaleta das 119 estirpes.

PRÉ-PROCESSAMENTO DOS DADOS

Com este passo tenta-se extrair a informação relevante das canaletas, a qual é a distribuição das suas bandas, procurando atenuar diversas perturbações presentes nas imagens como: deformações das bandas, variabilidade de iluminação, manchas, entre outros.

O procedimento seguido baseia-se no método usado em Villanueva (2007), onde primeiramente, cada imagem de canaleta é transformada em uma sequência discreta, a qual é chamada *eletroferograma*. Seguidamente é calculada a sequência média das colunas da matriz de tons de cinza, e em seguida é determinado um limiar que identifica a informação relevante (picos). Com este limiar desloca-se a sequência de tal forma que os picos, que representam as bandas, fiquem no lado positivo. Finalmente, obtém-se a sequência de *eletroferograma* anulando os valores negativos e normalizando os valores positivos com respeito a seu máximo valor. A Figura 4.8 ilustra o processo de formação de *eletroferograma*.

Figura 4.8: Processo de formação do electroferograma a partir de uma canaleta. Adaptado de Villanueva (2007).

A matriz de distância é calculada a partir dos eletroferogramas calculados previamente. Para isso, foi escolhido o coeficiente de correlação de *Pearson*, o qual mede a associação linear entre duas seqüências sem depender da unidade de medida. Quanto maior o valor do coeficiente, maior a similaridade entre as seqüências. Para dois eletroferogramas x_p e x_q , o coeficiente de correlação é definido como:

$$r_{x_p x_q} = \frac{1}{N} \sum_{i=1, N} \left(\frac{x_p[i] - \bar{x}_p}{\sigma_{x_p}} \right) \left(\frac{x_q[i] - \bar{x}_q}{\sigma_{x_q}} \right) \quad (4.1)$$

onde N indica o tamanho dos eletroferogramas, σ representa o desvio padrão e $\bar{x}_p$, $\bar{x}_q$ são as respectivas médias de x_p e x_q .

Os elementos da matriz de distância Δ são formados a partir dos recíprocos dos coeficientes de correlação calculados entre todos os pares de eletroferogramas, isto é, $\Delta = (\delta_{ij})$, onde $\delta_{ij} = 1 - r_{ij}$. Desta forma se encontra uma matriz de distância, em que cada elemento δ_{ij} representa a distância entre duas estirpes (i, j). Na Figura 4.9 é ilustrado o processo de formação de uma matriz de distância a partir dos eletroferogramas obtidos de 119 estirpes de uma determinada região ribossomal e enzima de restrição.

Figura 4.9: Ilustração do processo de formação de uma matriz de distância.

4.3. FERRAMENTA MdsEXPLORER

Durante este trabalho foi desenvolvida uma ferramenta em Matlab^{®2} denominada *MDSExplorer*, a qual integra os 4 algoritmos MDS analisados: CMDS, SMACOF, FASTMAP e LANDMARK. Esta ferramenta também integra a opção de visualização de dados, visando complementar o processo iniciado com o estudo de técnicas de redução dimensional e análise dos mapeamentos realizados pelos diferentes algoritmos MDS estudados neste trabalho.

O funcionamento da ferramenta *MDSExplorer* envolve três fases: a seleção das opções de análise, o processo de mapeamento a partir do algoritmo MDS escolhido, a matriz de distância introduzida e a visualização interativa no espaço euclidiano dos resultados do mapeamento com os índices de desempenho associados ao mesmo, conforme é apresentado na Figura 4.10. Assim, esta ferramenta pode ser usada para apoiar ao usuário na análise e extração de informação mais rapidamente, representando em forma gráfica os objetos mapeados.

A Figura 4.11 apresenta a janela de opções de análise da ferramenta *MDSExplorer*. Esta janela permite a seleção da matriz de distância a mapear e a seleção de diversas opções de análise relativas ao algoritmo MDS (tipo de algoritmo MDS, número de dimensões para o algoritmo FASTMAP, número de dimensões e número de pontos *landmark* para o algoritmo LANDMARK).

² <http://www.mathworks.com/>

Figura 4.10: Funcionamento da ferramenta MDSExplorer gerando mapeamentos de dados mediante a seleção de uma matriz de distância.

Figura 4.11: Interface de seleção de matrizes distância do banco de estirpes de *Bradyrhizobium*.

A Figura 4.12 apresenta um exemplo da interface da ferramenta *MDSExplorer* apresentando o mapeamento gerado pelo algoritmo LANDMARK a partir da matriz de distância do banco de dados Câncer. Esta janela possui um espaço para seleção de Ferramentas de controle, um espaço para visualização dos dados outro para visualização de análise de *stress* e outro para visualização de análise de autovalores.

Figura 4.12: Exemplo de interface do *MDSExplorer* apresentado a análise do banco de dados Câncer.

As ferramentas de controle permitem a seleção de opções sobre o espaço de visualização, permitindo a escolha de visualização dos dados em 2D ou 3D, projeções sobre os planos X, Y ou Z, visualização de eixos, visualização de malhas e visualização de etiquetas identificadoras dos pontos.

O espaço de Análise de *stress* apresenta três curvas que representam a evolução do *stress* com o número de dimensões. Essa análise ajuda a determinar qual é o número mais adequado de dimensões para dar uma completa representação dos dados. Indicando-se o ponto no qual o *stress* é estabilizado pode ser observada a otimização realizada no mapeamento. Para um número dado de dimensões, quanto menor é o *stress*, melhor a qualidade da representação e o ajuste encontrado entre distâncias e dissimilaridades.

O espaço de Análise de Autovalores indica a proporção da distribuição dos

dados. O objetivo deste gráfico é apresentar quanta informação (porcentagem da variância total) fornece cada dimensão no conjunto total de dimensões mapeadas.

O espaço de visualização apresenta uma correspondência biunívoca entre os objetos no espaço original e os pontos no espaço mapeado. Cada objeto de um conjunto de dados subjacente em um espaço multidimensional é mapeado para um ponto no espaço tridimensional (espaço de visualização) seguindo a construção de mapeamentos da técnica *Scaterplot* explicada no Capítulo 2. Os eixos do espaço de visualização correspondem às dimensões e são escalados linearmente pelos valores de mínimo e máximo de cada dimensão correspondente.

A geração de cores seguiu o esquema do RGB, segundo o apresentado na Figura 4.13. Cada ponto no espaço de visualização corresponde a uma coordenada deste espaço de cores.

Figura 4.13: Imagem do espaço RGB representado em um cubo.

VISUALIZAÇÃO EM VTK

Como ferramenta complementar ao MDSExplorer foi desenvolvido uma interface de visualização na biblioteca de *Visualization ToolKit* (VTK) (SCHROEDER *et al.*, 2002). Esta ferramenta lê e visualiza em 3D os mapeamentos gerados pelo *MDSExplorer* (armazenados em um arquivo *txt*) e assim realiza uma representação dos dados com melhor iteratividade, aproveitando os recursos do VTK. Esses recursos possibilitam que o usuário “navegue” através dos gráficos criados pelo VTK, possibilitando simulação de movimentação, aproximação e afastamento (zoom), interação com iluminação e câmeras, isto é, visualização interativa de dados gerados

em 3D.

O VTK processa filtros, mapeadores, iluminadores (*Lights*), câmeras e atores que são instanciados para criar uma representação gráfica dos dados. O modelo de programação do VTK adota o paradigma de fluxo de dados. Nesse paradigma, módulos são conectados para formar uma rede que descreve um canal (*pipeline*) de processamento de dados. Os módulos executam operações algorítmicas sobre os dados enquanto eles fluem pelo *pipeline*.

Ferramentas para visualização de dados como o VTK, podem ser utilizadas gratuitamente no desenvolvimento de aplicativos gráficos para a visualização de dados como os estudados neste trabalho. O algoritmo desenvolvido para visualização do mapeamento de dados em 3D, foi escrito na linguagem C++ utilizando o *Microsoft Visual C++ 2005 Express Edition*. A versão do VTK utilizada foi a 5.0, instalada na plataforma *Win32*.

A Figura 4.14 apresenta um exemplo de visualização dos bancos de estirpes de *Bradyrhizobium* com o VTK. Como pode ser observado foram criados contornos e planos separadores no eixo central, sendo que esses planos ajudam a separar os pontos facilitando a visualização de agrupamentos. Cabe ressaltar que cada ponto está associado a uma etiqueta descritiva do ponto visualizado e a uma cor correspondente ao mapeamento de coordenada.

Figura 4.14: Visualização do banco *Bradyrhizobium* em VTK apresentando o resultado do mapeamento gerado no *MDSExplorer*.

