Universidade Federal de Pernambuco
Centro de Ciências Exatas e da Natureza
Centro de Informática

Doutorado em Matemática Computacional

Bi-clustering de Dados Genéticos Binários Baseado em Modelos de Classificação Logística

Carla Claudia da Rocha Rego Monteiro

Tese de Doutorado

Recife
25 de Novembro de 2009

Universidade Federal de Pernambuco
Centro de Ciências Exatas e da Natureza
Centro de Informática

Carla Claudia da Rocha Rego Monteiro

Bi-clustering de Dados Genéticos Binários Baseado em Modelos de Classificação Logística

Trabalho apresentado ao Programa de Doutorado em Matemática Computacional do Centro de Ciências Exatas e da Natureza Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Doutor em Matemática Computacional .

Orientadora: *Profa. Dra. Katia S. Guimarães*

Recife
25 de Novembro de 2009

Monteiro, Carla Claudia da Rocha Rego
Bi-clustering de dados genéticos binários baseado em modelos de classificação logística / Carla Claudia da Rocha Rego Monteiro. - Recife: O Autor, 2009.
xix, 84 p. : il., fig., tab.

Tese (doutorado) – Universidade Federal de Pernambuco. CCEN. Matemática Computacional, 2009.

Inclui bibliografia.

1. Clustering. 2. Bi-clustering. 3. Regressão logística. I. Título.

519.5 CDD (22. ed.) MEI2010 – 052

UFPE 60 ANOS

UNIVERSIDADE FEDERAL DE PERNAMBUCO
Centro de Ciências Exatas e da Natureza
Doutorado em Matemática Computacional

PARECER DA BANCA EXAMINADORA DE DEFESA DE TESE DE DOUTORADO

CARLA CLAUDIA DA ROCHA REGO MONTEIRO

"BI-CLUSTERING DE DADOS GENÉTICOS BINÁRIOS BASEADO EM MODELOS DE CLASSIFICAÇÃO LOGÍSTICA"

A Banca composta pelos Professores: KATIA SILVA GUIMARÃES, do Centro de Informática da UFPE, MARCÍLIA CAMPOS ANDRADE, do Centro de Informática da UFPE; SÍLVIO DE BARROS MELO, do Centro de Informática da UFPE; MARIA EMÍLIA MACHADO TELLES WALTER, do Departamento de Ciência da Computação da UnB; e RENATA MARIA CARDOSO RODRIGUES DE SOUZA, do Centro de Informática da UFPE; considera a Tese da candidata:

(X) APROVADA ( ) REPROVADA ( )EM EXIGÊNCIA

Secretaria do Programa de Pós-Graduação em Matemática Computacional do Centro de Ciências Exatas e da Natureza da Universidade Federal de Pernambuco, aos 25 dias do mês de novembro de 2009.

PROFª. KATIA SILVA GUIMARÃES
PRESIDENTE E 1º EXAMINADOR

PROFª. MARCÍLIA CAMPOS ANDRADE
2º EXAMINADOR

PROF. SÍLVIO DE BARROS MELO
3º EXAMINADOR

PROFª. MARIA EMÍLIA MACHADO TELLES WALTER
4º EXAMINADOR

PROFª. RENATA MARIA CARDOSO RODRIGUES DE SOUZA
5º EXAMINADOR

Aos meus amores Rafael, Juliana e Cristiano.

Agradecimentos

Agradeço a todos aqueles que contribuíram para a conclusão deste trabalho.

Ao meu esposo Cristiano pelo apoio e presença em todos os momentos.

Ao meu filho Rafael pela compreensão e paciência com meus períodos de ausência.

À minha querida filha Juliana pelos sorrisos de animação todas as manhãs.

Aos meus pais Vilma e José pelo incentivo e apoio que foram essenciais para realização deste trabalho.

Aos meus sogros, irmãos e cunhados pelo incentivo.

Às minhas eternas amigas Jaqueline, Fabíola e Roberta pelo apoio e incentivo durante toda a jornada.

Aos amigos "amesianos" Jonilda, Antônio, Patrícia, Vicente, Claudia e João, pela força nos momentos difíceis e alegria nos momentos de descontração.

Aos colegas e funcionários do Departamento de Estatística da UFPE.

Aos amigos Cristina Raposo, Audrey Cysneiros, Francisco Cysneiros, Cláudia Lima, Issac Xavier, Carla Vivacqua e André Pinho pelo incentivo e apoio.

À Katia Guimarães pela orientação e apoio.

Aos professores Silvio Melo e Francisco Cribari pelo apoio.

Ao Yamanishi pelo fornecimento dos dados biológicos.

Ao Eduardo Gusmão pela implementação do programa dos gráficos de bi-clustering.

Ao CNPq pelo suporte financeiro.

Resumo

Informações de interações de proteínas são fundamentais para a compreensão dos processos celulares. Por esta razão, várias abordagens têm sido propostas para inferir sobre pares de proteínas de redes de todos os tipos de dados biológicos. Nesta tese é proposto um método de bi-clustering, Lbic, baseado num modelo de classificação logística, para analisar dados biológicos binários. O Lbic é comparado com outros dois métodos de bi-clustering apresentados na literatura, mostrando melhores resultados. Seu desempenho também é comparado àqueles de um método supervisionado, análise de correlação canônica com Kernel, aplicado aos mesmos conjuntos de dados. Os resultados mostram que o Lbic alcança desempenho superior aos da aborgadem supervisionada treinada com até 25% do conhecimento da rede alvo.

Palavras-chave: bi-clustering, modelo de classificação logística, dados binários, interação proteína-proteína, combinação de métodos.

Abstract

Protein interaction information is fundamental to understand the cellular processes. Due to that, much is being done to automatically infer protein networks from all types of biological data. This work proposes a biclustering method, Lbic, based on a logistic model, to analyze biological data in binary format. The proposed method is compared with two other biclustering methods from literature, showing improved results. Its performance is also compared to the ones from a supervised method, kernel canonical correlation analysis, applied to the same data sets. The experiments show that the proposed method achieves a performance very close to the supervised approach trained with up to 25% of knowledge of the target network.

Keywords: biclustering, logistic classification model, binary data, protein-protein interaction, methods combination.

Sumário

1 Introdução **1**

2 Trabalhos Relacionados **5**

2.1 Métodos de Agrupamento 5

2.1.1 Clustering 6

2.1.2 Bi-Clustering 9

2.1.2.1 Aspectos do Bi-cluster 10

2.1.2.2 Abordagens de Bi-clustering para Dados de Expressão de Genes [MO04] 15

2.1.2.3 Abordagem Bi-clustering para Dados de Expressão de Genes Método Plaid [TBKH05] 23

2.1.2.4 Abordagem de Biclustering para Dados Genômicos Binários Método Bicbin [UW08] 27

2.2 Métodos de Inferência de Proteína-Proteína 30

2.2.1 Análise de Correlação Canônica com Kernel Supervisionada ACCKS [YVK04] 33

2.3 Conclusões 36

3 Modelos Estatísticos **37**

3.1 Modelos lineares 37

3.1.1 Caso 1: Modelo de Regressão Múltipla 39

3.1.2 Caso 2: Modelo de Classificação 39

3.1.2.1 Modelo de Análise de Covariância 46

3.2 Modelos Não Lineares 48

3.2.1 Caso 1: Modelos de Regressão Logística 48

3.2.2 Caso 2: Modelo de Classificação Logística 52

3.2.2.1 Modelo de Classificação Logística com uma Covariável 54

3.3 Conclusões 56

4 Método Lbic **57**

4.1 Modelo Lbic 57

4.2 Algoritmo Lbic 59

4.3 Conclusões 63

5 Aplicações **65**
5.1 Dados Experimentais 65
5.2 Parâmetros 66
5.3 Comparações 68
5.3.1 ACCKS com várias supervisões 71
5.3.2 Lbic versus Bicbin para dados artificiais 72
5.3.3 Lbic versus Bicbin para dados filogenéticos 73
5.3.4 Combinação (Lbic e Plaid) versus Lbic e Plaid 73
5.3.5 Lbib versus ACCKS para dados filogenéticos 74
5.3.6 Combinação (Lbib, Plaid) versus Integração (ACCKS) 75
5.4 Conclusões 76

6 Considerações Finais **79**
6.1 Contribuições 79
6.2 Resultados 79
6.3 Trabalhos Futuros 80

Referências Bibliográficas **81**

Lista de Figuras

1.1 Exemplo de rede de proteínas 1
1.2 Identificação de Bi-cluster 3

2.1 Matriz original com três bi-clusters 9
2.2 Aspectos do Bi-cluster - Plaid 27
2.3 Aspectos do Bi-cluster - Bicbin 30

4.1 Aspectos de bi-clusters a serem identificados 57
4.2 Algoritmo Lbic 62

5.1 Exemplo de gráfico VPP$\times$ S 69
5.2 ACCKS com supervisões 10%, 25%, 50% e 90% 71
5.3 Lbic e BicBin para dados artificiais 72
5.4 Lbic (dados filogenéticos), Plaid (dados de expressões) e Combinação (Lbic, Plaid) 74
5.5 Métodos Lbic e ACCKS para dados filogenéticos 75
5.6 Combinação (Lbic$_{PHY}$ e Plaid$_{EXP}$) e Integração ACCKS$_{EXP,PHY}$ 76

Lista de Tabelas

2.1 Matriz de Dados Genômicos 6
2.2 Resumo dos Métodos de Bi-Clustering 22

5.1 Método Plaid para dados de expressões de genes 67
5.2 Método Bicbin para dados artificiais 67
5.3 Método Lbic para dados filogenéticos 67
5.4 Método Lbic para dados artificiais 67
5.5 Método ACCKS para dados filogenéticos e de expressões de genes 68

CAPÍTULO 1

Introdução

A inferência de redes de proteínas tem sido tema de grande relevância na biologia computacional, uma vez que as interações entre proteínas, chamadas de interações proteína-proteína, estão relacionadas com as suas funções biológicas. A compreensão destas funções é de grande importância, pois permite, por exemplo, o desenvolvimento de medicamentos que agem, em determinadas doenças, focando apenas nos genes específicos, possíveis causadores do mal, não interferindo naqueles que são importantes para o paciente e que não se relacionam com a enfermidade. Uma rede de proteína pode ser representada através de um grafo, cujos vértices são as proteínas e as arestas são as interações proteína-proteína. Um exemplo de uma rede de proteínas é apresentado na Figura 1.1.

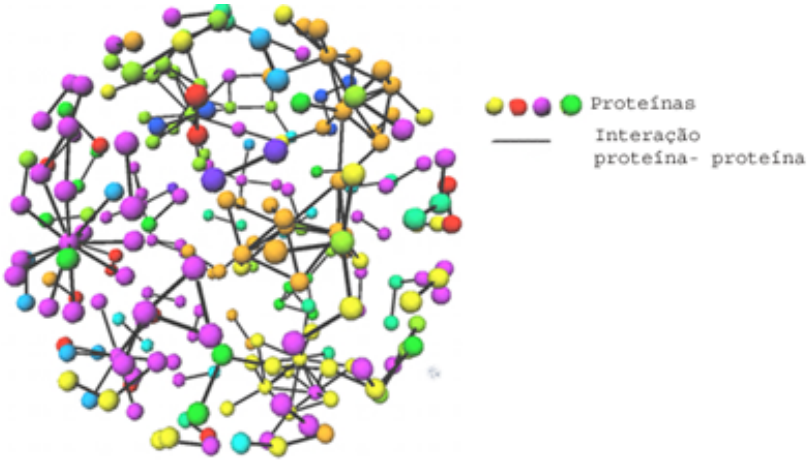

Figura 1.1 Exemplo de rede de proteínas

As proteínas são codificadas através de trechos dos DNA's chamados de genes. Os genes podem ser expressos de várias maneiras gerando conjuntos que podem representar informações genômicas relevantes.

Um conjunto de dados genômicos é usualmente apresentado como uma matriz $n \times c$, onde em cada linha são observados resultados de uma transcrição de um gene particular em relação a várias condições (colunas). Estas condições podem descrever várias situações dependendo da observação a ser transcrita. Portanto, pode-se pensar na matriz como um conjunto de n vetores, representando os genes, com c resultados cada. Exemplos de dois conjuntos genômicos são dados de expressão de genes e dados filogenéticos. Em dados de expressão de genes usualmente em cada linha da matriz observam-se as expressões de um gene particular em relação a várias

condições. Essas condições são, por exemplo, o tipo de experimento utilizado para obter as expressões dos genes, ou o tempo em que a expressão foi gerada, em experimentos envolvendo estudos de ciclos biológicos. O resultado medido para obter cada expressão usualmente assume valores reais.

Em dados filogenéticos, um exemplo pode ser tal que em cada linha da matriz é observado se determinado gene do organismo em estudo é ortólogo a genes de outros organismos representados pelas colunas. Genes ortólogos são genes em organismos diferentes que são similares pelo fato de se originarem de um antepassado comum. O resultado obtido em cada coluna (organismo), para cada gene, é o valor um (1), se o gene está presente no organismo representado por aquela coluna, e o valor (0), se o gene não está presente no organismo representado por aquela coluna. Neste trabalho os dados filogenéticos utilizados serão designados desta maneira.

Uma grande parte dos trabalhos relacionados à inferência de redes biológicas faz uso de dados de expressão representados por valores pertencentes ao conjunto dos números reais, $\mathbb{R}$ [MO04]. Outra parte faz uso de uma combinação de diferentes tipos de dados genômicos que assumem valores reais ou binários; neste caso, diz-se que está-se fazendo uso de uma técnica de integração de dados. Também diz-se que ocorre integração de dados quando são combinados dados genômicos de mesma natureza, porém referentes a organismos diferentes. Um exemplo do primeiro caso é encontrado no artigo de Yamanishi e colegas [YVK04], cujo trabalho apresenta uma abordagem supervisionada da análise de correlação canônica de Kernel (ACCKS) para fazer inferência sobre redes de proteínas considerando a informação de quatro tipos de dados genômicos para o mesmo organismo. O segundo caso é aplicado, por exemplo, por Reiss e colegas [RBB06], cujo trabalho emprega bi-clustering para fazer inferência sobre redes de proteínas considerando somente dados de expressão de genes, mas usando a informação de quatro organismos.

Em 1963, os biólogos Sokal e Sneath já estudavam métodos que classificam organismos similares na intenção de determinar se estes grupos representavam diferentes espécies biológicas, abrindo caminho para uma melhor compreensão do processo de evolução das espécies [SS63]. O método de agrupamento empregado na época foi o de clustering, e desde então, vários artigos na área têm sido publicados.

É sabido que genes que possuem padrões de resultados semelhantes sob as mesmas condições, são genes que potencialmente podem apresentar uma relação biológica, isto é, são genes que potencialmente podem interagir. Para se fazer inferência de uma rede de proteínas, pode-se verificar para cada par de genes, dentre todos possíveis pares pertencentes à rede, se eles interagem segundo alguma medida de similaridade. Uma medida de similaridade é usualmente definida por algum tipo de medida de distância ou correlação entre dois itens (genes) representados por vetores. Quanto maior a distância, menor a similaridade; quanto maior a correlação, maior a similaridade.

Agrupar genes usando uma abordagem de clustering pode levar a concluir erroneamente pela não existência de clusters. Uma situação em que isso ocorreria seria quando um grupo de genes não satisfaz ao critério de similaridade apresentado para todas as condições, mas satisfaz a um subgrupo delas. A figura 1.2 mostra a identificação de um bi-cluster (retângulo em verde) com quatro genes: G3, G4, G5, e G6. Esses genes não seriam agrupados na abordagem de clustering onde seriam identificados dois clusters: o primeiro cluster (em azul) agruparia os

genes G1, G5, G6, G7, G8 e o segundo cluster (em vermelho) agruparia os genes G2, G3 e G4.

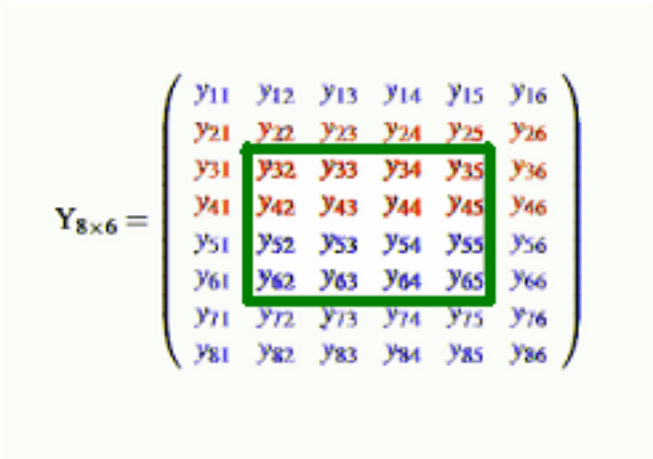

Figura 1.2 Identificação de Bi-cluster

A metodologia que classifica grupos de genes segundo um subgrupo de condições, isto é, que simultaneamente realiza clustering com os genes e clustering com as condições é chamada de bi-clustering. Madeira e colegas [MO04] fizeram uma triagem de vários métodos de bi-clustering aplicados a dados biológicos. Esses métodos foram formulados para dados genômicos que assumem valores reais. Recentemente, Koyutürk e colegas [KSG04], Preli'c e colegas [PBZea06] e Uitert e colegas [UW08] propuseram metodologias de bi-clustering para dados genômicos binários.

Embora a maioria dos métodos utilizem dados genômicos cujos resultados pertencem ao conjunto dos reais, dados genômicos cujos resultados assumem valores binários podem fornecer informações importantes. O foco principal deste trabalho foi encontrar um método que trabalhasse com dados genômicos binários. Assim, nesta tese é proposta uma metodologia de bi-clustering, chamada Lbic, para dados genômicos que assumem valores binários, uma vez que o pouco que tem sido trabalhado nessa direção ainda não representa avanço suficiente para a realização de inferência de interação de proteínas. Com base nos bi-clusters encontrados pela nova metodologia, serão propostas regras de decisões para fazer inferência de redes de proteínas. Também serão discutidas formas de combinar os resultados obtidos sob a nova metodologia proposta com resultados de uma metodologia de bi-clustering para dados genômicos que assumem valores reais. As análises realizadas utilizam, para efeito de validação da inferência de interação de proteínas-proteínas, dados filogenéticos e de expressão de genes, obtidos da levedura *Saccharomyces cerevisiae* apresentados no artigo de Yamanishi e colegas [YVK04]. Dados artificiais também foram gerados para comparação do método proposto Lbic com o método de bi-clustering Bicbin.

Os estudos desenvolvidos são apresentados ao longo de seis capítulos, incluindo esta introdução.

O Capítulo 2 relata algumas abordagens de clustering e bi-clustering apresentadas na literatura. Especial atenção é dada a metodologias de clustering K-means [Mac67], de bi-clustering Plaid [TBKH05] e de bi-clutering Bicbin [UW08]. Ainda neste capítulo são descrito alguns dos trabalhos relacionados à inferência de redes de proteínas. Especial atenção é dada a metodologia de análise de correlação canônica com kernel supervisionada [YVK04].

O Capítulo 3 apresenta conceitos fundamentais relativos à teoria estatística de modelos lineares e não lineares, que servirão de base teórica para os métodos de bi-clustering apresentados neste trabalho.

O Capítulo 4 apresenta o método Lbic, principal contribuição desta tese.

O Capítulo 5 mostra as comparações do método Lbic com as metodologias Bicbin [UW08], Plaid [TBKH05] e SKCCA [YVK04] sob diversas situações.

O Capítulo 6 apresenta as contribuições e os resultados desta tese, e indica os estudos a serem trabalhados no futuro .

CAPÍTULO 2

Trabalhos Relacionados

Conhecer funções biológicas de proteínas nos seres vivos continua sendo um dos principais desafios da biologia. Métodos de agrupamentos de genes têm sido um tema de grande relevância para este conhecimento. Clustering e bi-clustering são os métodos de agrupamento mais aplicados a dados genômicos, que são apresentados em forma de matrizes $(n \times c)$, onde as linhas e colunas representam genes e condições, respectivamente. Estas abordagens identificam grupos de genes e (ou) condições com base em suas similaridades. Sabendo que o conhecimento de interações proteína-proteína fornece informações importantes de como funcionam as funções biológicas dessas proteínas, vários estudos para detectar interações usando agrupamentos e outras técnicas de inferência foram propostos. Neste capítulo serão discutidos alguns métodos de clustering e bi-clustering, dentre eles o Plaid e o Bicbin que serão comparados com o método proposto Lbic. Ainda serão comentados alguns métodos de inferência de interações proteína-proteína, dentre eles o ACCKS que também será comparado com o Lbic.

2.1 Métodos de Agrupamento

Agrupamento é uma metodologia que classifica itens considerando alguma medida de similaridade. Medidas de similaridades são propostas segundo o objetivo e característica dos genes estudados, bem como a natureza dos dados (discreta, contínua, binária). Nesta seção é de interesse agrupar genes similares de acordo com uma característica particular. Para facilitar o entendimento das abordagens de clustering e de bi-clustering algumas notações são introduzidas.

Seja $Y_{n\times c} = \{y_{ij}\}$ uma matriz $(n \times c)$ de dados genômicos onde y_{ij} é a resposta do i-ésimo gene sob a j-ésima condição, com $i \in I = \{1,...,n\}$ e $j \in J = \{1,...,c\}$. Seja $I' \subset I$ um subconjunto de n' genes e $J' \subset J$ um subconjunto de c' condições, então pode-se definir as seguintes sub-matrizes de $Y_{n\times c}$:

$Y_{n'\times c} = \{y_{ij}\}$, com ß $\in I' = \{1,...,n'\}$ e $j \in J = \{1,...,c\}$ que representa um subconjunto de n' genes; e

$Y_{n\times c'} = \{y_{ij}\}$, com $i \in I = \{1,...,n\}$ e $j \in J' = \{1,...,c'\}$ que representa um subconjunto de c' condições.

A estrutura de uma matriz de dados genômicos com n genes e c condições pode ser visualizada na Tabela 2.1.

Tabela 2.1 Matriz de Dados Genômicos

	Condição 1	...	Condição j	...	Condição c
Gene 1	y_{11}	...	y_{1j}	...	y_{1c}
Gene ...	...	...	...	...	...
Gene i	y_{i1}	...	y_{ij}	...	y_{ic}
Gene ...	...	...	...	...	...
Gene n	y_{n1}	...	y_{nj}	...	y_{nc}

2.1.1 Clustering

Clustering é uma metodologia não supervisionada que agrupa genes considerando alguma medida de similaridade. Aos genes classificados num mesmo grupo é dado o nome de cluster. Os genes são usualmente descritos como vetores de tamanho c, onde cada elemento do vetor representa um resultado sob uma condição particular.

Sejam $(G_1, G_2, G_3, G_4, G_5, G_6, G_7, G_8)$ um conjunto de $n = 8$ genes e $(C_1, C_2, C_3, C_4, C_5, C_6)$ um conjunto de $c = 6$ condições. Então, a matriz de dados pode ser representada por:

$$\mathrm{Y}_{8\times 6} = \begin{pmatrix} y_{11} & y_{12} & y_{13} & y_{14} & y_{15} & y_{16} \\ y_{21} & y_{22} & y_{23} & y_{24} & y_{25} & y_{26} \\ y_{31} & y_{32} & y_{33} & y_{34} & y_{35} & y_{36} \\ y_{41} & y_{42} & y_{43} & y_{44} & y_{45} & y_{46} \\ y_{51} & y_{52} & y_{53} & y_{54} & y_{55} & y_{56} \\ y_{61} & y_{62} & y_{63} & y_{64} & y_{65} & y_{66} \\ y_{71} & y_{72} & y_{73} & y_{74} & y_{75} & y_{76} \\ y_{81} & y_{82} & y_{83} & y_{84} & y_{85} & y_{86} \end{pmatrix}.$$

Dois possíveis clusters (subconjuntos) de genes e condições desta matriz são respectivamente $(G_1, G_2, G_4, G_6, G_7)$ e (C_2, C_3, C_5), onde $n' = 5$ e $c' = 3$. Portanto pode-se obter as seguintes sub-matrizes:

$$\mathrm{Y}_{5\times 6} = \begin{pmatrix} y_{11} & y_{12} & y_{13} & y_{14} & y_{15} & y_{16} \\ y_{21} & y_{22} & y_{23} & y_{24} & y_{25} & y_{26} \\ y_{41} & y_{42} & y_{43} & y_{44} & y_{45} & y_{46} \\ y_{61} & y_{62} & y_{63} & y_{64} & y_{65} & y_{66} \\ y_{71} & y_{72} & y_{73} & y_{74} & y_{75} & y_{76} \end{pmatrix} \quad \text{e} \quad \mathrm{Y}_{8\times 3} = \begin{pmatrix} y_{12} & y_{13} & y_{15} \\ y_{22} & y_{23} & y_{25} \\ y_{32} & y_{33} & y_{35} \\ y_{42} & y_{43} & y_{45} \\ y_{52} & y_{53} & y_{55} \\ y_{62} & y_{63} & y_{65} \\ y_{72} & y_{73} & y_{75} \\ y_{82} & y_{83} & y_{85} \end{pmatrix},$$

as quais representam clusters de genes ($n' = 5$) e condições ($c' = 3$), respectivamente.

Três abordagens de clustering amplamente conhecidas na literatura são citadas a seguir.

1. Clustering Hierárquico [JW00]

Esta abordagem inicialmente transforma a matriz de dados, $Y_{n\times c}$, numa matriz (simétrica) de similaridades, $D_{n\times n}$, onde a medida de similaridade pode ser, por exemplo, o coeficiente de correlação de Fisher. Depois agrupam os genes, baseados em $D_{n\times n}$, criando um diagrama de árvore (dendograma) até restar só um nó. Os ramos (branches) na árvore representam os clusters. Um algoritmo básico é apresentado abaixo.

Algoritmo 2.1

1. Aloque cada gene a um cluster, formando n clusters com um único gene. Compute a distância (similaridade) entre os clusters (dois à dois) como a distância entre os genes que os clusters contêm.

2. Encontre o par de clusters mais similar e junte em um único cluster, de maneira que resultem um cluster a menos.

3. Compute as distâncias entre o novo cluster e os demais clusters.

4. Repita os passos 2 e 3 até todos genes estarem em um único cluster de tamanho n (se for interesse obter K clusters, deve-se escolher as maiores K-1 ligações).

O passo 3 pode ser feito de três maneiras:

1. Simples (*single - linkage*)- a distância entre dois clusters é a menor distância entre qualquer par dos seus elementos.

2. Completa (*complete - linkage*)- a distância entre dois clusters é a maior distância entre qualquer par dos seus elementos.

3. Média (*average - linkage*)- a distância entre dois clusters é a distância média entre qualquer par dos seus elementos.

2. *K*-means [JW00]

Esta abordagem é uma das mais simples e conhecidas. Ela agrupa os genes em um número fixo K de clusters representados pelos seus centróides. Um centróide é uma medida que resume os resultados dos elementos do cluster. Ele pode ser, por exemplo, a média aritmética. Inicialmente K centróides são escolhidos de tal maneira que eles estão o mais distante possível uns dos outros. Depois, esses centróides são re-calculados à medida que um gene é inserido em um cluster. Sejam $\mathbf{y}_k^i$ o vetor do i-ésimo gene do cluster k e c_k o k-ésimo centróide. O método K-means minimiza uma função erro quadrático, EQ, dada por:

$$EQ = \sum_{k=1}^{K} \sum_{i=1}^{n} \left\| y_k^i - c_k \right\|^2, \tag{2.1}$$

onde $\| \quad \|$ representa uma medida de similaridade (usualmente dada pela distância Euclidiana). O método proposto nesta tese faz uso da metodologia de K-means, por este motivo apresentaremos os passos de seu algoritmo.

Algoritmo 2.2

1. Apresente K pontos no espaço representado pelos genes que serão agrupados como os K centróides iniciais.

2. Aloque cada gene ao grupo que tem o centróide mais próximo, baseado na função erro quadrático.

3. Após alocar todos os n genes, recalcule os valores dos K centróides.

4. Aloque um gene por vez ao grupo que tem o centróide mais próximo (baseado na função erro quadrático). Recalcule o centróide do grupo que recebeu o novo gene e o centróide do grupo que cedeu o gene.

5. Repita o passo 4 até que nenhum gene mude de grupo.

3. Mapa Auto-Organizável
Self-Organized Map (SOM) [Koh82]

O SOM é um algoritmo de clustering que mapeia dados multi-dimensionais em uma superfície bi-dimensional. O mapa é construído através de uma ordenação topológica de clusters. O mapa pode ser usado para identificar grupos e relações entre genes projetando os dados numa imagem bi-dimensional que indica regiões de similaridades.

O algoritmo denota os clusters como nós, os quais são arranjados em uma grade retangular onde são apresentadas larguras e alturas. Inicialmente o SOM escolhe uma amostra aleatória de genes para ser os nós (clusters) iniciais e em seguida redefine estes nós usando um processo sistemático.

Tal método é similar ao método de K-means. O número de clusters a ser identificado também é pré-determinado, mas diferente do método de K-means, o SOM não força para a quantidade de clusters ser igual a de nós, pois é possível que um nó não esteja associado a nenhum gene quando o mapa for completo.

2.1.2 Bi-Clustering

A maioria das abordagens de clustering tem restrição de que um gene não pode participar de mais de um cluster nem de nenhum cluster, além dos genes não serem agrupados usando apenas um subconjunto de condições. Na abordagem de bi-clustering é possível obter subconjuntos dos dados que contemplem estas situações. Um bi-cluster da matriz $Y_{n\times c}$ é representado pela submatriz:

$Y_{n'\times c'} = \{y_{ij}\}$, com $i \in I' = \{1,...,n'\}$ e $j \in J' = \{1,...,c'\}$.

