UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE
INSTITUTO METRÓPOLE DIGITAL
IMD0039 - ESTRUTURAS DE DADOS BÁSICAS II
IMD0040 - LINGUAGEM DE PROGRAMAÇÃO II

PROJETO 3ª UNIDADE - 2015.1
Versão 0.2

Análise de Expressão Gênica

1. Introdução

Através da análise do padrão de expressão individual de cada gene é possível agrupar amostras com um mesmo perfil gênico e calcular a distância de similaridade entre grupos de amostras com um mesmo perfil e assim torna possível identificar padrões de expressão associados a condições específicas (por exemplo resposta à quimioterapia ou indivíduos doentes versus sadios). Estudos de microarray e/ou de sequenciamento de segunda geração, possibilitam a quantificação simultânea da expressão de milhares de genes de uma determinado amostra biológica. As tecnicas de quantificação de expressão gênica são bastante utilizadas em experimentos genômicos de diversas espécies animais e vegetais, e têm sido gradativamente incorporados em diferentes áreas de pesquisa: como crescimento e metabolismo, resposta imune a doenças, reprodução e resposta a fatores de estresse não-infecciosos (restrição alimentar, exposição a elementos tóxicos e outras condições ambientais desfavoráveis), bem como melhoramento genético animal. A realização dos experimentos com expressão gênica, desde a coleta das amostras, até a obtenção dos valores de expressão, envolve uma série de procedimentos laboratoriais de alta complexidade, que frequentemente introduzem variações adicionais aos resultados obtidos. Desta forma a condução de ensaios estatísticos rigorosos, se faz necessário para a eliminação de passives viés.

O objetivo deste projeto é identificar uma assinatura estável de expressão gênica capaz de predizer respostas a tratamentos através da análise combinada de dados de expressão gênica e dados clínicos.

Uma assinatura de expressão gênica é obtida após comparar valores representativos de expressão gênicas entre grupos de amostras (por exemplo grupo de indivíduos sadios versus grupo de indivíduos doentes). Ao se escolher um teste estatístico e estabelecer um critério de corte (baseado no p-valor <= 0.05 por exemplo), consegue-se identificar genes diferencialmente expressos. Uma assinatura gênica consiste no conjunto de genes diferencialmente expressos entre os grupos de amostras. Genes que não possuem significância estátistica (p-valor >= 0.05) entre os grupos analisados são genes não informativos e não devem compor a assinatura. Se o método estatístico e o critério de corte

forem apropriados, uma analise de clusterização deve separar perfeitamente os dois grupos de amostras originalmente testados estatisticamente, a não ser que uma das amostras sejam um falso-positivo (uma amostra classificada erroneamente, muito comum de acontecer em prontuários clínicos). Por fim, uma assinatura de expressão gênica é validada e considerada estável após a utilização de uma técnica de validação cruzada chamada de leave-one-out. O método consiste na remoção de uma amostra qualquer e reanálise de clusterização, se a estrutura de clusterização se mantém, a amostra escolhida retorna ao dado original e uma nova amostra é removida e uma nova clusterização é realizada, até que todas as amostras sejam testadas. Uma assinatura gênica é considerada estável se após as N configurações (remoções de amostras, uma a uma com reposição), a clusterização inicial das amostras se mantém.

2. Descrição Geral

O trabalho consiste em desenvolver um software para a análise de expressão gênica e obtenção de uma assinatura estável de expressão a partir de dados de obtidos de experimentos de microarray e/ou de sequenciamento de segunda geração.

A abordagem proposta conta com a varredura de vasta quantidade de dados de expressão gênica disponíveis, possibilitando a identificação de uma assinatura de expressão gênica confiável. Assim, esta estratégia deve identificar genes com padrões de expressão diferencial entre os grupos de amostras e determinar uma assinatura proveniente do estudo individual de cada gene. Seu grupo deverá propor ainda métodos para melhorar a precisão e a confiança da assinatura resultante.

3. Objetivos específicos:

O projeto apresenta os seguinte objetivos específicos que deverão ser contemplados pelo seu programa:

1. Propor e aplicar uma estratégia para eliminar genes não informativos e obtenção da assinatura genica.
2. Aplicar um algoritmo de agrupamento (Clustering) e obter uma matriz de distâncias.
3. Dada a matriz de distancias obtidas no item 2, desenhar um dendograma em formato de árvores para representar as distancias.
4. Aplicar técnica de validação cruzada: leave-one-out, para checar a confiabilidade da assinatura gerada.

De modo a atender a esses objetivos, detalhamos abaixo os diversos requisitos que devem ser considerado.