4.4. EXPERIMENTOS

Os experimentos realizados estão divididos em duas partes. A primeira parte consiste em avaliar os algoritmos MDS nos bancos de dados de avaliação mediante a análise de diversos índices de desempenho. Logo, nessa análise é escolhido o algoritmo MDS que apresenta o melhor desempenho com o objetivo de realizar o mapeamento e representação visual do banco de estirpes de *Bradyrhizobium*. Os índices de desempenho utilizados são os seguintes:

- Índices de *stress*, os quais compreendem o Stress-1 (Equação 3.12), Stress-2 (Equação 3.13). Com estes índices pretende-se medir a precisão do mapeamento. Menor valor de *stress* indica maior proximidade da matriz de distância obtida dos pontos resultantes com respeito à matriz de distância original.
- Tempo computacional, refere-se ao tempo em segundos, usado pelos algoritmos MDS, desde o instante em que são invocados no programa principal até o instante em que retornam os resultados.
- Pureza do agrupamento induzido, este é um índice usado para avaliar a proximidades dos agrupamentos encontrados por algum método de agrupamento nos dados mapeados com respeito às classes reais. Para um banco de dados expresso em tuplas $(\mathbf{x}_i, cl_i)$, onde $\mathbf{x}_i$ indica os atributos do dado i , e cl_i indica a classe à qual pertence o dado, $cl_i \in \{cl_1, \dots, cl_m\}$, a pureza de um agrupamento C com grupos $\{c_1, c_2, \dots, c_g\}$ obtido por algum método de agrupamento é definido como:

$$Pureza(C) = \frac{\sum_{c_i \in C} \max(N_{c_i}(cl_1), N_{c_i}(cl_2), \dots, N_{c_i}(cl_m))}{n} \quad (4.1)$$

onde $N_{c_i}(cl_j)$ denota o número de elementos com classe cl_j dentro do grupo c_i e n denota o número de dados. O índice de pureza indica a percentagem de elementos que possuem a classe majoritária em cada grupo.

Na Figura 4.15 é mostrado o pseudocódigo dos experimentos realizados para obter estes índices de desempenho. Como podem ser observados, os algoritmos SMACOF, FASTMAP e LANDMARK são testados para um número crescente de dimensões. O

número máximo de dimensões analisado foi escolhido de forma que fosse maior do que o número real de dimensões. A limitação que se tem aqui é que nem todos os algoritmos MDS conseguem encontrar mapeamentos com dimensionalidade maior que o real. No caso do banco Iris (4 dimensões), a faixa de dimensões analisadas foi entre 1 e 8. Nos bancos Câncer (9 dimensões) e Sintético (3 dimensões) foi entre 1 e 10 dimensões, já para o banco de Imagens, a dimensionalidade foi variada entre 1 e 19 dimensões (sua verdadeira dimensionalidade).

Para calcular os índices de desempenho dos algoritmos SMACOF, FASTMAP e LANDMARK em um determinado banco de dados, o procedimento consiste em executar iterativamente o algoritmo respectivo calculando seu tempo computacional, seus índices de *stress* (*CalcStress*) e o índice de pureza do agrupamento induzido nessa dimensionalidade. O algoritmo de agrupamento (*Agrupar*) escolhido neste trabalho é o *K-Means* (DUDA *et al.*, 2001), o qual requer como entrada o número de grupos (*num_clusters*). Nesta avaliação usou-se como valor deste parâmetro o número de classes conhecido de cada banco de dados. O algoritmo CMDS, diferentemente dos outros algoritmos, é executado somente uma vez por cada banco de dados. Isto porque CMDS sempre retorna o número máximo de dimensões possíveis (correspondentes aos autovalores positivos). Portanto, o cálculo do tempo computacional é realizado somente uma vez. O cálculo dos outros índices de desempenho é similar aos outros métodos, variando-se o número de dimensões dos pontos obtidos. As saídas dos experimentos em cada método e banco de dados correspondem a um vetor de tempo **T**, dois vetores de *stress*: **S1** (Stress-1), **S2** (Stress-2), e um vetor **P** contendo as purezas dos agrupamentos induzidos. Cada elemento destes vetores corresponde a um índice numa determinada dimensionalidade.

Note-se que para o algoritmo LANDMARK é requerido o número de pontos *landmark* (*p*). Para avaliar a influência deste parâmetro nos índices de desempenho foram testados três valores, os quais são expressos como porcentagem do tamanho do banco de dados. Estas porcentagens foram escolhidas em 10%, 20% e 30%, como se mostra na Tabela 4.2.

Entradas: Δ = matriz distância, k = # dimensões, p = # pontos *landMark* (caso LANDMARK), $num_clusters$ = # de classes, **labels** = vetor de rótulos de classe de cada dado .

Saídas: **X** = matriz de pontos, n = # iterações (caos SMACOF), **T** = vetor de tempo computacional, **S1,S2** = vetores de *stress*, **P** = vetor de purezas.

Caso: CMDS

```

Tic //Inicia contador de tempo
X = CMDS(  $\Delta$  ) // chama ao algoritmo MDS
T = Toc // Salva o tempo usado pelo algoritmo
Para (  $i = 1 : k$  ) fazer // Por para cada dimensão
    D = DistEuc(X(1:i)) // Calcula a matriz distância tomando  $i$  dimensões de X
    [S1(i),S2(i)] = CalcStress(D, $\Delta$ ) // Calcula os índices de stress
    C = Agrupar(X(1:i),num_clusters) //Agrupa os pontos tomando  $i$  dimensões
    P(i) = Pureza(C,labels) // Calcula a pureza do agrupamento
fim

```

Caso: SMACOF // Otimiza com SMACOF o **X** obtido por MDS

```

Para (  $i = 1 : k$  ) fazer
    Tic
    [X,D] = SMACOF(  $\Delta$ , X(1:i) )
    T(i) = Toc
    [S1(i),S2(i)] = CalcStress(D, $\Delta$ )
    C = Agrupar(X,num_clusters)
    P(i) = Pureza(C,labels)
fim

```

Caso: FASTMAP

```

Para (  $i = 1 : k$  ) fazer
    Tic
    X = FASTMAP(  $\Delta$ ,  $i$  )
    T(i) = Toc
    D = DistEuc(X)
    [S1(i),S2(i)] = CalcStress(D, $\Delta$ )
    C = Agrupar(X,num_clusters)
    P(i) = Pureza(C,labels)
fim

```


Caso: LANDMARK**Para** ($i = 1 : k$) **fazer**

Tic

 $\mathbf{X} = \text{LANDMARK}(\mathbf{\Delta}, i, p)$ $\mathbf{T}(i) = \text{Toc}$ $\mathbf{D} = \text{DistEuc}(\mathbf{X})$ $[\mathbf{S1}(i), \mathbf{S2}(i)] = \text{CalcStress}(\mathbf{D}, \mathbf{\Delta})$ $C = \text{Agrupar}(\mathbf{X}, \text{num_clusters})$ $\mathbf{P}(i) = \text{Pureza}(C, \text{labels})$ **fim**

Figura 4.15: Pseudocódigo do processo de cálculo dos índices de desempenho dos algoritmos MDS nos bancos de dados de avaliação.

Tabela 4.2 – Tamanho de amostras apresentadas ao algoritmo LANDMARK.

Banco de Dados	No. De Dados	10%	20%	30%
Iris	150	15	30	45
Câncer	683	68	137	205
Imagens	2101	210	420	630
Sintético	1000	100	200	300
Bradyrhizobium	119	12	24	36

Os experimentos foram efetuados em um computador AMD 64 Athlon(tm), operando a uma velocidade de 3 GHz, com 2 GBytes de memória, e sistema operacional *MS Windows XP*.

Capítulo 5

RESULTADOS E DISCUSSÕES

Este Capítulo está estruturado em duas partes. Na primeira parte são apresentados e discutidos os resultados obtidos nos experimentos dos bancos de dados de avaliação, e na segunda parte são apresentados e discutidos os resultados do banco de dados de aplicação. Com os resultados dos bancos de avaliação pretende-se analisar o desempenho dos diversos métodos MDS estudados quanto à sua precisão de mapeamento (através do *stress*), seu tempo computacional, e seu índice de pureza do agrupamento resultante. Com o resultado desta análise pretende-se escolher o algoritmo MDS que apresenta os melhores índices de desempenho para mapear o banco de aplicação (banco de estirpes de *Bradyrhizobium*).

5.1. RESULTADOS NOS BANCOS DE DADOS DE AVALIAÇÃO

BANCO DE DADOS IRIS

Nas Figuras 5.1 e 5.2 são mostrados os valores medidos dos 2 critérios de *stress* analisados. Cada figura corresponde à análise de um determinado critério, na qual são mostradas as 6 curvas de *stress* correspondentes aos algoritmos MDS estudados: CMDS, CMDS_SMACOF, FASTMAP e LANDMARK (com números de pontos *landmark* igual a: 10%, 20% e 30% do tamanho do banco de dados). Estas curvas são apresentadas como função do número de dimensões. Pôde ser observado que nos 2 critérios de *stress* e em todos os algoritmos MDS os valores de *stress* estão próximos de 0, para 4 ou mais dimensões, o que indica que a matriz de distância construída a partir dos pontos mapeados com 4 ou mais dimensões é muito próxima da matriz de

distância original. Isto concorda com o fato de que a verdadeira dimensionalidade do banco Iris é 4 e, portanto, tomar uma maior dimensionalidade não acrescenta maior informação. É observado também que nos 2 critérios, o algoritmo CMDS_SMACOF destaca (com menores valores de *stress*) sobre os demais algoritmos nas 3 primeiras dimensões analisadas. Estes resultados mostram que a otimização realizada por CMDS_SMACOF sobre os resultados de CMDS resulta em uma melhor precisão de mapeamento, medido através do índice de *stress*.