A metodologia de bi-clustering identifica um conjunto de b bi-clusters, tal que, cada um obedeça a alguma característica de homogeneidade, isto é, ela identifica b subconjuntos de I, $I'_1,...,I'_b$, onde $I'_k \in I$, $k = 1,...,b$, e b subconjuntos de J, $J'_1,...,J'_b$, onde $J'_k \in J$, $k = 1,...,b$, seguindo um critério de homogeneidade.

Considerando os conjuntos de $n = 8$ genes e $c = 6$ condições apresentados na seção anterior, pode-se gerar três bi-clusters, $(b = 3)$ tais que $I'_1 = \{1,5,7,8\}$, $I'_2 = \{2,4,8\}$ e $I'_3 = \{3,5,7,\}$ são três subconjuntos de genes e $J'_1 = \{1,3,6\}$, $J'_2 = \{2,4,6\}$ e $J'_3 = \{1,3,4,5\}$ são três subconjuntos de condições que formam os três bi-clusters. A ilustração desses bi-clusters (submatrizes) é apresentada abaixo.

$$\mathrm{Y}_{4\times3} = \begin{pmatrix} y_{11} & y_{13} & y_{16} \\ y_{51} & y_{53} & y_{56} \\ y_{71} & y_{73} & y_{76} \\ y_{81} & y_{83} & y_{86} \end{pmatrix}, \mathrm{Y}_{3\times3} = \begin{pmatrix} y_{22} & y_{24} & y_{26} \\ y_{42} & y_{44} & y_{46} \\ y_{82} & y_{84} & y_{86} \end{pmatrix} \text{ e } \mathrm{Y}_{3\times4} = \begin{pmatrix} y_{31} & y_{33} & y_{34} & y_{35} \\ y_{51} & y_{53} & y_{54} & Y_{55} \\ y_{71} & y_{73} & y_{74} & Y_{75} \end{pmatrix}.$$

Observa-se que os genes e condições classificados em um bi-cluster não precisam estar juntos na matriz original. A figura 2.1 mostra os três bi-clusters na matriz original.

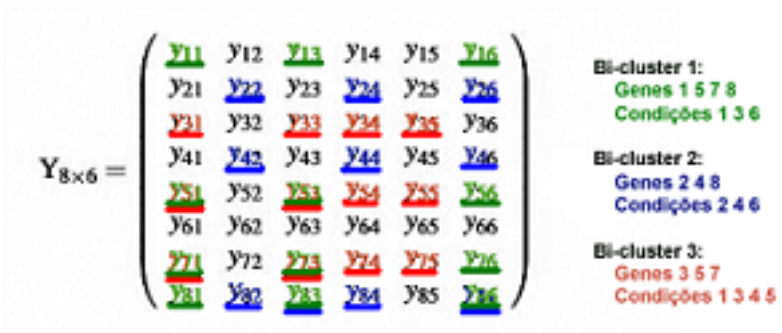

Figura 2.1 Matriz original com três bi-clusters

Várias abordagens de bi-clustering têm sido propostas para analisar dados de expressão de genes com a intenção de atingir os seguintes objetivos:

- Agrupar genes de acordo com suas expressões sob múltiplas condições;
- Classificar um novo gene dada a sua expressão e as expressões dos outros genes;
- Agrupar as condições baseadas nas expressões dos genes; e
- Classificar uma nova condição dada a expressão do gene sob aquela condição.

Na Biologia o conhecimento de como os genes ou as condições se relacionam é importante para entender como os processos biológicos funcionam. Considerando tal fato, a metodologia de bi-clustering pode ser aplicada em situações tais como:

- Somente um pequeno conjunto de genes participa de um processo celular de interesse;
- Um processo celular de interesse é ativo somente num subconjunto de condições; e
- Um único gene pode participar de múltiplos processos que podem ou não ser co-ativos sob todas as condições.

2.1.2.1 Aspectos do Bi-cluster

As metodologias de bi-clustering definem alguns aspectos necessários para a identificação dos bi-clusters. Descrições detalhadas desses aspectos são apresentadas.

Aspecto 1: Tipo de Homogeneidade do Bi-cluster

O aspecto 1 considera o tipo de homogeneidade que se deseja encontrar em um bi-cluster, isto é, como serão definidos os subconjuntos de genes e de condições. Os tipos de homogeneidade podem ser:

1(a) Bi-cluster com todos os elementos com valores constantes;

1(b) Bi-cluster com elementos com valores constantes em cada gene (ou condições);

1(c) Bi-cluster com elementos com valores coerentes (*coherent values*); e

1(d) Bi-cluster com evoluções coerentes (*coherent evolutions*).

Os valores (ou símbolos) são ditos coerentes se obedecem à mesma relação nas linhas e (ou) colunas. As homogeneidades dos tipos **1**(a), **1**(b) e **1**(c), analisam os valores numéricos da matriz de dados genômicos segundo os comportamentos das linhas e colunas. Suponha que o bi-cluster identificado seja aquele apresentado anteriormente pela matriz $Y_{5\times3}$, então, os seguintes resultados podem ser obtidos:

$$\mathrm{y}_{5\times3} = \begin{pmatrix} y & y & y \\ y & y & y \\ y & y & y \\ y & y & y \\ y & y & y \end{pmatrix} \text{ representando } \mathbf{1}(\text{a}),\ y \in \mathbb{R};$$

$$Y_{5\times 3} = \begin{pmatrix} y_1 & y_1 & y_1 \\ y_2 & y_2 & y_2 \\ y_4 & y_4 & y_4 \\ y_6 & y_6 & y_6 \\ y_7 & y_7 & y_7 \end{pmatrix}$$ representando **1**(b) com genes com valores constantes, $y_i \in \mathbb{R}$;

$$Y_{5\times 3} = \begin{pmatrix} y_2 & y_3 & y_5 \\ y_2 & y_3 & y_5 \\ y_2 & y_3 & y_5 \\ y_2 & y_3 & y_5 \\ y_2 & y_3 & y_5 \end{pmatrix}$$ representando **1**(b) com condições com valores constantes, $y_i \in \mathbb{R}$;

$$Y_{5\times 3} = \begin{pmatrix} y_1 & y_1+a & y_1+b \\ y_2 & y_2+a & y_2+b \\ y_4 & y_4+a & y_4+b \\ y_6 & y_6+a & y_6+b \\ y_7 & y_7+a & y_7+b \end{pmatrix}$$ representando **1**(c) com modelo aditivo, y_i, a e $b \in \mathbb{R}$;

$$Y_{5\times 3} = \begin{pmatrix} y1 & y1\times a & y1\times b \\ y2 & y2\times a & y2\times b \\ y4 & y4\times a & y4\times b \\ y6 & y6\times a & y6\times b \\ y7 & y7\times a & y7\times b \end{pmatrix}$$ representando **1**(c) com modelo multiplicativo, a, b e $y_i \in \mathbb{R}$.

Enquanto os tipos de homogeneidades **1**(a), **1**(b) e **1**(c) assumem que os elementos da matriz são valores pertencentes ao conjunto dos reais, a homogeneidade **1**(d) observa a natureza desses elementos representado-os por símbolos. As três matrizes seguintes apresentam exemplos de homogeneidade **1**(d).

$$Y_{5\times 3} = \begin{pmatrix} s & s & s \\ s & s & s \\ s & s & s \\ s & s & s \\ s & s & s \end{pmatrix}$$ representa elementos com único resultado (símbolo);

$$Y_{5\times 3} = \begin{pmatrix} s1 & s1 & s1 \\ s2 & s2 & s2 \\ s4 & s4 & s4 \\ s6 & s6 & s6 \\ s7 & s7 & s7 \end{pmatrix}$$ representa evoluções coerentes nas linhas;

$$Y_{5\times 3} = \begin{pmatrix} s2 & s3 & s5 \\ s2 & s3 & s5 \\ s2 & s3 & s5 \\ s2 & s3 & s5 \\ s2 & s3 & s5 \end{pmatrix}$$ representa evoluções coerentes nas colunas.

Aspecto 2: Tipo de Estrutura do Bi-cluster

Outro ponto importante em um algoritmo que identifica bi-clusters é o aspecto 2, o qual indica a estrutura das relações das linhas e das colunas no bi-cluster. Existem várias estruturas que podem ser consideradas, algumas delas são:

2(a) Bi-clusters com linhas e colunas exclusivas;

2(b) Bi-clusters com linhas exclusivas;

2(c) Bi-clusters com colunas exclusivas;

2(d) Bi-clusters sem sobreposições e não exclusivos;

2(e) Bi-clusters sem sobreposições mas com estrutura de tabuleiro;

2(f) Bi-cluster sem sobreposições mas com estrutura de árvore;

2(g) Bi-clusters com sobreposições hierárquicas; e

2(h) Bi-clusters com sobreposições arbitrárias;

Exemplos dessas estruturas são representados abaixo:

$$Y_{8\times6} = \begin{pmatrix} b_1 & b_1 & . & . & . & . \\ b_1 & b_1 & . & . & . & . \\ b_1 & b_1 & . & . & . & . \\ . & . & b_2 & b_2 & . & . \\ . & . & b_2 & b_2 & . & . \\ . & . & . & . & b_3 & b_3 \\ . & . & . & . & b_3 & b_3 \\ . & . & . & . & b_3 & b_3 \end{pmatrix}$$ com 3 bi-clusters com linhas e colunas exclusivas, **2**(a);

$$Y_{8\times6} = \begin{pmatrix} . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & b_1 & b_1 & b_1 & . \\ . & . & b_1 & b_1 & b_1 & . \\ . & . & b_1 & b_1 & b_1 & . \\ . & . & b_1 & b_1 & b_1 & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \end{pmatrix}$$ com um único bi-cluster, caso particular de **2**(a);

$$Y_{8\times6} = \begin{pmatrix} b_1 & b_1 & b_1 & . & . & . \\ b_1 & b_1 & b_1 & . & . & . \\ b_1 & b_1 & b_1 & . & . & . \\ . & . & b_2 & b_2 & b_2 & . \\ . & . & b_2 & b_2 & b_2 & . \\ . & . & . & . & b_3 & b_3 \\ . & . & . & . & b_3 & b_3 \\ . & . & . & . & b_3 & b_3 \end{pmatrix}$$ com 3 bi-clusters com linhas exclusivas, **2**(b);

$$Y_{8\times 6} = \begin{pmatrix} b_1 & b_1 & . & . & . & . \\ b_1 & b_1 & . & . & . & . \\ b_1 & b_1 & . & . & . & . \\ b_1 & b_1 & b_2 & b_2 & . & . \\ b_1 & b_1 & b_2 & b_2 & . & . \\ . & . & b_2 & b_2 & b_3 & b_3 \\ . & . & . & . & b_3 & b_3 \\ . & . & . & . & b_3 & b_3 \end{pmatrix}$$ com 3 bi-clusters com colunas exclusivas, **2**(c);

$$Y_{8\times 6} = \begin{pmatrix} b_1 & b_1 & b_3 & b_3 & b_3 & b_3 \\ b_1 & b_1 & b_3 & b_3 & b_3 & b_3 \\ b_1 & b_1 & b_3 & b_3 & b_3 & b_3 \\ b_1 & b_1 & . & . & b_4 & b_4 \\ b_1 & b_1 & . & . & b_4 & b_4 \\ b_2 & b_2 & b_2 & b_2 & b_4 & b_4 \\ b_2 & b_2 & b_2 & b_2 & b_4 & b_4 \\ b_2 & b_2 & b_2 & b_2 & b_4 & b_4 \end{pmatrix}$$ com 4 bi-clusters sem sobreposição e não exclusivos, **2**(d);

$$Y_{8\times 6} = \begin{pmatrix} b_1 & b_1 & b_2 & b_2 & b_3 & b_3 \\ b_1 & b_1 & b_2 & b_2 & b_3 & b_3 \\ b_4 & b_4 & b_5 & b_5 & b_6 & b_6 \\ b_4 & b_4 & b_5 & b_5 & b_6 & b_6 \\ b_7 & b_7 & b_8 & b_8 & b_9 & b_9 \\ b_7 & b_7 & b_8 & b_8 & b_9 & b_9 \\ b_{10} & b_{10} & b_{11} & b_{11} & b_{12} & b_{12} \\ b_{10} & b_{10} & b_{11} & b_{11} & b_{12} & b_{12} \end{pmatrix}$$ com 12 bi-clusters com estrutura de tabuleiro, **2**(e);

$$Y_{8\times 6} = \begin{pmatrix} b_1 & b_1 & . & . & . & . \\ b_1 & b_1 & . & . & . & . \\ b_1 & b_1 & . & . & . & . \\ b_2 & b_2 & . & . & . & . \\ b_2 & b_2 & . & . & . & . \\ b_2 & b_2 & b_3 & b_3 & b_4 & b_4 \\ b_2 & b_2 & b_3 & b_3 & b_4 & b_4 \\ b_2 & b_2 & b_3 & b_3 & b_4 & b_4 \end{pmatrix}$$ com 4 bi-clusters com estrutura de árvore, **2**(f);

$$Y_{8\times 6} = \begin{pmatrix} b_1 & b_1 & b_1 & b_3 & b_3 & b_3 \\ b_1 & b_{1,2} & b_{1,2} & b_3 & b_{3,4} & b_{3,4} \\ b_1 & b_{1,2} & b_{1,2} & b_3 & b_{3,4} & b_{3,4} \\ b_1 & b_1 & b_1 & b_3 & b_{3,4} & b_{3,4} \\ . & . & . & b_3 & b_{3,4} & b_{3,4} \\ . & . & . & b_3 & b_3 & b_3 \\ b_5 & b_5 & . & b_3 & b_3 & b_3 \\ b_5 & b_5 & . & b_3 & b_3 & b_3 \end{pmatrix}$$ com 5 bi-clusters com sobreposições hierárquicas, **2**(g);

$$Y_{8\times 6} = \begin{pmatrix} . & . & . & . & . & . \\ . & b_1 & b_1 & b_1 & b_2 & b_2 \\ . & b_1 & b_{1,3} & b_{1,3} & b_{2,3} & b_{2,3} \\ . & b_1 & b_{1,3} & b_{1,3} & b_{2,3} & b_{2,3} \\ . & . & b_3 & b_3 & b_3 & b_3 \\ . & . & b_3 & b_3 & b_3 & b_3 \\ . & b_4 & b_4 & b_4 & . & . \\ . & b_4 & b_4 & b_4 & . & . \end{pmatrix}$$ com 4 bi-clusters com sobreposições arbitrárias, **2**(h);

onde b_k representa o k-ésimo bi-cluster contendo o elemento de uma linha i $(i = 1, ..., 8)$ e de uma coluna j $(j = 1, ..., 6)$, e $b_{k,k'}$ representa os bi-clusters k e k' que contêm o elemento de uma linha i $(i = 1, ..., 8)$ e de uma coluna j $(j = 1, ..., 6)$, simultaneamente.

Aspecto 3: Quantidade Procurada de Bi-clusters

O aspecto 3 indica a quantidade de bi-clusters a ser identificado em cada iteração do algoritmo. Essas quantidades podem ser:

3(a) Um por iteração;

3(b) Vários por iteração; e

3(c) Simultâneos.

Aspecto 4: Tipo de Algoritmo de Procura

Dependendo dos aspectos 1, 2 e 3 selecionados, vários algoritmos de procura (aspecto 4) são propostos na literatura [MO04].

4(a) Combinação de algoritmos de clustering;
(*Iterative Row and Column Clustering combination*)

4(b) Divisão e Combinação do problema ;
(*Divide and Conquer*)

4(c) Escolhendo ótimo local;
(*Greedy Iterative Search*)

4(d) Enumeração Exaustiva ; e
(*Exhaustive Bi-cluster Enumeration*)

4(e) Processo iterativo / modelos estatísticos.
(*Distribution Parameter Identification*)

Métodos baseados nos quatros aspectos de descrição dos bi-clusters a serem identificados são apresentados nas seções seguintes.

2.1.2.2 Abordagens de Bi-clustering para Dados de Expressão de Genes [MO04]

Madeira e colegas [MO04] colocam em perspectiva um conjunto de abordagens de bi-clustering para dados de expressão de genes proposto entre os anos de 1991 e 2003. Algumas de suas revisões são apresentadas a seguir.

1. Padrões *ks* de δ-válido
δ-valid *ks*-patterns [CST00]

Califano e colegas definem padrões *ks* de δ-válido como um subconjunto de genes, I', de tamanho k e um subconjunto de condições (suporte), J', de tamanho s, tal que para cada gene i

$$\max(y_{ij}) - \min(y_{ij}) < \delta, \forall j \in J'.$$

O objetivo do método é encontrar um δ-válido máximo que exiba bi-clusters com valores coerentes para as respostas dos genes em certas condições e nenhuma coerência nos valores das respostas nas outras condições. O padrão *ks* de δ-válido é máximo se nenhum gene ou condição pode ser adicionado a ele, isto é, não existe $k' > k$ ou $s' > s$ tal que o padrão *ks* de δ-válido seja válido. Após pré-processar os dados, é usado um algoritmo de descoberta de padrões. Um teste de significância estatística é usado para avaliar a qualidade desses padrões. Os padrões encontram subconjuntos de genes e de condições candidatos a um bi-cluster estatisticamente significante, descartando os demais candidatos. Depois é escolhido um conjunto de padrões ótimos dentre aqueles estatisticamente significantes usando um algoritmo guloso, o qual adiciona linhas e colunas aos padrões existentes de modo que eles tenham padrão máximo. O algoritmo de descoberta de padrões considera que cada coluna (condição) da matriz de dados é um vetor, e identifica os padrões dos vetores permitindo todos os alinhamentos de vetores possíveis.

Uma restrição é usada para limitar o impacto de emparelhamentos aleatórios ocorridos por grandes distâncias entre os vetores. Os vetores são pré-alinhados antes de serem usados.

O algoritmo começa com um único padrão com nenhuma linha (gene), com todas as colunas e valores zero para cada coluna. Os valores de cada coluna são ordenados e todos subconjuntos de valores contínuos que são δ-válido são selecionados. Subconjuntos que estão completamente contidos dentro de outros subconjuntos são removidos.

Cada subconjunto é considerado um potencial super-padrão de um padrão máximo. Todas as possíveis combinações máximas desses super-padrões são obtidas iterativamente. Como resultado, todos os padrões que existem na matriz de dados são gerados hierarquicamente por uma combinação padrão.

2. Clustering de dois Fatores Duplos
Couple Two-Way Clustering (CTWC) [GLD00]

O método CTWC combina clustering de um fator aplicado às linhas (características) e às colunas (objetos) e realiza agrupamento hierárquico aplicado à matriz de similariedades das linhas (matriz Y transformada segundo um conjunto de colunas estáveis).

O algoritmo de CTWC inicia com dois conjuntos: F_0 (com todas as linhas) e O_0 (com todas as colunas), e um parâmetro ajustável, T, assumindo valor zero ($T = 0$). Depois aplica o algoritmo de clustering hierárquico em cada conjunto. Esse processo é repetido apresentando um incremento de T. Quando T cresce, o cluster é subdividido em vários subclusters. O processo continua até T não crescer mais e ser formado um único cluster.

Uma medida de estabilidade, ΔT, é calculada pelo tamanho do intervalo em que o cluster permanece inalterado. Logo, um cluster é dito estável se ele permanece inalterado através de um ΔT dado. Durante a execução do algoritmo são geradas duas listas de clusters estáveis de linhas e colunas (um bi-cluster por vez / iteração), e depois são geradas duas listas de pares de subconjuntos de linhas e colunas. Em cada iteração, um subconjunto de linhas é pareado a um subconjunto de colunas. Quando um novo cluster estável é formado, ele é adicionado à lista de linhas e colunas, e é criado um apontador identificando de onde o par se originou (nó pai). A iteração continua até que nenhum novo cluster satisfaça o critério de estabilidade ou o tamanho crítico seja achado.

3. δ *bi-clusters* [CC00]

Cheng e Church definem um bi-cluster como subconjuntos de linhas e colunas com altos escores de similaridades. O escore de similaridade é dado pela média dos quadrados dos resíduos, H. Seja $Y_{I\times J}$, a matriz de dados, então $Y_{I'\times J'}$ ($I' < I$ e $J < J'$) é definido como um δ-bi-cluster se $H(I',J') < \delta$, para algum $\delta \geq 0$. Para encontrar um δ-bi-cluster, um modelo aditivo é formulado para cada bi-cluster $Y_{I'\times J'}$ dado por:

$$y_{ij} = \mu + \alpha_i + \beta_j + e_{ij}$$

onde:

μ é a média geral;

α_i é o efeito da i-ésima linha;

β_j é o efeito da j-ésima coluna; e

e_{ij} é o termo residual aleatório.

Os estimadores desses parâmetros são iguais àqueles estimados no Capítulo 3. Então o estimador do resíduo é expresso como:

$$\widehat{e}_{ij} = y_{ij} - y_{iJ'} - y_{I'j} + y_{I'J'},$$

onde: $y_{iJ'} = \bar{y}_{i.} = \sum_{j=1}^{|J'|} y_{ij} / |J'|$;

$y_{I'j} = \bar{y}_{.j} = \sum_{i=1}^{|I'|} y_{ij} / |I'|$; e

$y_{I'J'} = \bar{y}_{..} = \sum_{i=1}^{|I'|} \sum_{j=1}^{|J'|} y_{ij} / |I'|\,|J'|$.

Portanto a média dos quadrados dos resíduos pode ser calculada por:

$$H(I'J') = \frac{1}{|I'||J'|} \sum_{i\in I', j\in J'} \widehat{e}_{ij}^2.$$

Para identificar δ-bi-clusters com $H(I',J')$ menor que um δ, são combinados vários algoritmos do tipo guloso que removem e adicionam linhas e colunas da matriz Y. Primeiro são removidas as linhas e as colunas enquanto a média dos quadrados dos resíduos é superior a um limiar. Depois adicionam linhas e colunas enquanto a média dos quadrados dos resíduos não aumenta. Desta forma um bi-cluster é identificado por vez.

O método sempre começa com a matriz Y até H não decrescer mais ou $H < \delta$. Em cada iteração o bi-cluster é marcado com um número aleatório. O método permite encontrar sobreposições entre os bi-clusters desde que as adições de linhas e colunas são feitas com os valores originais, mas não é provável identificar muitas sobreposições.

4. Modelos Relacionais Probabilístico
Probabilistic Relational Models (PRMs) [STG$^+$01]

Segal e colegas usam PRMs para modelar a distribuição conjunta dos valores do gene. Para tanto, um modelo aditivo em cada bi-cluster k é definido por:

$$\theta_{ijk} = \mu_k + \alpha_{ik} + \beta_{jk} + e_{ijk}$$

onde:

μ_k é a média do bi-cluster k;

α_i é o efeito da i-ésima linha do k-ésimo bi-cluster;

β_j é o efeito da j-ésima coluna do k-ésimo bi-cluster; e

e_{ijk} é um termo aleatório independente e identicamente distribuído por N(0,σ_k^2).

Portanto, um elemento da matriz de dados é modelado como:

$$y_{ij} = \sum_k \theta_{ijk}\rho_{ik}\kappa_{jk} \ ,$$

onde:

ρ_{ik} assume 1 se a linha i pertence ao bi-cluster k, e 0, caso contrário; e

κ_{jk} assume 1 se a coluna j pertence ao bi-cluster k, e 0, caso contrário.

A idéia do método é estimar (usando Algoritmo EM) a atividade de cada coluna em cada bi-cluster minimizando a expressão:

$$\sum_k (y_{ijk} - \theta_{ijk}\rho_{ik}\kappa_{jk})/\sigma_k^2.$$

5. Clustering com Dois Fatores Interrelacionado
Interrelated Two-Way Clustering (ITWC) [TZZR01]

Tang e colegas combinam resultados de clustering com um Fator nas linhas e outro nas colunas de Y para produzir os bi-clusters. O algoritmo primeiro aplica clustering nas linhas da matriz Y, selecionando K clusters exclusivos com n_1 linhas de Y , I_i, $i = 1,...,K$.

Para exemplificação, suponha que seja usado o método de clustering K-means, para $K = 2$, então têm-se os clusters I_1 e I_2. Em seguida o algoritmo aplica clustering nas colunas para cada cluster I_i, selecionando dois clusters, J_1 e J_2. Os resultados dos quatro clusters são combinados dividindo as colunas em grupos, C_i, $i = 1,...4$, onde $C_1 = J_{1,a}, C_2 = J_{1,b}, C_3 = J_{2,a}$ e $C_4 = J_{2,b}$. Para os quatro grupos são encontrados pares heterogêneos (C_s, C_t), $s,t = 1,...4$, isto é, são encontrados pares de grupos de colunas que não compartilham as mesmas linhas no seu cluster.

Por fim, o algoritmo ordena as linhas decrescentemente segundo a distância do cosseno entre cada linha e a linha representada em cada bi-cluster. Após o passo final o número de linhas n_1 é reduzido a n_2 e então, são repetidos todos os passos até as condições de término do algoritmo serem satisfeitas.

6. Modelo Plaid

Plaid Model [LO02]

A idéia do modelo Plaid é representar os elementos da matriz de dados genômicos por cores, e então ordenar as linhas e colunas da matriz de forma que elementos com cores semelhantes fiquem próximos, formando uma aparência de uma colcha de b retalhos. Um possível arranjo desses retalhos é a existência de b retalhos exclusivos e exaustivos na diagonal da matriz, representando clusters de genes e clusters de condições, isto é, cada elemento da matriz pode ser explicado através do gene, da condição e do retalho a que ele pertence. Esses retalhos são chamados de bi-clusters se eles atendem a um critério particular. O nome Plaid descreve a aparência de um bi-cluster. O modelo Plaid é definido como:

$$y_{ij} = \sum_{k=0}^{b} \theta_{ijk} \rho_{ik} \kappa_{jk}, \tag{2.2}$$

onde:

$i = 1,...,n, j = 1,...,c$ e $k = 1,...,b$;

y_{ij} é a resposta da expressão do i-ésimo gene e j-ésima condição;

θ_{ijk} é o efeito do k-ésimo bi-cluster;

ρ_{ik} assume valor 1 se o i-ésimo gene está no k-ésimo bi-cluster e 0, caso contrário; e

κ_{jk} assume valor 1 se a j-ésima condição está no k-ésimo bi-cluster e 0, caso contrário.

Lazzaroni e Owen identificam os bi-clusters através de um algoritmo iterativo que minimiza a seguinte função:

$$\frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{c} (y_{ij} - \widehat{\theta}_{ij0} - \sum_{k=1}^{b} \widehat{\theta}_{ijk} \rho_{ik} \kappa_{jk})^2. \tag{2.3}$$

7. Clustering Conjugado Duplo

Double Conjugated Clustering (DCC) [BJK02]

DCC é uma abordagem de clustering com dois fatores que permite o uso de qualquer algoritmo de clustering. Neste trabalho Busygin e colegas apresentam um algoritmo que calcula as similaridades entre linhas ou colunas de uma matriz de dados através de SOM e de métrica angular. Tal algoritmo primeiro designa cada nó da linha a um nó da coluna, ou vice-versa, formando um nó conjugado (através da métrica angular), isto é, ele fornece dois espaços conjugados: espaço característico representando as linhas (com tamanho n) e espaço amostral representando as colunas (com tamanho c). Em seguida é aplicado o clustering SOM nos dois espaços, alternando a ordem.

Os resultados de um espaço são usados para corrigir posições dos nós correspondentes no outro espaço. O processo é repetido até que o número de características e amostras movidas sejam menores que um limiar dado em ambos espaços. O algoritmo retorna um grupo de linhas para cada grupo de colunas (conjunto de bi-clusters).

8. Biclustering com Sobreposição Flexivel

Flexible Overlapped biClustering (FLOC) [YWWY02]
(*generalized δ-bi-clusters*)

FLOC é uma generalização do δ-bi-clustering de Cheng e Church. O FLOC define um limiar de ocupação, ϑ, tal que,

para cada linha $i \in I'$, $\frac{J''_i}{J'} > \vartheta$,

e para cada coluna $j \in J'$, $\frac{I''_j}{J'} > \vartheta$,

onde J''_i e I''_j são os conjuntos de elementos especificados pela linha i e pela coluna j, repectivamente. A abordagem ainda define o volume do δ-bi-cluster, $\upsilon_{I'J'}$, como o número de valores específicos de y_{ij}. O método de Cheng e Church é um caso especial do FLOC quando $\vartheta = 1$. Sejam as médias das linhas, colunas e δ-bi-cluster são definidos, respectivamente, por:

$$y_{iJ'} = \frac{1}{|J''_i|} \sum_{j \in J''_i} y_{ij};$$

$$y_{I'j} = \frac{1}{|I''_j|} \sum_{i \in I''_j} y_{ij}; \text{ e}$$

$$y_{I'j} = \frac{1}{\upsilon_{I'J'}} \sum_{i \in I''_j, j \in J''_i} y_{ij}.$$

Então, o estimador do resíduo é calculado como:

$$\widehat{e}_{ij} = y_{ij} - y_{iJ'} - y_{I'j} + y_{I'J'},$$

para y_{ij} especificado e 0, caso contrário.