3.1. Entrada de dados

A entrada de dados de seu software deverá ser feita através de arquivos de texto, considerando as seguintes informações:

- Arquivo de texto em formato tabular onde cada coluna representa uma amostra e cada linha representa um gene, onde os valores das linhas são representativos do grau de expressão de cada gene em cada amostra.

- Arquivo de texto informando o grupo que cada amostra pertence. Cada linha contém 2 elementos separados por espaço, onde o primeiro valor representa o identificador da amostra, e o segundo valor representa o identificador do grupo ao qual a amostra pertence.

A expressão de um gene em uma determinada amostra é representado por um número em ponto flutuante (double).

3.2. Eliminação de genes não informativos e obtenção de assinatura gênica

De maneira a eliminar genes não informativos e obter a assinatura gênica emprega-se testes estatísticos.

[1]Estatística (disciplina): É um conjunto de métodos para planejar experimento, obter dados e organizá-los, analisá-los, interpretá-los e deles extrair conclusões. A estatística pode ser dividida em duas partes:

- *Estatística Descritiva: Que trata da descrição tabular, gráfica e paramétrica dos dados provenientes de populações e amostras.*
- *Estatística Inferencial: Parte dos resultados obtidos nas amostras e faz inferências para a população. Estuda a estimação e os testes sobre os parâmetros populacionais.*

População: Em termos estatísticos define-se uma população como sendo um conjunto de informações que tenham, pelo menos, uma característica em comum.

Amostra: É um subconjunto da população.

Parâmetro: É uma medida numérica que descreve uma característica de uma população. Os símbolos são apresentados por letras gregas. Ex.: μ=Média populacional, σ^2=Variância populacional; σ = Desvio Padrão populacional.

---

[1] Trecho extraído do mini-curso “Ferramentas estatísticas para análise experimental de algoritmos”, de autoria de Lívia Silva, Elizabeth Goldbarg e Marco Golgbarg, apresentado no EPOCA 2008.

Estatística (métrica): É uma medida obtida através dos elementos das amostras. Ex.: Média da Amostra = X; Variância da Amostra = S^2; Desvio padrão da Amostra = S;

Teste de Hipótese: É um procedimento que conduz a uma decisão acerca das hipóteses com base numa amostra. Quando queremos avaliar um parâmetro populacional, sobre o qual não possuímos nenhuma informação com respeito a seu valor, não resta outra alternativa a não ser estimá-lo através do intervalo de confiança [Hines, William et al, 2006]. No entanto, se tivermos alguma informação com respeito ao valor do parâmetro que desejamos avaliar, podemos testar esta informação no sentido de aceitá-la como verdadeira ou rejeitá-la. Chamaremos de "Hipótese Nula" e indicaremos por H0 a informação a respeito do valor do parâmetro que queremos avaliar. Chamaremos de "Hipótese Alternativa" e indicaremos por H_a a afirmação a respeito do valor do parâmetro que aceitaremos como verdadeira caso H0 seja rejeitada.

Existem varias maneiras de se eliminar genes não informativos e de identificar assinaturas gênicas, alguns exemplos de testes estatísticos que podem ser usados para tal:

1- Variância: pode-se utilizar um teste de variância simples, sem considerar os grupos de amostras, identificando os tops 50 genes mais variáveis. Desta forma os top 50 genes formariam a assinatura gênica e os restante são não informativos.

2- Teste t de Student: análise paramétrica, teste de hipótese que usa conceitos estatísticos para rejeitar ou não uma hipótese nula. Os genes que passarem no critério de corte: p-valor <= 0,05, formam a assinatura gênica e os restante são não informativos.

3- Teste wilcoxon: análise não paramétrica, teste de hipótese estatística usada quando se comparam duas amostras relacionadas. Os genes que passarem no critério de corte: p-valor <= 0,05, formam a assinatura gênica e os restante são não informativos.

Ps. Existem outras formas, até mais robustas de obtenção de assinaturas gênicas, como parte da resolução desta tarefa os alunos devem ainda propor métodos para melhorar a precisão e a confiança da assinatura resultante.

Incentivamos que seu grupo utilize uma biblioteca externa para os cálculos estatísticos. O objetivo aqui é que seu grupo não perca tempo implementando testes estatísticos, mas utilize uma das bibliotecas disponíveis na Internet. A escolha da biblioteca fica a critério do grupo. Uma simples busca na Internet permite encontrar um número de bibliotecas, tais como Apache Commons Math (http://commons.apache.org/proper/commons-math/). O site http://math.nist.gov/javanumerics/ apresenta uma lista de pacotes estatísticos que podem ser utilizados. Seu grupo deverá pesquisar e escolher pacotes por conta própria.

