BRÁULIO ROBERTO GONÇALVES MARINHO COUTO

USO DA ÁLGEBRA LINEAR PARA ANÁLISE DE SIMILARIDADES E EXTRAÇÃO DE PADRÕES EM SEQUÊNCIAS PROTÉICAS

Belo Horizonte

Novembro de 2010

UNIVERSIDADE FEDERAL DE MINAS GERAIS

PROGRAMA DE PÓS-GRADUAÇÃO EM BIOINFORMÁTICA

BRÁULIO ROBERTO GONÇALVES MARINHO COUTO

USO DA ÁLGEBRA LINEAR PARA ANÁLISE DE SIMILARIDADES E EXTRAÇÃO DE PADRÕES EM SEQUÊNCIAS PROTÉICAS

Tese apresentada ao Programa de Pós-graduação em Bioinformática da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do título de Doutor em Bioinformática.

Orientador — Prof. Dr. Marcos Augusto dos Santos

Departamento de Ciência da Computação, Instituto de Ciências Exatas – ICEx, Universidade Federal de Minas Gerais – UFMG, Belo Horizonte – MG.

Co-Orientador — Prof. Dr. Marcelo Matos Santoro

Laboratório Marcos Luiz dos Mares-Guia de Enzimologia e Físico-Química de Proteínas, Departamento de Bioquímica-Imunologia, Instituto de Ciências Biológicas – ICB, Universidade Federal de Minas Gerais – UFMG, Belo Horizonte – MG.

Universidade Federal de Minas Gerais
Instituto de Ciências Biológicas
Programa de Doutorado em Bioinformática
Avenida Presidente Antônio Carlos, 6627 – Pampulha
31270-901 - Belo Horizonte – MG
Página Eletrônica: www.pgbioinfo.icb.ufmg.br
Endereço eletrônico: bioinfo@icb.ufmg.br – Telefax: 55 31 3409-2554

ATA DA DEFESA DA TESE DE DOUTORADO DE Bráulio Roberto Gonçalves Marinho Couto. Aos vinte e três dias do mês de novembro de 2010 às 09h00 horas, reuniu-se no Instituto de Ciências Biológicas da Universidade Federal de Minas Gerais a Comissão Examinadora da tese de doutorado, indicada *ad referundum* do Colegiado do Programa, para julgar, em exame final, o trabalho intitulado "Uso da álgebra linear para análise de similaridades e extração de padrões em seqüências protéicas", requisito final para a obtenção do grau de Doutor em Ciências, Área de Concentração: Bioinformática. Abrindo a sessão o Presidente da Comissão, Prof. Dr. Marcos Augusto dos Santos da Universidade Federal de Minas Gerais, após dar a conhecer aos presentes o teor das Normas Regulamentares do Trabalho Final, passou a palavra ao candidato para apresentação de seu trabalho. Seguiu-se a argüição pelos examinadores, com a respectiva defesa do candidato. Logo após a Comissão se reuniu sem a presença do candidato e do público para julgamento e expedição do resultado final. Foram atribuídas as seguintes indicações: Prof. Dr. PhD. Mohammed J. Zaki do Rensselaer Polytechnic Institute, aprovado; Prof. Dr. Carlos H. da Silveira da Universidade Federal de Itajubá, aprovado; Prof. Dr. Frederico Ferreira Campos Filho da Universidade Federal de Minas Gerais, aprovado; Prof. Dr. José Miguel Ortega da Universidade Federal de Minas Gerais, aprovado; Prof. Dr. Marcelo Matos Santoro, coorientador, da Universidade Federal de Minas Gerais, aprovado; Prof. Dr. Marcos Augusto dos Santos, orientador, da Universidade Federal de Minas Gerais, aprovado. Pelas indicações o candidato foi considerado APROVADO. O resultado final foi comunicado publicamente ao candidato pelo Presidente da Comissão. Nada mais havendo a tratar o Presidente da Comissão encerrou a reunião e lavrou a presente ata que será assinada por todos os membros participantes da Comissão Examinadora. Belo Horizonte, 23 de novembro de 2010.

Prof. PhD. Mohammed J. Zaki – (Rensselaer Polytechnic Institute)

Prof. Dr. Carlos H. da Silveira – (Universidade Federal de Itajubá)

Prof. Dr. Frederico Ferreira Campos Filho (UFMG)

Prof. Dr. José Miguel Ortega – UFMG

Prof. Dr. Marcelo Matos Santoro - coorientador - UFMG

Prof. Dr. Marcos Augusto dos Santos – orientador – UFMG

Prof. Dr. Wagner Meira Júnior
Coordenador do Programa de Doutorado em Bioinformática - UFMG / Portaria DAP nº 2065

"*Since the measuring device has been constructed by the observer … we have to remember that what we observe is not the nature itself but nature exposed to our method of questioning*."

Physics and Philosophy [1958]
Wener Karl Heisenberg, 1901-1976.