Portanto a média dos quadrados dos resíduos pode ser calculada por:

$$H(I'J') = \frac{1}{\upsilon_{I'J'}} \sum_{i \in I'_j, j \in J'_i} \widehat{e}^2_{ij},$$

Diferentemente do δ-bi-clustering, o FLOC encontra bi-clusters simultâneos que lidam com dados faltantes. Tal método também descobre K possíveis bi-clusters sobrepostos. O algoritmo de FLOC primeiro gera K bi-clusters pela adição de linhas e colunas com independente probabilidade p. Cada coluna e linha é examinada para encontrar uma ação em que diminua o H. Então K bi-clusters levam a K ações para cada linha e coluna. A que tem máximo ganho é executada. O ganho de uma ação é uma função da redução de H e do aumento de número de genes no bicluster. O processo de otimização pára quando as ações em potenciais não melhoram a qualidade do bicluster, dada pela expressão

$$\frac{1}{K} \sum_{k=1}^{K} H(I', J')_k.$$

9. δ *p-Clusters* [WWYY02]

Wang e colegas também utilizam um modelo aditivo para encontrar δp-Clusters. Dada a matriz $Y_{I \times J}$, então o método define submatrizes 2×2 por:

$$M_{2\times 2} = (I_{i1,i2}, J_{j1,j2})$$

onde $i1, i2 \in I$ e $j1, j2 \in J$ e calcula um $pscore(M)$ para cada submatriz

$$pscore(M) = |(a_{i_1 j_1} a_{i_1 j_2}) - (a_{i_2 j_1} a_{i_2 j_2})|$$

A submatriz (I', J') é um δp-Cluster se para qualquer submatriz

$$M_{2\times 2} \subset (I', J') \text{ tem-se } pscore(M) < \delta.$$

O algoritmo δp-Clustering realiza uma enumeração exaustiva de bi-clusters sujeita a restrição de ter um limite mínimo de linhas e um limite mínimo de colunas. Uma árvore de sufixo é usada para acelerar o processo de enumeração e evitar repetições. Tal algoritmo começa encontrando um conjunto com candidatos ao conjunto de máxima dimensão (MDS) para cada par de linhas e para cada par de colunas. Um par de linhas MDS $(\mathbf{y}^i, \mathbf{y}^{i'})$ é um conjunto de colunas que definem um bicluster de tamanho máximo que inclui as linhas $\mathbf{y}^i$ e $\mathbf{y}^{i'}$, enquanto um par de colunas MDS $(\mathbf{y}_j, \mathbf{y}_{j'})$ é um conjunto de linhas que definem um bicluster de tamanho máximo que incluem as colunas $\mathbf{y}_j$ e $\mathbf{y}_{j'}$. O conjunto de candidatos MDSs é calculado usando um método que gera todos os possíveis MDS para cada par de linhas e cada par de colunas e é reestimado usando propriedades que relacionam os pares de linhas MDS com os pares de colunas MDS.

10. Sub-Matriz com Ordem Preservada
Order-Preserving Sub-Matrix (OPSM) [BDCKY02]

Ben-Dor e colegas definem o bi-cluster como uma submatriz com ordem preservada. Eles dão ênfase à ordem relativa das colunas quando identificam OPSMs com máxima significância estatística, a qual é calculada pelo limite superior de uma probabilidade de que uma matriz aleatória de dados, $n \times c$, contenha um modelo completo de tamanho s com k ou mais linhas . O método primeiro define um modelo completo, (J,Π) onde J são subconjuntos de s colunas e $\Pi = (j_1,...,j_s)$ é ordenação linear das colunas em J.

Uma submatriz é determinada por subconjuntos de genes e condições, onde para cada condição os genes têm mesma ordem linear. Depois são definidos modelos parciais de ordem (a,b), tais que $< j_1,...,j_a >$ são os a menores elementos e $< j_{s-(b-1)},...,j_s >$ são os b maiores elementos. Estes modelos são expandidos iterativamente até o modelo tornar-se completo. O algoritmo primeiro avalia todos os modelos parciais de ordem (1,1), mantendo os l melhores. Depois expande os l modelos naqueles de ordem (2,1), mantendo os l melhores. Em seguida expande os l modelos naqueles de ordem (2,2), depois de ordem (3,2) ,..., até os modelos completos $(s/2, s/2)$.

11. Distribuição Multinomial/ Amostragem Gibbs
Mutinomial Distribution/ Gibbs Sampling [SMDM03]

Sheng e colegas usam distribuições multinomiais independentes para modelar os dados de cada coluna num bi-cluster. Assim os dados num bi-cluster são consistentes através das linhas para cada coluna, apesar desses valores poderem diferir entre colunas. O mesmo pode ser feito com as linhas. A amostragem Gibbs é usada para estimar os parâmetros dessas distribuições. O algoritmo acha um bi-cluster por vez. Primeiro são designados aleatoriamente os rótulos 1 ou 0 às linhas e colunas, onde 1 significa que a linha (ou coluna) pertence ao bi-cluster e 0 que não pertence.

Depois, para cada linha i ($i = 1,...,n$), a distribuição de Bernoulli é computada (fixando os rótulos para todas as outras linhas) por amostragem Gibbs, obtendo a estimativa do rótulo desta linha. Em seguida, similarmente ao passo descrito para as linhas, os rótulos para as colunas são estimados. O algoritmo estima estes rótulos iterativamente até um número pré-determinado de vezes. Para achar múltiplos bi-clusters, as linhas que pertencem ao bi-cluster encontrado anteriormente recebem rótulos permanentemente iguais a zero no bicluster encontrado. Logo linhas num bi-cluster selecionado anteriormente não podem pertencer a bi-clusters futuros, no entanto, o modelo ainda usa todos os dados para calcular as estimativas dos rótulos, mesmo aqueles marcados.

12. *Spectral* [KBCG03]

O método Spectral proposto por Kluger e colegas identifica bi-clusters com valores coerentes e com estrutura de tabuleiro incorporando tipos de normalização da matriz de dados. O método assume que a contribuição de um bi-cluster é dada por um modelo multiplicativo após uma normalização dos dados. Para acessar a qualidade do bi-cluster, os resultados são testados

contra a hipótese nula de não existência de uma estrutura na matriz de dados Y. O método usa abordagem espectral para bi-clustering assumindo que após a normalização da matriz, ela tem estrutura de tabuleiro. Seja a matriz normalizada, Y_t, então as soluções das equações:

$$Y_t' Y_t x_1 = \lambda^2 x_1$$

$$Y_t Y_t' x_2 = \lambda^2 x_2$$

fornecem autovetores e autovalores. Se as constantes num autovetor podem ser ordenadas produzindo uma estrutura de escada, então os clusters de colunas e linhas podem ser identificados. O padrão de tabuleiro na matriz Y é refletido nas estruturas constantes de pares de autovetores x_1 e x_2 que solucionam aquelas equações onde x_1 e x_2 tem um mesmo autovalor. Três tipos de normalização são propostas pelos autores do artigo: *Independent re-scaling of rows and columns*, *Bi-stochastization* e *Log-interactions*.

13. Resumo: Métodos Bi-clustering [MO04]

Os métodos apresentados nessa seção são descritos na Tabela 2.2 apresentando os quatro aspectos dos bi-clusters citados nesse capítulo: Aspecto 1 - Tipo de Homogeneidade, Aspecto 2 - Tipo de Estrutura, Aspecto 3 - Algoritmo de Procura e Aspecto 4 - Quantidade Procurada.

Tabela 2.2 Resumo dos Métodos de Bi-Clustering

Método	Referência	Homogeneidade	Estrutura	Algoritmo	Quant. Procurada
$\delta Patterns$	[CST00]	1(b)	2(h)	3((c)	4(c)
CTWC	[GLD00]	1(c)	2(h)	3(b)	4(a)
$\delta bi-cluster$	[CC00]	1(c)	2(h)	3(a)	4(c)
PRMs	[STG$^+$01]	1(b)/1(c)	2(a)/2(h)	3(c)	4(e)
ITWC	[TZZR01]	1(c)	2(b)/2(e)	3(b)	4(a)
Plaid	[LO02]	1(c)	2(h)	3(a)	4(e)
DCC	[BJK02]	1(c)	2(a)/2(e)	3(c)	4(a)
FLOC	[YWWY02]	1(c)	2(h)	3(c)	4(c)
$\delta p-cluster$	[WWYY02]	1(c)	2(d)	3(c)	4(d)
OPSM	[BDCKY02]	1(d)	2(a)/2(h)	3(a)	4(c)
D.M./ A. Gibbs	[SMDM03]	1(b)	2(b)/2(c)	3(a)	4(e)
Spectral	[KBCG03]	1(c)	2(e)	3(c)	4(c)

2.1.2.3 Abordagem Bi-clustering para Dados de Expressão de Genes Método Plaid [TBKH05]

O método Plaid proposto por Tuner e colegas [TBKH05] identifica bi-clusters baseado no modelo Plaid [LO02] formulado para dados genômicos que assumem valores pertencentes ao conjunto dos reais, como por exemplo, os de expressão de genes. Atenção especial foi dada a este método pois através de seu algoritmo foi gerada a idéia do método Lbic proposto nesta tese. Resultados deste método foram comparados aos do método Lbic. Neste trabalho também foi proposta uma alternativa de combinação de resultados dos métodos Plaid e Lbic para a inferência sobre interações de pares de proteínas. O modelo e o algoritmo de implementação do método são apresentados a seguir.

Seja Y uma matriz de dados genômicos, onde seus elementos $\{y_{ij}\}$, com i=1,..., n e j=1, ...,c, assumem valores pertencentes ao conjunto dos reais. O modelo Plaid considerando b bi-clusters é dado por:

$$\begin{aligned} y_{ij} &= \mu_0 + \sum_{k=1}^{b} (\mu_k + \alpha_{ik} + \beta_{jk})\rho_{ik}\kappa_{jk} + e_{ij} \\ &= \mu_0 + \sum_{k=1}^{b} \theta_{ijk}\rho_{ik}\kappa_{jk} + e_{ij} \end{aligned} \tag{2.4}$$

onde:

$i = 1,...,n, j = 1,...,c$ e $k = 1,...,b$;

y_{ij} é a resposta da expressão do i-ésimo gene e j-ésima condição;

$\mu_0 = \mu_0 + \alpha_{i0} + \beta_{j0}$ é a média geral que representa a matriz de dados (bi-cluster base);

μ_k é a média do k-ésimo bi-cluster;

α_{ik} é o efeito fixo do i-ésimo gene no k-ésimo bi-cluster;

β_{jk} é o efeito fixo do j-ésimo condição no k-ésimo bi-cluster;

e_{ij} é o erro aleatório do i-ésimo gene e j-ésima condição;

$\theta_{ijk} = \mu_k + \alpha_{ik} + \beta_{jk}$ é o efeito do k-ésimo bi-cluster;

ρ_{ik} assume valor 1 se o i-ésimo gene está no k-ésimo bi-cluster e 0, caso contrário; e

κ_{jk} assume valor 1 se a j-ésima condição está no k-ésimo bi-cluster e 0, caso contrário.

As definições de ρ e κ implicam nas seguintes restrições:

$$\sum_{k=1}^{b} \rho_{ik} = 1 \qquad \text{para todo } i,$$

e

$$\sum_{k=1}^{b} \kappa_{jk} = 1 \qquad \text{para todo } j, \tag{2.5}$$

indicando que cada gene e cada condição pertence a um único bi-cluster.

A estrutura de bi-clusters descrita acima é raramente encontrada em dados reais, pois em geral, os genes participam de mais de uma função biológica e portanto um gene pode pertencer a mais de um bi-cluster. A situação mais provável de acontecer será aquela em que os bi-clusters, se sobrepõem em alguns momentos. Portanto, as restrições (2.5) podem ser relaxadas para:

$$\sum_{k=1}^{b} \rho_{ik} \geq 2 \qquad \text{para algum } i,$$

e

$$\sum_{k=1}^{b} \kappa_{jk} \geq 2 \qquad \text{para algum } j. \tag{2.6}$$

Por outro lado, existem situações em que, além de identificarem bi-clusters sobrepostos, é possível existir algum gene ou alguma condição que não pertence a qualquer bi-cluster. Neste caso as restrições para ρ e κ são:

$$\sum_{k=1}^{b} \rho_{ik} = 0 \qquad \text{para algum } i,$$

e

$$\sum_{k=1}^{b} \kappa_{jk} = 0 \qquad \text{para algum } j. \tag{2.7}$$

A última situação é a adotada nesta tese. Considerando o modelo (2.4) sob as restrições (2.7), observa-se que o valor da expressão do gene i sob a condição j é expresso como uma soma de efeitos aditivos de bi-clusters mais um termo aleatório, seguindo a estrutura do modelo de classificação (3.7), onde os efeitos dos genes e das condições são substituídos pela soma dos efeitos dos bi-clusters. Portanto as restrições $\sum$ utilizadas para estimação dos parâmetros naquele modelo são expressas no modelo Plaid como:

$$\sum_{i=1}^{n} \rho_{ik}\alpha_{ik} = 0 \qquad \text{para todo } k,$$

e

$$\sum_{j=1}^{c} \kappa_{jk}\beta_{jk} = 0 \qquad \text{para todo } k, \tag{2.8}$$

A estimativa dos parâmetros deste modelo é obtida pelo método de mínimos quadrados e é calculada da mesma forma apresentada na seção de modelo de classificação, no Capítulo 3. Portanto o efeito estimado do k-ésimo bi-cluster, sob as restrições (2.8) é calculado por:

$$\widehat{\theta}_{ijk} = \widehat{\mu}_k + \widehat{\alpha}_{ik} + \widehat{\beta}_{jk} = \bar{y}_{..k} + (\bar{y}_{i.k} - \bar{y}_{..k}) + (\bar{y}_{.jk} - \bar{y}_{..k}) = \bar{y}_{i.k} + \bar{y}_{.jk} - \bar{y}_{..k}, \tag{2.9}$$

e a média do bi-cluster base é estimada como:

$$\widehat{\mu}_0 = \bar{y}_{...},$$

onde:

$$\bar{y}_{..k} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{c} y_{ijk}}{nc}$$

e

$$\bar{y}_{...} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{c}\sum_{k=1}^{b} y_{ijk}}{ncb}.$$

O algoritmo proposto por Turner e colegas[TBKH05] identifica bi-clusters que minimizam a soma de quadrados dos resíduos do modelo Plaid, calculada por:

$$SQR_P = \sum_{i=1}^{n}\sum_{j=1}^{c} \widehat{e}_{ij}^2 = \sum_{i=1}^{n}\sum_{j=1}^{c} (y_{ij} - \widehat{\mu}_0 - \sum_{k=1}^{b} \widehat{\theta}_{ijk}\rho_{ik}\kappa_{jk})^2. \tag{2.10}$$

O método identifica um bi-cluster por vez, seguindo o Algoritmo 2.3 apresentado adiante. Para iniciar a procura por um bi-cluster k, um retalho inicial é formado através de um subconjunto de genes e de um subconjunto de condições, representados pelos parâmetros que assumem $\rho_{ik} = 1$ e $\kappa_{jk} = 1$, respectivamente. Estes parâmetros são inicialmente estimados através do método de clustering K-means, K=2, aplicado independentemente aos genes e às condições. Os clusters com menos genes e com menos condições definem o retalho inicial. Os retalhos ajustados através da reestimação dos parâmetros ρ_{ik} e κ_{jk} como segue.

$$\widetilde{\rho}_{ik} = \begin{cases} 1, & \text{se } \widehat{\rho}_{ik} = 1 \text{ e } \sum_{j:\widehat{\kappa}_{jk}=1} (\widehat{e}_{ijk} - \widehat{\kappa}_{jk}(\widehat{\mu}_k + \widehat{\alpha}_{ik} + \widehat{\beta}_{jk}))^2 < (1-\tau_1)\sum_{j:\widehat{\kappa}_{jk}=1} \widehat{e}_{ijk}^2, \\ 0, & \text{caso contrário,} \end{cases}$$

e

$$\widetilde{\kappa}_{jk} = \begin{cases} 1, & \text{se } \widehat{\kappa}_{jk} = 1 \text{ e } \sum_{i:\widehat{\rho}_{ik}=1} (\widehat{e}_{ijk} - \widehat{\rho}_{ik}(\widehat{\mu}_k + \widehat{\alpha}_{ik} + \widehat{\beta}_{jk}))^2 < (1-\tau_2)\sum_{i:\widehat{\rho}_{ik}=1} \widehat{e}_{ijk}^2, \\ 0, & \text{caso contrário,} \end{cases} \tag{2.11}$$

onde τ_1 e τ_2 representam as menores reduções na soma de quadrados dos resíduos para os subconjuntos de genes e condições, respectivamente. Tuner e colegas atribuem valores entre

0.5 e 0.7 para τ_1 e τ_2. A reestimação desses parâmetros é feita primeiramente nos ρ_i's, para todo i pertencente ao retalho, e depois nos κ_j's, para todo j pertencente ao retalho. Os novos parâmetros resultam num novo retalho. O retalho continua sendo reestimado até um número pré-determinado S de vezes ou até os estimadores de seus parâmetros se estabilizarem, isto é, até o retalho reestimado ser o igual ao estimado anteriormente. O retalho estabilizado é k-ésimo bi-cluster encontrado. O bi-cluster é adicionado ao modelo e os valores de y_{ij}, para todo i e j, são reajustados. Com base nesses novos valores, um novo bi-cluster é procurado, repetindo o processo até atingir um número pré-determinado de bi-clusters, ou nenhum retalho encontrado atender ao critério do teste de permutação. O teste de permutação é realizado permutanto aleatoriamente os elementos da matriz de dados e executando os passos descritos para identificar os bi-clusters na matriz original.

O Algoritmo 2.3, apresentado a seguir, representa a estrutura desse algoritmo. Por simplicidade, o índice k que representa o retalho é eliminado.

Algoritmo 2.3

1. Calcule a matriz de resíduos por $\widehat{e}_{ij} = Y_{ij} - \widehat{\theta}_{ij0}$.
2. Calcule valores iniciais $\widehat{\rho}_i^0$ e $\widehat{\kappa}_j^0$ através de clustering K-means, K=2.
3. Faça s=1.
4. Calcule os efeitos do retalho usando e^*: sub-matriz de $\widehat{e}_{ij}$ obtida por $\widehat{\rho}_i^{s-1}$ e $\widehat{\kappa}_j^{s-1}$.

$$\widehat{\mu}^s = \overline{e}_{..}^*$$

$$\widehat{\alpha}_i^s = \begin{cases} \overline{e}_{i.}^* - \widehat{\mu}^s, & \forall i : \widehat{\rho}_i^{s-1} = 1, \\ 0, & \text{caso contrário,} \end{cases}$$

$$\widehat{\beta}_j^s = \begin{cases} \overline{e}_{.j}^* - \widehat{\mu}^s, & \forall j : \widehat{\kappa}_j^{s-1} = 1, \\ 0, & \text{caso contrário.} \end{cases}$$

5. Calcule os parâmetros do retalho s:

$$\widetilde{\rho}_i^s = \begin{cases} 1, & \text{se } \sum_j (\widehat{e}_{ij} - \widehat{\kappa}_j^{s-1}(\widehat{\mu^s} + \widehat{\alpha}_i^s + \widehat{\beta}_j^s))^2 < (1-\tau_1)\sum_j \widehat{e}_{ij}^2, \\ 0, & \text{caso contrário,} \end{cases}$$

$$\widetilde{\kappa}_j^s = \begin{cases} 1, & \text{se } \sum_i (\widehat{e}_{ij} - \widehat{\rho}_i^{s-1}(\widehat{\mu^s} + \widehat{\alpha}_i^s + \widehat{\beta}_j^s))^2 < (1-\tau_e)\sum_i \widehat{e}_{ij}^2, \\ 0, & \text{caso contrário.} \end{cases}$$

6. Repita os passos 4 e 5 para s=2, ..., S, onde S é o número de iterações para estabilizar o retalho.
7. Calcule $\widehat{\mu}^{S+1}$, $\widehat{\alpha}_i^{S+1}$ e $\widehat{\beta}_j^{S+1}$, como no passo 4.
8. Calcule $\widehat{\rho}_i^{S+1}$ e $\widehat{\kappa}_j^{S+1}$ como no passo 5.
9. Calcule a soma dos quadrados dos valores ajustados do retalho,
   $SQr = \sum_{ij} (\widehat{\mu} + \widehat{\alpha}_i + \widehat{\beta}_j)\widehat{\rho}_i\widehat{\kappa}_j$.
10. Permute $\widehat{e}_{ij}$ e siga os passos de 2 a 9.
11. Repita o passo 10 T vezes, onde T é o número de retalhos permutados a serem usados no teste de permutação.
12. Aceite o retalho como um bi-cluster se seu SQr é menor que todos os T SQr calculados para as T matrizes permutadas. Caso contrário, pare.
13. Reajuste todos os bi-clusters no modelo R vezes.

Revisão

A figura 2.2 apresenta os quatro aspectos a serem atendidos pelos bi-clusters procurados pelo método Plaid.

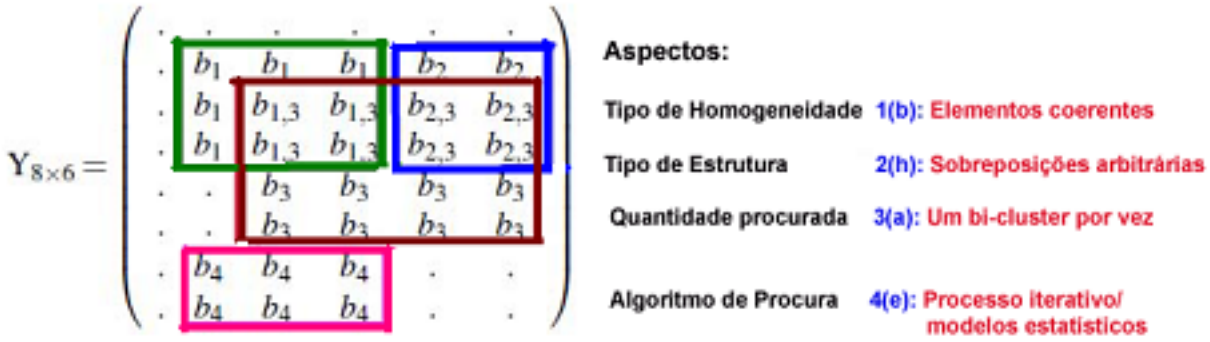

Figura 2.2 Aspectos do Bi-cluster - Plaid

2.1.2.4 Abordagem de Biclustering para Dados Genômicos Binários Método Bicbin [UW08]

Métodos de Biclustering para dados binários foram pouco estudados. Segundo Uitert e Wessels [UW08], além do método Bicbin, apenas dois métodos, *Cmnk* proposto por Koyuturk e colegas [KSG04], e Bicmax proposto por Preli'c e colegas [PBZea06] trataram deste tema. O método

Bicmax foi comparado com alguns daqueles do artigo de Madeira e colegas apresentados na Seção 2.1.2.2, mostrando a importância de um método próprio para dados binários. Ainda segundo Uitert e Wessels, o método Bicbin mostrou ser superior a estes dois métodos. Assim, nesta tese foi abordado apenas o método Bicbin.

O método Bicbin identifica bi-clusters em matrizes de dados binários através de uma função escore. Seja $Y_{n\times c}$ uma matriz de dados binários com proporção de uns igual a p. O Bicbin assume que cada elemento da matriz é um resultado de uma prova de Bernoulli com probabilidade de sucesso p. Seja $X_{n',c'}$ uma variável aleatória denotando o número de uns numa submatriz de tamanho $n' \times c'$. Então, $X_{n',c'}$ segue uma distribuição Binomial com parâmetros $n'c'$ e p. O método visa encontrar submatrizes com baixos valores da probabilidade $P(X_{n',c'} > u)$, onde u é o número de uns na matriz. Como a proporção de uns em matrizes binárias esparsas é pequena, a probabilidade $P(X_{n',c'} > u)$ usualmente é muito pequena, alcançando rapidamente o limite de precisão da máquina usada para fazer sua computação. Uitert e colegas propõem uma versão multiplicativa do limite de Chernoff para calcular tal probabilidade. Assim, a probabilidade é limitada como

$$P(X_{n',c'} > u) \leq \begin{cases} \mathrm{e}^{-\frac{(u-n'c'p)^2}{3n'c'p}}, & u \in [n'c'p, 2n'c'p]\,; \\ \mathrm{e}^{-\frac{(u-n'c'p)^2}{u+n'c'p}}, & u > 2n'c'p. \end{cases}$$

Baseado no limite desta probabilidade, em vez do método procurar por matrizes com baixa probabilidade de uns, ele procura por matrizes com altos valores para o expoente da função apresentada acima.

$$\widetilde{C}(n',c',u) = \begin{cases} \frac{(u-n'c'p)^2}{3n'c'p}, & u \in [n'c'p, 2n'c'p]\,; \\ \frac{(u-n'c'p)^2}{u+n'c'p}, & u > 2n'c'p. \end{cases}$$

O escore $\widetilde{C}(n',c',u)$ leva em consideração os tamanhos das linhas e colunas simultaneamente. Por tal motivo, este escore é normalizado por um fator $n'^{\alpha}c'^{\beta}$ com α e $\beta \in [0,1]$.Assim o Bicbin propõe encontrar submatrizes de tamanho $n' \times c'$, com u elementos uns e que tenha máximo escore para a função

$$C_{\alpha,\beta} = \frac{\widetilde{C}(n',c',u)}{n'^{\alpha}c'^{\beta}} = \begin{cases} \frac{(u-n'c'p)^2}{n'^{\alpha}c'^{\beta}3n'c'p}, & u \in [n'c'p, 2n'c'p]\,; \\ \frac{(u-n'c'p)^2}{n'^{\alpha}c'^{\beta}(u+n'c'p)}, & u > 2n'c'p. \end{cases}$$

O algoritmo Bicbin é apresentado em três etapas. A primeira etapa, apresentada no Algoritmo 2.4, encontra um bi-cluster com $C_{\alpha,\beta}$ máximo. A segunda, apresentada no Algoritmo 2.5, encontra um p_1-bi-cluster com $C_{\alpha,\beta}$ máximo. E a última, apresentada no Algoritmo 2.6, encontra todos p_1-bi-clusters com $C_{\alpha,\beta}$ máximo. Os Esquemas do algoritmo são apresentados a seguir.

Algoritmo 2.4

1. Selecione um subconjunto de linhas e crie um vetor ρ tal que $\rho_i = 1$ se a linha i é selecionada e, $\rho_i = 0$, caso contrário $(i = 1, ..., n)$.

2. Compute as somas das colunas para as linhas selecionadas: $s = Y'\rho$.

3. Ordene decrescentemente os valores do vetor **s** e os rotule por $\pi_1, ..., \pi_c$, onde π_1 é a coluna com maior soma e π_c é a coluna com menor soma.

4. Considere o subconjunto de colunas $C_{c'} := \{\pi_1, ..., \pi_{c'}\}$. Para cada $c' = \{1, ..., c\}$ é formado um bi-cluster com linhas $R := \{i | \rho_i = 1\}$ e um subconjunto de colunas $C_{c'}$. Para cada um dos c bi-clusters calcule:

$$C_{\alpha,\beta}(c') = \begin{cases} \frac{(\sum_{j=1}^{c} s(\pi_j) - n'c'p)^2}{3n'^{1+\alpha}c'^{1+\beta}p}, & \sum_{j=1}^{c} s(\pi_j) \in [n'c'p, 2n'c'p]; \\ \frac{(\sum_{j=1}^{c} s(\pi_j) - n'c'p)^2}{n'^{\alpha}c'^{\beta}(\sum_{j=1}^{c} s(\pi_j) + n'c'p)}, & \sum_{j=1}^{c} s(\pi_j) > 2n'c'p. \end{cases}$$

5. Determine $c'^* = \arg\max_{c'} \left\{C_{\alpha,\beta}(c')\right\}$, obtendo um conjunto de colunas $C_{c'^*} = \{\pi_1, ..., \pi_{c'^*}\}$.

6. Construa um vetor κ tal que $\kappa_j = 1$ se $j \in C_c'^*$ e 0, caso contrário.

7. Volte ao o passo 2 e repita o processo para as colunas com $s = Y\kappa$. Pare se $C_{\alpha,\beta}(c'^*)$ não aumenta. Linhas com $\rho = 1$ e colunas com $\kappa = 1$ formam um bi-cluster com $C_{\alpha,\beta}$ obtido do máximo de $C_{\alpha,\beta}(c'^*)$.

8. Repita os passos de 1 a 7 I vezes. Retorne o bi-cluster com máximo $C_{\alpha,\beta}$ para todas I execuções.

Algoritmo 2.5

1. Execute o Algoritmo 2.4 para a matriz de dados Y e obtenha um bi-cluster B com $C_{\alpha,\beta}$ máximo.

2. Calcule p_B, a proporção de uns em B.

3. Pare se $p_B \geq p_1$, B é um bi-cluster com $C_{\alpha,\beta}$ máximo. Se $p_B < p_1$, execute o Algoritmo 2.4 em B, denote o novo bi-cluster por B e volte ao passo 2.

Algoritmo 2.6

1. Execute o Algoritmo 2.5 em Y e obtenha um p_1-bi-cluster com $C_{\alpha,\beta}$ máximo.
2. Coloque o bi-cluster numa lista de bi-clusters. Coloque seus elementos em Y iguais a zero.
3. Verifique se Y ainda contém elementos não zeros. Se não, pare, senão volte ao passo 1.

Revisão

A figura 2.3 apresenta os quatro aspectos a serem atendidos pelos bi-clusters procurados pelo método Bicbin.

Figura 2.3 Aspectos do Bi-cluster - Bicbin

2.2 Métodos de Inferência de Proteína-Proteína

Nesta seção serão apresentados alguns métodos de inferência de proteína-proteína.