3.3. Algoritmo de agrupamento e matriz de distância

Um algoritmo de agrupamento tem como objetivo colocar objetos similares em um mesmo grupo, enquanto que objetos não similares são colocados em grupos diferentes. Para este trabalho, seu grupo deverá escolher ao menos um algoritmo de agrupamento para implementar em seu programa. Entretanto, a implementação de múltiplos algoritmos (assim como permitir ao usuário escolher qual algoritmo utilizar) valem pontos extras na avaliação.
Em seguida, apresentamos diversos links que podem ser usados como ponto de partida em sua procura.

- http://fbarth.net.br/materiais/docs/am/agrupamento.pdf
- http://www.fsma.edu.br/si/edicao4/FSMA_SI_2009_2_Tutorial.pdf
- http://www.maxwell.vrac.puc-rio.br/14382/14382_4.PDF
- http://www.angelfire.com/ab/cias/chem9.html

3.4. Detalhamento sobre o desenho do dendograma

Um dendograma é um tipo específico de diagrama que representa a organização de determinados fatores e variáveis. Normalmente empregado em métodos quantitativos que levam a agrupamentos e à sua ordenação hierárquica ascendente, um dendograma ilustra o arranjo de agrupamentos derivado da aplicação de um algoritmo de clustering. Em termos gráficos se assemelha aos ramos de uma árvore que se vão dividindo noutros sucessivamente. A Figura 1 apresenta um exemplo de dendograma.

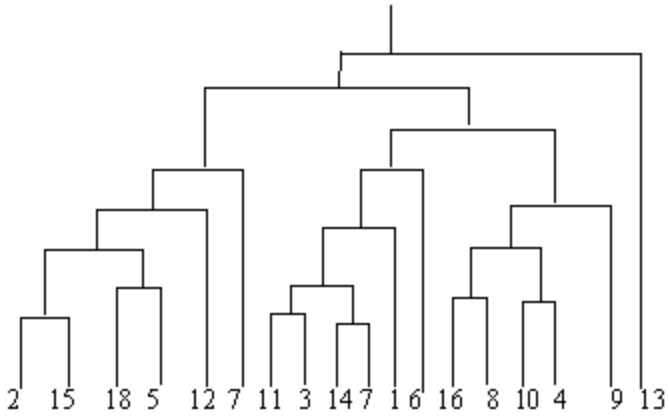

Figura 1: Exemplo de dendograma.
Fonte: http://www.angelfire.com/ab/cias/chem9.html

Seu dendograma deverá ser representado através de uma árvore, onde nós folha representam indivíduos, e nós internos representam agrupamentos com seu respectivo valor. Para o cálculo dos valores dos nós internos, pode-se utilizar a mediana dos valores dos nós filhos.

3.5. Validação cruzada

O conceito central das técnicas de validação cruzada é o particionamento do conjunto de dados em subconjuntos mutualmente exclusivos, e posteriormente, utilizasse alguns destes subconjuntos para a estimação dos parâmetros do modelo e o restante dos subconjuntos são empregados na validação do modelo. Existem diversas formas de realizar o particionamento dos dados, sendo as três mais utilizadas: o método holdout, o k-fold e o leave-one-out.

O método *leave-one-out* é um caso específico do *k-fold*, com *k* igual ao número total de dados *N*. Nesta abordagem são realizados *N* cálculos de erro, um para cada amostra, sendo que cada amostra deve ser removida uma a uma com reposição (onde a amostra removida deve ser recolocada em sua posição original para que a próxima amostra possa ser removida) e recalculado a matriz de distância a cada uma das N amostras. Uma assinatura gênica é considerada estável se após as N configurações (remoções de amostras, uma a uma com reposição), a raiz principal separa os dois grupos de amostras.

Seu programa deverá implementar a técnica leave-one-out de modo a fazer a validação cruzada dos dados analisados.

3.6. Salvar em arquivo

Seu programa deverá incluir mecanismos que permitam salvar os dados processados em arquivos.

Inclua mecanismos que permitam salvar em arquivo:
1- Uma lista com os genes que compõem a assinatura genica.
2- A matriz de distância resultante da análise de Clustering.
3- Uma árvore representando o dendograma.

4. Interface com o usuário

Seu programa deverá possuir uma interface de interação com o usuário.