Figura 5.1: Stress-1 para o banco Iris

Figura 5.2: Stress-2 para o banco Iris

Na Figura 5.3 mostram-se as curvas do tempo computacional de todos os algoritmos analisados como função do número de dimensões. Essas curvas são mostradas em escala logarítmica devido a diferenças muito grandes entre algoritmos. Como é observada, a curva do algoritmo CMDS_SMACOF apresenta tempos computacionais significativamente maiores que os outros algoritmos, isto é devido à natureza iterativa de SMACOF. Note-se também neste algoritmo que o tempo computacional cai significativamente quando a dimensionalidade analisada é a verdadeira dimensionalidade do banco de dados (4 dimensões). Este comportamento pode ser devido a que o mapeamento CMDS otimizado por SMACOF está muito próximo do ótimo e, portanto, a convergência é alcançada rapidamente. Por outro lado, CMDS apresenta a segunda maior curva de tempo computacional, note-se que esta curva é constante devido a que CMDS é executado somente uma vez, não dependendo do número de dimensões. O tempo computacional do CMDS foi replicado em todas as dimensões no gráfico por razões de comparação. As curvas dos algoritmos FASTMAP e LANDMARK apresentam os menores tempos computacionais, no entanto, o FASTMAP apresenta um tempo computacional crescente com o número de dimensões. As curvas de LANDMARK são menores para todas as dimensões sendo quase insensíveis à dimensionalidade e ao número de pontos *landmark*. Este resultado pode ser devido ao banco de dados Iris ter poucos dados e, portanto, os tempos computacionais são tão pequenos que poderiam estar lidando com a mínima resolução da função usada para medir o tempo computacional.

Figura 5.3: Tempo Computacional do banco Iris (em escala logarítmica).

A Figura 5.4 apresenta o índice de pureza de agrupamentos realizados com número de grupos igual ao verdadeiro número de classes (3), variando-se a dimensionalidade dos dados mapeados pelos distintos algoritmos MDS. Pode ser observado que quase todos os algoritmos induzem agrupamentos com purezas iguais a 89.33%, nas distintas dimensões analisadas. Somente o algoritmo FASTMAP apresenta um índice de pureza de agrupamento ligeiramente menor (88.7%) em 2 dimensões. Isto significa que os dados mapeados por FASTMAP em 2 dimensões induzem agrupamentos de menor qualidade, que os demais algoritmos na mesma dimensionalidade.

Figura 5.4: Índice de pureza de agrupamento do banco Iris

BANCO DE DADOS DE CÂNCER

Nas Figuras 5.5 e 5.6 são mostradas respectivamente as curvas de *stress* para cada um dos critérios analisados. Como é observado, nos 2 critérios de *stress* e em todos os algoritmos MDS, os valores de *stress* estão próximos de 0, para 9 e 10 dimensões, indicando que a matriz de distância construída a partir dos pontos mapeados nestas dimensões é muito próxima da matriz de distância original. Esse resultado era esperado, já que a verdadeira dimensionalidade deste banco de dados é 9 e, portanto, não se consegue uma maior precisão de mapeamento ao tomar uma maior dimensionalidade. Similar ao banco Iris é observado que o algoritmo CMDS_SMACOF apresenta os menores valores de *stress* nos 2 critérios analisados.

Por outro lado, o algoritmo FASTMAP mostra os maiores valores de *stress* em todos os critérios. Já, os algoritmos CMDS, e LANDMARK com os distintos números de pontos *landmark* mostram curvas sobrepostas, o que significa que a precisão de mapeamento nestes métodos é quase insensível ao número de pontos *landmark*.

Figura 5.5: Stress-1 para o banco Câncer

Figura 5.6: Stress-2 para o banco Câncer

A Figura 5.7 mostra as curvas do tempo computacional dos algoritmos MDS como função do número de dimensões. Semelhante ao banco Iris, as curvas são mostradas em escala logarítmica por existir diferenças muito grandes entre os algoritmos. Como se observa, a curva do algoritmo CMDS_SMACOF apresenta tempos computacionais significativamente maiores do que os demais algoritmos.

Note-se também neste algoritmo que a curva do tempo computacional cai significativamente quando a dimensionalidade analisada é a verdadeira (9 dimensões), o que pode ser devido aos pontos CMDS otimizados por SMACOF estarem muito próximos do ótimo e, portanto, a convergência é alcançada rapidamente. No outro extremo se encontra a curva de tempo computacional do algoritmo LANDMARK com 10% de pontos *landmark*, o qual apresenta tempos computacionais significativamente menores dos outros algoritmos. As outras curvas de LANDMARK com 20% e 30% de pontos *landmark* apresentam tempos computacionais superiores. Isto era esperado já que à medida que se toma um maior número de pontos *landmark* o tempo computacional começa a ser dominado pelo cálculo CMDS que se realiza internamente sobre os pontos *landmark*. No limite, quando a porcentagem de pontos *landmark* tende para 100%, a curva do tempo computacional aproxima-se para a curva do CMDS. Note-se também que, semelhante ao banco Iris, o algoritmo LANDMARK é quase insensível ao número de dimensões. Já no algoritmo FASTMAP, o tempo computacional é crescente com o número de dimensões.

Figura 5.7: Tempo Computacional do banco Câncer (em escala logarítmica).

A Figura 5.8 apresenta o índice de pureza de agrupamentos realizados com número de grupos igual ao verdadeiro número de classes (2), variando-se a dimensionalidade dos dados mapeados pelos distintos algoritmos MDS. Pode ser observado que somente o algoritmo FASTMAP apresenta valores de pureza inferiores

aos outros algoritmos nas primeiras 4 dimensões. Isto pode significar que FASTMAP gera mapeamentos que induzem agrupamentos de menor qualidade em baixas dimensionalidades. Os outros algoritmos apresentam um comportamento similar no índice de pureza, sendo que em 2 e 3 dimensões o índice é igual a 96.05% e em dimensionalidades maiores a 3 o índice de pureza atinge a 96.2%. Nesses algoritmos poder-se-ia indicar que com dimensionalidade maior do que 4 não se consegue melhores qualidade de agrupamentos.

Figura 5.8: Índice de pureza de agrupamento do banco Câncer

BANCO DE DADOS DE IMAGENS

Nas Figuras 5.9 e 5.10 são mostradas as respectivas curvas de *stress* para cada um dos critérios analisados. Como poder ser observado, nos 2 critérios de *stress* e em todos os algoritmos MDS, os valores de *stress* estão próximos de 0 para dimensionalidades maiores ou iguais a 14. Este resultado indica que se pode reproduzir com uma alta precisão a matriz de distância original a partir dos pontos mapeados em uma dimensionalidade significativamente inferior à verdadeira (19), o que significa que este banco de dados possui várias dimensões irrelevantes que não acrescentam informação. É observado também que o algoritmo CMDS-SMACOF, diferentemente dos resultados nos bancos anteriores, não se destaca sobre os outros algoritmos quanto aos seus índices de *stress*. Sua curva de *stress* está praticamente

sobreposta com as curvas dos algoritmos CMDS e LANDMARK, assim o SMACOF neste banco de dados não consegue melhorar significativamente a precisão do mapeamento realizado por CMDS. Por outro lado o algoritmo FASTMAP apresenta o pior desempenho quanto ao *stress* (como nos bancos anteriores), com curvas acima das outras. Notá-se também que o desempenho em *stress* do algoritmo LANDMARK é quase insensível ao número de pontos *landmark*.

Figura 5.9: Stress-1 para o banco Imagens

Figura 5.10: Stress-2 para o banco Imagens

Na Figura 5.11 são apresentadas às curvas em escala logarítmica do tempo computacional dos diversos algoritmos como função do número de dimensões para o banco de dados Imagens. Igual aos bancos anteriores, observa-se que a curva do

algoritmo CMDS_SMACOF apresenta tempos computacionais significativamente maiores do que o resto dos algoritmos. É observado também nesta curva que o tempo computacional cai significativamente a partir de 14 dimensões, o que pode ser devido a que essa é a dimensionalidade em que o CMDS gera mapeamentos próximos do ótimo e, portanto, o SMACOF alcança a convergência mais rapidamente. Isto também concorda com a análise de *stress* a qual mostra que com 14 dimensões se consegue mapeamentos com baixo *stress* em todos os algoritmos. Semelhantemente aos resultados nos bancos anteriores, observa-se que o algoritmo LANDMARK com 10% de pontos *landmak*, apresenta uma curva com tempos computacionais significativamente menores do que os outros algoritmos, e estes tempos são quase insensíveis à dimensionalidade. O contrário é observado no algoritmo FASTMAP, no qual o tempo computacional cresce com o número de dimensões.