1. Integração de Dados Filogenéticos e de Expressão [HKC+04]

Em seu artigo Haugen e colegas [HKC+04] falam sobre a importância da integração de dados para realizar a inferência de pares de proteínas. Eles integram dados de expressão de genes e dados filogenéticos que medem o impacto do metal arsênico aplicado à levedura *Saccharomyces cerevisiae*. O metal arsênico é um poluente do meio ambiente e um cancerígeno humano. Ele

ainda é usado no tratamento de leucemia, o que gera a necessidade de conhecimento do impacto do mesmo no organismo humano. No entanto ainda não existe um modelo matemático que descreva a doença no ser humano de maneira eficiente.

Desde que proteínas afetadas pelo metal arsênico têm sido descritas em organismos como o *Saccharomyces cerevisiae*, os autores propõem uma metodologia que utiliza os dados integrados para inferir sobre a rede de proteínas usando a levedura citada. O método constrói dois tipos de redes de proteínas: regulatória e metabólica. A construção da rede regulatória é realizada via o algoritmo *ActiveModules*, o qual usa um escore de vizinhança, enquanto a construção da rede metabólica é realizada através de um algoritmo proposto pelos autores, o qual usa a técnica de programação dinâmica.

2. Integração de Múltiplas Espécies [SSK$^+$04]

Redes de proteínas construídas com informação de uma única espécie podem apresentar interações proteína-proteína que são consideradas falso-positivas. Essas interações, muitas vezes complexas, poderiam ser captadas através de uma abordagem que utilizasse informações de várias espécies. Por isso Sharan e colegas [SSK$^+$04] consideram três espécies distintas na construção de redes de proteínas: *Caenoshabditis elegans*, *Drosophila melanogaster* e *Saccharomyces cerevisiae*.

A metodologia proposta utiliza uma extensão do método *PATHBLAST* [KSK$^+$03]. Eles representam a rede de proteínas por um grafo, onde cada nó consiste de um grupo de proteínas com sequências similares nas três espécies, e cada ligação entre dois nós re-presenta a interação entre os grupos de proteínas. A idéia é procurar sub-redes que identifiquem dois tipos de estruturas conservadas: caminho linear curto de interação de proteínas, a qual modela os caminhos de traduções de sinais; e clusters densos de interações, a qual modela complexos de proteínas.

A procura das sub-redes é guiada pelas estimativas da confiança de cada interação de proteínas propostas em Bader e colegas [BCRC04] e calculadas através de um modelo probabilístico. Em uma sub-rede real é assumido que cada interação deve estar presente independentemente e com alta probabilidade. O método compara esta rede a uma sub-rede artificial gerada aleatoriamente com probabilidade de interação entre duas proteínas quaisquer dependendo do número total de conexões de sua rede. A comparação entre sub-rede real ajustada a uma estrutura desejada (caminho ou cluster) e a sub-rede artificial, é realizada através do logaritmo da razão das verossimilhanças entre o modelo probabilístico ajustado a sub-rede desejada e o modelo ajustado a sub-rede artificial.

3. Combinação Redes Físicas e Genéticas [KI05]

Alguns estudos sobre interações de proteínas têm mostrado que duas proteínas aparecendo numa mesma região de redes genéticas são prováveis de interagir fisicamente. Kelley e Ideker [KI05] combinam redes físicas e genéticas em regiões onde estas proteínas são correlacionadas, sugerindo a possibilidade de interpretar relações do tipo *synthetic-lethal* usando interações físicas: entre caminhos e dentro dos caminhos. Esses tipos de interações em redes genéticas têm as seguintes interpretações: interação entre caminhos (a qual acontece entre dois genes de dois diferentes caminhos) afirma que a eliminação de um gene muda a função do caminho ao qual o

gene está associado, mas não muda a função do outro caminho; por outro lado, a interação dentro do caminho (que acontece entre dois genes de um mesmo caminho) afirma que a eliminação de um gene não muda a função do seu caminho, mas os efeitos de vários genes eliminados são letais.

A verificação se um determinado grupo (sub-rede desejada) de proteínas é mais conectado, isto é, tem mais interações entre as proteínas do que seria esperado em um grupo conectado aleatoriamente (sub-rede gerada artificialmente), é realizada através do escore calculado do logaritmo da razão das verossimilhanças entre o modelo probabilístico ajustado a sub-rede desejada e o modelo ajustado a sub-rede artificial. Este escore é obtido tanto para a rede física, quanto para a genética para os dois tipos de interações.

4. Modelos de Redes Bayesianas [SG05]

Santos e Guimarães [SG05] apresentam outra abortagem para fazer inferência de interações de proteínas. Tal abordagem faz uso de modelos de redes bayesianas utilizando conhecimentos biológicos a priori e combinações de regressões não paramétricas.

5. Análise de Correlação Canônica com Kernel (ACCK) [YVNK03, YVK04, YVK05]

Yamanishi e colegas [YVNK03] descrevem duas alternativas para medir a correlação entre diferentes conjuntos de dados genômicos, de modo a encontrar subconjuntos de genes que dividem similaridades com respeito a fatores biológicos desses diferentes conjuntos. Ambas as abordagens utilizam uma extensão de análise de correlação canônica com Kernel (ACCK) [Aka01], método que encontra direções entre dois espaços característicos com máxima correlação. Estes espaços característicos são definidos por funções de kernel que são aplicadas às matrizes dos conjuntos genômicos e portanto, transformando-as em matrizes de kernels onde seus elementos representam similaridades entre genes.

A primeira abordagem, chamada de análise de correlação canônica múltipla, é uma generalização do método ACCK para mais de dois conjuntos de dados genômicos. A segunda, chamada de análise de correlação canônica integrada, maximiza a correlação entre dois grupos de dados genômicos combinados através da soma de suas matrizes de kernel. Esses métodos são aplicados a três conjuntos correspondendo às relações funcionais entre genes em caminhos metabólicos utilizados para encontrar *operons* em *Escherichia coli*.

Ainda fazendo inferência sobre redes de proteínas baseado em ACCK, Yamanishi e colegas acrescentam supervisão ao método, considerando o conhecimento de redes de proteínas consideradas confiáveis [YVK04]. Os autores aplicam a metodologia a quatro tipos de dados genômicos da levedura *Saccharomyces cerevisiae*. Este método é comparado com o método Lbic no capítulo de aplicação, por essa razão é detalhado na Seção 2.2.1.

Em novo artigo, Yamanishi e colegas [YVK05] além de integrarem diferentes tipos de dados genômicos e utilizarem métodos supervisionados, incorporam informações químicas ao método de inferência de rede de proteínas. O artigo ainda mostra uma maneira alternativa de combinar os tipos de dados genômicos. Em vez de ser utilizada a média aritmética dos diferentes kernels, é calculada uma média ponderada das matrizes transformadas pelos kernels. A ponderação é dada pelos escores proporcionais estimados através da área abaixo da curva ROC (Receiver Operating Characteristic) de cada conjunto de dados. A curva ROC avalia a variação entre a sensibilidade e a especificidade para diferentes limiares [Met78].

2.2.1 Análise de Correlação Canônica com Kernel Supervisionada ACCKS [YVK04]

O método ACCKS é supervisionado por parte de uma rede de proteínas conhecida, chamada de rede padrão ouro. A idéia central do método é encontrar um espaço característico especial, ajustado à parte supervisionada, onde os clusters são mais fáceis de serem identificados e então realizar a análise clássica de clustering, isto é, analisar as proteínas numa região que é mais fácil de encontrar proteínas que apresentem algum tipo de relação, isto é, proteínas que interagem.

O espaço característico é definido como o espaço gerado pelos S primeiros autovetores da matriz de similaridades entre as proteínas (definida por uma função kernel). Os autovetores são obtidos através do método de análise de componentes principais.

Seja $\chi = \{\mathbf{y}^1, ..., \mathbf{y}^N\}$ um conjunto de N vetores representando N proteínas, uma função kernel

$$\begin{aligned} &\mathrm{K}(\mathbf{y}^l, \mathbf{y}^{l'}) = \mathrm{K}(\mathbf{y}^{l'}, \mathbf{y}^l) && \text{para quaisquer duas proteínas, e} \\ &\sum_{l=1}^{N} \alpha_l \alpha_{l'} \mathrm{K}(\mathbf{y}^l, \mathbf{y}^{l'}) \geq 0 && \text{para } N \text{ inteiro,} \end{aligned} \tag{2.12}$$

onde:

$\mathbf{y}^l = [y_{l1}, ..., y_{lc}]$ e $\mathbf{y}^{l'} = [y_{l'1}, ..., y_{l'c}]$ representam vetores de c observações das proteínas l e l', respectivamente , para $l, l' = 1, ..., N$; e

$\alpha_1, ..., \alpha_N$ assumem valores pertencentes ao conjunto dos reais.

Como os dados genômicos utilizados no artigo dado apresentam formas diferentes de expressar as proteínas, os mesmos, representados por matrizes, são transformados através de funções de kernel. Dessa maneira, todos os tipos de conjuntos de dados podem ser tratados da mesma forma.

Sejam P conjuntos de dados e $K_1, ..., K_P$ matrizes de kernels que representam as similaridades das proteínas com respeito ao p-ésimo conjunto. Os autores propõem uma integração desses conjuntos através da média aritmética das suas matrizes de kernels, dada por:

$$\bar{K}_{\text{int}} = \sum_{p=1}^{P} \frac{\mathrm{K}_p}{P}. \tag{2.13}$$

Baseado em uma função de kernel pode-se gerar o conjunto de funções:

$$H = \{f(\mathbf{y}) = \textstyle\sum_{l=1}^{N} \alpha_l \mathrm{K}(\mathbf{y}^l, \mathbf{y}), (\alpha_1, ..., \alpha_N) \in \mathbb{R}^N\},$$

dada a norma:

$$\|f\|_H = \textstyle\sum_{l,l'} \alpha_l \alpha_{l'} \mathrm{K}(\mathbf{y}^l, \mathbf{y}^{l'}).$$

Portanto a projeção sobre a primeira direção principal é definida, a menos de um escalar, como uma função $f^{(1)} \in H$ que minimiza $\left\| f^{(1)} \right\|_H$ sob a restrição:

$$\sum_{l=1}^{N} f^{(1)}(\mathbf{y}^l)^2 = 1, \tag{2.14}$$

enquanto as projeções sobre as direções principais posteriores são definidas recursivamente, de maneira similar à primeira, mas com a adicional restrição de ortogonalidade:

$$\sum_{l=1}^{N} f^{(s)}(\mathbf{y}^l) f^{(s')}(\mathbf{y}^l) = 0 \qquad \text{se} \qquad s < s'. \tag{2.15}$$

O método ACCK representa cada proteína ($\mathbf{y}^l$) pelo vetor:

$$\left[f^{(1)}(\mathbf{y}^l), ..., f^{(S)}(\mathbf{y}^l) \right]^{\mathrm{T}},$$

onde $l = 1, ..., N$, $S < c$ e $f^{(s)}(\mathbf{y})$ é a projeção de $\mathbf{y}$ sobre o s-ésimo componente principal.

Portanto, para verificar se as proteínas l e l' interagem, deseja-se que haja similaridade entre $f^{(s)}(\mathbf{y}^l)$ e $f^{(s)}(\mathbf{y}^{l'})$ para $s = 1, ..S$.

As projeções no espaço característico ideal não podem ser computadas uma vez que a rede completa das proteínas não é conhecida. O método SKCCA restringe este espaço a um espaço característico ideal ajustado (pelo menos na parte conhecida a priori).

Seja um conjunto de dados genômicos com N proteínas, onde $\mathbf{y}^1, ..., \mathbf{y}^n$ são os vetores das proteínas pertencentes à rede conhecida a priori, e $\mathbf{y}^{n+1}, ..., \mathbf{y}^N$ os vetores das demais proteínas. Sejam K_1 a matriz de kernels do conjunto dados genômicos, $n \times n$, relativa àquelas n primeiras proteínas e K_2 a matriz de kernels, $n \times n$, relativa à rede de proteína conhecida a priori. Para qualquer função f escolhida para as n proteínas são definidas as normas $\|f_1\|$ e $\|f_2\|$ para K_1 e K_2, respectivamente. Portanto ACCKS procura duas funções que satisfazem:

$$\sum_{l=1}^{N} f_k(\mathbf{y}^l)^2 = 1 \qquad k = 1, 2, \tag{2.16}$$

e que maximizam a expressão:

$$\mathrm{Corr}(f_1, f_2) \times \frac{1}{\sqrt{1 + \lambda_1 \|f_1\|^2}} \times \frac{1}{\sqrt{1 + \lambda_2 \|f_2\|^2}}, \tag{2.17}$$

onde:

λ_1 e λ_2 são parâmetros de regularização positivos; e

Corr(f_1, f_2) é o coeficiente de correlação entre f_1 e f_2.

As características posteriores podem ser definidas recursivamente maximizando a Expressão (2.17) adicionada das restrições de ortogonalidade. Tal procedimento é equivalente a resolver a expressão de autovalor generalizado dada por:

$$\begin{pmatrix} 0 & K_1K_2 \\ K_1K_2 & 0 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix} = \rho \begin{pmatrix} (K_1 + \lambda_1 I)^2 & 0 \\ 0 & (K_2 + \lambda_2 I)^2 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha \end{pmatrix}, \tag{2.18}$$

onde I é uma matriz identidade, α_1 e α_2 são autovetores, e ρ é um autovalor. Logo, as funções de proteínas procuradas pela ACCKS podem ser dadas por:

$$f_1 = K_1\alpha_1 \text{ e } f_2 = K_2\alpha_2,$$

onde α_1 e α_2 são os autovetores da Equação (2.18).

Sejam $\alpha_1^{(1)}, ..., \alpha_1^{(S)}$ as S primeiras soluções da Equação (2.18), obtidas através de valores decrescentes de ρ. Então as S características de interesse do K_1 são:

$$f_1^{(s)} = K_1\alpha_1^{(s)}, \text{ para s = 1,..., S.}$$

Estas características podem ser generalizadas para qualquer proteína $\mathbf{y}$ como segue:

$$f^{(s)}(\mathbf{y}) = \sum_{l=1}^{n} \alpha_1^{(s)}(\mathbf{y}^l)\mathrm{K}(\mathbf{y}^l, \mathbf{y}). \tag{2.19}$$

O método de ACCKS representa cada proteína ($\mathbf{y}^l$) pelo vetor:

$$\mathbf{u} = \left[f^{(1)}(\mathbf{y}^l), ..., f^{(S)}(\mathbf{y}^l) \right]^{\mathrm{T}}.$$

Diz-se que duas proteínas (representadas por u e u') interagem se o coeficiente de correlação entre u e u' é maior que um limiar dado. O coeficiente de correlação é dado por:

$$\mathrm{Corr}(\mathbf{u}, \mathbf{u}') = \frac{\mathrm{Cov}(\mathbf{u}, \mathbf{u}')}{\sqrt{\mathrm{Var}(\mathbf{u})}\sqrt{\mathrm{Var}(\mathbf{u}')}}. \tag{2.20}$$

2.3 Conclusões

Neste capítulo foram apresentadas revisões dos métodos de agrupamento e de inferência de interação de proteína-proteína. Nas Seções 2.1.2.3 e 2.1.2.4 foram detalhados os métodos Plaid e Bicbin uma vez que os mesmos serão comparados com Lbic no Capítulo 5. Os métodos apresentados na Seção 2.2 mostram a importância de alternativas para fazer inferência de redes de proteínas usando: combinações de tipos de dados genômicos para uma mesma espécie; combinações de dados genômicos de mesmo tipo, mas com diferentes espécies; e combinações de tipos de redes. A maioria desses métodos procura encontrar regiões onde as proteínas sejam mais similares quanto a alguma característica. O método ACCKS foi detalhado para compreenssão na comparação do mesmo com o Lbic no Capítulo 5. Na literatura, a idéia de combinar informações tem mostrado ser eficiente, por isso nesta tese também é proposta uma alternativa de combinação de informações dos dados de expressão de genes e filogenéticos através dos métodos Lbic e Plaid.

CAPÍTULO 3

Modelos Estatísticos

A importância de trabalhar com diferentes tipos de dados genômicos está na oportunidade de captar uma maior quantidade de relações entre os genes. Nesta tese são analisados dados de expressão de genes, cujo resultado medido é um valor contínuo, e dados filogenéticos, cujo resultado medido para cada gene sob uma condição é binário. Por causa da natureza distinta de tais dados, é necessário encontrar ferramentas que tratem adequadamente suas informações. Neste capítulo são revisados os principais conceitos de modelos lineares e modelos não lineares, que serão úteis para este estudo. Estes modelos são usados como base teórica para as metodologias Plaid e Lbic.

3.1 Modelos lineares

Modelos lineares são amplamente aplicados nas mais diversas áreas da ciência para investigar relações entre variáveis. É frequente na literatura representar um modelo linear geral em forma matricial, como:

$$\mathbf{y} = X\beta + \varepsilon, \tag{3.1}$$

onde:

$\mathbf{y} = [y_1, ..., y_N]^{\mathrm{T}}$ é um vetor coluna $N \times 1$ de observações da variável resposta;

$X = [\mathbf{1}, \mathbf{x}_1, ..., \mathbf{x}_p]$ é a matriz do modelo $N \times (p+1)$ contendo observações de p variáveis independentes, também chamadas explicativas, denotadas por $\mathbf{x}_1, ..., \mathbf{x}_p$, e um vetor unitário $N \times 1$ de uns, denotado por $\mathbf{1}$;

$\beta = [\beta_0, \beta_1, ..., \beta_p]^{\mathrm{T}}$ é o vetor $(p+1) \times 1$ de parâmetros cujos valores são desconhecidos e fixos; e

$\varepsilon = [e_1, ..., e_N]^{\mathrm{T}}$ é um vetor coluna $N \times 1$ de termos aleatórios chamados de erros.

Este modelo é dito linear pois a equação é linear nos parâmetros. Portanto modelos quadráticos ou polinomiais mantendo a linearidade nos parâmetros são considerados modelos lineares.

O modelo (3.1) é chamado de Gauss-Markov se ele atende às seguintes condições (de Gauss-Markov):

1. $\mathrm{E}(\varepsilon_l) = 0$, $l = 1, ..., N$, o que implica $\mathrm{E}(\mathbf{y}) = X\beta$;
2. $\mathrm{Var}(\varepsilon_l) = \sigma^2$, $l = 1, ..., N$, o que implica $\mathrm{Var}(\mathbf{y}) = \sigma^2 \mathrm{I}$, onde I é a matriz identidade; e
3. $\mathrm{Cov}(\varepsilon_l, \varepsilon_{l'}) = 0$, $l \neq l'$, o que implica $\mathrm{Cov}(y_l, y_{l'}) = 0$. (3.2)

O modelo linear de Gauss-Markov é flexível bastante para acomodar diversas situações de apelo prático, permitindo a realização de inferências relativas a comparações de médias, predição, identificação de variáveis explicativas relevantes e de clustering, dentre outras. Tais situações práticas surgem naturalmente dos objetivos específicos de estudos científicos, e dependem, por exemplo, das naturezas da variável resposta $\mathbf{y}$ e das variáveis explicativas $\mathbf{x}_1, ..., \mathbf{x}_p$; e da estrutura da matriz X do modelo.

Frequentemente há interesse em realizar inferências para combinações lineares dos componentes de β, representadas por $\lambda^{\mathrm{T}}\beta$, onde λ é um vetor de constantes $(p+1)\times 1$. Para tanto, é necessário que $\lambda^{\mathrm{T}}\beta$ seja estimável. A definição de estimabilidade é apresentada abaixo.

Definição 3.1.1

Considere o modelo de Gauss-Markov e $\lambda = [\lambda_0, ..., \lambda_p]^{\mathrm{T}}$ um vetor de constantes $(p+1)\times 1$. A combinação linear $\lambda^{\mathrm{T}}\beta$ é estimável se existe um estimador de $\lambda^{\mathrm{T}}\beta$ centrado e função linear de $\mathbf{y}$, isto é, se existe um vetor $\mathbf{a}$, $(p+1)\times 1$, tal que E($\mathbf{a}^{\mathrm{T}}\mathbf{y}$) $=\lambda^{\mathrm{T}}\beta$.

Uma consequência imediata da definição é que $\mathbf{a}^{\mathrm{T}}\mathbf{y}$ é um estimador de $\lambda^{\mathrm{T}}\beta$, se e somente se, $\lambda^{\mathrm{T}} = \mathbf{a}^{\mathrm{T}}X$, isto é, se e somente se, λ^{T} pertence ao espaço linha de X [Har98, Sea71].

Existem outras condições necessárias e suficientes para $\lambda^{\mathrm{T}}\beta$ ser estimável. Uma delas é descrita a seguir.

Proposição 3.1.1

Considere o modelo de Gaus-Markov. Sejam λ um vetor de constantes $(p+1)\times 1$; $s = (p+1) - r(X)$, onde $r(X)$ é o número de colunas linearmente independentes na matriz do modelo X (chamado de posto de X); e $\mathbf{c}_1, ..., \mathbf{c}_s$ s vetores, $(p+1)\times 1$, linearmente independentes, tais que $X\mathbf{c}_1 = ... = X\mathbf{c}_s = \mathbf{0}$. Então uma combinação linear $\lambda^{\mathrm{T}}\beta$ é dita estimável se, e somente se, os componentes de λ são tais que $\lambda^{\mathrm{T}}\mathbf{c}_1 = ... = \lambda^{\mathrm{T}}\mathbf{c}_s = 0$.

A prova dessa proposição pode ser encontrada na literatura [Har98, Sea71]. Tal resultado apresenta uma maneira prática de investigação da estimabilidade de $\lambda^{\mathrm{T}}\beta$, especialmente quando o valor de s é pequeno. Por exemplo, se $s = 1$, pode-se evidenciar apenas uma relação entre as colunas de X e verificar se a mesma relação ocorre para os correspondentes elementos do vetor λ.

Algumas propriedades das funções estimáveis são listadas abaixo.

1. Quando a matriz X do modelo é de posto coluna completo, isto é, $r(X) = (p+1)$, então, todas as funções $\lambda^{\mathrm{T}}\beta$ são estimáveis, inclusive $\beta_0, ..., \beta_p$.
2. Quando a matriz X do modelo não é de posto coluna completo $(r(X) < (p+1))$, então, pelo menos um dos elementos de β não é estimável.
3. Independente da estrutura da matriz X do modelo, $X\beta$ é sempre estimável.
4. Se λ^{T} pertence ao espaço linha de X, isto é, se $\lambda^{\mathrm{T}} = c^{\mathrm{T}}X$ então $\lambda^{\mathrm{T}}\beta$ é estimável.

A seguir são considerados dois casos para ilustrações de aplicações de modelos lineares de Gauss-Markov.

3.1.1 Caso 1: Modelo de Regressão Múltipla

O modelo de regressão múltipla é um caso particular do modelo de Gauss-Markov, quando tanto a variável resposta quanto as variáveis explicativas são contínuas (assumem valores pertencentes ao conjunto dos reais). Neste caso, um dos principais interesses é o de realizar predições dos valores de $\mathbf{y}$. Algumas propriedades dos estimadores dos parâmetros do vetor β devem ser atendidas para identificar variáveis explicativas importantes. Em estudos de regressão múltipla, a matriz X do modelo é de posto coluna completo, $r(X) = p+1$, e como consequência os parâmetros de β são estimáveis [Sea82, Ren00]. Um método que usualmente é aplicado para estimar esses parâmetros é o de mínimos quadrados, que não requer nenhuma suposição sobre a distribuição dos erros [Sea71, NWKN96, Ren00]. O método de mínimos quadrados procura estimadores dos parâmetros de β, definidos por $\widehat{\beta}$, que minimizem a soma de quadrados dos desvios entre os valores observados, y_i, e seus valores preditos,$\widehat{y}_i$.

Seja $X\widehat{\beta}$ o preditor de $\mathbf{y}$. A soma dos quadrados dos desvios pode ser escrita em forma matricial como segue.

$$Q(\widehat{\beta}) = (\mathbf{y} - X\widehat{\beta})^{\mathrm{T}}(\mathbf{y} - X\widehat{\beta}). \tag{3.3}$$

Sejam as derivadas parciais de $Q(\widehat{\beta})$, com respeito a $\widehat{\beta}$, dadas por:

$$\frac{\partial Q(\widehat{\beta})}{\partial \widehat{\beta}} = -2X^{\mathrm{T}}\mathbf{y} + 2X^{\mathrm{T}}X\widehat{\beta}. \tag{3.4}$$

Uma maneira de achar o mínimo de $Q(\widehat{\beta})$, é resolver o sistema de $p+1$ equações lineares, chamadas de equações normais, expressas em forma matricial como:

$$-2X^{\mathrm{T}}\mathbf{y} + 2X^{\mathrm{T}}X\widehat{\beta} = \mathbf{0}. \tag{3.5}$$

As soluções das equações normais são os estimadores de mínimos quadrados do vetor β, $\widehat{\beta}$, dado por:

$$\widehat{\beta} = (X^{\mathrm{T}}X)^{-1}X^{\mathrm{T}}\mathbf{y}. \tag{3.6}$$

Sob as condições de Gauss-Markov (3.2), os estimadores de mínimos quadrados, $\widehat{\beta}$, são BLUE (Best Linear Unbiased Estimators), isto é, têm menor variância dentre todos estimadores lineares centrados [Sea71, NWKN96, Ren00]. A análise de modelos de regressão é um assunto amplamente explorado na literatura. Para saber mais é possível consultar, por exemplo, Searle [Sea71], e Neter e colegas [NWKN96].

3.1.2 Caso 2: Modelo de Classificação

Em alguns estudos a variável explicativa é determinada através de uma subdivisão das unidades observadas em classes, baseada em algum critério. Estas classes são disjuntas e exaustivas. Ao critério usado para gerar classes é dado o nome de fator, enquanto as classes de unidades correspondem aos níveis do fator. Por exemplo, suponha que uma variável explicativa em estudo seja

a localização do gene na célula [HFG+03] com quatro possíveis resultados: mitocôndria, golgi, núcleo e outras. O fator neste exemplo é a localização, que possui quatro níveis: mitocôndria, golgi, núcleo e outras. Variáveis cujas respostas são categorias ou classes são chamadas de categorizadas ou classificatórias. O modelo de classificação é um caso particular do modelo de Gauss-Markov, quando a variável resposta é contínua e as variáveis explicativas são classificatórias.

Ao trabalhar com dados de expressão de genes é desejado explicar as similaridades dos genes quanto às expressões obtidas nas diferentes condições de um experimento. Quanto mais similares os genes são, maior a possibilidade de interação entre eles. Entender como os níveis de expressão de cada gene se comportam é de grande interesse para poder fazer inferência sobre as interações dos genes. O método Plaid proposto por Tuner e colegas [TBKH05], por exemplo, utiliza o modelo de classificação com duas variáveis classificatórias (genes e condições) para modelar as expressões dos genes e identificar bi-clusters que minimizam a soma dos quadrados dos resíduos de um modelo similar ao de classificação.

Suponha que uma variável classificatória 1, chamada de fator 1, tenha I níveis, e uma variável classificatória 2, chamada de fator 2, tenha J níveis. Então, tem-se o seguinte modelo de classificação:

$$y_{ij} = \mu + \alpha_i + \beta_j + e_{ij}, \tag{3.7}$$

onde:

y_{ij} é a resposta observada pelo i-ésimo nível do fator 1 e j-ésimo nível do fator 2;

μ é a média geral;

α_i é o efeito fixo do i-ésimo nível do fator 1;

β_j é o efeito fixo do j-ésimo nível do fator 2; e

e_{ij} é o erro aleatório que tem o i-ésimo nível do fator 1 e j-ésimo nível do fator 2.

Representando o modelo como apresentado em (3.1), tem-se:

$$\mathbf{y} = X\beta + \varepsilon, \tag{3.8}$$

onde:

$\mathbf{y} = [y_{11}, ..., y_{IJ}]^{\mathrm{T}}$ é um vetor $(IJ \times 1)$ observações da variável resposta;

$X = [\mathbf{1}, \mathbf{x}_1, ..., \mathbf{x}_I, \mathbf{x}_{I+1}, ..., \mathbf{x}_{I+J}]$ é a matriz do modelo, $IJ \times (I + J + 1)$, representada por vetores tais que:

$$\mathbf{x}_{lm} = \begin{cases} 1, & \text{se } m\text{-ésimo nível do fator 1 está presente na } l\text{-ésima observação,} \\ 0, & \text{caso contrário,} \end{cases}$$

para m=1,...,I e l=1,...,IJ, e

$$\mathbf{x}_{lm} = \begin{cases} 1, & \text{se o nível } (m\text{-}I) \text{ do fator 2 está presentena } l\text{-ésima observação ,} \\ 0, & \text{caso contrário,} \end{cases}$$

para m=I+1,...,I+J e l=1,...,IJ;

β=$[\mu, \alpha_1, ..., \alpha_I, \beta_1, ..., \beta_J]^{\mathrm{T}}$ é o vetor de (1+I+J) parâmetros desconhecidos e fixos; e

$\varepsilon = [e_{11}, ..., e_{IJ}]^{\mathrm{T}}$ é um vetor residual aleatório.