Sua interface de interação deverá permitir ao usuário executar as seguintes operações.
- abrir arquivo de entrada com amostras;
- abrir arquivo de entrada com grupo de cada amostra;
- Rodar o algoritmo de agrupamento;
- Rodar o processo de validação cruzada;
- Alterar critério de corte de p-valores para a obtenção da assinatura genica;
- Alterar teste estatísticos para a obtenção da assinatura genica (extra);

- Alterar método de construção da tabela de distâncias (extra);
- Exibir a lista de genes, assinatura genica;
- Exibir o dendograma gerado para o usuário;
- Salvar em arquivo;

Esta interface poderá ser feita de modo textual ou gráfica, a escolha do grupo. Independente da forma de interação escolhida, seu grupo deverá aplicar um padrão de projeto adequado para a definição das classes que implementem sua interface.

No caso de interfaces de texto, seu programa deverá receber comandos textuais do usuário. Para cada ação que pode ser realizada, um determinado comando precisa ser definido. Além disso, deverá existir um comando de ajuda, que exiba a lista de comandos para o usuário.

No caso de interface gráfica, seu programa deverá definir botões ou menus para as diversas ações que podem ser executadas por um usuário.

Em ambos os casos, textual ou gráfica, seu grupo deverá definir uma maneira de apresentar o dendograma gerado para o usuário.

5. Manual do usuário

É importante que o grupo prepare um manual de operações do programa desenvolvido. Este manual será utilizado pelo usuário final do programa para rodar seus experimentos.

6. Entrega e Avaliação

Seu grupo deverá submeter um arquivo compactado contendo os seguintes entregáveis:

- Código fonte do software desenvolvido, incluindo um documento README.TXT contendo instruções de como se pode compilar o código fonte.
- Manual de usuário mostrando como instalar, executar e utilizar seu programa.
- Relatório técnico.

O relatório técnico deverá conter pelo menos as seguintes seções:
- Introdução
- Descrição do problema abordado
- Descrição geral das estruturas de dados utilizadas
- Descrição da abordagem de solução do problema (algoritmos)
- Detalhes de projeto OO (diagramas de classes destacando as decisões de projetos tomada, bem como identificando padrões de projeto aplicados)

- Explicação detalhada dos algoritmos utilizados (com pseudocódigo e análise de complexidade)
- Conclusão
- Referências

O relatório deverá ser feito seguindo o template da Sociedade Brasileira de Computação (SBC) que pode ser encontrado no seguinte endereço:

http://www.sbc.org.br/index.php?option=com_jdownloads&Itemid=195&task=view.download&cid=38

6.1. Avaliação de EDB2

A avaliação da disciplina EDB2 considerará os seguintes itens:

- Funcionamento do software da maneira solicitada, atendendo a todos os requisitos.

- Modelagem adequada do problema do ponto de vista algorítmico, o que engloba:
  - Escolha das estruturas de dados estudadas em EDB2, de forma a maximizar a eficiência dos algoritmos (por exemplo, como será modelado o dendograma? como construir o dendograma de forma eficiente?).
  - Definição de método(s) para eliminação de genes não informativos.
  - Escolha de métodos(s) para cálculo de matriz de distâncias entre amostras (pessoas).

- Implementação correta do software, nos seguintes aspectos:
  - Estruturas de dados empregadas.
  - Leitura dos dados de entrada.
  - Algoritmo de eliminação de genes não informativos.
  - Algoritmo de cálculo de matriz de distâncias.
  - Algoritmo de clusterização para geração do dendograma.
  - Algoritmo de validação cruzada.
  - Saída de dados, com correta identificação de uma assinatura estável de expressão gênica.

- Completude do relatório técnico no que se refere aos itens solicitados na seção 6, incluindo a adequação ao modelo proposto pela SBC. O texto do relatório técnico deverá ser coerente, coeso e objetivo, contemplando as informações suficientes e necessárias para o entendimento do software desenvolvido.

- Apresentação.

Acarretarão diminuição na nota:

- Presença de erros de compilação e/ou execução.
- Falta de análise de complexidade dos algoritmos implementados.
- Falta de documentação do programa em JavaDoc.
- Entrega incompleta.
- Mal uso da norma culta da Língua Portuguesa.

6.2 Avaliação de LP2

A avaliação para a disciplina de LP2 levará em conta os seguintes aspectos:

- Acompanhamento do projeto
- Qualidade do projeto desenvolvido
- Relatório técnico
- Apresentação

Cada um desses elementos serão detalhados a seguir:

6.2.1. Acompanhamento do projeto

Ao longo do andamento do projeto serão definidas datas de checkpoint onde cada grupo deverá fazer uma breve apresentação de seu progresso. Cada checkpoint terá um entregável esperado.