Figura 5.11: Tempo Computacional do banco Imagens (em escala logarítmica).

A Figura 5.12 apresenta o índice de pureza de agrupamentos realizados com número de grupos iguais ao verdadeiro número de classes (7), variando-se a dimensionalidade dos dados mapeados pelos distintos algoritmos MDS. Semelhante aos resultados obtidos nos bancos anteriores, pode ser observado que o algoritmo FASTMAP apresenta os piores valores de pureza com respeito aos outros algoritmos nas primeiras 11 dimensões. Os outros algoritmos apresentam índices de pureza similares, sendo que a partir de 7 dimensões atingem-se índices de pureza relativamente estáveis próximos a 66.5%.

Figura 5.12: Índice de pureza de agrupamento do banco Imagens

BANCO DE DADOS SINTÉTICO

Nas Figuras 5.13 e 5.14 são apresentadas as respectivas curvas de *stress* para os 2 critérios de *stress* analisados. Pode-se observar que, de maneira semelhante aos resultados obtidos nos bancos anteriores, o algoritmo CMDS-SMACOF destaca-se apresentando menores valores de *stress* nas 3 primeiras dimensões com respeito aos demais algoritmos. Já para dimensionalidades maiores ou iguais a 3, todos os algoritmos nos 2 critérios de *stress* atingem a 0, confirmando o fato de que a verdadeira dimensionalidade deste banco é 3. Da mesma forma observa-se que o algoritmo FASTMAP mostra curvas com os maiores valores de *stress*, e que os algoritmos CMDS e LANDMARK apresentam similares valores de *stress*, corroborando-se também os resultados dos bancos anteriores.

Figura 5.13: Stress-1 para o banco Sintético

Figura 5.14: Stress-2 para o banco Sintético

Na Figura 5.15 são apresentadas as curvas em escala logarítmica do tempo computacional dos diversos algoritmos como função do número de dimensões. Semelhante aos resultados nos bancos anteriores, observa-se que a curva do algoritmo CMDS_SMACOF apresenta tempos computacionais significativamente maiores do resto de algoritmos e que seu *stress* cai consideravelmente na verdadeira dimensionalidade (3), devido ao mapeamento (CMDS) estar próximo do ótimo e, portanto, a convergência é alcançada rapidamente. Isto se verifica também com a análise de *stress*, a qual mostra que com 3 dimensões conseguem-se mapeamentos com baixo *stress* em todos os algoritmos. De maneira similar aos resultados anteriores,

o algoritmo LANDMARK com 10% de pontos *landmark* destaca-se com os menores tempos computacionais e quase insensíveis à dimensionalidade. O contrario é observado no algoritmo FASTMAP, o qual mostra tempos computacionais crescentes com o número de dimensões.

Figura 5.15: Tempo Computacional do banco Sintético (em escala logarítmica).

A Figura 5.16 apresenta o índice de pureza de agrupamentos realizados com número de grupos igual ao verdadeiro número de classes (5), variando-se a dimensionalidade dos dados mapeados pelos distintos algoritmos MDS. Diferentemente dos resultados obtidos nos bancos anteriores, pode ser observado que o algoritmo FASTMAP apresenta um ligeiro destaque em relação aos outros algoritmos, porém estas diferenças são muito pequenas (0.3%). Os maiores índices de purezas (próximos de 96.7%) são alcançadas a partir de 3 dimensões na maioria dos algoritmos, o qual coincide com a verdadeira dimensionalidade do banco de dados.

Figura 5.16: Índice de pureza de agrupamento do banco Sintético

DISCUSSÃO

Sumarizando, os resultados obtidos nos bancos de avaliação mostram que o desempenho dos algoritmos MDS quanto ao seu índice de *stress* não varia significativamente, destacando-se o algoritmo CMDS-SMACOF, o qual apresenta o melhor índice de *stress* ou equivalentemente a melhor precisão no mapeamento. No entanto este algoritmo apresenta um alto custo computacional quando comparado com os outros algoritmos. Em relação à capacidade de induzir melhores agrupamentos, todos os algoritmos apresentam índices semelhantes, sendo o FASTMAP o que apresenta o pior índice em baixas dimensões.

Com base nestes resultados, o algoritmo LANDMARK com 10% de pontos *landmark* iniciais estaria apresentando o melhor compromisso dos 3 índices de desempenho analisados. Isto é, uma precisão de mapeamento aceitável (unicamente superado por CMDS-SMACOF), o melhor desempenho em custo computacional e a capacidade de induzir agrupamentos nos dados mapeados tão bons quanto os outros algoritmos (exceto o FASTMAP que apresenta agrupamentos de pior qualidade em baixas dimensões)

A Tabela 5.1 apresenta uma comparação entre os índices de pureza obtidos nos bancos de avaliação quando foram submetidos ao algoritmo de classificação *K-Means*

(antes de seu ingresso no MDS) conhecendo-se *a priori* seu número de grupos. Também se apresenta nesta tabela os índices de pureza obtidos depois de utilizar-se o algoritmo LANDMARK com 10% de pontos iniciais. Nota-se que os índices de pureza são praticamente os mesmos, mostrando uma recuperação muito fiel das estruturas contidas nos dados originais. Isto estaria reforçando a seleção deste algoritmo para ser usado com o banco de estirpes *Bradyrhizobium*.

Tabela 5.1 - Tabela de Índices de purezas dos bancos de avaliação analisados com o algoritmo *K-Means*.

<i>Banco de dados</i>	<i>Índice de Pureza antes de MDS (original)</i>	<i>Índice de Pureza depois de LANDMARK 10%</i>
Iris	89.33%	89.33%
Câncer	96.19%	96.19%
Sintético	96.60%	96.70%
Imagens	66.90%	66.90%

5.2. RESULTADOS NO BANCO DE DADOS DE ESTIRPES *BRADYRHIZOBIUM*

Com o algoritmo LANDMARK com 10% de pontos *landmark* foram realizados os mapeamentos das 9 matrizes de distâncias do banco de estirpes de *Bradyrhizobium*. As Figuras 5.17 até 5.25 mostram respectivamente os mapeamentos de cada matriz de distância tomando as 3 primeiras dimensões. Cada figura mostra a distribuição de autovalores (em porcentagem) para cada mapeamento. As cores de cada ponto nesses mapeamentos correspondem à combinação de cores das coordenadas, segundo explicado na Seção 4.3. Abaixo de cada mapeamento são apresentados os 2 critérios de *stress* analisados para cada matriz. Note-se também que cada matriz é identificada pelo rótulo correspondente ao número da canaleta, região ribossomal e enzima de restrição, assim, por exemplo, a matriz D1-(16S-Cfo) identifica-se como: D(número da canaleta) - (região ribossomal - enzima de restrição). Em todas as matrizes analisadas no critério de *stress* resalta-se com um quadro verde o número que representa o menor valor de *stress* encontrado nos 2 critérios. Indica-se dessa forma que esta seria a dimensionalidade com a qual se pode reproduzir com uma alta

precisão a matriz de pontos mapeados. Isto também indicaria que a matriz analisada possui dimensões irrelevantes que não acrescentam maior informação. O fato de que os três critérios estejam representados na mesma figura não indica que se pretende fazer uma comparação entre critérios de *stress*, senão, encontrar a concordância de ambos os critérios na determinação da dimensionalidade com menor valor de *stress*.

No Apêndice 2 são apresentados os critérios de *stress* para os mapeamentos realizados com os outros algoritmos (CMDS, SMACOF, FASTMAP e LANDMARK). Somente são apresentados os resultados do critério de *stress* S1, já que o critério de *stress* S2 mostrou resultados muito similares em todos os casos.

Figura 5.17: (a) Visualização das três primeiras dimensões do mapeamento da Matriz D1 (16S- Cfo), (b) Distribuição de Autovalores por dimensionalidade, (c) Critérios de *stress* para a matriz D1 de *Bradyrhizobium*.

A Figura 5.17(a) mostra o mapeamento da matriz D1. Observa-se que nesta figura são distinguidos 3 possíveis grupos. O valor representado pelas 3 primeiras dimensões corresponde a 77% do total de autovalores (Ver Figura 5.17 b). A Figura 5.17(c) apresenta os 2 critérios de *stress*, ressaltando em um quadro verde o valor 5 como o menor valor de *stress* encontrado, correspondente ao critério S1 (Stress-1).