Um ponto importante a observar é que somando as colunas da matriz do modelo $\mathbf{x}_1, ..., \mathbf{x}_I$ ou $\mathbf{x}_{I+1}, ..., \mathbf{x}_{I+J}$ obtém-se o vetor coluna unitário $\mathbf{1}$. Logo, existe apenas $1+(I-1)+(J-1)$ colunas de X que são linearmente independentes. Neste caso, $p = I+J$ e $r(X) = 1+(I-1)+(J-1) = p-1 < p+1$. Assim verifica-se que X não tem posto coluna completo implicando que pelo menos um dos parâmentros de β não é estimável (propriedade 3). Pela Proposição 3.1.1 tem-se que $s = (p+1) - r(X) = (p+1) - (p-1) = 2$. Logo, têm-se duas relações entre as colunas de X. Estas relações ocorrem nos correspondentes elementos do vetor λ de maneira que $\lambda^{\mathrm{T}}\beta$ seja estimável:

$$\lambda = [\textstyle\sum_{m=1}^{I} \lambda_m = \sum_{m=I+1}^{I+J} \lambda_m, \lambda_1, ..., \lambda_I, \lambda_{I+1}, ..., \lambda_{I+J}]^{\mathrm{T}}.$$

Sejam $I = 2$ e $J = 3$, então tem-se:

$$\mathbf{y} = [y_{11}, y_{12}, y_{13}, y_{21}, y_{22}, y_{23}]^{\mathrm{T}},$$

$$X = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

$$\beta = [\mu, \alpha_1, \alpha_2, \beta_1, \beta_2, \beta_3]^{\mathrm{T}} \quad \text{e} \quad \varepsilon = [e_{11}, e_{12}, e_{13}, e_{21}, e_{22}, e_{23}]^{\mathrm{T}}.$$

Observa-se que a soma das colunas $\mathbf{x}_1$ e $\mathbf{x}_2$, e a soma das colunas $\mathbf{x}_3$, $\mathbf{x}_4$ e $\mathbf{x}_5$ da matrix X resultam na coluna $\mathbf{x}_1 = \mathbf{1}$. Logo, $p = 5$, $r(X) = 1+(2-1)+(3-1) = 4 < p+1$ e $s = 5+1-4 = 2$.

Como consequência da Proposição 3.1.1 as combinações lineares do tipo $\lambda^{\mathrm{T}}\beta$ são estimáveis se os elementos de λ atendem a seguinte relação:

$$\lambda = [\textstyle\sum_{m=1}^{2} \lambda_m = \sum_{m=3}^{5} \lambda_m, \lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5]^{\mathrm{T}}.$$

Por exemplo,

$\lambda = [1,1,0,1,0,0]^{\mathrm{T}} \rightarrow \lambda^{\mathrm{T}}\beta = \mu + \alpha_1 + \beta_1$;
$\lambda = [0,1,-1,0,0,0]^{\mathrm{T}} \rightarrow \lambda^{\mathrm{T}}\beta = \alpha_1 - \alpha_2$; e
$\lambda = [0,0,0,0,1,-1]^{\mathrm{T}} \rightarrow \lambda^{\mathrm{T}}\beta = \beta_2 - \beta_3$,

seguem as relações acima e portanto são estimáveis, no entanto,

$$\begin{aligned}
\lambda = [1,0,0,0,0,0]^{\mathrm{T}} &\rightarrow \quad \lambda^{\mathrm{T}}\beta = \mu 1; \\
\lambda = [0,1,0,0,0,0]^{\mathrm{T}} &\rightarrow \quad \lambda^{\mathrm{T}}\beta = \alpha_1; \\
\lambda = [0,0,1,0,0,0]^{\mathrm{T}} &\rightarrow \quad \lambda^{\mathrm{T}}\beta = \alpha_2; \\
\lambda = [0,0,0,1,0,0]^{\mathrm{T}} &\rightarrow \quad \lambda^{\mathrm{T}}\beta = \beta_1; \\
\lambda = [0,0,0,0,1,0]^{\mathrm{T}} &\rightarrow \quad \lambda^{\mathrm{T}}\beta = \beta_2; \text{ e} \\
\lambda = [0,0,0,0,0,1]^{\mathrm{T}} &\rightarrow \quad \lambda^{\mathrm{T}}\beta = \beta_3,
\end{aligned}$$

não são estimáveis pois não obedecem às relações citadas.

A matrix X não possui posto coluna completo, logo, tem-se que a matriz $(X^{\mathrm{T}}X)$ não é inversível. Uma alternativa para ainda utilizar-se a expressão (3.6) é substituir a inversa $(X^{\mathrm{T}}X)^{-1}$ por uma inversa generalizada $(X^{\mathrm{T}}X)^{-}$ [Har98], mas isso implica que $\widehat{\beta}$ não tem solução única (os elementos de β continuam a ser não estimáveis). Existem várias maneiras de encontrar inversas generalizadas, uma delas é adicionar restrições aos parâmetros. Porém, os parâmetros são estimáveis somente sob as restrições escolhidas [NWKN96, Sea82].

Analisando modelos com esse tipo de estrutura de X, usualmente o principal objetivo não é estimar os parâmetros de β para fazer predições da variável resposta como nos modelos de regressão, mas, observar combinações lineares $\lambda^{\mathrm{T}}\beta$, tais que λ pertence ao espaço linha de X (atendem as relações de λ citadas), como por exemplo, diferenças entre efeitos de um fator que verificam como os níveis dos fatores afetam a variável resposta. Felizmente essas combinações lineares dos parâmetros são estimáveis, isto é, suas estimativas sempre apresentam o mesmo resultado independentemente da restrição escolhida. Este fato deve-se à propriedade de invariância de $\lambda^{\mathrm{T}}(X^{\mathrm{T}}X)^{-}X^{\mathrm{T}}$ para λ pertencente ao espaço linha de X [Har98]. Para exemplificar, seguem resultados de três inversas generalizadas de $X^{\mathrm{T}}X$ usadas para estimar as diferenças: $\alpha_1 - \alpha_2$ e $\beta_2 - \beta_3$. Para simplificar os resultados são definidas as notações:

$\bar{y}_{..} = \frac{\sum_{i=1}^{I}\sum_{j=1}^{J} y_{ij}}{IJ}$, como a média geral;

$\bar{y}_{i.} = \frac{\sum_{j=1}^{J} y_{ij}}{J}$, como a média do nível i do fator 1; e

$\bar{y}_{.j} = \frac{\sum_{i=1}^{I} y_{ij}}{I}$, como a média do nível j do fator 2.

1. Inversa generalisada baseada nas restrições: $\alpha_1 = 0$ e $\beta_1 = 0$

Sob estas restrições a matriz do modelo pode ser reparametrizada como:

$$X_{(1)} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix},$$

enquanto o vetor β passa a ser;

$$\beta^{(1)} = \begin{pmatrix} \mu \\ \alpha_2 \\ \beta_2 \\ \beta_3 \end{pmatrix}.$$

Logo, a inversa para ${X_{(1)}}^{\mathrm{T}} X_{(1)}$ é dada por:

$$X_{(1)}^{\mathrm{T}} X_{(1)}^{-} = \begin{pmatrix} 2/3 & -1/3 & -1/2 & -1/2 \\ -1/3 & 2/3 & 0 & 0 \\ -1/2 & 0 & 1 & 1/2 \\ -1/2 & 0 & 1/2 & 1 \end{pmatrix}.$$

Baseada nesta inversa, os estimadores de β dados pela Expressão (3.6) são:

$\hat{\mu} = \bar{Y}_{..} + \bar{Y}_{1.} + \bar{Y}_{.1}$,
$\hat{\alpha}_1 = 0$,
$\hat{\alpha}_2 = \bar{Y}_{2.} - \bar{Y}_{1.}$,
$\hat{\beta}_1 = 0$,
$\hat{\beta}_2 = \bar{Y}_{.2} - \bar{Y}_{.1}$, e
$\hat{\beta}_3 = \bar{Y}_{.3} - \bar{Y}_{.1}$.

Portanto,

$\hat{\alpha}_1 - \hat{\alpha}_2 = \bar{Y}_{1.} - \bar{Y}_{2.}$, e
$\hat{\beta}_2 - \hat{\beta}_3 = \bar{Y}_{.2} - \bar{Y}_{.3}$

2. Inversa generalisada baseada nas restrições: $\alpha_2 = 0$ e $\beta_3 = 0$

Sob estas restrições a matriz do modelo pode ser reparametrizada como:

$$X_{(2)} = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix},$$

enquanto o vetor β passa a ser;

$$\beta^{(2)} = \begin{pmatrix} \mu \\ \alpha_1 \\ \beta_1 \\ \beta_2 \end{pmatrix}.$$

Logo, a inversa para ${X_{(2)}}^{\mathrm{T}} X_{(2)}$ é dada por:

$$X_{(2)}^{\mathrm{T}} X_{(2)}^{-} = \begin{pmatrix} 2/3 & -1/3 & -1/2 & -1/2 \\ -1/3 & 2/3 & 0 & 0 \\ -1/2 & 0 & 1 & 1/2 \\ -1/2 & 0 & 1/2 & 1 \end{pmatrix}.$$

Baseada nesta inversa, os estimadores de β dados pela Expressão (3.6) são:

$$\hat{\mu} = \bar{Y}_{..} + \bar{Y}_{2.}\bar{Y}_{.3},$$
$$\hat{\alpha}_1 = \bar{Y}_{1.} - \bar{Y}_{2.},$$
$$\hat{\alpha}_2 = 0,$$
$$\hat{\beta}_1 = \bar{Y}_{.1} - \bar{Y}_{.3},$$
$$\hat{\beta}_2 = \bar{Y}_{.2} - \bar{Y}_{.1}, \text{ e}$$
$$\hat{\beta}_3 = 0.$$

Portanto,

$$\hat{\alpha}_1 - \hat{\alpha}_2 = \bar{Y}_{1.} - \bar{Y}_{2.}, \text{ e}$$
$$\hat{\beta}_2 - \hat{\beta}_3 = \bar{Y}_{.2} - \bar{Y}_{.3}$$

3. Inversa generalisada baseada nas restrições: $\alpha_1 + \alpha_2 = 0$ e $\beta_1 + \beta_2 + \beta_3 = 0$

Esta restrição é chamada sigma (restrição $\sum$). Ela é a restrição mais utilizada na literatura, pois permite interpretações intuitivas dos estimadores dos parâmetros. A restrição $\sum$ impõe que a soma dos parâmetros (referentes aos níveis de um fator) seja igual a zero. Considerando o modelo (3.7) ou (3.8), as restrições $\sum$ são tais que:

$$\sum_{i=1}^{I} \alpha_i = 0 \;\; e \;\; \sum_{j=1}^{J} \beta_j = 0. \tag{3.9}$$

No exemplo citado, sob as restrições $\sum$, tem-se que α_2 pode ser estimado por $-\hat{\alpha}_1$ e β_3 pode ser estimado por $-(\hat{\beta}_1 + \hat{\beta}_2)$. Logo, a matriz do modelo pode ser reparametrizada como:

$$X_{(3)} = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & 0 \\ 1 & -1 & 0 & 1 \\ 1 & -1 & -1 & -1 \end{pmatrix},$$

enquanto o vetor β passa a ser;

$$\beta^{(3)} = \begin{pmatrix} \mu \\ \alpha_1 \\ \beta_1 \\ \beta_2 \end{pmatrix}.$$

Logo, a inversa para ${X_{(3)}}^{\mathrm{T}} X_{(3)}$ é dada por:

$$X_{(3)}^{\mathrm{T}} X_{(3)}^{-} = \begin{pmatrix} 1/6 & 0 & 0 & 0 \\ 0 & 1/6 & 0 & 0 \\ 0 & 0 & 1/3 & -1/6 \\ 0 & 0 & -1/6 & 1/3 \end{pmatrix}.$$

Baseada nesta inversa, os estimadores de β dados pela Expressão (3.6) são:

$$\hat{\mu} = \bar{Y}_{..},$$
$$\hat{\alpha}_1 = \bar{Y}_{1.} - \bar{Y}_{..},$$
$$\hat{\alpha}_2 = \bar{Y}_{2.} - \bar{Y}_{..},$$
$$\hat{\beta}_1 = \bar{Y}_{.1} - \bar{Y}_{..},$$
$$\hat{\beta}_2 = \bar{Y}_{.2} - \bar{Y}_{..}, \text{ e}$$
$$\hat{\beta}_3 = \bar{Y}_{.3} - \bar{Y}_{..}.$$

Portanto,

$$\hat{\alpha}_1 - \hat{\alpha}_2 = \bar{Y}_{1.} - \bar{Y}_{2.}, \text{ e}$$
$$\hat{\beta}_2 - \hat{\beta}_3 = \bar{Y}_{.2} - \bar{Y}_{.3}.$$

Como é verificado, para as três inversas (restrições) escolhidas, as diferenças $\alpha_1 - \alpha_2$ e $\beta_2 - \beta_3$ são estimadas da mesma maneira, enquanto μ, os α's e os β's apresentam diferentes estimadores.

As restrições (3.9) permitem, através do método de mínimos quadrados, encontrar estimadores que têm as seguintes interpretações:

$\widehat{\mu} = \bar{y}_{..}$ é a média geral das observações;

$\widehat{\alpha}_i = \bar{y}_{i.} - \bar{y}_{..}$ é a diferença entre a média do nível i do fator 1 e a média geral;

$\widehat{\beta}_j = \bar{y}_{.j} - \bar{y}_{..}$ é a diferença entre a média do nível j do fator 2 e a média geral; e

$\widehat{e}_{ij} = y_{ij} - \bar{y}_{i.} - \bar{y}_{.j} + \bar{y}_{..}$ é a diferença entre o valor observado com o i-ésimo nível do fator 1 e o j-ésimo nível do fator 2, e o valor predito com o i-ésimo nível do fator 1 e o j-ésimo nível do fator 2.

Assim, baseado no modelo de classificação, a soma dos quadrados dos resíduos pode ser expressa por:

$$SQR = \sum_{i=1}^{I} \sum_{j=1}^{J} \widehat{e}_{ij}^2 = \sum_{i=1}^{I} \sum_{j=1}^{J} (y_{ij} - \bar{y}_{i.} - \bar{y}_{.j} + \bar{y}_{..})^2. \tag{3.10}$$

O método Plaid apresentado no capítulo anterior formula um modelo semelhante ao de classificação com o objetivo de identificar subconjuntos de genes e condições que minimizem a soma dos quadrados dos resíduos do modelo Plaid.

Nos modelos (3.7) e (3.8) observa-se ainda que, para cada combinação do nível i do fator 1 com o nível j do fator 2, apenas uma observação é obtida, fazendo necessária a suposição de aditividade do modelo para obtenção dos estimadores dos resíduos. Caso tenham-se mais de uma observação por combinação dos níveis dos fatores, essa suposição pode ser relaxada e o resíduo passa a ser a diferença entre o valor observado numa dada combinação e a média das observações desta combinação.

Em algumas situações, ao modelo de classificação são incorporadas variáveis explicativas contínuas, chamadas covariáveis. A seguir é apresentado tal modelo, chamado de modelo de análise de covariância.

3.1.2.1 Modelo de Análise de Covariância

O modelo de análise de covariância incorpora características dos modelos de classificação e de regressão. Na maioria das vezes, os objetivos de análises de covariância coincidem com os de modelos de classificação. No entanto, o uso de covariáves (aspectos do modelo de regressão) permite, por exemplo, a comparação de grupos, controlando o efeito de alguma variável secundária relevante. Através desse modelo, comparações são realizadas isolando o efeito da covariável (variável secundária). Além dessa vantagem, o uso do modelo de análise de covariância pode acarretar em melhor ajuste do modelo linear. Tradicionalmente, a análise de covariância tem sido usada como uma ferramenta em análises de experimentos.

O modelo de análise de covariância é expresso incorporando covariáveis ao modelo apresentado em (3.7). Suponha que apenas uma covariável, z_{ij}, seja de interesse, então o modelo de covariância é formulado como:

$$y_{ij} = \mu + \alpha_i + \beta_j + \omega\, z_{ij} + \varepsilon_{ij}, \tag{3.11}$$

onde:

y_{ij} é resposta observada pelo i-ésimo nível do fator 1 e j-ésimo nível do fator 2;

μ é a média geral;

α_i é o efeito fixo do i-ésimo nível do fator 1;

β_j é o efeito fixo do j-ésimo nível do fator 2;

z_{ij} é a covariável observada pelo i-ésimo nível do fator 1 e j-ésimo nível do fator 2;

ω é o coeficiente de regressão relativo à covariável; e

e_{ij} é o resíduo aleatório que tem o i-ésimo nível do fator 1 e j-ésimo nível do fator 2.

Reescrevendo a expressão (3.11) em forma matricial:

$$\mathbf{y} = X\beta + \mathbf{z}\omega + \varepsilon, \tag{3.12}$$

onde:

$\mathbf{y} = [y_{11}, ..., y_{IJ}]^{\mathrm{T}}$ é um vetor $(IJ \times 1)$ observações da variável resposta;

$X = [\mathbf{1}, \mathbf{x}_1, ..., \mathbf{x}_I, \mathbf{x}_{I+1}, ..., \mathbf{x}_{I+J}]$ é a matriz do modelo, $IJ \times (I+J+1)$, representada por vetores tais que:

$$\mathbf{x}_{lm} = \begin{cases} 1, & \text{se o } m\text{-ésimo nível do fator 1 está presente na } l\text{-ésima observação,} \\ 0, & \text{caso contrário,} \end{cases}$$

para m=1,...,I e $l = 1, ..., IJ$, e

$$\mathbf{x}_{lm} = \begin{cases} 1, & \text{se o nível } (m\text{-}I) \text{ do fator 2 está presente na } l\text{-ésima observação,} \\ 0, & \text{caso contrário,} \end{cases}$$

para m=I+1,...,I+J e $l = 1,...,IJ$;

$\beta = [\mu, \alpha_1, ..., \alpha_I, \beta_1, ..., \beta_J]^{\mathrm{T}}$ é o vetor de (1+I+J) parâmetros desconhecidos e fixos;

$\mathbf{z}$ é um vetor de observações da covariável;

ω é o coeficiente de regressão relativo a covariável; e

$\varepsilon = [e_{11}, ..., e_{IJ}]^{\mathrm{T}}$ é um vetor de erros aleatórios.

Seja $U = [X, \mathbf{z}]$ e $\theta = [\beta^{\mathrm{T}}, \omega]^{\mathrm{T}}$ pode-se reescrever 3.12 como:

$$\mathbf{y} = U\theta + \varepsilon. \tag{3.13}$$

Portanto as equações normais para (3.13) podem ser expressas por:

$$U^{\mathrm{T}}U\widehat{\theta} = U^{\mathrm{T}}\mathbf{y}, \tag{3.14}$$

ou

$$\begin{pmatrix} X^{\mathrm{T}} \\ \mathbf{z}^{\mathrm{T}} \end{pmatrix} (X, \mathbf{z}) \begin{pmatrix} \beta \\ \omega \end{pmatrix} = \begin{pmatrix} X^{\mathrm{T}} \\ \mathbf{z}^{\mathrm{T}} \end{pmatrix} \mathbf{y}, \tag{3.15}$$

ou

$$\begin{pmatrix} X^{\mathrm{T}}X & X^{\mathrm{T}}\mathbf{z} \\ \mathbf{z}^{\mathrm{T}}X & \mathbf{z}^{\mathrm{T}}\mathbf{z} \end{pmatrix} \begin{pmatrix} \beta \\ \omega \end{pmatrix} = \begin{pmatrix} X^{\mathrm{T}}y \\ \mathbf{z}^{\mathrm{T}}y \end{pmatrix}, \tag{3.16}$$

as quais podem ainda ser particionadas como:

$$X^{\mathrm{T}}X\widehat{\beta} + \mathbf{X}^{\mathrm{T}}\mathbf{z}\widehat{\omega} = \mathbf{X}^{\mathrm{T}}\mathbf{y} \tag{3.17}$$

e

$$\mathbf{z}^{\mathrm{T}}X\widehat{\beta} + \mathbf{z}^{\mathrm{T}}\mathbf{z}\widehat{\omega} = \mathbf{z}^{\mathrm{T}}\mathbf{y}, \tag{3.18}$$

Uma vez que a matriz X tem a mesma estrutura da matriz do modelo de classificação, $(X^{\mathrm{T}}X)$ não tem posto completo e consequentemente não é inversível. Portanto, sob as restrições (3.9), as Equações normais (3.17) e (3.18) fornecem os seguintes estimadores:

$$\widehat{\beta} = (\mathbf{z}^{\mathrm{T}}\mathbf{z})^{-1}\mathbf{z}^{\mathrm{T}}\mathbf{y} - (\mathbf{z}^{\mathrm{T}}\mathbf{z})^{-1}\mathbf{z}^{\mathrm{T}}X\widehat{\omega}, \tag{3.19}$$

$$\widehat{\omega} = [\mathbf{z}^{\mathrm{T}}(I-P)\mathbf{z}]^{-1}X^{\mathrm{T}}(I-P)\mathbf{y}, \tag{3.20}$$

onde:

$P = X(X^{\mathrm{T}}X)^{-}X^{\mathrm{T}}$ é a matriz de projeção sobre o espaço coluna de X; e

$(X^{\mathrm{T}}X)^{-}$ é uma inversa generalizada de $(X^{\mathrm{T}}X)$.

Simirlamente ao modelo de classificação logística pode-se obter $(X^{\mathrm{T}}X)^{-}$ através das restrições $\sum$.

Definições gerais dos modelos de classificação e de covariância, como por exemplo propriedades estatísticas, soma de quadrados, testes de hipóteses e diagnósticos são facilmente encontradas na literatura [NWKN96, Ren00].

3.2 Modelos Não Lineares

Na literatura, a identificação de genes similares em dados de microarranjo tem sido bastante abordada, mas quando os dados são filogenéticos ou de outra natureza, como dicotômica ou quali-tativa, pouco tem sido trabalhado [UW08]. Frequentemente, dados binários são usados como um suporte a mais para os dados de expressão em vez de serem as principais informações. Os modelos apresentados na seção anterior assumiam que a variável resposta era uma variável contínua. Quando a variável resposta é classificatória, estes modelos não apresentam bons ajustes, no sentido que não é garantido aos valores preditos que eles tenham seus resultados dentro do intervalo esperado. Por exemplo, modelando uma variável resposta que assume valores 0 ou 1, isto é, modelando uma variável binária usando modelos (3.1), não se tem garantias de que as observações preditas assumirão valores zeros ou uns, ou mesmo dentro do intervalo (0,1). Modelos que ajustam variáveis respostas classificatórias são modelos não lineares. O método proposto nesta tese é aplicado a dados de natureza binária e é baseado em modelos não lineares. Simirlarmente aos modelos da Seção 3.1, dois casos para ilustração são apresentados a seguir.

3.2.1 Caso 1: Modelos de Regressão Logística

O modelo de regressão logística é aplicado a variáveis respostas binárias e a variáveis explicativas contínuas. Estes modelos ajustam os dados utilizando a função (transformação) logit dada por:

$$\mathrm{logit}(\pi) = \log[\frac{\pi}{(1-\pi)}], \tag{3.21}$$

onde $\pi = P(y = 1)$ é a probabilidade da variável resposta assumir o valor 1, chamada de probabilidade de sucesso.

Representando a função logit como sendo uma função de p variáveis explicativas tem-se a equação:

$$\mathrm{logit}(\pi(\boldsymbol{x},\beta)) = \log[\frac{\pi(\mathbf{x},\beta)}{(1-\pi(\mathbf{x},\beta))}] = \mathbf{x}^{\mathrm{T}}\beta, \tag{3.22}$$

onde:

$\mathbf{x} = [1, x_1, ..., x_p]^{\mathrm{T}}$ é um vetor cujos elementos são valores das variáveis explicativas contínuas;

$\beta = [\beta_0, \beta_1, ..., \beta_p]^{\mathrm{T}}$ é um vetor de parâmetros desconhecidos de dimensão (p+1); e

$\pi(\mathbf{x},\beta)$ é a probabilidade da variável resposta assumir o valor um, dada $\mathbf{x}^{\mathrm{T}}\beta$.

Aplicando a função exponencial em ambos os lados da expressão (3.22) obtém-se a seguinte relação:

$$\frac{\pi(\mathbf{x},\beta)}{(1-\pi(\mathbf{x},\beta))} = \exp(\mathbf{x}^{\mathrm{T}}\beta) \tag{3.23}$$

Logo, pode-se expressar o modelo de regressão logística como:

$$\pi(\mathbf{x},\beta) = \frac{\exp(\mathbf{x}^{\mathrm{T}}\beta)}{1+\exp(\mathbf{x}^{\mathrm{T}}\beta)} \tag{3.24}$$

Este modelo pode ainda ser expresso como um modelo de regressão como segue:

$$y = \pi(\mathbf{x},\beta) + e, \tag{3.25}$$

onde:

e é o erro aleatório com média zero e variância $\pi(\mathbf{x},\beta)[1-\pi(\mathbf{x},\beta)]$; e

y é a variável resposta binária, a qual assume valor zero ou um.

Consequentemente, a esperança da variável resposta y é a probabilidade de sucesso $\pi(\mathbf{x},\beta)$. Portanto o erro segue a distribuição binomial com média zero e variância $\pi(\mathbf{x},\beta)(1-\pi(\mathbf{x},\beta))$. O modelo de regressão logística $\pi(\mathbf{x},\beta)$ é uma função não linear de $\mathbf{x}$ que garante que o preditor de $\pi(\mathbf{x},\beta)$ assuma valores no intervalo de 0 até 1, como desejado.

Em modelos lineares um método frequentemente usado para estimar os parâmetros é o de mínimos quadrados, o qual calcula os estimadores através da minimização da soma dos quadrados dos resíduos. Tal método, quando aplicado à regressão linear, sob usuais suposições de Gauss-Markov, apresenta estimadores com propriedades desejáveis, tais como a apresentada na Seção 3.1.1. Infelizmente este método não apresenta as mesmas propriedades estatísticas quando aplicado à regressão logística.

Quando a suposição de normalidade dos erros é garantida em modelos lineares, o método de estimação chamado de máxima verossimilhança [Ren00] também é empregado a estes modelos. Este método é frequentemente utilizado para estimar os parâmetros em regressão logística [Chr97, HL00]. Para poder aplicar este método, primeiro é necessário construir um função chamada de função de verossimilhança, a qual é expressa por uma função da probabilidade de sucesso como uma função dos parâmetros desconhecidos.

Seja $\mathbf{x}^l = (x_{l0}, x_{l1}, ..., x_{lp})$ um vetor de p variáveis explicativas e do termo constante $x_{l0} = 1$ para a l-ésima variável resposta. Suponha que uma amostra de N observações independentes da variável resposta seja disponível, $y_1, ..., y_N$, então a expressão (3.24) para a l-ésima observação é dada por:

$$\pi(\mathbf{x}^l,\beta) = \frac{\exp(\sum_{m=0}^{p}\beta_m x_{lm})}{1+\exp(\sum_{m=0}^{p}\beta_m x_{lm})}. \tag{3.26}$$

Logo a contribuição da l-ésima observação para a função de verossimilhança é expressa como:

$$\pi(\mathbf{x}^l,\beta)^{y_l}[1-\pi(\mathbf{x}^l,\beta)]^{1-y_l}. \tag{3.27}$$

Uma vez que as observações são assumidas independentes, a função de verossimilhança é obtida como o produto das contribuições de cada observação:

$$l(\beta) = \prod_{l=1}^{N} \pi(\mathbf{x}^l, \beta)^{y_l} [1 - \pi(\mathbf{x}^l, \beta)]^{1-y_l}. \tag{3.28}$$

O método de máxima verossimilhança calcula os estimadores dos $\beta' s$ de modo que eles maximizem $l(\beta)$. Por facilidade algébrica, usualmente é maximizado o logaritmo da verossimilhança, dado por:

$$L(\beta) = \ln[l(\beta)] = \sum_{l=1}^{N} y_l \ln[\pi(x^l, \beta)] + (1 - y_l) \ln[1 - \pi(x^l, \beta)]. \tag{3.29}$$

Para achar os estimadores de máxima verossimilhança para β, $\widehat{\beta}$, $L(\beta)$ é diferenciada com respeito a cada parâmetro de β e então suas expressões são igualadas a zero. Isto é, p+1 equações não lineares, chamadas de equações normais, são construídas como segue:

$$\frac{\partial L(\beta)}{\partial \beta_m} = \sum_{l=1}^{N} y_l x_{lm} - \sum_{l=1}^{N} \pi(x^l, \beta) x_{lm} = 0, \tag{3.30}$$

para m=0,..., p. Considerando a equação (3.30) em forma matricial, temos:

$$\frac{\partial L(\beta)}{\partial \beta} = X^{\mathrm{T}}(\mathbf{y} - \pi) = \mathbf{0}, \tag{3.31}$$

onde:

$\mathbf{y}^{\mathrm{T}} = [y_1, ..., y_N]$ é um vetor de N observações independentes da variável resposta;

$X = [\mathbf{x}_0, \mathbf{x}_1, ..., \mathbf{x}_p]$ é a matriz do modelo $N \times (p+1)$; e

$\pi = [\pi(\mathbf{x}^1, \beta), ..., \pi(\mathbf{x}^N, \beta)]$ é o vetor de N probabilidades, onde $\mathbf{x}^l$ é um vetor dos elementos da l-ésima linha da matrix X.