Checkpoint 1: Tarefas a serem desenvolvidas pelo grupo para o andamento do projeto com alocação de responsáveis por cada tarefa.	Aula 28 (13/05/2015)
Checkpoint 2: Projeto OO (diagrama de classes) inicial da solução.	Aula 29 (25/05/2015)
Checkpoint 3: Implementação - fase 1	Aula 31 (01/06/2015)
Checkpoint 4: Implementação - fase 2	Aula 33 (08/06/2015)

6.2.2. Qualidade do projeto desenvolvido

Neste aspecto serão considerados os diversos conceitos abordados ao longo do semestre. Cada critério abaixo tem uma pontuação máxima de 10 pontos.

Critérios a serem considerados

Qualidade de Projeto OO	
Modelagem em Classes	
Herança	
Polimorfismo	
Coesão e Acoplamento	
Tratamento de exceções	
Documentação (Javadoc)	
Apresentação	
Padrão de projeto	
Uso adequado de biblioteca externa	

6.2.3. Relatório técnico

Serão considerados os mesmos critérios relacionados à completude do relatório técnico estabelecidos para a avaliação de EDB2.

6.2.4. Apresentação

Cada grupo terá 20 minutos para fazer sua apresentação. Nesta apresentação, os grupos deverão demonstrar o funcionamento do programa desenvolvido. Além disso, deverão estar aptos a responder questões sobre o desenvolvimento do projeto.

Importante: Não será dada uma única nota ao grupo. Cada componente do grupo receberá uma nota de acordo com seu desempenho durante a apresentação.

O programa será avaliado como um todo, ou seja, os requisitos não receberão pontuações individualmente. Dessa forma, a falta de um ou mais requisitos acarretará na perda de pontos, que poderá ser compensada (não totalmente, claro) através de outros componentes bem desenvolvidos.

Componentes adicionais serão muito bem vistos, desde que implementados de maneira racional. Lembre-se de usar o bom senso para não transformar criatividade em bagunça.

7. Dúvidas e Dicas

Se durante o desenvolvimento do projeto tiver alguma dúvida sobre a tarefa solicitada tente duas possibilidades. Primeiro leia este documento, ou novamente ou pela primeira vez. Segundo, procure na Internet por mais informações. Por último, pergunte ao professor. Se a sua dúvida for interessante ela e a resposta serão acrescentadas neste documento.

8. Autoria, Política de Colaboração, Plágio e Duplicação de Material

Este trabalho poderá ser desenvolvido em grupos com três ou, excepcionalmente, quatro integrantes. Eventualmente, alguns grupos poderão ser convocados para uma entrevista. O objetivo de tal entrevista é comprovar a verdadeira autoria do código entregue. Assim, qualquer um dos componentes do grupo deve ser capaz de explicar qualquer trecho do código do projeto.

O trabalho em cooperação entre alunos da turma é estimulado. Porém, esta interação não deve ser entendida como permissão para utilização de código ou parte de código de outras equipes, o que pode caracterizar a situação de plágio. Trabalhos plagiados receberão nota ZERO automaticamente, independente de quem seja o verdadeiro autor dos trabalhos infratores.

- Um dos motivos mais comuns para problemas de plágio em trabalhos de programação é deixar para fazer o trabalho de última hora. Evite isso, e tenha certeza de descobrir o que você tem que fazer (que não significa necessariamente como fazer) o mais cedo o possível. Em seguida, decida o que você precisará fazer para completar o trabalho. Isto provavelmente envolverá alguma leitura e prática de programação. Se estiver em dúvida sobre o que foi pedido pelo trabalho, pergunte ao professor da disciplina.
- Outra razão muito comum é trabalhar em conjunto com outros alunos da disciplina. Não faça trabalhos de programação em conjunto, ou seja, não utilizem um único PC, ou sentem lado a lado, principalmente, digitando código ao mesmo tempo. Discutam as diversas partes do trabalho, mas não submetam o mesmo código.
- Não é aceitável a submissão de código com diferenças em comentários e nomes de variáveis, por exemplo. É muito fácil para nós detectar quando isso for feito, e verificaremos esse caso.
- Nunca deixe outra pessoa ter uma cópia de seu código, não importando o quão desesperado eles possam estar. Sempre aconselhe alguém nesta situação a buscar ajudar com o professor da disciplina.

Versão 0.2 - 13/05/2015 - 10:30 - Acrescentado algumas informações na seção 3. Correção na seção 6. Adicionado critérios de correção EDB2 e nova seção (seção 8).