Figura 5.18: (a) Visualização das três primeiras dimensões do mapeamento da Matriz D2-(16S-Dde), (b) Distribuição de Autovalores por dimensionalidade, (c) Critérios de *stress* para a matriz D2 de *Bradyrhizobium*.

A Figura 5.18(a) de forma semelhante à análise da Figura 5.17, representa o mapeamento da matriz D2 e se observa 3 possíveis grupos. O valor representado pelas

3 primeiras dimensões corresponde a 82% do total de autovalores (Ver Figura 5.18 b). A Figura 5.18(c) apresenta os 3 critérios de *Stress*, ressaltando o valor 5 como o menor valor de *Stress* encontrado, correspondente ao critério S2 (Stress-2).

Figura 5.19: (a) Visualização das três primeiras dimensões do mapeamento da Matriz D3-(16S-Msp), (b) Distribuição de Autovalores por dimensionalidade, (c) Critérios de *stress* para a matriz D3 de *Bradyrhizobium*.

A Figura 5.19(a) de forma semelhante à análise das figuras anteriores representa o mapeamento da matriz D3 e observam-se 3 ou 4 possíveis grupos. O valor representado pelas 3 primeiras dimensões corresponde a 79% do total de autovalores (Ver Figura 5.19 b). A Figura 5.19 (c) apresenta os 2 critérios de *stress*, ressaltando o

valor 5 como o menor valor de *stress* encontrado, correspondente ao critério S1 (Stress-1).

Figura 5.20: (a) Visualização das três primeiras dimensões do mapeamento da Matriz D4-(23S-Hha), (b) Distribuição de Autovalores por dimensionalidade, (c) Critérios de *stress* para a matriz D4 de *Bradyrhizobium*.

A Figura 5.20(a) mostra o mapeamento da matriz D4 e observam-se 2 possíveis grandes grupos. O valor representado pelas 3 primeiras dimensões corresponde a 83% do total de autovalores (Ver Figura 5.20 b). A Figura 5.20(c) apresenta os 2 critérios de *stress*, ressaltando o valor 3 como o menor valor de *stress* encontrado, correspondente ao critério S1 (Stress-1).

Figura 5.21: (a) Visualização das três primeiras dimensões do mapeamento da Matriz D5-(23S-Hha), (b) Distribuição de Autovalores por dimensionalidade, (c) Critérios de *stress* para a matriz D5 de *Bradyrhizobium*.

A Figura 5.21(a) apresenta o mapeamento da matriz D5, observam-se 3 possíveis grupos. O valor representado 3 três primeiras dimensões corresponde a 80% do total de autovalores (Ver Figura 5.21 b). A Figura 5.21(c) apresenta os 2 critérios de *stress*, ressaltando o valor 4 como o menor valor de *stress* encontrado, correspondente ao critério S1 (Stress-1).

Figura 5.22: (a) Visualização das três primeiras dimensões do mapeamento da Matriz D6-(23S-Hinf), (b) Distribuição de Autovalores por dimensionalidade, (c) Critérios de *stress* para a matriz D6 de *Bradyrhizobium*.

A Figura 5.22(a) mostra o mapeamento da matriz D6 e observa-se 3 ou mais possíveis grupos, não sendo clara a separação entre grupos. O valor representado pelas 3 primeiras dimensões corresponde a 85% do total de autovalores (Ver Figura 5.22 b). A Figura 5.22(c) apresenta os 2 critérios de *stress*, ressaltando o valor 5 como o menor valor de *stress* encontrado, correspondente ao critério S1 (Stress-1).

Figura 5.23: (a) Visualização das três primeiras dimensões do mapeamento da Matriz D7-(IGS-Dde), (b) Distribuição de Autovalores por dimensionalidade, (c) Critérios de *stress* para a matriz D7 de *Bradyrhizobium*.

A Figura 5.23(a) representa o mapeamento da matriz D7 e observam-se 3 ou mais possíveis grupos. O valor representado pelas 3 primeiras dimensões corresponde a 63% do total de autovalores (Ver Figura 5.23 b). A Figura 5.23(c) apresenta os 2 critérios de *stress*, ressaltando o valor 9 como o menor valor de *stress* encontrado, correspondente ao critério S1 (Stress-1). Sendo este um valor alto, explica-se o porquê do menor valor encontrado nos autovalores, o qual indicaria que a informação contida nos dados esteja distribuída numa maior dimensionalidade.

Figura 5.24: (a) Visualização das três primeiras dimensões do mapeamento da Matriz D8-(IGS-Hae), (b) Distribuição de Autovalores por dimensionalidade, (c) Critérios de *stress* para a matriz D8 de *Bradyrhizobium*.

A Figura 5.24(a) mostra o mapeamento da matriz D8 e observam-se 4 ou mais possíveis grupos. O valor representado pelas 3 primeiras dimensões corresponde a 78% do total de autovalores (Ver Figura 5.24 b). A Figura 5.24(c) apresenta os 2 critérios de *stress*, ressaltando o valor 9 como o menor valor de *stress* encontrado, correspondente ao critério S1 (Stress-1). Semelhante à Figura 5.23(c) o alto valor no critério de *stress*, estaria explicando que a informação contida nos dados esteja distribuída em uma maior dimensionalidade e que a informação observada nas 3 primeiras dimensões não estaria apresentando muita informação.

Figura 5.25: (a) Visualização das três primeiras dimensões do mapeamento da Matriz D9-(IGS-Msp), (b) Distribuição de Autovalores por dimensionalidade, (c) Critérios de *stress* para a matriz D9 de *Bradyrhizobium*.

A Figura 5.25(a) apresenta o mapeamento da matriz D9, observam-se 4 ou mais possíveis grupos. O valor representado pelas 3 primeiras dimensões corresponde a 69% do total de autovalores (Ver Figura 5.25 b). A Figura 5.25(c) apresenta os 2 critérios de *stress*, ressaltando o valor 8 como o menor valor de *stress* encontrado, correspondente ao critério S1 (Stress-1). No entanto, esse valor é muito próximo dos valores 5, 6, ou 7 o qual pode indicar que também nestas dimensionalidades a informação estaria bem representada.

Na Figura 5.26 apresenta-se o mapeamento realizado com LANDMARK 10% em 3 dimensões para as 9 matrizes do banco *Bradyrhizobium*. Com um traço manual foram selecionados possíveis agrupamentos que capturam a percepção de agrupamentos de semelhança entre estirpes.

Figura 5.26: Visualização em três dimensões do banco *Bradyrhizobium*.

Tabela 5.2 - Tabela resumo de porcentagem de Autovalores acumulados nas 3 primeiras dimensões para o banco *Bradyrhizobium*.

R. Ribossomal	16S			23S			IGS		
Dimensão	D1	D2	D3	D4	D5	D6	D7	D8	D9
1	40	46	50	49	42	60	32	38	30
2	23	26	18	23	25	15	16	27	23
3	14	10	11	12	13	9	14	13	16
%Total Autovalores	77	82	79	83	80	85	63	78	69

Tabela 5.3 - Tabela resumo de mínimo valor do critério de *stress* encontrado no banco *Bradyrhizobium*.

R. Ribossomal	16S			23S			IGS		
Matriz	D1	D2	D3	D4	D5	D6	D7	D8	D9
Stress	5	5	5	3	4	5	9	9	8

DISCUSSÃO

A Figura 5.26 e as Tabelas 5.2 e 5.3 sumarizam os resultados obtidos no banco de estirpes *Bradyrhizobium*, mostrando que quanto ao desempenho, o algoritmo LANDMARK (com 10% de pontos iniciais *landmark*) apresento eficiência quanto a auxiliar na representação destes dados em 3 dimensões.

Como se pode ver a Figura 5.26 mostra que na maioria das estirpes encontra-se em 3 ou 4 grupos. Esses resultados de agrupamentos visuais concordam com análises de classificação, realizados em trabalhos de Villanueva (2007), Milagre (2003) e Christ (2007) onde foram encontrados semelhantes números de grupos, no entanto, a classificação nestes trabalhos não foi visual.

A Tabela 5.2 mostra que em 3 dimensões na maioria das matrizes de distância analisadas, conserva-se a informação acima de 63% nos autovalores, o que indica que a maior variância destes dados está nestas 3 primeiras dimensões. Isto também estaria indicando que o mapeamento visual realmente auxiliaria muito no processo de encontrar agrupamentos nesses dados, já que na maioria das vezes estaria mostrando uma boa representação.

A Tabela 5.3 resume o critério de menor valor de *stress* encontrado e estaria indicando que provavelmente a melhor representação está na matriz D4 - 23S - Hae e D5 - 23S - Hha . Também estaria indicando que as matrizes D7 - IGS - Dde , D8 - IGS - Hae e D9 - IGS - Msp não estariam sendo bem representadas no seu mapeamento visual em três dimensões dado que seus melhores critérios de *stress* caem em 8 ou 9 dimensões.