Observa-se que as equações normais (3.30) e (3.31) são não lineares em β, logo são necessários métodos especiais para solucioná-las. O método de Newton-Raphson [Ren00] é um método iterativo para solucionar equações não lineares através da maximização de uma função $f(\beta)$. O algoritmo começa com uma atribuição inicial para os parâmetros, β^0, e então define uma sequência de $\beta' s$, β^1, β^2, ..., que converge para um $\widehat{\beta}$ que maximize $f(\beta) = 0$. A sequência é definida recursivamente, onde o valor de β^{t+1} é obtido conhecendo o valor de β^t. Sabe-se via Teorema de Taylor[PMJ65] que se β^t é aproximadamente β^{t+1} e $\eta^t = \beta^{t+1} - \beta^t$, então

$$f(\beta^{t+1}) \cong f(\beta^t) + \partial f(\beta^t) \eta^t. \tag{3.32}$$

Newton-Raphson iguala (3.32) a zero, de forma que

$$\eta^t = -(\partial f(\beta^t))^{-1} f(\beta^t), \tag{3.33}$$

então

$$\beta^{t+1} = \beta^t + \eta^t, \tag{3.34}$$

para η^t dado em (3.33).

Seja a função de β, $f(\beta)$, definida como as equações normais:

$$f(\beta) = \frac{\partial L(\beta)}{\partial \beta} = X^{\mathrm{T}}(\mathbf{y} - \pi). \tag{3.35}$$

Seja a segunda derivada de $f(\beta)$, a matriz Hessiana:

$$\partial f(\beta) = \frac{\partial^2 L(\beta)}{\partial \beta \partial \beta^{\mathrm{T}}} = -\sum_{l=1}^{N} x^l x^{l'} \pi(x^l, \beta)(1 - \pi(x^l, \beta)), \tag{3.36}$$

representada em forma matricial como:

$$\partial f(\beta) = \frac{\partial^2 L(\beta)}{\partial \beta \partial \beta^{\mathrm{T}}} = -X^{\mathrm{T}} W X, \tag{3.37}$$

onde W é uma matriz diagonal, ($N \times N$), com o l-ésimo elemento igual a $\pi(x^l, \beta)(1 - \pi(x^l, \beta))$.

Substituído as expressões (3.35) e (3.37) naquelas em (3.33) e (3.34), o método de Newton-Raphson fornece as seguintes expressões:

$$\eta^t = (X^{\mathrm{T}} W^t X)^{-1} X^{\mathrm{T}}(\mathbf{y} - \pi^t), \tag{3.38}$$

e

$$\begin{aligned} \beta^{t+1} &= \beta^t + (X^{\mathrm{T}} W^t X)^{-1} X^{\mathrm{T}}(\mathbf{y} - \pi^t) \\ &= (X^{\mathrm{T}} W^t X)^{-1} X^{\mathrm{T}} W^t (X\beta^t + W^{t\,-1}(\mathbf{y} - \pi^t)) \\ &= (X^{\mathrm{T}} W^t X)^{-1} X^{\mathrm{T}} W^t \upsilon^t, \end{aligned} \tag{3.39}$$

onde:

$$\upsilon^t = X^{\mathrm{T}} \beta^t + W^{t\,-1}(\mathbf{y} - \pi^t).$$

Baseados nestas expressões são apresentados dois algoritmo para o método de Newton-Raphson para encontrar os estimadores dos parâmetros do modelo de regressão logística.

Algoritmo 3.1

1. Faça $\beta = \mathbf{0}$.
2. Calcule os elementos de π através de $\pi(\mathbf{x}^l, \beta) = \frac{\exp(\mathbf{x}^{l\,\mathrm{T}}\beta)}{1+\exp(\mathbf{x}^{l\,\mathrm{T}}\beta)}$, $l = 1, ..., N$.
3. Calcule os elementos da diagonal da matriz W através de $\pi(\mathbf{x}^l, \beta)(1 - \pi(\mathbf{x}^l, \beta))$, $l = 1, ..., N$.

4. Calcule υ como $X^{\mathrm{T}}\beta + W^{-1}(\mathbf{y} - \pi)$.

5. Calcule o novo β pela expressão $(X^{\mathrm{T}}WX)^{-1}X^{\mathrm{T}}W\upsilon$.

6. Caso o critério de parada seja encontrado, pare. Caso contrário refaça os passos de 2 a 5.

Uma vez que W é uma matriz diagonal, operações com ela podem ser ineficientes. Para corrigir tais problemas, este algoritmo pode ser ligeiramente alterado da seguinte forma:

Algoritmo 3.2

1. Faça $\beta = \mathbf{0}$.

2. Calcule os elementos de π dados por $\pi(\mathbf{x}^l, \beta) = \frac{\exp({\mathbf{x}^l}^{\mathrm{T}}\beta)}{1+\exp({\mathbf{x}^l}^{\mathrm{T}}\beta)}$, $l = 1, ..., N$.

3. Calcule os elementos da nova matriz

   $$\widetilde{X} = \begin{pmatrix} \widetilde{\mathbf{x}}^1 \\ \widetilde{\mathbf{x}}^2 \\ \cdots \\ \widetilde{\mathbf{x}}^N \end{pmatrix},$$

   onde $\widetilde{\mathbf{x}}^l = \pi(\mathbf{x}^l, \beta)(1 - \pi(\mathbf{x}^l, \beta)){\mathbf{x}^l}^{\mathrm{T}}, l = 1, ..., N$.

4. Calcule o novo β pela expressão $(X^{\mathrm{T}}\widetilde{X})^{-1}X^{\mathrm{T}}(\mathbf{y} - \pi)$.

5. Caso o critério de parada seja encontrado, pare. Caso contrário, refaça os passos de 2 a 4.

Os Algoritmos 3.1 e 3.2 são frequentemente utilizados por usuários de regressão logística, porém existe uma variedade de novas abordagens para otimizar os resultados. Quando as variáveis explicativas são classificatórias, o número de parâmetros a estimar é igual à soma do total de níveis de cada variável classificatória. Portanto, se o total de níveis de cada variável aumenta, as máquinas usadas para implementar os algoritmos para estimar estes parâmetros necessitam de mais memórias.

Na seção a seguir tem-se o caso de modelos onde tanto a variável resposta quanto as variáveis explicativas são classificatórias.

3.2.2 Caso 2: Modelo de Classificação Logística

A análise de classificação logística é similar à de regressão logística, onde as variáveis explicativas neste caso são classificatórias. Nesta tese situações com apenas duas variáveis explicativas são exemplificadas, por isso, esta seção assume duas variáveis explicativas classificatórias.

Seja $\pi(\mu, \alpha_i, \beta_j)$ a probabilidade que a variável resposta (com o i-ésimo nível do fator 1 e o j-ésimo nível do fator 2) assuma o valor 1. Então a expressão (3.22) pode ser reescrita como:

$$\text{logit}(\pi(\mu,\alpha_i,\beta_j)) = \log[\frac{\pi(\mu,\alpha_i,\beta_j)}{(1-\pi(\mu,\alpha_i,\beta_j))}] = \mu+\alpha_i+\beta_j. \tag{3.40}$$

onde:

μ é a média geral;

α_i é o efeito fixo do i-ésimo nível do fator 1, $i = 1,...,I$; e

β_j é o efeito fixo do j-ésimo nível do fator 2, $j = 1,...,J$.

Portando, similarmente ao modelo de regressão logística, o modelo de classificação logística pode ser expresso como:

$$\pi(\mu,\alpha_i,\beta_j) = \frac{\exp(\mu+\alpha_i+\beta_j)}{1+\exp(\mu+\alpha_i+\beta_j)}. \tag{3.41}$$

Seja β=$[\mu,\alpha_1,...,\alpha_I,\beta_1,...,\beta_J]^{\mathrm{T}}$ o vetor de $(1+I+J)$ parâmetros do modelo (3.41). O método de estimação para β segue aquele apresentado na seção de modelos de regressão logística. Portando, $f(\beta)$ é definida como:

$$f(\beta) = \frac{\partial L(\beta)}{\partial \beta} = X^{\mathrm{T}}(\mathbf{y}-\pi) = \mathbf{0}, \tag{3.42}$$

onde:

$\mathbf{y} = [y_{11},...,y_{IJ}]^{\mathrm{T}}$ é um vetor $(IJ\times 1)$ observações da variável resposta;

$\pi = [\pi(\mu,\alpha_1,\beta_1),...,\pi(\mu,\alpha_I,\beta_J)]^{\mathrm{T}}$ é o vetor de IJ probabilidades; e

$X = [\mathbf{1},\mathbf{x}_1,...,\mathbf{x}_I,\mathbf{x}_{I+1},...,\mathbf{x}_{I+J}]$ é a matriz do modelo representada por vetores tais que:

$$\mathbf{x}_{lm} = \begin{cases} 1, & \text{se o } m\text{-ésimo nível do fator 1 está presente na } l\text{-ésima observação,} \\ 0, & \text{caso contrário,} \end{cases}$$

para m=1,..., I, l=1,..., IJ, e

$$\mathbf{x}_{lm} = \begin{cases} 1, & \text{se o nível } (m\text{-}I) \text{ do fator 2 está presente na } l\text{-ésima observação,} \\ 0, & \text{caso contrário,} \end{cases}$$

para m=I+1,...,I+J, l=1,..., IJ.

Neste contexto, a matriz X do modelo possui colunas que não são linearmente independentes e consequentemente não tem posto completo. Então, similarmente aos modelos de classificação, as restrições (3.9) também são incorporadas ao modelo. Portanto, os estimadores de $\mu,\alpha_1,...,\alpha_{I-1},\beta_1,...,\beta_{J-2}$ e β_{J-1} são calculados através dos Algoritmos 3.1 e 3.2 substituindo a matriz X do modelo pela matriz reparametrizada (baseada nas restrições (3.9)), e os estimadores de α_I e β_J podem ser obtidos por:

$$\widehat{\alpha}_I = \sum_{i=1}^{I-1}\alpha_i \;\; e \;\; \widehat{\beta}_J = \sum_{j=1}^{J-1}\beta_j. \tag{3.43}$$

3.2.2.1 Modelo de Classificação Logística com uma Covariável

Similarmente ao modelo de análise de covariância apresentado na Seção 3.1.2, o modelo de classificação logística com uma covariável incorpora uma covariável ao modelo de classificação logística, isto é, a expressão (3.40) pode ser escrita com:

$$\text{logit}(\pi(\mu,\alpha_i,\beta_j,\omega,z_{ij})) = \log[\frac{\pi(\mu,\alpha_i,\beta_j,\omega,z_{ij})}{(1-\pi(\mu,\alpha_i,\beta_j,\omega,z_{ij})}] = \mu+\alpha_i+\beta_j+\omega\, z_{ij}, \tag{3.44}$$

onde:

μ é a média geral;

α_i é o efeito fixo do i-ésimo nível do fator 1, $i = 1,...,I$;

β_j é o efeito fixo do j-ésimo nível do fator 2, $j = 1,...,J$;

z_{ij} é a covariável observada pelo i-ésimo nível do fator 1 e j-ésimo nível do fator 2; e

ω é o coeficiente de regressão relativo à covariável.

Logo, o modelo de classificação logística com uma covariável é dado por:

$$\pi(\mu,\alpha_i,\beta_j,\omega,z_{ij}) = \frac{\exp(\mu+\alpha_i+\beta_j+\omega\, z_{ij})}{1+\exp(\mu+\alpha_i+\beta_j+\omega\, z_{ij})} \tag{3.45}$$

Portanto as equações normais (3.42) passam a ser escritas como:

$$\frac{\partial L(\theta)}{\partial\theta} = U^{\mathrm{T}}(\mathbf{y}-\pi) = \mathbf{0}, \tag{3.46}$$

onde:

$\mathbf{y} = [y_{11},...,y_{IJ}]^{\mathrm{T}}$ é um vetor $(IJ\times 1)$ observações da variável resposta;

$\theta = [\beta^{\mathrm{T}},\omega]^{\mathrm{T}}$, com β=$[\mu,\alpha_1,...,\alpha_I,\beta_1,...,\beta_J]^{\mathrm{T}}$;

$\pi = [\pi(\mu,\alpha_1,\beta_1,\omega,z_{11}),...,\pi(\mu,\alpha_I,\beta_J,\omega,z_{IJ})]^{\mathrm{T}}$ é o vetor de IJ probabilidades;

$U = [X,\mathbf{z}] = [\mathbf{1},\mathbf{x}_1,...,\mathbf{x}_I,\mathbf{x}_{I+1},...,\mathbf{x}_{I+J},\mathbf{z}]$ é a matriz do modelo tal que:

$$\mathbf{x}_{lm} = \begin{cases} 1, & \text{se o } m\text{-ésimo nível do fator 1 está presente na } l\text{-ésima observação,} \\ 0, & \text{caso contrário,} \end{cases}$$

para m=1,...,I, l=1,..., IJ, e

$$\mathbf{x}_{lm} = \begin{cases} 1, & \text{se o nível } (m\text{-}I) \text{ do fator 2 está presente na } l\text{-ésima observação,} \\ 0, & \text{caso contrário,} \end{cases}$$

para m=I+1,...,I+J, l=1,..., IJ; e

$\mathbf{z}$ é um vetor de observações da covariável.

Desde que a matriz U do modelo não é de posto completo, ela é reparametrizada considerando as restrições (3.9). Logo a expressão (3.46) pode ser reescrita como:

$$\frac{\partial L(\breve{\theta})}{\partial \breve{\theta}} = \breve{U}^{\mathrm{T}}(\mathbf{y} - \pi) = \mathbf{0}, \tag{3.47}$$

onde:

$\breve{\theta} = [\mu, \alpha_1, ..., \alpha_{I-1}, \beta_1, ..., \beta_{J-1}, \omega]^{\mathrm{T}}$; e

$\breve{U} = [\breve{X}, \mathbf{z}] = [\mathbf{1}, \mathbf{x}_1, ..., \mathbf{x}_{I-1}, \mathbf{x}_I, ..., \mathbf{x}_{I+J-1}, \mathbf{z}]$ é a matriz do modelo reparametrizado tal que:

$$\mathbf{x}_{lm} = \begin{cases} 1, & \text{se o } m\text{-ésimo nível do fator 1 está presente na } l\text{-ésima observação,} \\ -1, & \text{se o } I\text{-ésimo nível do fator 1 está na } l\text{-ésima observação,} \\ 0, & \text{caso contrário,} \end{cases}$$

para m=1,...,I-1,l=1,..., IJ, e

$$\mathbf{x}_{lm} = \begin{cases} 1, & \text{se o nível } (m\text{-}I{+}1) \text{ do fator 2 está presente na } l\text{-ésima observação,} \\ -1, & \text{se o nível } (J) \text{ do fator 2 está presente na } l\text{-ésima observação,} \\ 0, & \text{caso contrário,} \end{cases}$$

para m=I,...,I+J-1 e l=1,..., IJ.

Seguindo os passos do método estimação da Seção (3.2.1), podem-se obter as seguintes expressões:

$$f(\breve{\theta}) = \frac{\partial L(\breve{\theta})}{\partial \breve{\theta}} = \breve{U}^{\mathrm{T}}(\mathbf{y} - \pi) = \mathbf{0}, \tag{3.48}$$

$$\partial f(\breve{\theta}) = \frac{\partial^2 L(\breve{\theta})}{\partial \breve{\theta} \partial \breve{\theta}^{\mathrm{T}}} = -\breve{U}^{\mathrm{T}} W \breve{U}, \tag{3.49}$$

onde W é uma matriz diagonal, $(IJ \times IJ)$, com o m-ésimo elemento igual a $\pi(\breve{u}^l, \breve{\theta})(1 - \pi(\breve{u}^l, \breve{\theta}))$, e $\breve{u}^l$ correspondendo à l-ésima linha da matriz $\breve{U}$.

Portanto, as expressões para implementar o algoritmo de Newton-Raphson são calculadas como segue.

$$\eta^t = (\breve{U}^{\mathrm{T}} W^t \breve{U})^{-1} \breve{U}^{\mathrm{T}} (\mathbf{y} - \pi^t), \tag{3.50}$$

e

$$\begin{aligned} \breve{\theta}^{t+1} &= \breve{\theta}^t + (\breve{U}^{\mathrm{T}} W^t \breve{U})^{-1} \breve{U}^{\mathrm{T}} (\mathbf{y} - \pi^t) \\ &= (\breve{U}^{\mathrm{T}} W^t \breve{U})^{-1} \breve{U}^{\mathrm{T}} W^t (\breve{U} \breve{\theta}^t + \breve{U}^{t-1} (\mathbf{y} - \pi^t)) \\ &= (\breve{U}^{\mathrm{T}} W^t \breve{U})^{-1} \breve{U}^{\mathrm{T}} W^t \upsilon^t, \end{aligned} \tag{3.51}$$

onde:

$$v^t = \breve{U}^{\mathrm{T}} \breve{\theta}^t + W^{t\,-1}(\mathbf{y} - \pi^t).$$

Utilizando o Algoritmo (3.1) ou (3.2), os parâmetros de $\breve{\theta}$ são estimados substituindo a matriz X do modelo por $\breve{U}$. Enquanto os estimadores de α_I e β_J podem ser calculados como aqueles em (3.43).

3.3 Conclusões

Neste capítulo foram apresentados alguns modelos que serviram de suporte aos modelos aplicados nesta tese. O método Plaid [TBKH05] apresentado na comparação e combinação com o método proposto Lbic utiliza um modelo semelhante ao de classificação com dois fatores. O método Lbic utiliza um modelo semelhante ao de classificação logística com dois fatores e uma covariável.

CAPÍTULO 4

Método Lbic

Neste capítulo é apresentado o método de bi-clustering originalmente proposto nesta tese, o Lbic. Tal método classifica genes de conjuntos de dados genômicos cujas respostas são binárias. Uma vez que os dados analisados são binários, como foi citado no Capítulo 2, métodos do tipo Plaid, que são aplicados à dados contínuos e utilizam modelos semelhantes aos de análise de classificação, não apresentam resultados rozoáveis para esses tipos de dados. Por esta razão e ainda observando a idéia do Plaid, surgiu a proposta de utilizar modelos semelhantes aos de classificação logística, os quais são adequados a dados binários. Por causa da escolha do modelo, foi dado ao método o nome Lbic, L de logística e bic de bi-cluster. A idéia é identificar bi-clusters cujas respostas observadas dos genes apresentem probabilidades similares sob as mesmas condições. O método identifica bi-clusters que atendem aos aspectos apresentados na Figura 4.1.

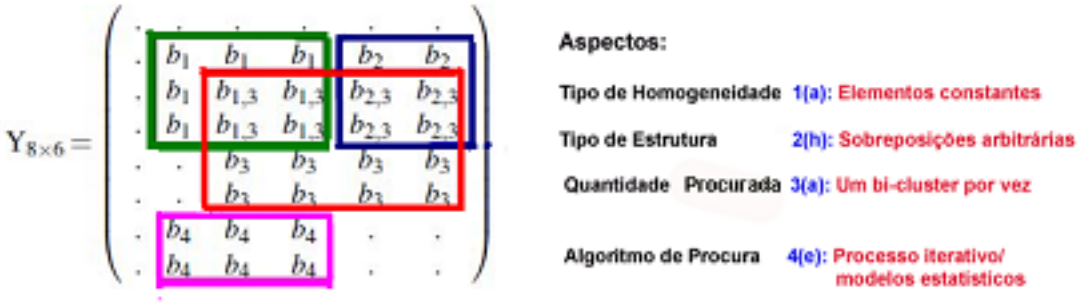

Figura 4.1 Aspectos de bi-clusters a serem identificados

4.1 Modelo Lbic

Considere a situação em que o conjunto de dados genômicos é descrito por uma matriz $Y = \{y_{ij}\}$, onde as linhas representam os genes, i =1, ..., n; as colunas, as condições, j =1, ..., c; e os y_{ij} correspondem as respostas observadas do i-ésimo gene sob a j-ésima condição, tais que

$$y_{ij} = \begin{cases} 1, & \text{se o gene } i \text{ tem sucesso na condição } j, \\ 0, & \text{caso contrário,} \end{cases}$$

onde um gene é dito ter sucesso se ele apresenta a característica estudada em uma particular condição. O modelo Lbic define a probabilidade de sucesso do gene i sob a condição j para cada bi-cluster por $\pi_{ij} = P(y_{ij} = 1)$. Tal modelo segue a estrutura de classificação logística, onde cada π_{ij} é expresso como uma função dos efeitos de bi-clusters aditivos, similarmente ao descrito pelo método Plaid no Capítulo 2.

Seja θ_{ijk} o efeito do k-ésimo bi-cluster para o gene i ($i = 1, ..., n$) e condição j ($j = 1, ..., c$), onde $k = 1, ..., b$. Uma forma de expressar θ_{ijk} em função de fatores importantes como o efeito do bi-cluster é dada por:

$$\theta_{ijk} = \mu_k + \alpha_{ik} + \beta_{jk}, \tag{4.1}$$

onde:

μ_k é a média do efeito do bi-cluster k;
α_{ik} é o efeito do gene i do bi-cluster k; e
β_{jk} é o efeito da condição j do bi-cluster k.

Seja $\theta_{ij(b)} = [\theta_{ij1}, ..., \theta_{ijb}]^{\mathrm{T}}$ o vetor de efeitos dos b bi-clusters. A probabilidade de sucesso do gene i sob a condição j dado o efeito de todos bi-clusters é expressa como:

$$\pi(\theta_{ij(b)}) = P(y_{ij} = 1 | \theta_{ij(b)}). \tag{4.2}$$

Logo a função logit (3.21) escrita em termos dos efeitos dos bi-clusters é dada por:

$$\text{logit}(\pi(\theta_{ij(b)})) = \sum_{k=1}^{b} \theta_{ijk}\rho_{ik}\kappa_{jk} = \sum_{k=1}^{b} (\mu_k + \alpha_{ik} + \beta_{jk})\rho_{ik}\kappa_{jk}, \tag{4.3}$$

onde:

$$\rho_{ik} = \begin{cases} 1, & \text{se o } i\text{-ésimo gene está no } k\text{-ésimo bi-cluster,} \\ 0, & \text{caso contrário;} \end{cases}$$

e

$$\kappa_{jk} = \begin{cases} 1, & \text{se a } j\text{-ésima condição está no } k\text{-ésimo bi-cluster,} \\ 0, & \text{caso contrário.} \end{cases}$$

Portanto, o modelo Lbic, obtido a partir da expressão (4.3) é formulado como:

$$\pi(\theta_{ij(b)}) = \frac{\exp[\sum_{k=1}^{b} (\mu_k + \alpha_{ik} + \beta_{jk})\rho_{ik}\kappa_{jk}]}{1 + \exp[\sum_{k=1}^{b} (\mu_k + \alpha_{ik} + \beta_{jk})\rho_{ik}\kappa_{jk}]}. \tag{4.4}$$

Dessa forma é possível reescrever o modelo linear (3.25) como:

$$y_{ij} = \pi(\theta_{ij(b)}) + e_{ij}, \tag{4.5}$$

onde e_{ij} é uma variável residual aleatória. A variável e_{ij} é estimada através de:

$$\widehat{e}_{ij} = y_{ij} - \pi(\widehat{\theta}_{ij(b)}), \tag{4.6}$$

onde o estimador $\widehat{\theta}_{ij(b)}$ é obtido através do método de máxima verossimilhança, utilizando o algoritmo de Newton-Raphson apresentado no Capítulo 3.

O método Lbic identifica bi-clusters que minimizam a soma de quadrados dos resíduos, (SQR_L), dado por:

$$\begin{aligned} SQR_L = \sum_{ij} \widehat{e}_{ij}{}^2 &= \sum_{ij} (y_{ij} - \pi(\widehat{\theta}_{ij(b)}))^2 \\ &= \sum_{ij} \left(y_{ij} - \frac{\exp[\sum_{k=1}^{b} (\widehat{\mu}_k + \widehat{\alpha}_{ik} + \widehat{\beta}_{jk}) \widehat{\rho}_{ik} \widehat{\kappa}_{jk}]}{1 + \exp[\sum_{k=1}^{b} (\widehat{\mu}_k + \widehat{\alpha}_{ik} + \widehat{\beta}_{jk}) \widehat{\rho}_{ik} \widehat{\kappa}_{jk}]} \right)^2. \end{aligned} \tag{4.7}$$

Na seção seguinte é descrito o algoritmo que implementa o método proposto baseado no modelo Lbic.

4.2 Algoritmo Lbic

O algoritmo Lbic identifica um bi-cluster por vez. Para identificar um bi-cluster, o algoritmo Lbic age similarmente ao algoritmo Plaid, no sentido de iniciar a busca através de um retalho inicial que é reestimado várias vezes até que seja encontrado um bi-cluster em potencial. Mas, diferente do método Plaid que no final do processo de identificação de um k-ésimo bi-cluster reestima os efeitos dos $(k-1)$ bi-clusters encontrados anteriormente, o método Lbic reestima os efeitos dos $(k-1)$ bi-clusters encontrados anteriormente durante o processo de identificação do k-ésimo bi-cluster. Para isso a soma dos efeitos dos $(k-1)$ bi-clusters é reescrita como uma covariável no modelo Lbic 4.4. Logo a Expressão 4.4 para o k-ésimo bicluster é dada por:

$$\pi(\theta_{ij(k)}) = \frac{\exp[(\mu_k + \alpha_{ik} + \beta_{jk}) \rho_{ik} \kappa_{jk} + \omega_k \sum_{l=1}^{k-1} (\mu_l + \alpha_{il} + \beta_{jl}) \rho_{il} \kappa_{jl}]}{1 + \exp[(\mu_k + \alpha_{ik} + \beta_{jk}) \rho_{ik} \kappa_{jk} + \omega_k \sum_{l=1}^{k-1} (\mu_l + \alpha_{il} + \beta_{jl}) \rho_{il} \kappa_{jl}]}. \tag{4.8}$$

Um retalho é definido como um subconjunto de genes e condições para os quais os parâmetros ρ_{ik} e κ_{jk} assumem simultaneamente o valor 1. Para formar um retalho, inicialmente os parâmetros ρ_{ik} e κ_{jk} são estimados através do método de clustering K-means (apresentado no Capítulo 2) aplicado a uma matriz de probabilidades P. Primeiro o K-means é aplicado às linhas da matriz P para obter um cluster de genes e depois independentemente, é novamente aplicado às colunas da matriz P para obter um cluster de condições. Formado o retalho inicial, o modelo Lbic é ajustado e os parâmetros ρ_{ik} e κ_{jk} são reestimados de maneira que qualquer gene ou condição que não reduza a soma de quadrados dos resíduos seja retirado do retalho. As reestimações de ρ_{ik} e κ_{jk} para o k-ésimo bi-cluster são obtidas através das seguintes relações:

$$\widehat{\rho}_{ik} = \begin{cases} 1, \quad \text{se} \quad \widetilde{\rho}_{ik} = 1, \text{ e} \\ \qquad \sum_{j:\widetilde{\kappa}_{jk}=1} \left(y_{ij} - \frac{\exp[(\widehat{\mu}_k + \widehat{\alpha}_{ik} + \widehat{\beta}_{jk})\widetilde{\kappa}_{jk} + \widehat{\omega}_k * \sum_{l=1}^{k-1} (\widetilde{\mu}_l + \widetilde{\alpha}_{il} + \widetilde{\beta}_{jl})\widetilde{\kappa}_{jk}]}{1+\exp[(\widehat{\mu}_k + \widehat{\alpha}_{ik} + \widehat{\beta}_{jk})\widetilde{\kappa}_{jk} + \widehat{\omega}_k * \sum_{l=1}^{k-1} (\widetilde{\mu}_l + \widetilde{\alpha}_{il} + \widetilde{\beta}_{jl})\widetilde{\kappa}_{jk}]} \right)^2 \leq \\ \qquad (1-\gamma_l) \sum_{j:\widetilde{\kappa}_{jk}=1} \left(y_{ij} - \pi(\widetilde{\theta}_{ij(k-1)})\widetilde{\kappa}_{jk} \right)^2, \\ 0, \quad \text{caso contrário,} \end{cases}$$

e

$$\widehat{\kappa}_{jk} = \begin{cases} 1, \quad \text{se} \quad \widetilde{\kappa}_{jk} = 1, \text{ e} \\ \qquad \sum_{i:\widetilde{\rho}_{ik}=1} \left(y_{ij} - \frac{\exp[(\widehat{\mu}_k + \widehat{\alpha}_{ik} + \widehat{\beta}_{jk})\widehat{\rho}_{ik} + \widehat{\omega}_k * \sum_{l=1}^{k-1} (\widetilde{\mu}_l + \widetilde{\alpha}_{il} + \widetilde{\beta}_{jl})\widetilde{\rho}_{ik}]}{1+\exp[(\widehat{\mu}_k + \widehat{\alpha}_{ik} + \widehat{\beta}_{jk})\widetilde{\rho}_{ik} + \widehat{\omega}_k * \sum_{l=1}^{k} (\widetilde{\mu}_l + \widetilde{\alpha}_{il} + \widetilde{\beta}_{jl})\widetilde{\rho}_{ik}]} \right)^2 \leq \\ \qquad (1-\gamma_c) \sum_{i:\widetilde{\rho}_{ik}=1} \left(y_{ij} - \pi(\widetilde{\theta}_{ij(k-1)})\widetilde{\rho}_{ik} \right)^2, \\ 0, \quad \text{caso contrário,} \end{cases} \tag{4.9}$$

onde:

γ_l e γ_c representam a menor redução na soma de quadrados dos resíduos para os subconjuntos de genes e condições, respectivamente;

$\widetilde{\rho}$=1 e $\widetilde{\kappa}$=1 indicam os genes e condições pertencentes ao retalho atual;

$\widetilde{\mu}$, $\widetilde{\alpha}$ e $\widetilde{\beta}$ são estimativas dos bi-clusters identificados; e

$\widehat{\mu}$, $\widehat{\alpha}$, $\widehat{\beta}$ e $\widehat{\omega}$ são estimadores do modelo Lbic com uma covariável, onde a covariável é dada pela soma das estimativas dos $(k-1)$ bi-cluters identificados. Estes estimadores são calculados como aqueles apresentados na Seção 3.2.2.1,

O retalho continua sendo reestimado até um número pré-determinado R de vezes ou até as reestimações dos seus parâmetros se estabilizarem, isto é, até o retalho reestimado ser o igual ao estimado no passo anterior. Em todas as simulações realizadas no contexto desta tese, o número R de iterações necessárias foi pequeno, inferior a 10.