Capítulo 6

CONCLUSÕES E SUGESTÕES

Após a análise dos resultados, conclui-se de forma geral que as técnicas de redução dimensional MDS podem ser usadas de forma efetiva na redução dimensional e representação visual de dados genômicos obtidos pela técnica RFLP-PCR.

A análise comparativa dos diversos algoritmos MDS estudados sugerem que o algoritmo LANDMARK realiza mapeamentos com o menor tempo computacional e uma precisão de mapeamento comparável com os demais algoritmos, avaliados através dos critérios de *stress* adotados. Foi verificado também que o LANDMARK apresenta uma boa tendência de manter os grupos existentes nos dados após do mapeamento. Enfatiza-se que estes resultados são válidos para os critérios de avaliação propostos no presente trabalho. Comparações com dados de literatura devem ser cuidadosas, levando-se em consideração diferenças nas condições experimentais praticadas.

Com base na análise comparativa, escolheu-se o algoritmo LANDMARK com (10%) de pontos *landmark* para realizar a representação visual do banco de estirpes de *Bradyrhizobium*. Encontrou-se que o número de dimensões significativas neste banco de dados varia de acordo com a região ribossomal analisada; assim, para a região ribossomal 16S (matrizes D1, D2 e D3) a informação relevante estaria nas 5 primeiras dimensões. Para a região ribossomal 23S (matrizes D4, D5 e D6) a informação relevante estaria nas 4 primeiras dimensões. Já para a região ribossomal IGS (matrizes D7, D8 e D9) a dimensionalidade relevante seria 9. Isso significaria que a região IGS apresenta a maior riqueza de informação, o que concorda com o fato de que esta é a região de maior variabilidade genética intra-espécie. Também foi observado que tomando as 3 primeiras dimensões nos respectivos mapeamentos das 9 matrizes

obtêm-se uma perda máxima de informação de 40% (matriz D7), o qual pode ser aceitável para uma representação visual exploratória inicial.

A ferramenta MDSExplorer criada para realizar a análise comparativa das técnicas MDS serviu não somente para o processo de geração de testes e avaliação dos algoritmos MDS, mas também, devido à sua funcionalidade de visualização dos mapeamentos em 2D e 3D, foi usada para realizar a exploração visual dos dados no banco de estirpes de *Bradyrhizobium*. É importante notar, que os resultados obtidos com esta ferramenta são dependentes da métrica de distância selecionada, necessária para gerar a matriz de distância. Se esta métrica for capaz de gerar boas diferenças entre as informações dos dados então o algoritmo de redução dimensional tenderá a gerar boas representações. Também, os resultados dependem da escolha do número de dimensões. Existe um equilíbrio entre o número de dimensões e a evolução dos valores de *stress*, já que, quanto menor é o *stress* melhor é qualidade da representação, mais quase sempre maior é o número de dimensões. Todos estes inconvenientes influenciam os resultados, precisando-se sempre da habilidade do usuário para interpretar os mapeamentos gerados. Precisamente esta ferramenta apresenta diferentes opções para que o usuário interaja e observe a sua influencia nos resultados. Assim, esta ferramenta constitui uma contribuição do presente trabalho e sugere-se sua utilização em bancos de dados genômicos semelhantes.

SUGESTÕES PARA TRABALHOS FUTUROS

E sugerido para trabalhos futuros o estudo de algoritmos MDS que realizem um único mapeamento a partir de várias matrizes de distância. Esta necessidade foi encontrada no banco de estirpes de *Bradyrhizobium*, no qual existem 9 matrizes de distâncias obtidas a partir da análise de 3 regiões ribossomais com 3 enzimas de restrição. Nesse, caso foram obtidos 9 mapeamentos do mesmo conjunto de bactérias, mas para o especialista é importante visualizar uma única representação das bactérias. Seria interessante implementar um algoritmo que represente visualmente o consenso ou desacordo entre todos os mapeamentos que se dispõe.

REFERÊNCIAS BIBLIOGRÁFICAS

- ABDI, H.; VALENTIN, D.; O'TOOLE, A. J.; CHOLLET, S.; CHREA, C. (2007). Analyzing Assessors and Products in Sorting Task: DISTATIS, **Theory and Applications. Food Quality and Preference**. Volume 18, Issue 4, pp. 627-640, 2007.
- ABDI, H.; VALENTIN, D.; O'TOOLE, A. J.; EDELMAN, B. (2005). DISTATIS: The Analysis of Multiple Distance Matrices. San Diego, CA, USA .IEEE Computer Society. **Conference on Computer Vision and Pattern Recognition (CVPR'05) Workshops - Volume 03**, pp 42–47, 2005.
- AGRAFIOTIS, D. K.; RASSOKHIN, D. N.; LOBANOV, V.S. (2000). Multidimensional Scaling and Visualization of Large Molecular Similarity Tables. **Journal of Computational Chemistry**, Vol. 22, No. 5, pp. 488–500 (2001).
- ARAÚJO, F. F.; HUNGRIA, M. (1999). Nodulação e rendimento de soja co-infectada com *Bacillus Subtilis* e *Bradyrhizobium Japonicum* / *Bradyrhizobium Elkanii*. **Revista de Pesquisa Agropecuária**. Bras., Set 1999, vol.34, no.9, p.1633-1643. ISSN 0100-204X
- ASUNCION, A.; NEWMAN, D. J. (2007). **UCI Machine Learning Repository** [<http://www.ics.uci.edu/~mlern/MLRepository.html>]. Irvine, CA: University of California, School of Information and Computer Science, 2007.
- BASALAJ, W. (2000). **Proximity Visualization of Abstract Data**. PhD thesis, University of Cambridge, 2000.
- BECKER, R. A.; CLEVELAND, W.S.; SHYU, M. (1996). The Visual Design and Control of Trellis Display, **Journal of Computational and Graphical Statistics**, Vol. 5(2), pp. 123-155, 1996.
- BECKMANN, H.; GATTAZ, W. F. (2002). Multidimensional analysis of the

- concentrations of 17 substances in the CSF of schizophrenics and controls. **Journal of Neural Transmission**, Volume 109, Numbers 5-6 / May, 2002
- BEDDOW, J. (1990). Shape coding of multidimensional data on a microcomputer display. In Arie Kaufman, editor, **Proceedings of IEEE Visualization '90**, pp. 238-246, 1990.
- BEVILACQUA, S. (2004). O emprego da Multidimensional Scaling: estudo de caso envolvendo seis instituições de ensino superior do noroeste paulista, uma contribuição para a Qualidade em Serviços. **Revista Pesquisa e Desenvolvimento Engenharia de Produção**. N.3, pp. 43-53, 2004.
- BISHOP, C. M. (1995). **Neural Networks for Pattern Recognition**. Oxford University, 1995.
- BORG, I.; GROENEN, P. (2005). **Modern Multidimensional Scaling: Theory and Applications**. Second Edition. Springer Press, 2005.
- BORGES, H. B. (2006). **Redução de Dimensionalidade de Atributos em Bases de Dados de Expressão Gênica**. Dissertação (Mestrado) – Pontifícia Universidade Católica do Paraná. Programa de Pós Graduação em Informática. Curitiba, 2006.
- CARD, S. K.; MACKINLAY, J. D.; SHNEIDERMAN, B. (1999). Information Visualization. Readings **In Information Visualization: Using Vision to Think**, Written and edited by Stuart K. Card, Jock D. Mackinlay, Ben Shneiderman, 1999.
- CARMO, M. B. (2003). **Visualização de Informação Modelo Integrado para o Tratamento de Filtragem e Múltiplas Representações**. Dissertação (Doutorado). Departamento de Informática. Faculdade de Ciências da Universidade de Lisboa. Campo Grande, 1749-016 Lisboa Portugal, 2003.
- CARREIRA, P. M. (1997). **A Review of Dimension Reduction Techniques**. Technical Report CS-9609. Dept. of Computer Science. University of Sheffield, 1997.

- CARROL, D. J.; GREEN, P. (1997). Psychometric Methods in Marketing Research: Part II, Multidimensional Scaling. **Journal of Marketing Research**, Vol. 34, No 2, pp. 193-204, 1997.
- CHERNOFF, H. (1973). The use of faces to represent points in k-dimensional space graphically. **Journal of the American Statistical Association**, Vol. 68, No 342, pp. 361-368, 1973.
- CHIZI, B.; MAIMON, O. (2005). Dimension reduction and Feature Selection. In Odded Maimon and Lios Rokach, editors. **The Data Mining and Knowledge Discovery Handbook**. Springer, pp 93-111, 2005.
- CHRIST, R. E. (2007). **Classificação de bactérias do gênero Bradyrhizobium usando uma rede neural ART2 com dados de eletroforese de genes ribossomais**. Dissertação (Mestrado) apresentada à Escola de Engenharia de São Carlos da Universidade de São Paulo. 2007.
- CHUEIRI, W. A.; PAJARA, F.; BOZZA, D. (2005). Importância da Inoculação e Nodulação na Cultura da Soja. **Revista de divulgação Técnica MANAH**. Nro.169, Set, Out e Nov de 2005.
- COX, D. J. (1990). **The Art of Scientific Visualization**, Academic Computing, vol.4, nro. 6, pp. 20, 1990.
- COX, T. F.; COX, M. A. A. (2000). **Multidimensional scaling**. Second Edition. London: Chapman & Hall, 2000.
- DAMIANCE, A. P. G. (2006). **Desenvolvimento de modelos dinâmicos para a formação de clusters aplicados em dados biológicos**. Dissertação (Mestrado) apresentada ao Instituto de Ciências Matemáticas e Computação ICMC/USP São Carlos, 2006.
- DASARATHY, B. (1980). Nosing around the neighborhood: A new system structure and classification rule for recognition in partially exposed environments. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, Vol.2, No.1, pp.