Após encontrar o retalho final é verificado se as proporções de uns (e zeros) de cada condição (coluna) é maior que um dado limiar φ_c. É de se esperar que φ_c seja um valor próximo a 1, significando que os genes selecionados têm os mesmos resultados para aquela condição. Se a proporção da coluna j for inferior ao φ_c, descarta-se a condição j fazendo $\kappa_{jk} = 0$.

Depois é verificado se a proporção de colunas com respostas iguais a um são maiores que um outro limiar dado φ_l, pois pretende-se agrupar genes que tenham uma quantidade mínima

$(c * \varphi_l)$ de condições com a característica estudada. Isto é, deseja-se que $\sum_j y_{ijk}$ seja próxima ao número de condições do retalho k. Se esta proporção for maior ou igual ao φ_l, o retalho obedece a regra de proporções de uns (e zeros), e é considerado como o k-ésimo bi-cluster, caso contrário ele é descartado.

A decisão do número de clusters, K, a ser agrupado depende do tamanho da matriz Y, n*c, e da capacidade da máquina onde o algoritmo será implementado. O método permite selecionar aleatoriamente um dos K clusters formados, ou escolher o maior (ou menor) cluster gerado.

Nas situações em que as probabilidades estimadas de $\pi(\widehat{\theta}_{ij(k)})$ assumem valores um ou zero, um pequeno valor δ é adicionado (para probabilidades zero) ou subtraído (para probabilidade um) a essas estimativas. Tal manobra permite que os valores das estimativas de μ, α_i e β_j sejam viáveis(finitos). É definido um δ= 1e-35. Para k=1, $\pi(\widehat{\theta}_{ij(0)})$ é definido por P_{ij}, o qual é expresso em (4.10) adiante.

Na identificação do primeiro bi-cluster os elementos da matriz P são definidos por:

$$P_{ij} = \frac{(\#\text{ uns da linha } i) * (\#\text{ zeros da coluna } j)}{(\#\text{ uns da matriz } Y)^2}, \tag{4.10}$$

e para o k-ésimo bi-cluster $(k = 2, ..., b)$, a matriz P é reestimada pelas probabilidades ajustadas, $\pi(\widehat{\theta}_{ij(k)})$, através do modelo Lbic.

O algoritmo Lbic identifica até b bi-clusters baseados nos parâmetros $(Y, P, \gamma_l, \gamma_c, \varphi_c, \varphi_l, K, R)$. O Algoritmo 4.1, apresentado a seguir, representa a estrutura para identificar o k-ésimo bi-cluster.

Algoritmo 4.1

1. Calcule (ou ajuste, se não for o primeiro bi-cluster) a matriz de probabilidades P.
2. Calcule os valores iniciais de $\widehat{\rho}_{ik}^0$ selecionando um cluster de linhas de P obtido do clustering K-means.
3. Calcule os valores iniciais de $\widehat{\kappa}_{jk}^0$ escolhendo um cluster de colunas de P obtido do clustering K-means.
4. Faça r=1.
5. Calcule os efeitos do modelo Lbic para o retalho indicado por : $\widehat{\rho}_{ik}^{r-1}$ e $\widehat{\kappa}_{jk}^{r-1}$
6. Calcule os parâmetros do retalho r como:

$$\widehat{\rho}_{ik}^r = \begin{cases} 1, \quad \text{se} \quad \sum_j \left(y_{ij} - \frac{\exp[(\widehat{\mu}_k^r + \widehat{\alpha}_{ik}^r + \widehat{\beta}_{jk}^r)\widetilde{\kappa}_{jk}^{r-1} + \widehat{\omega}_k^r * \sum_{l=1}^{k-1} (\widetilde{\mu}_l^r + \widetilde{\alpha}_{il}^r + \widetilde{\beta}_{jl}^r)\widetilde{\kappa}_{jk}^{r-1}]}{1+\exp[(\widehat{\mu}_k^r + \widehat{\alpha}_{ik}^r + \widehat{\beta}_{jk}^r)\widetilde{\kappa}_{jk}^{r-1} + \widehat{\omega}_k^r * \sum_{l=1}^{k-1} (\widetilde{\mu}_l^r + \widetilde{\alpha}_{il}^r + \widetilde{\beta}_{jl}^r)\widetilde{\kappa}_{jk}^{r-1}]} \right)^2 \leq \\ \qquad\qquad (1-\gamma_l) \sum_j \left(y_{ij} - \pi(\widetilde{\theta}_{ij(k-1)})\widetilde{\kappa}_{jk}^{r-1} \right)^2, \\ 0, \quad \text{caso contrário,} \end{cases}$$

e

$$\widehat{\kappa}_{jk}^{r} = \begin{cases} 1, \quad \text{se} \quad \sum_i \left(y_{ij} - \frac{\exp[(\widehat{\mu}_k^r + \widehat{\alpha}_{ik}^r + \widehat{\beta}_{jk}^r)\widehat{\rho}_{ik}^{r-1} + \widehat{\omega}_k^r * \sum_{l=1}^{k-1} (\widetilde{\mu}_l^r + \widetilde{\alpha}_{il}^r + \widetilde{\beta}_{jl}^r)\widetilde{\rho}_{ik}^{r-1}]}{1+\exp[(\widehat{\mu}_k^r + \widehat{\alpha}_{ik}^r + \widehat{\beta}_{jk}^r)\widetilde{\rho}_{ik}^{r-1} + \widehat{\omega}_k^r * \sum_{l=1}^{k} (\widetilde{\mu}_l^r + \widetilde{\alpha}_{il}^r + \widetilde{\beta}_{jl}^r)\widetilde{\rho}_{ik}^{r-1}]} \right)^2 \leq \\ \qquad (1-\gamma_c) \sum_i \left(y_{ij} - \pi(\widetilde{\theta}_{ij(k-1)})\widetilde{\rho}_{ik}^{r-1} \right)^2, \\ 0, \quad \text{caso contrário.} \end{cases}$$

7. Repita os passos 5 e 6 para r=2, ..., (R-1), onde R é o número de iterações para estabilizar o retalho.
8. Calcule os efeitos do retalho R como no passo 5.
9. Calcule $\widehat{\rho}_{ik}^{R}$ e $\widehat{\kappa}_{jk}^{R}$ como no passo 6.
10. Aceite o retalho R como o k-ésimo bi-cluster se ele atende aos critérios de proporções de zeros e uns para as colunas do retalho, caso contrário pare.

Este algoritmo pode ser visualizado na Figura 4.2.

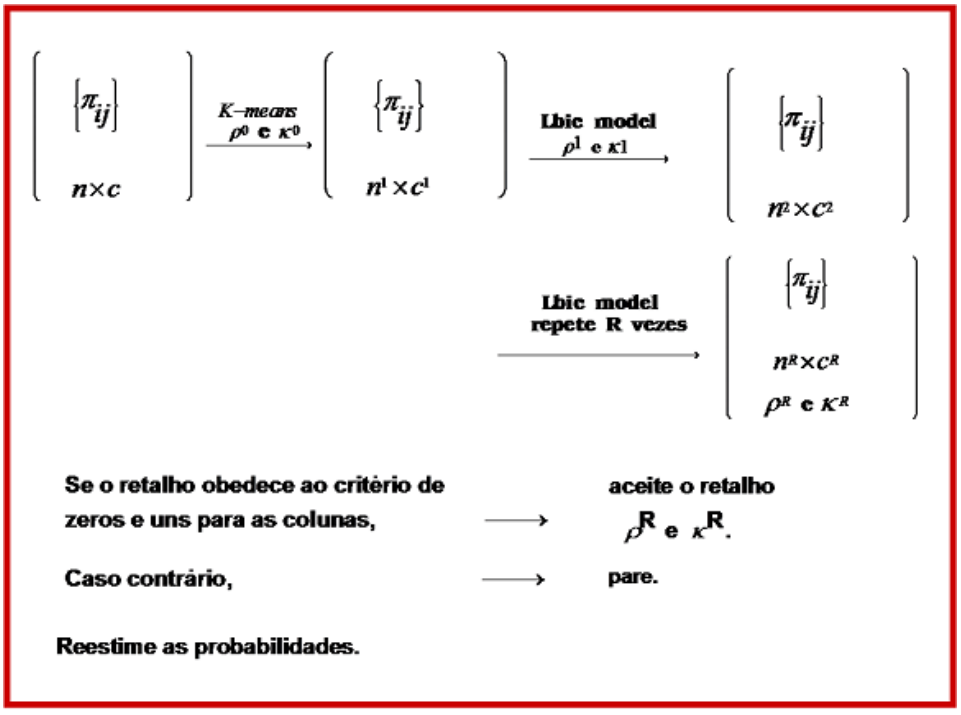

Figura 4.2 Algoritmo Lbic

4.3 Conclusões

Neste trabalho a metodologia de bi-clustering Lbic é utilizada como uma ferramenta para fazer inferência sobre interações de pares de genes. Assim, o método proposto Lbic não só visa identificar bi-clusters, mas também ser capaz de inferir sobre interações de pares de genes.

Resultados de diferentes tipos de dados genômicos podem ser combinados captando interações de pares dos genes que não seriam identificadas usando os dados genômicos individualmente. Por esta razão, no capítulo seguinte é apresentada um alternativa de combinação dos resultados dos métodos de bi-clustering Lbic e Plaid aplicados a dados filogenéticos e de expressão de genes, respectivamente. O algoritmo foi implementado na Linguagem R.

CAPÍTULO 5

Aplicações

Neste capítulo o método proposto, Lbic, é comparado com outros métodos de bi-clustering, Plaid [TBKH05] e Bicbin [UW08], e um método de análise de correlação canônica, ACCKS [YVK04]. Nas comparações são usados dados genômicos do artigo de Yamanishi et al. [YVK04] e dados gerados artificialmentes.

5.1 Dados Experimentais

Dois tipos de dados genômicos com 769 proteínas (genes) da levedura *Saccharomyces cerevisiae* são usados para fazer inferência de pares de proteínas (representadas por genes): dados de expressões de genes e dados filogenéticos. Um conjunto com todas as interações destes genes também é apresentado como uma rede confiável a ser inferida pelos dados genômicos. Estes dados são apresentados a seguir:

1. Dados de Expressões de Genes (*EXP*): Para cada gene, 157 condições são usadas. Estes dados são representados em forma de uma matriz $Y_{EXP:769\times157}$, onde os elementos $\{y_{ij}\}$ para *i*=1,..., 769 e *j*=1, ..., 157 assumem valores reais.

2. Dados Filogenéticos (*PHY*): Para cada gene é verificado se ele é ortólogo a genes de 145 organismos. Genes ortólogos apresentam a mesma função em organismos diferentes, em geral. Dos 145 organismos, 11 são eucariontes, 16 são arqueas e 118 são bactérias. Estes dados são representados em forma de uma matriz $Y_{PHY:769\times145}$, onde os elementos $\{y_{ij}\}$ para *i*=1,..., 769 e *j*=1, ..., 145 assumem valores dicotômicos:

$$y_{ij} = \begin{cases} 1, & \text{se o gene } i \text{ está presente no organismo } j, \\ 0, & \text{se o gene } i \text{ não está presente no organismo } j. \end{cases}$$

3. Rede de Proteínas Padrão Ouro: considerada uma parte confiável da rede global de proteínas a ser inferida. Esta rede é usada para verificar se a rede encontrada (baseadas nos dados de expressões de genes e (ou) dados filogenéticos) apresentam resultados similares a ela. A rede padrão ouro contém os mesmos 769 genes da levedura *Saccharomyces cerevisiae* apresentadas nos dados genômicos anteriores, onde 3702 pares desses genes interagem. Redes de proteínas podem ser definidas como um grafo onde os vértices correspondem aos genes e as arestas, correspondem às interações entre os pares de genes. Tal gráfo pode ser representado em forma de uma matriz de adjacências $A_{n\times n}$, onde os elementos $\{a_{ll'}\}$, para $l, l' = 1, ..., n$, assumem valores dicotômicos:

$$a_{ll'} = \begin{cases} 1, & \text{se o gene } l \text{ interage com o gene } l', \\ 0, & \text{se o gene } l \text{ não interage com o gene } l'. \end{cases}$$

Além dos dados citados acima, vários conjuntos de dados foram gerados artificialmente para comparação do Lbic com o Bicbin, como por exemplo: 100 matrizes (100×100) com três bi-clusters, 100 matrizes (200×200) com cinco bi-clusters, 100 matrizes (500×500) com cinco bi-clusters, usando proporções de uns iguais a 0.15, 0.20, 0.30, 0.40 e 0.50. Um desses conjuntos é representado neste trabalho como segue.

4. Dados Artificiais: matriz 100×100 de zeros e uns gerados por uma distruição de Bernoulli (com probabilidade 0.25) e sobreposição de três bi-clusters (blocos de uns) gerando uma proporção de uns na matriz igual p = 40.

Para aplicar a metodologia ACCKS são necessário transformar as matrizes de dados de expressão de genes e dados filogenéticos, e a rede conhecida (matriz padrão ouro) pelas suas matrizes de kernels.

Seja $\mathbf{y}^i$ um vetor de observações do gene i. Então, para os dados de expressões de genes, onde os elementos de $\mathbf{y}^i$, para i=1,..., n assumem valores reais, é empregada a função gaussiana dada por:

$$K(\mathbf{y}^i, \mathbf{y}^{i'}) = \exp(-\left\|\mathbf{y}^i - \mathbf{y}^{i'}\right\|^2 / 2\sigma^2,$$

e para os dados filogenéticos, onde os elementos $\mathbf{y}^i$, para i=1,..., n, assumem valores 0 ou 1, é aplicada a função linear obtida por:

$$K(\mathbf{y}^i, \mathbf{y}^{i'}) = \mathbf{y}^i \times \mathbf{y}^{i'}.$$

Seja A a matriz de adjacências da rede padrão ouro. A função aplicada a esta matriz é a função difusão, dada por

$$K = \exp(\beta H),$$

onde $\beta > 0$, $H = (A - D)$ e D é uma matriz diagonal de conectividade dos nós.

5.2 Parâmetros

Para cada método considerado neste trabalho, uma variedade de parâmetros é necessária para a execução de seu algoritmo. Os parâmetros foram escolhidos depois de vários estudos através de tentativas para melhorar o desempenho dos métodos. A seguir são apresentadas tabelas com os parâmetros utilizados em cada método aplicado.

Tabela 5.1 Método Plaid para dados de expressões de genes

Parâmetro	Descrição	valor
τ_1	critério de eliminação de linha	0.6
τ_2	critério de eliminação de coluna	0.6
T	número de retalhos permutados a serem usados no teste de permutação	2
R	número de reajustes de cada retalho	4
S	número de iterações para achar um retalho final	6
b	número máximo de bi-clusters	15

Tabela 5.2 Método Bicbin para dados artificiais

Parâmetro	Descrição	valor
α	critério de eliminação de linha	0.50
β	critério de eliminação de coluna	0.70
p.g	procura pelas linhas	True
p	proporção de uns na matriz de dados	0.40

Tabela 5.3 Método Lbic para dados filogenéticos

Parâmetro	Descrição	valor
τ_1	critério de eliminação de linha	0.70
τ_2	critério de eliminação de coluna	0.60
R	número de iterações para achar um retalho final	4
b	número máximo de bi-clusters	10
K_l	número de clusters das linhas para o K-means	4
K_c	número de clusters das colunas para o K-means	2
φ_l	proporção mínima de colunas com valores iguais a 1	0.60
φ_c	proporção mínima de uns (ou zeros) por coluna	0.85

Tabela 5.4 Método Lbic para dados artificiais

Parâmetro	Descrição	valor
τ_1	critério de eliminação de linha	0.70
τ_2	critério de eliminação de coluna	0.60
R	número de iterações para achar um retalho final	4
b	número máximo de bi-clusters	5
K_l	número de clusters das linhas para o K_means	2
K_c	número de clusters das colunas para o K_means	2
φ_l	proporção mínima de colunas com valores iguais a 1	0.60
φ_c	proporção mínima de uns (ou zeros) por coluna	0.80

Tabela 5.5 Método ACCKS para dados filogenéticos e de expressões de genes

Parâmetro	Descrição	valor
β	parâmetro do Kernel difusão	1
σ	parâmetro do Kernel gaussiano	5
L	número de componentes principais	True
λ_1	parametro de regularização 1	0.40
λ_2	parametro de regularização 2	0.40

Para Lbic aplicado aos dados filogenéticos a quantidade K = 4 foi escolhida de modo que o tamanho do retalho, dado pelo produto entre o número de genes e o número de condições, $n * c$, atendesse ao tamanho da memória necessária para o programa executar o modelo Lbic naquele retalho. Uma vez que o total de pares de genes que de fato interagem é muito pequeno comparado com o total de pares de genes estudados (somente pequenos grupos de genes apresentam algum tipo de relação), então, K=4 é uma escolha razoável para selecionar clusters de genes para o retalho inicial.

O desempenho de ACCKS é altamente dependente da proporção de informação conhecida na fase de treinamento. Por tal motivo são selecionados três percentuais de supervisão da rede padrão ouro : 10%, 25% e 50%. Isto significa que os dados são treinados com 10%, 25% e 50% e então testados com 90%, 75% e 50% respectivamente.

Para o primeiro percentual, 10%, os genes são divididos em 10 partes. Uma vez que uma parte é escolhida, as demais partes são treinadas. Isto é repetido para as 10 partes, gerando 10 matrizes de correlações dos genes, $M_i,\ i = 1,...,10$.

Para o segundo percentual, 25%, os genes são divididos em 4 partes. Para uma parte treinada, as três demais são testadas. Repete-se o processo para as 4 partes gerando 4 matrizes de correlações dos genes, $M_i,\ i = 1,...,14$.

Simirlamente, para o terceiro percentual, 50%, os genes são divididos em duas partes. Quando uma parte é treinada, a outra é testada, gerando duas matrizes de correlações dos genes, $M_i,\ i = 1,2$.

Para os percentuais 10%, 25% e 50% são calculadas as médias das matrizes $\bar{M}_{10\%j} = \sum_{i=1}^{10} M_i/10$, $\bar{M}_{25\%j} = \sum_{i=1}^{4} M_i/4$ e $\bar{M}_{50\%j} = \sum_{i=1}^{2} M_i/2$, respectivamente. Este processo é repetido 30 vezes e novamente é calculada a média das 30 matrizes (de médias de correlações) geradas. Para os 10%, 25% e 50% de supervisões são calculadas as médias $\bar{M}_{10\%} = \sum_{j=1}^{30} M_{10\%j}/30$, $\bar{M}_{25\%} = \sum_{j=1}^{30} M_{25\%j}/30$ e $\bar{M}_{50\%} = \sum_{j=1}^{30} M_{50\%j}/30$, respectivamente, que são usadas para gerar os gráficos na comparação com o Lbic.

5.3 Comparações

Todas as comparações realizadas nesta tese foram feitas graficamente. Para comparar os método Lbic e Bicbin (aplicados aos dados artificiais) são construídos gráficos que representam as matrizes de dados com seus bi-clusters identificados. Para as outras comparações são gerados gráficos dos valores preditivos positivos versus os valores das sensibilidades que verificam os desempenhos das inferências feitas para interações de pares de proteínas (genes. Neste capítulo

quando se fizer referência a interações será a respeito de interações de pares de proteínas. A sensibilidade (S) representa a proporção entre as interações verdadeiras encontradas e todas as interações na rede padrão ouro, e é calculada por:

$$\mathrm{S} = \frac{\mathrm{VP}}{\mathrm{VP+FN}} \in [0,1]\,,$$

onde:

VP (Verdadeiro Positivo) é a quantidade de interações verdadeiras encontradas;

FN (Falso Negativo) é a quantidade de não interações falsas encontradas.

O valor preditivo positivo (VPP) representa a proporção entre as interações verdadeiras encontradas e todas as interações encontradas, e é obtida por:

$$\mathrm{VPP} = \frac{\mathrm{VP}}{\mathrm{VP+FP}} \in [0,1]\,,$$

onde FP (Falso positivo) é a quantidade de interações falsas encontradas.

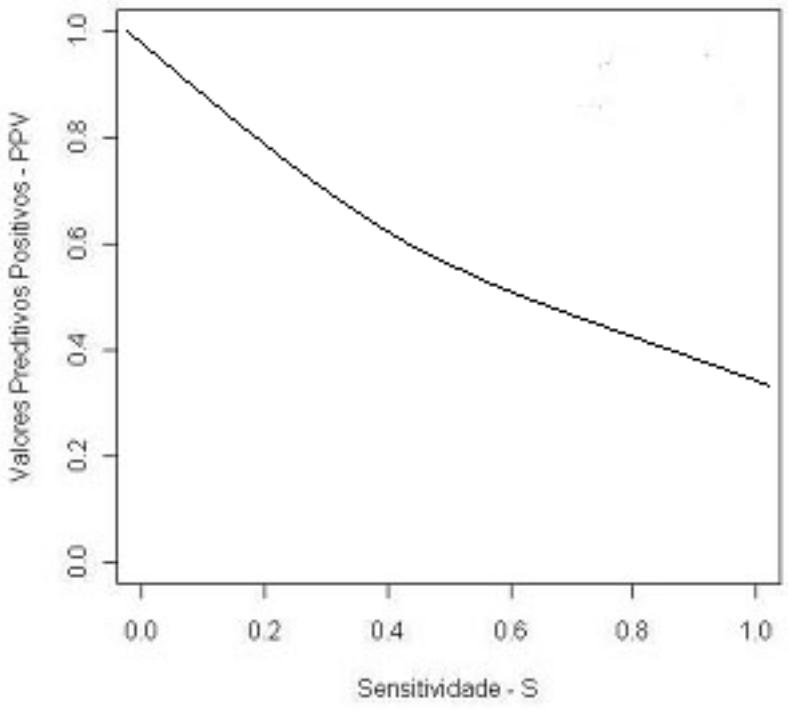

Figura 5.1 Exemplo de gráfico VPP$\times$ S

Quanto maiores os valores do VPP melhores são os resultados. Na prática, a curva do gráfico S $\times$ VPP tem altos valores de VPP para baixos valores de S e baixos valores de VPP

para altos valores de S. Frequentemente a sensibilidade do método atinge valores medianos, entre 0.4 e 0.7. Assim, nesta faixa de valores para sensibilidade é desejado ter valores de VPP o mais alto possíveis. A curva com os valores de S fora deste intervalo deve ser ignorada. A figura 5.1 mostra um exemplo de curva desejada.

Interação de Proteínas: Regra aplicada aos métodos Plaid e Lbic

Para decidir se um par de genes interage foram estudados vários critérios. Dois deles que melhor segregam os pares de genes que interagem daqueles que não interagem, aplicados nos métodos Plaid e Lbic, são apresentados abaixo.

Critério 1: Proporção de bi-cluters comuns ao par (gene l, gene l'), dada por:

$$\frac{\text{\# bi-clusters com o par (gene } l \text{ , gene } l' \text{) presente}}{\text{máximo(\# bi-clusters com o gene } l \text{ presente, \# bi-clusters com o gene } l' \text{ presente)}}$$

Critério 2: Quantidade de colunas exclusivas em todos os bi-clusters nas quais o gene l e o gene l' aparecem juntos.

Um par de genes interage se o valor calculado para esse par, segundo um critério, é maior que um limiar dado. Uma combinação dos Critérios 1 e 2 permite uma melhor segregação entre os pares que interagem e não interagem. Assim, para os métodos Lbic e Plaid é adotada a regra que um par (gene l , gene l') interage se ele atende a pelo menos um dos critérios apresentados.

Os métodos Lbic e Plaid não identificam necessariamente bi-clusters que englobam todos os genes (nem todos pares) associados a rede padrão ouro, isto é, existem pares de genes que não pertencem a nenhum bi-cluster identificado. Portanto, a regra de decisão só é aplicada a pares de genes que pertencem a pelo menos um bi-cluster.

Usualmente a proporção de pares de genes que interagem na rede padrão ouro é muito pequena (1.5% no caso dos 769 genes aplicados nesta tese). Assim, existe uma grande diferença entre o número de pares que interagem daqueles que não interagem. Para detectar diferenças entre os métodos aplicados, são escolhidos alguns pares de genes da rede padrão ouro. Dentro dos pares de genes identificados pelos bi-clusters são selecionados todos aqueles que interagem na rede padrão ouro, e selecionados aleatoriamente o dobro dos pares de genes que não interagem, isto é, são escolhidos na proporção de um para dois (1:2) pares de genes que interagem e que não interagem na rede padrão ouro. Este procedimento é repetido 100 vezes e então as médias dos valores de S e VPP sob os mesmos limiares são analisadas graficamente.

Os limiares foram definidos como proporções do número de pares a serem inferidos. Os limiares escolhidos são: 1% , 2.5%, 5%, 7.5%, 10%, 15%, 20%, 30%, 50 % e 70% dos pares com maiores valores calculados pelo critério determinado.

Interação de Proteínas: Regra aplicada ao método ACCKS

Critério 3: Coeficiente de Correlação de Pearson

Para o método ACCKS é adotada a regra que um par (gene l , gene l') interage se o coeficiente de correlação de Pearson (Critério 3) calculado para esse par é maior que um limiar dado. Similarmente ao procedimento realizado com os métodos Plaid e Lbic, são escolhidos

aleatoriamente na proporção de um para dois (1:2) pares de genes que interagem e que não interagem na rede padrão ouro. Este procedimento é repetido 100 vezes e são calculados as médias dos valores de S e VPP sob os mesmos limiares. Uma vez que são geradas várias matrizes de correlações de genes (segundo sua supervisão), então, para cada supervisão, o procedimento anterior é aplicado a cada uma dessas matrizes e as médias das médias de S e VPP são analisadas graficamente.

5.3.1 ACCKS com várias supervisões

O Método ACCKS aplicado aos dados filogenéticos e de expressões mostra claramente (ver figura 5.2) que seu desempenho cresce com o aumento da percentagem de conhecimento. Estes desempenhos tornam-se irreais, uma vez que o atual conhecimento sobre interações de pares de genes é muito limitado. Um percentual de 25% de supervisão é considerado razoável, por este motivo o método Lbic é comparado com ACCKS com supervisões de até 25%.

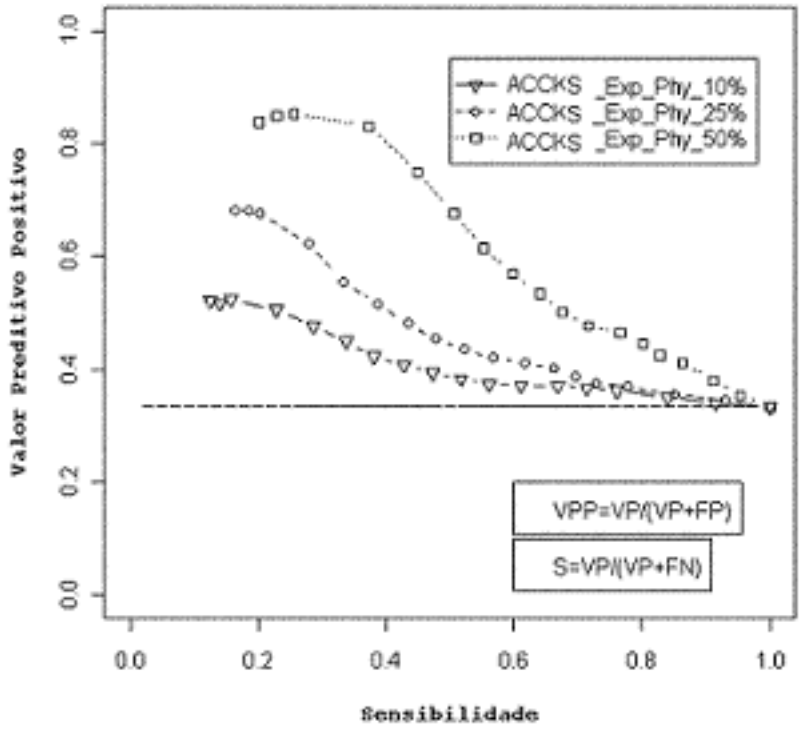

Figura 5.2 ACCKS com supervisões 10%, 25%, 50% e 90%

5.3.2 Lbic versus Bicbin para dados artificiais

Os métodos Lbic e Bicbin são aplicados aos vários conjuntos de dados artificiais citados na Seção 5.1. Para 83% deles, o desempenho do Lbic supera a de Bicbin, identificando bi-clusters que apresentam sobreposições e que são pequenos em relação aos demais na respectiva matriz. Em algumas situações o método Bicbin identifica bi-clusters com colunas onde a proporção de uns é em torno de 60%. Para os demais conjuntos os dois métodos apresentam resultados similares. Resultado típico dos métodos aplicados aos dados artificiais é mostrado na Figura 5.3. A Figura 5.3(a) mostra a matriz original com três bi-clusters gerados artificialmente. As Figuras 5.3(b) e 5.3(c) mostram os bi-clusters identificados pelo Bicbin, e as Figuras 5.3(d) e 5.3(e) mostram os bi-clusters identificados pelo Lbic. As cores pretas e brancas representam os zeros e uns, respectivamente, da matriz original. Bi-clusters com cores cinza escura indicam que eles não têm sobreposição e com cores cinza claro idicam que eles têm sobreposições. O método Bicbin aplicado a matriz original também identifica um terceiro bi-cluster, mas ele é a própria matriz original. Este é um resultado típico para este método. Por causa da proporção de zeros estabelecida pelo Bicbin, pequenos bi-clusters não são identificados, pois em poucas iterações o algoritmo identifica a matriz original. Mudando os valores dos parâmetros α e β leva a piores resultados, isto é, só dois biclusters são identificados: uma submatriz da matriz original e a própria matriz. O método Lbic, no entanto, identifica três bi-clustes. A Figura 5.3(e) mostra que Lbic encontra bi-clusters que se sobrepoem, como é desejado, enquanto o Bicbin não encontra.