67-71, 1980.

DAVISON (2006). Department of Biology, Davidson College. Disponível em: <<http://www.bio.davidson.edu/COURSES/genomics/method/RFLP.html>>. Acesso em: 5 nov., 2006.

DUDA, R. O.; HART, P. E.; STORK, D. G. (2001). **Pattern Classification**. Second Edition. A Wiley-Interscience Publication John Wiley and Sons, Inc. 2001.

EVERITT, B. S.; LANDAU, S.; LEESE, M. (2001). **Cluster Analysis**. Oxford University Press Inc., New York, 4 ed., 2001.

FALOUTSOS, C.; LIN, K. (1995). FastMap: A fast algorithm for indexing, data-mining and visualization of traditional and multimedia datasets. **In Proc. ACM SIGMOD**, Vol. 24, No. 2, pp. 163–174, 1995.

FRUCHTERMAN, T.; REINGOLD, E. (1991). Graph drawing by force-directed placement. **Software- Practice and Experience**, Vol. 21, No. 11, pp. 1129–1164, 1991.

GERMANO, M. G.; MENNA, P.; MOSTASSO, F. L.; HUNGRIA, M. (2006). RFLP analysis of the RNA operon of a Brazilian collection of Bradyrhizobial strains from 33 legume species. **International Journal of Systematic and Evolutionary Microbiology**, Vol. 56, No. 1, pp. 217–229, 2006.

GROENEN, P. J. F.; WINSBERG, S.; RODRÍGUEZ, O.; DIDAY, E. (2006). I-Scal: Multidimensional scaling of interval dissimilarities. **Computational Statistics & Data Analysis**, pp. 360–378, 2006.

GROENEN, P. J.; VELDEN, V. V. (2004). **Multidimensional Scaling**. Econometrix Institute Report EI 2004-15

HÉBERT, P. A.; MASSON, M. H.; DENŒUX, T. (2006). Fuzzy multidimensional scaling. **Computational Statistics & Data Analysis**, pp. 335–359, 2006.

HUANG, S.; WARD M.; RUNDENSTEINER, E. (2005). Exploration of

- dimensionality reduction for text visualization. Coordinated and Multiple Views in Exploratory Visualization, Proceedings: **Third International Conference on Digital Object Identifier**, Vol. 10.1109/CMV, pp. 63-74, 2005.
- HUNGRIA, M.; VARGAS, M.A.T.; CAMPO, R.J. (1997). A inoculação da soja. Londrina: Embrapa-CNPSO, 1997. 20p. (Embrapa-CNPSO. Circular técnica, 17).
- INSELBERG, A. (1985). The Plane with Parallel Coordinates. **Special Issue on Computational Geometry**. Visual Computer, Vol. 1, pp. 69-91, 1985.
- INSELBERG, A.; DIMSDALE, B. (1990). Parallel Coordinates: A Tool for Visualizing Multidimensional Geometry, in **Proceedings of IEEE Visualization'90**, pp. 361- 375, 1990.
- IZMAILOV, C.A.; SKOLOV, E. N.; KORSHUNOVA, S.G. (2005). Multidimensional scaling reliability in similarity judgments about environmental sentences. **The Spanish Journal of Psychology**, Volume 8, No. 2, pp. 119-133, 2005. ISSN 1138-7416
- JAIN, A. K.; DUIN, P. W.; MAO, J. (2000). Statistical pattern recognition: A review. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, Vol. 22, No. 1, pp. 4-37, 2000.
- LEEuw, J.; HEISER, W. (1982). **Theory of multidimensional scaling**. North-Holland, Krishnaiah P.R. and Kanal L.N Handbook of statistics, Vol. 2, pp. 285–316, 1982.
- LEVKOWITZ, H. (1991). Color Icons: Merging Color and Texture Perception for Integrated Visualization of Multiple Parameters, **IEEE Visualization '91 Proceedings**, pp. 164–170, 1991.
- LIMA, A. S.; ANDRADE, J. P.; MOREIRA, F. M. (2005). Diversidade fenotípica e eficiência simbiótica de estirpes de *Bradyrhizobium* SPP. de solos da Amazônia. **Revista de Pesquisa agropecuária Brasileira**, Nov 2005, vol.40, no.11, p.1095-1104. ISSN 0100-204X

- MAIMOM, O; ROKACH, L. (2005). **The Data Mining and Knowledge Discovery Handbook**. Springer. 2005.
- MARTIN, K.; SHROEDER, W.; LORENSEN, B. (2003) The Visualization Toolkit An Objet-Oriented Approach To 3D Graphics. Kitware Inc., 3rd edition, 2003 <http://public.kitware.com/VTK/>
- MILAGRE, S. T. (2003). **Análise de Estabilidade de CLUSTER em uma coleção brasileira de bactérias siazotróficas do BRADYRHIZOBIUM**. Dissertação (Mestrado) apresentada ao Departamento de Engenharia Elétrica da Universidade Estadual de Londrina, 2003.
- MOINS, S. (2002). **Implementation of a simulated annealing algorithm for Matlab** (Thesis), Institutionen för Systemteknik. University Linköping. Sweden, 2002.
- MÜLLER, W; ALEXA, M. (1998). Using Morphing for Information Visualization. **CIKM 98 Workshop on New Paradigms in Information Visualization and Manipulation (NPIV 98)**, 1998.
- NASCIMENTO, H. A.; FERREIRA C. B. (2005). Visualização de Informações - Uma Abordagem Prática. **XXV Congresso da Sociedade Brasileira de Computação**. Unisinos, São Leopoldo-RS, Cap. 2, pp. 1262 -1312, 2005.
- NAUD, A. (2001). **Neural and Statistical Methods for the Visualization of Multidimensional Data**. Ph.D. thesis, University of Mikolaja Kopernika Toruniu (Nicolaus Copernicus University), Poland 2001.
- NÓBREGA, R. S. A.; MOREIRA, F. M. S.; SIQUEIRA, J. O.; LIMA, A. S. (2004). Caracterização fenotípica e diversidade de bactérias diazotróficas associativas isoladas de solos em reabilitação após a mineração de bauxita. **Revista Brasileira de Ciência do Solo**, mar./apr. 2004, vol.28, no.2, p.269-279. issn 0100-0683
- PICKETT, R. M., GRIENSTEIN, G. (1988). Iconographic Displays for Visualizing Multidimensional Data, **Proceedings of IEEE International Conference on Systems, Man, and Cybernetics**, China, pp. 514–519, 1988.

- PICKETT, R. M. (1970). **Visual Analyses of Texture in the Detection and Recognition of Objects, Picture Processing and Psycho-Pictories**, Academic Press, New York, 1970.
- PLATT, J. C. (2005). FastMap, MetricMap, and Landmark MDS are all Nyström Algorithms. Proc. **10th International Workshop on Artificial Intelligence and Statistics**, pp. 261-268, (2005).
- PRIEGO, J.L.O. (2003). A Vector Space Model as a methodological approach to the Triple Helix dimensionality: A comparative study of Biology and Biomedicine Centres of two European National Research Councils from a Webometric view. **Journal of Scientometrics**, Volume 58, Number 2 / October, 2003.
- RIPLEY, B. D. (1996). **Pattern Recognition and Neural Networks**. Cambridge University Press, Cambridge. England, pp 305-311, 1996.
- SAMMON, J. W. (1969). A nonlinear mapping for data analysis. **IEEE Transactions on Computers**, C-18(5), pp. 401–409, 1969.
- SCHROEDER, M.; GILBERT, D; HELDEN, J. V.; NOY, P. (2001). Approaches to visualization in bioinformatics: from dendograms to Space Explorer. **Journal of Information Sciences**, Vol. 139, No 1-2, pp. 19-57, 2001.
- SCHROEDER, W. J.; MARTIM, K.; LORENSEN, B. (2002). **The Visualization Toolkit - An Object-Oriented Approach to 3D Graphics**, 3 ed., Prentice-Hall, 2002.
- SCOTT, D. W. (1992). **Multivariate Density Estimation: Theory, Practice, and Visualization**. Wiley Series in Probability and Mathematical Statistics. New York, London, Sydney, 1992.
- SILVA, V.; TENENBAUM, J. B. (2003). Global versus local methods in nonlinear dimensionality reduction. In S. Becker, S. Thrun, and K. Obermayer, editors, **Proc. NIPS**, Vol. 15, pp. 721–728, 2003.

- SILVA, V; TENENBAUM, J. B. (2004). **Sparse multidimensional scaling using landmark points**. Stanford Mathematics Technical Report, 2004.
- SNEATH, P. H. A.; SOKAL, R. R. (1973). **Numerical Taxonomy**. San Francisco: W. H. Freeman & Co.
- TAKANE, Y; YOUNG, F. W.; LEEUW, J. (1977). Nonmetric individual differences multidimensional scaling: an alternating least squares method with optimal scaling features. **Journal of Psychometrika**, Vol. 42, pp. 7-67, 1977.
- TENG, L.; LI, H.; FU, X.; CHEN, W.; SHEN, I. (2005). Dimension Reduction of Microarray Data Based on Local Tangent Space Alignment. **In Fourth IEEE Conference on Cognitive Informatics, 2005 (ICCI 2005)**. pp. 154-159. Digital Object Identifier: 10.1109/COGINF.2005.1532627
- VARGAS, L. K.; LISBOA, B. B.; SCHOLLES, D.; SILVEIRA, J. R. P.; JUNG, G. C.; GRANADA, C. E.; NEVES, A. G.; BRAGA, M. M.; NEGREIROS, T. (2007). Diversidade genética e eficiência simbiótica de rizóbios noduladores de acácia-negra de solos do rio grande do sul. **Revista Brasileira de Ciência do Solo**, Nro. 31, pp. 647-654, 2007.
- VENKATARAJAN, M. S.; BRAUN, W. (2001). New quantitative descriptors of amino acids based on multidimensional scaling of a large number of physical-chemical properties. **Journal of Molecular Modeling**. ISSN 1610-2940, Volume 7, Number 12, pp 445-453. December, 2001
- VILLANUEVA, E. R. (2007). **Métodos Bayesianos aplicados em taxonomia molecular**. Dissertação (Mestrado) apresentada à Escola de Engenharia de São Carlos da Universidade de São Paulo, 2007.
- ZAHA, A (2000). **Biologia Molecular Básica**, 2da. Ed., Porto Alegre: Mercado Aberto, Brasil, 2000.

APÊNDICES

APÊNDICE 1: Tabelas de tempo computacional

As tabelas a seguir apresentam o tempo computacional medido em segundos, para os bancos de dados de avaliação.

Tabela 6.1: Tempo computacional para o banco de dados Iris.

	<i>FASTMAP</i>	<i>LANDMARK</i> 15 Pontos (10%)	<i>LANDMARK</i> 30 Pontos (20%)	<i>LANDMARK</i> 45 Pontos (30%)	<i>CMDS</i>	<i>SMACOF</i>
1	0.02	0.01	0.02	0.02	0.09	0.14
2	0.02	0.01	0.01	0.01	0.09	1.41
3	0.02	0.01	0.01	0.02	0.09	3.36
4	0.02	0.02	0.01	0.01	0.09	0.70
5	0.03	0.01	0.02	0.02	0.09	4.52
6	0.03	0.01	0.01	0.01	0.09	4.34
7	0.03	0.01	0.01	0.01	0.09	4.62
8	0.05	0.01	0.01	0.02	0.09	4.67
9	0.05	0.01	0.02	0.01	0.09	4.69
10	0.05	0.01	0.01	0.02	0.09	4.70

Tabela 6.2: Tempo computacional para o banco de dados Câncer.

	<i>FASTMAP</i>	<i>LANDMARK</i> 68 Pontos (10%)	<i>LANDMARK</i> 137 Pontos (20%)	<i>LANDMARK</i> 205 Pontos (30%)	<i>CMDS</i>	<i>SMACOF</i>
1	0.09	0.02	0.09	0.28	7.55	10.06
2	0.22	0.02	0.09	0.28	7.55	104.38
3	0.33	0.03	0.11	0.27	7.55	116.43
4	0.42	0.02	0.09	0.27	7.55	117.64
5	0.53	0.03	0.09	0.27	7.55	118.61
6	0.66	0.03	0.09	0.27	7.55	119.24
7	0.78	0.03	0.09	0.27	7.55	120.38
8	0.91	0.03	0.09	0.27	7.55	120.89
9	1.02	0.02	0.09	0.27	7.55	21.08
10	1.17	0.03	0.09	0.27	7.55	122.41

Tabela 6.3: Tempo computacional para o banco de dados Imagens.

	<i>FASTMAP</i>	<i>LANDMARK</i> 210 Pontos (10%)	<i>LANDMARK</i> 420 Pontos (20%)	<i>LANDMARK</i> 630 Pontos (30%)	<i>CMDS</i>	<i>SMACOF</i>
1	1.266	1.515	2.375	8.344	335.67	51.766
2	2.11	0.422	2.406	8.391	335.67	767.17
3	3.016	0.406	2.375	8.453	335.67	1022.1
4	4.078	0.375	2.375	8.203	335.67	1031.9
5	5.156	0.375	2.391	8.422	335.67	1042.1
6	6.234	0.375	2.328	8.297	335.67	1052.4
7	7.265	0.375	2.328	8.766	335.67	1062
8	8.282	0.391	2.468	8.437	335.67	1066.8
9	9.313	0.391	2.391	8.203	335.67	1077.3
10	10.36	0.375	2.453	8.328	335.67	1083.5
11	11.485	0.391	2.39	8.61	335.67	1096.8
12	19.078	0.39	2.437	8.219	335.67	1100.4
13	28.484	0.391	2.453	8.406	335.67	1114.3
14	40.078	0.39	2.406	8.422	335.67	258.36
15	55.172	0.407	2.593	8.422	335.67	46.985
16	69.531	0.391	2.5	8.485	335.67	120.8
17	77.469	0.391	2.5	8.593	335.67	165.75
18	96.188	0.406	2.406	8.438	335.67	485.25
19	115.484	0.406	2.391	8.61	335.67	636.55
20		0.407	2.421	8.219	335.67	1174

Tabela 6.4: Tempo computacional para o banco de dados Sintético.

	<i>FASTMAP</i>	<i>LANDMARK</i> 100 Pontos (10%)	<i>LANDMARK</i> 200 Pontos (20%)	<i>LANDMARK</i> 300 Pontos (30%)	<i>CMDS</i>	<i>SMACOF</i>
1	0.25	0.06	0.28	0.92	33.20	41.48
2	0.45	0.06	0.28	0.91	33.20	66.81
3	0.67	0.06	0.31	0.91	33.20	39.91
4	0.91	0.06	0.28	0.91	33.20	72.72
5	1.14	0.06	0.30	0.88	33.20	272.04
6	1.38	0.06	0.28	0.95	33.20	274.56
7	1.64	0.06	0.28	0.92	33.20	276.04
8	1.88	0.08	0.30	0.94	33.20	277.37
9	2.11	0.06	0.28	0.89	33.20	279.11
10	2.34	0.06	0.30	0.97	33.20	280.98

APÊNDICE 2: Evolução do *stress* S1 para o banco *Bradyrhizobium*

As figuras a seguir apresentam a evolução do critério de *stress* S1 medido com 4 algoritmos MDS (CMDs, SMACOF, FASTMAP e LANDMARK) para o banco de dados *Bradyrhizobium*. Nesse critério, o algoritmo FASTMAP apresenta menor desempenho (respeito a os outros algoritmos), diferentemente do algoritmo SMACOF que como era esperado apresenta o melhor desempenho. O algoritmo LANDMARK apresenta desempenho similar quando são tomados 10, 20 e 30% de pontos iniciais *landmark*, incluso em muitas matrizes apresentam sobreposição (matrizes D1, D3, D5, D7, D9). No caso do algoritmo CMDs, apresenta em todas as figuras desempenho similar e sobreposição como o algoritmo LANDMARK.

Figura 6.1: Stress-1 para a matriz D1 do banco *Bradyrhizobium*.

Figura 6.2: Stress-1 para a matriz D2 do banco *Bradyrhizobium*.

Figura 6.3: Stress-1 para a matriz D3 do banco *Bradyrhizobium*.

Figura 6.4: Stress-1 para a matriz D4 do banco *Bradyrhizobium*.

Figura 6.5: Stress-1 para a matriz D5 do banco *Bradyrhizobium*.

Figura 6.6: Stress-1 para a matriz D6 do banco *Bradyrhizobium*.

Figura 6.7: Stress-1 para a matriz D7 do banco *Bradyrhizobium*.

Figura 6.8: Stress-1 para a matriz D8 do banco *Bradyrhizobium*.

Figura 6.9: Stress-1 para a matriz D9 do banco *Bradyrhizobium*