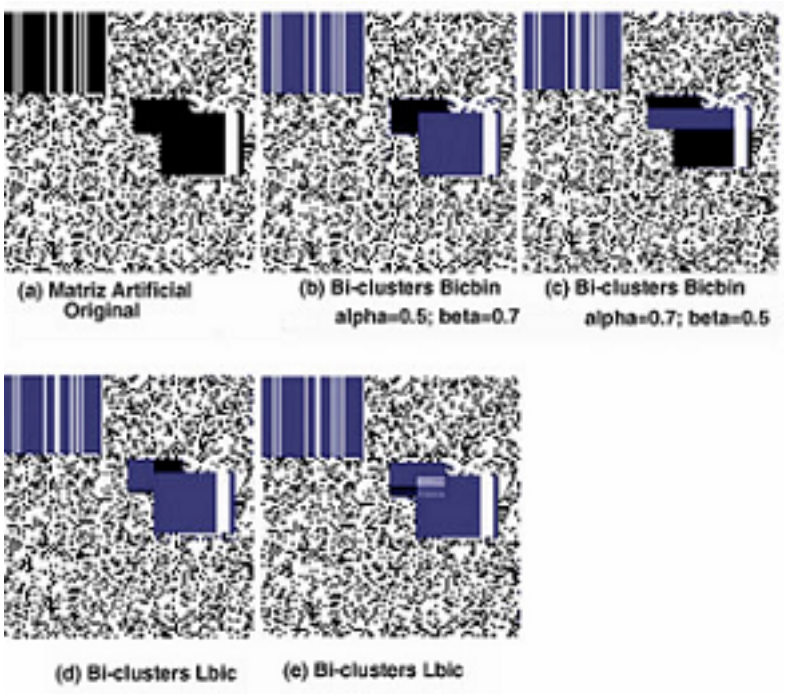

Figura 5.3 Lbic e BicBin para dados artificiais

5.3.3 Lbic versus Bicbin para dados filogenéticos

O método Bicbin quando aplicado aos dados filogenéticos só identifica um bi-cluster onde a maior parte dos genes é selecionada, levando a ter proporções de uns nas várias condições (colunas) muito distante do desejado 1. Uma possível explicação para este fato é que este método fixa uma quantidade de zeros para o bi-cluster. Similarmente ao encontrado nas matrizes artificiais, o Bicbin não encontra pequenos e sobrepostos bi-clusters nos dados filogenéticos, logo, ele não gera resultados que podem ser usados com o Critério 1 e 2 para decidir se um par de genes interage ou não. O método Lbic aplicado aos dados filogenéticos identifica vários bi-clusters que contém poucos genes, alguns bi-clusters que são sobrepostos e ainda, bi-clusters que têm colunas com mais de 80% de zeros.

5.3.4 Combinação (Lbic e Plaid) versus Lbic e Plaid

A proposta dessa tese é desenvolver um método que use dados binários para possibilitar fazer inferências sobre interação de pares de genes. A maioria dos métodos aplicados para fazer tal inferência faz uso de dados contínuos que por sua vez podem gerar resultados não tão satisfatórios. A Figura 5.4 mostra que a inferência realizada com dados de expressões via o método Plaid apresenta desempenho inferior àquela realizada com dados filogenéticos via o método Lbic. Toda a curva gerada pelo Lbic está acima daquela gerada pelo Plaid, mostrando que os dados filogenéticos são importantes para as compreensões das interações de pares de genes. Porém, para valores da sensibilidade entre 0.5 e 0.7 estas curvas ficam próximas e não apresentam elevados valores de VPP. Para melhorar esse resultado, uma combinação dos dados filogenéticos e de expressões através dos resultados obtidos por Lbic e Plaid, respectivamente, é proposta. A combinação dos resultados é feita seguindo uma nova regra de decisão para os pares de genes. Um par de genes é dito que interage se ele o faz em pelo menos um dos métodos (Lbic ou Plaid). Seja E a especificidade dada pela proporção entre as não interações verdadeiras encontradas e todas as não interações na rede padrão, calculada por:

$$\mathrm{E} = \frac{\mathrm{VN}}{\mathrm{VN+FP}} \in [0,1]\,,$$

onde:

VN (Verdadeiro Negativo) é a quantidade de não interações verdadeiras encontradas;

FP (Falso Positivo) é a quantidade de interações falsas encontradas.

Sejam S_L, E_L e VPP_L, respectivamente a sensibilidade, a especificidade e o valor preditivo positivo encontrados via Lbic; e S_P, E_P e VPP_P respectivamente a sensibilidade, a especificidade e o valor preditivo positivo encontrados via Plaid. Então, a sensibilidade combinada , a especificidade combinada e o valor preditivo positivo combinado são calculados como:

$$\mathrm{S}_C = \mathrm{S}_L + \mathrm{S}_P - \mathrm{S}_L * \mathrm{S}_P \in [0,1];$$

$$\mathrm{E}_C = \mathrm{E}_L * \mathrm{E}_P \in [0,1]; \text{ e}$$

$$\mathrm{VPP}_C = \frac{\mathrm{S}_{C*(1/3)}}{\mathrm{S}_{C*(1/3)+(1-\mathrm{E}_C)*(2/3)}} \in [0,1].$$

A curva dos métodos combinados é construída através de S_C versus VPP_C. Na Figura 5.4 verifica-se o desempenho da combinação dos métodos Lbic (dados filogenéticos) e Plaid (dados de expressões). Apesar desta curva estar abaixo da curva do método Lbic quando a sensibilidade atinge valores inferiores a 0.4, ela apresenta maiores valores de VPP para sensibilidades maiores que 0.4, como é desejado. Esse fato indica que a combinação dos resultados de identifição da integração dos pares de genes supera aqueles obtidos por Lbic, e como consequência, aqueles obtidos por Plaid.

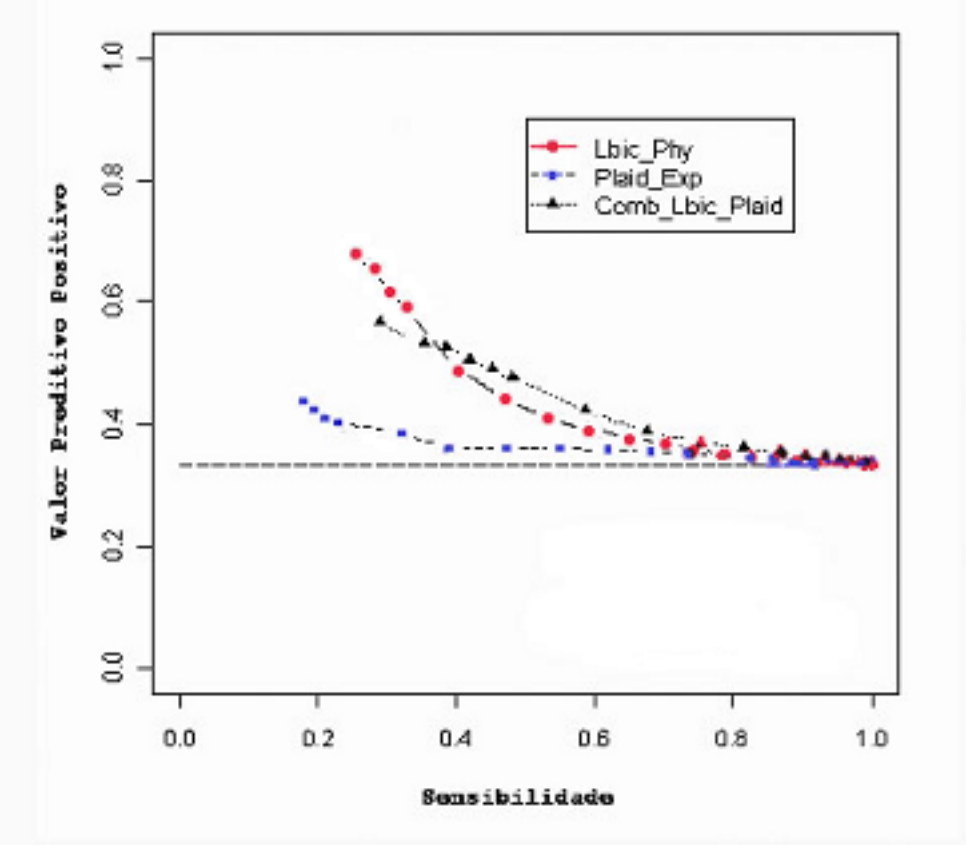

Figura 5.4 Lbic (dados filogenéticos), Plaid (dados de expressões) e Combinação (Lbic, Plaid)

5.3.5 Lbib versus ACCKS para dados filogenéticos

A Figura 5.5 mostra os desempenhos dos métodos Lbic e ACCKS (supervisão de 10%, 25% e 50%) aplicado aos dados filogenéticos. Nota-se que a curva do método Lbic está acima das curvas do método ACCKS supervisionado com 10% e 25% e da curva com 50% de supervisão quando as sensibilidades são menores que 0.55. Este resultado indica novamente que o método Lbic supera o ACCKS em termos de decidir sobre integração dos pares de genes mesmo sem

fazer uso de supervisão de até 25% da rede padrão ouro. Quando 50% dos dados são supervisionados, Lbic supera o ACCKS para sensibilidades menores que 0.55.

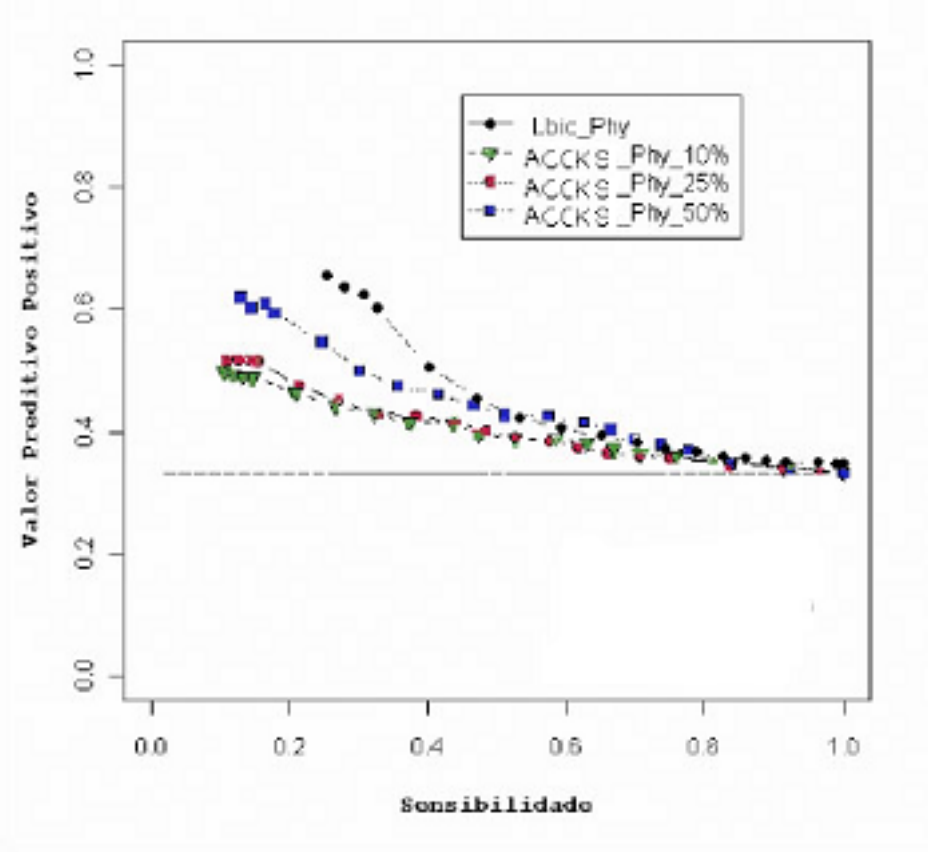

Figura 5.5 Métodos Lbic e ACCKS para dados filogenéticos

5.3.6 Combinação (Lbib, Plaid) versus Integração (ACCKS)

Como a combinação dos resultados de Lbic e Plaid parecem ajudar na inferência de pares de genes e o método ACCKS sugere integração de dados, uma comparação entre a combinação dos resultados dos métodos Lbic e Plaid e o método ACCKS com os dados filogenéticos e de expressões integrados é de interesse. A Figura 5.6 apresenta esta comparação usando os percentuais de dados de supervisão de 10%e 25%. A curva para os métodos combinados apresenta valores de VPP ligeiramente maiores quando as sensibilidades são maiores que 0.4. Logo verifica-se que os métodos Lbic e Plaid combinados apresentam desempenhos similares ao método ACCKS quando supervisionado com até 25% da rede padrão ouro.

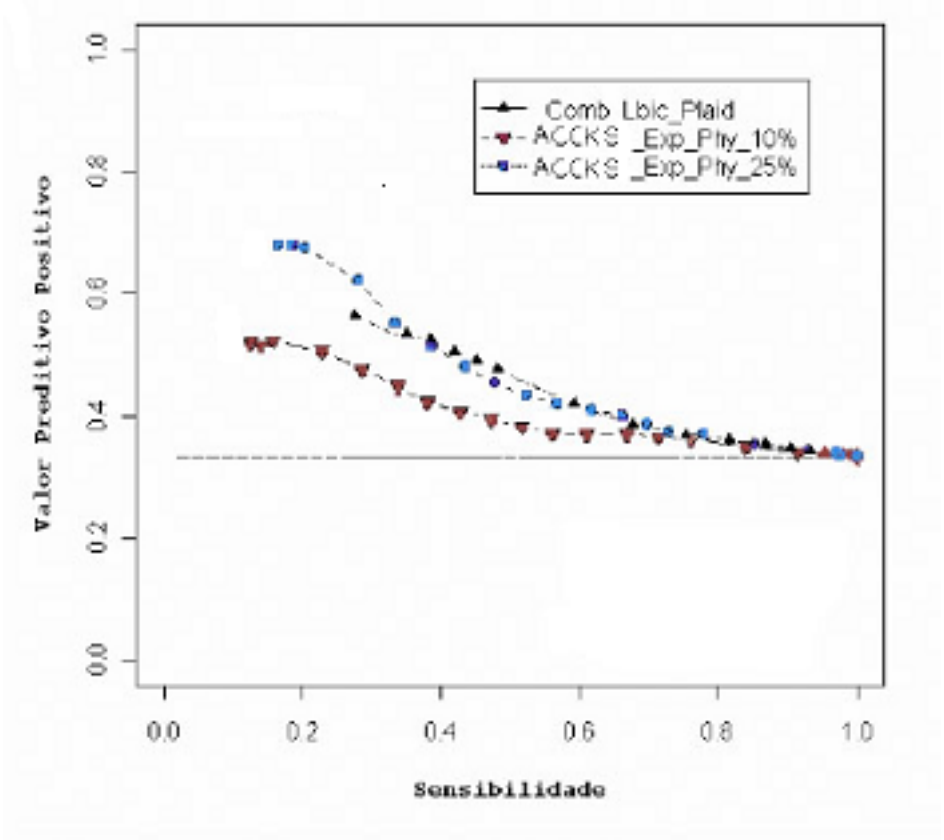

Figura 5.6 Combinação (Lbic$_{PHY}$ e Plaid$_{EXP}$) e Integração ACCKS$_{EXP,PHY}$

5.4 Conclusões

Neste capítulo apresentou-se comparações do método proposto nessa tese, Lbic, e os métodos Bicbin, Plaid e ACCKS. Em comparação com o Bicbin (bi-clustering para dados binários), usando dados artificiais, o Lbic apresentou melhores resultados, pois, diferente do Bicbin, ele identificou bi-clusters sobrepostos e de diferentes tamanhos (pequenos e grandes matrizes). Usando dados filogenéticos, diferente do Lbic, o Bicbin não conseguiu identificar bi-clusters de interesse.

Como Plaid é aplicado a dados de expressão e Lbic é aplicado a dados filogenéticos, a comparação é realizada no sentido de verificar qual método consegue obter mais informações sobre a rede de proteínas estudada. Lbic mostrou melhor desempenho, mas, combinando os resultados dos dois métodos o desempenho na inferência sobre as interações dos pares de proteínas da rede estudada aumentou.

Em comparação com ACCKS, Lbic decidiu mais corretamente sobre interação dos pares de genes, mesmo sem fazer uso de supervisão de até 25% da rede padrão ouro, quando ambos os métodos usaram dados filogenéticos.

Usando os resultados combinados dos métodos Lbic (usando dados filogenéticos) e Plaid

(usando dados de expressão) apresentou-se desempenho similar ao método ACCKS (usando dados filogenéticos e de expressão integrados) quando supervisionado com até 25% da rede padrão ouro.

Todos os métodos foram implementados em linguagem R. Os programas para executar os métodos Bicbin, Plaid e ACCKS foram obtidos com os autores. O programa para executar o método Lbic e construir os gráficos foram desenvolvidos neste trabalho.

CAPÍTULO 6

Considerações Finais

Neste capítulo são apresentados os resultados e contribuições desta tese, além de enumerar alguns dos trabalhos a serem feitos no futuro.

6.1 Contribuições

A agrupamento de genes é tema de grande relevância uma vez que o conhecimento de relações gênicas é importante em várias áreas da biotecnologia. Métodos de agrupamento têm sido bastante abordados para agrupar genes cujas respostas são de naturezas contínuas (assume valores no conjunto dos reais), mas pouco tem sido feito em relação a classificação de genes cujas respostas são de natureza binária. Tal fato pode levar a que conjuntos de dados genômicos importantes deixem de ser considerados na identificação de relações entre os genes. Por esta razão, nesta tese foi desenvolvida uma metodologia de bi-clustering, Lbic, que lida com dados genômicos binários. O método Lbic foi a principal contribuição deste trabalho.

No Capítulo 2 foram apresentados vários artigos que mostram que combinar informações de diferentes tipos de conjuntos de dados genômicos tornam as inferências sobre interações de pares de genes mais precisas. Por este motivo, neste trabalho também foi apresentado uma alternativa de combinação das informações de dados genômicos de natureza contínua e binária. Tal procedimento foi realizado através de um método de combinação dos resultados do método Lbic aplicado a dados binários e do método Plaid aplicado a dados contínuos.

Parte desse trabalho está no artigo publicado na conferência IEEE BIBE 2009 [MG09]. Outra parte está num artigo submetido a um periódico.

6.2 Resultados

Analisando os resultados do método Lbic aplicado a dados filogenéticos da levedura *Saccharomyces cerevisiae*, observa-se que os bi-clusters identificados detectam informações importantes para inferência de pares de genes daquela levedura, enquanto os resultados do método Plaid (proposto por Tuner e colegas [TBKH05]) aplicado a dados de expressão de genes da mesma levedura, não. Tal fato implica na importância de uma metodologia que considere dados binários. Também é observado que a combinação dos resultados desses dois métodos supera os resultados dos métodos realizados individualmente.

Nesta tese foi realizada uma inferência para redes de genes baseada na abordagem supervisionada SKCC (proposta por Yamanishi e colegas [YVK04]) aplicada aos dados filogenéticos, e aplicada aos dados de expressão e filogenéticos integrados para comparar com

a inferência realizada pelo Lbic. O método Lbic supera o método SKCCA em até 25% de supervisão.

Outro método também desenvolvido para dados binários é o Bicbin (proposto por Uitert e colegas [UW08]). Esta abordagem aplicada aos dados filogenéticos da levedura em estudo identifica apenas um bi-cluster contendo a maioria dos genes e não identifica bi-clusters pequenos e sobrepostos como faz o Lbic. O principal motivo da criação do método Lbic foi possibilitar a captura de informações de dados genômicos binários para fazer inferência sobre interação de pares de genes. Portanto, a falta de identificação de bi-clusters pelo Bicbin inviabiliza o procedimento de inferência para interação dos pares de genes proposto nesta tese. Por este motivo o Lbic e o Bicbin não foram comparados quando aplicados aos dados filogenéticos. Porém, o Lbic foi comparado ao Bicbin através de dados artificiais, mostrando que a identificação de bi-clusters de tamanhos distintos e com sobreposições é superior quando realizada pelo método Lbic.

6.3 Trabalhos Futuros

Alguns dos trabalhos a serem realizados no futuro são:

1. Encontrar alternativas de combinação dos métodos Lbic e Plaid.
2. Encontrar alternativas de estimação dos parâmetros em modelos de classificação logística.
3. Incorporar conhecimento biológico a priori ao modelo Lbic.
4. Incorporar variáveis relacionadas a vários organismos ao modelo Lbic.

Referências Bibliográficas

[Aka01] S. Akaho. A kernel method for canonical correlation analysis. *International Meeting of Psychometric Society-IMPS*, 2001.

[BCRC04] J. S. Bader, A. Chaudhuri, J. M. Rothberg, and J. Chant. Gaining confidence in high-throughput protein interaction networks. *Nature Biotechnology*, 22:78 – 85, 2004.

[BDCKY02] A. Ben-Dor, B. Chor, R. Karp, and Z. Yakhini. Discovering local structure in gene expression data: The order-preserving submatrix problem. *Proc. Sixth Int'l conf. Computational Biology*, pages 49–57, 2002.

[BJK02] S. Busygin, G. Jacobsen, and E. Kramer. Double conjugated clustering applied to leukemia microarray data. *Proc. Second SIAM Int'l Conf. Data Mining, Workshop Clustering High Dimensional Data*, 2002.

[CC00] Y. Cheng and G. Church. Biclustering of expression data. *Proc Int Conf Intell Syst Mol Biol.*, 8:93–103, 2000.

[Chr97] R. Christensen. *Log-Linear Models and Logistic Regression*. Spring-Verlag, New York, second edition, 1997.

[CM82] D. R. Cox and P. MacCullagh. Some aspects of analysis of covariance. *Biometrics*, 38(3):541–561, 1982.

[CST00] A. Califano, G. Stolovitzky, and Y. Tu. Analysis of gene expression microarrays for phenotype classification. *Proc. Int'l Conf. Computaional Molecular Biology*, 8, 2000.

[GLD00] G. Getz, E. Lavine, and E. Domany. Coupled two-way clustering analysis of gene microarray data. *Proc. Natural Academy of Sciences US*, 97(22):12079–12084, 2000.

[Har98] D. A. Harville. *Matrix Algebra From A Statistician's Perspective*. Spring-Verlag, New York, 1998.

[HFG+03] W-K. Huh, J. V. Falvo, L. C. Gerke, A. S. Carroll, R. W. Howson, J. S. Weissman, and E. K. O´Shea. Global analysis of protein localization in budding yeast. *Nature*, 425(6959):686–691, 2003.

[HKC+04] A. C. Haugen, R. Kelley, J. B. Collins, C. J. Tucker, C. Deng, C. A. Afshari, J. M. Brown, T. Ideker, and B. V. Houten. Integrating phenotypic and expression profiles to map arsenic-response networks. *Genome Biology*, 5(12):R95, 2004.

[HL00] D. W. Hosmer and S. Lemeshow. *Applied Logistic Regression*. John Wiley and Sons, INC, New York, second edition, 2000.

[JW00] R. A. Johnson and D. W. Wichern. *Applied multivariate statistical analysis*. Prentice Hall, New Jersey, 5th edition, 2000.

[KBCG03] Y. Klugar, R. Basri, J.T. Chang, and M. Gerstein. Spectral biclustering of microarray data: Coclustering genes and conditions. *Genome Research*, 13:703–716, 2003.

[KI05] R. Kelley and T. Ideker. Systematic interpretation of genetic interactions using proteins networks. *Nature Biotechnology*, 23:561 – 566, 2005.

[Koh82] T. Kohonen. Self-organized formation of topologically correct feature maps. *Biological Cybernatics.*, 43:59–69, 1982.

[KSG04] M. Koyutürk, W. Szpankowski, and A. Grama. Biclustering gene-feature matrices for statistically significant patterns. *Proc. IEEE Comput. Syst. Bioinform. Conf.*, pages 480–484, 2004.

[KSK+03] B.P. Kelley, R. Sharan, R. M. Karp, T. Sitter, D. E. Root, B. R. Stockwell, and T. Ideker. Conserved pathways within bacteria and yeast as revealed by global protein network alignment. *Proc. Natl. Acad. Sci.*, 100(20):11394–11399, 2003.

[LO02] L. Lazzeroni and A. Owen. Plaid models for gene expression data. *Statistica Sinica*, 12:61–86, 2002.

[Mac67] J. MacQueen. Some methods for classification and analysis of multivariate observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, 1:281–297, 1967.

[Met78] C. E. Metz. Basic principles of roc analysis. *Seminars in Nuclear Medicine*, VIII, 1978.

[MG09] C. C. R. R. Monteiro and K. S. Guimarães. Logistic biclustering models for protein network inference. *Ninth IEEE International Conference on Bioinformatics and Bioengineering*, pages 221–227, 2009.

[MO04] S. C. Madeira and A.L. Oliveira. Biclustering algorithms for biological data analysis: A survey. *IEEE Transactions on Computational Biology and Bioinformatics*, 1(1):24–45, 2004.

[NWKN96] J. Neter, W. Wasserman, M. H. Kutner, and C. J. Nachtsheim. *Applied Linear Statistical Models*. Mc Graw Hill, New York, fourth edition, 1996.

[PBZea06] A. Preli'c, S. Bleuler, and P. Zimmermann et al. A systematic comparison and evaluation of biclustering methods for gene expression data. *Bioinformatics*, 22(9):11221129, 2006.

[PMJ65] M. H. Protter and C. B. Morrey-Jr. *Modern Mathematical Analysis*. Addison-Wesley, Massachusetts, second edition, 1965.

[RBB06] D. J. Reiss, N.S. Baliga, and R. Bonneau. Integrated biclustering of heterogeneous genome-wide datasets for the inference of global regulatory networks. *BMC Bioinformatics*, 7(280), 2006.

[Ren00] A. C. Rencher. *Linear Models in Statistics*. Wiley Inter-Science, San Francisco, 2000.

[Sea71] S. R. Searle. *Linear Models*. John Wiley, New York, 1971.

[Sea82] S. R. Searle. *Matrix Algebra Useful for Statistics*. John Wiley, New York, 1982.

[SG05] G. B. Santos and K. S. Guimarães. Analyzing the effect of prior knowledge in genetic regulatory network inference. *Pattern Recognition and Machine Intelligence (PREMI),LNCS*, 3776:611–616, 2005.

[SMDM03] Q. Sheng, Y. Moreau, and B. De-Moor. Biclustering microarray data by gibbs sampling. *Bioinformatics*, 19:ii196–ii205, 2003.

[SS63] R. R. Sokal and P. H. A. Sneath. *Principles of Numerical Taxonomy*. W. H. Freeman, San Francisco, 1963.

[SSK+04] R. Sharan, S. Suthram, R. M. Kelley, T. Kuhn, S. McCuine, P. Uetz, T. Sittler, R. M. Karp, and T. Ideker. Conserved patterns of protein interaction in multiple species. *PNAS*, 102(6):1974–1979, 2004.

[STG+01] E. Segal, B. Taskar, A. Gasch, N. Friedman, and D. Koller. Rich probabilistic models for gene expression. *Bioinformatics*, 17:S243–S252, 2001.

[TBKH05] H. L. Tuner, T. C. Bailey, W. J. Krzanowski, and C. A Hemingway. Biclustering models for structured microarray data. *IEEE Transactions on Computational Biology and Bioinformatics.*, 2(4):316 – 329, 2005.

[TZZR01] C. Tang, L. Zhang, I. Zhang, and M. Ramanathan. Interrelated two-way clustering: An unsupervised approach for gene expression data analysis. *Proc. Second IEEE Int'l Symp. Bioinformatics and Bioeng.*, pages 41–48, 2001.

[UW08] M.V. Uitert and L. Wessels. Biclustering sparse binary genomic data. *Journal of Computational Biology*, 15(10):1329–1345, 2008.

[WWYY02] H. Wang, W. Wang, J. Yang, and P.S. Yu. Clustering by pattern similarity in large data sets. *Proc. ACM SIGMOD Int'l Conf.Management of Data*, pages 394–405, 2002.

[YVK04] Y. Yamanishi, J.-P. Vert, and M. Kanehisa. Protein network inference from multiple genomic data: a supervised approach. *Bioinformatics*, 20(1):i363–i370, 2004.

[YVK05] Y. Yamanishi, J.-P. Vert, and M. Kanehisa. Supervised enzyme network inference from the integration of genomic data and chemical information. *Bioinformatics*, 21(1):i468–i477, 2005.

[YVNK03] Y. Yamanishi, J.-P. Vert, A. Nakaya, and M. Kanehisa. Extraction of correlated gene clusters from multiple genomic data by generalized kernel canonical analysis. *Bioinformatics*, 19(1):i323–i330, 2003.

[YWWY02] J. Yang, W. Wang, H. Wang, and P. Yu. δ-clusters: Capturing subspace correlation in a large data set. *Proc. 18th IEEE Int'l Conf. Data Eng.*, pages 517–528, 2002.