Agradecimentos

Finalmente, estou terminando o meu doutorado. E tenho muito a agradecer! A minha jornada vem de longa data... para falar a verdade, preciso começar pela minha iniciação científica. Aos professores Eucler Paniago, Sandra Carvalho e Frederico Campos, muito obrigado por toda a orientação recebida. Ao amigo Hélio Duarte, pelo companheirismo. As decisões da minha vida profissional foram sustentadas, definidas conforme os ensinamentos que recebi de vocês durante a minha iniciação científica. Em especial, agradeço ao professor Frederico Campos: são mais de vinte e cinco anos de orientação!

Viajando no tempo, preciso agradecer aos companheiros da minha vida “hospital e controle de infecções”. Novamente, é longa a convivência... 20 anos para ser mais exato! Agradeço à amizade e ao apoio durante o tempo em que fiz especialização, mestrado e agora doutorado. Quanta coisa prometi fazer após o doutorado: reconstruir o SACIH é o mínimo! Carlos, Silma, Edna, Estevão, Raquel, José Antônio, Mônica, Hoberdan, Izabella, Karina, Fernanda, Nilza, Rose, Glorinha, Malu, Áurea, Isabel, Simone, Rafaela, Mariana, Clara, Jussara, Roberta, Helena, Hérika, Marilaine.

Quanto ao UNI-BH, são mais de 10 anos, dia-a-dia, vida pessoal e profissional no mesmo ambiente! Obrigado professoras Raquel Parreiras e Sueli Baliza! Muito obrigado, Miriam, Magali, Diva, Ana Paula e Sandra: conviver com vocês é um privilégio, uma honra.

Na FASEH, são mais de cinco anos de trabalho. O apoio incondicional do professor Assuero é outro privilégio que disponho.

Voltando ao doutorado, preciso agradecer muito ao amigo, professor e orientador Marcos Augusto dos Santos. Ele é o responsável por me trazer para esta coisa incrível chamada Bioinformática e por me mostrar toda a poesia, o poder da SVD. Eu realmente fui transformado ao longo deste aprendizado. Muito obrigado por esta oportunidade! Por meio da Bioinformática eu consegui me encontrar profissionalmente. Finalmente entendi porque estudei Engenharia Química, Estatística e Ciência da Computação.

Não posso deixar de falar do professor Marcelo Santoro. Pela sabedoria, pelo exemplo, pela orientação, pelas aulas maravilhosas e discussões durante a disciplina de Tópicos Especiais em Bioinformática I, II e III. Muito obrigado!

Pessoalmente, agradeço a Deus por ter me dado saúde. E, depois de estudar Bioquímica, Genética, Genoma, Proteoma e Transcriptoma posso afirmar categoricamente: não há como afirmar que Ele não exista. A complexidade da vida, a simplicidade e universalidade do código genético, não dá para aceitar que tudo isto seja resultado de um processo estocástico de longo prazo. E as leis da Física, as estrelas, o universo. Haja aleatoriedade! Em suma, obrigado Senhor, o maior programador, criador do programa principal e das subrotinas mais importantes da vida: o genoma e os genes.

Aos meus pais, Roberto e Dirce, aos meus irmãos Bruno, Robson e Cristiano: obrigado por acreditarem em mim!

À **minha filha Luiza**, adorável, carinhosa, “cuidadeira de quem ama”, admirável, amorosa, dedicada ao extremo a tudo que faz. Muito obrigado: acompanhar o seu crescimento é uma dádiva. Conviver com você me faz sentir eterno!

Gostaria de terminar estes agradecimentos enaltecendo a minha **mulher Ana Paula**. É a pessoa mais incrível que conheço! A sua inteligência, aliada à sua sensibilidade a tornaram especial. Não conheço ninguém tão brilhante, com tamanha capacidade de síntese e, ao mesmo tempo, tamanha dedicação aos seus. Haja entropia... e isto é ótimo! Preciso te agradecer por ter me incluído na sua vida. Muito obrigado meu amor.

Belo Horizonte, novembro de 2010.

Bráulio RGM Couto

Sumário

Lista de figuras

Lista de abreviaturas

SVD: decomposição em valores singulares;

BLAST: the basic local alignment search tool;

FASTA = Fast Alignment Search Tool;

PAM: Point Accepted Mutation matrix;

BLOSUM: BLOcks of Amino Acid SUbstitution Matrix.

Resumo

Extrair padrões de dados de seqüências de proteínas é um dos desafios da Biologia Computacional. Neste trabalho, é apresentada uma metodologia que usa técnicas de Álgebra Linear, Estatística e Otimização para a análise de sequências primárias de proteínas. Inicialmente, cada sequência é transformada num vetor de frequências de peptídeos de tamanho "*p*", considerando todas as combinações possíveis de aminoácidos para formarem um *p*-peptídeo. Com 20 aminoácidos, o modelo de espaço vetorial é formado por vetores de tamanho 20^p. Para avaliar a validade biológica do método, medidas de similaridade da SVD, distância Euclidiana e cosseno, foram comparadas com medidas de similaridade usadas por um programa de alinhamento de sequências (BLAST). A distância euclidiana foi negativamente correlacionada com *bit score* (r>-0,6) e positivamente correlacionado com *E value* (r>+0,7). Já o cosseno apresentou correlação negativa com *E value* (r>-0,7) e correlação positiva com *bit score* (r>+0,8). Foi obtida também uma estimava para o grau de concordância entre cosseno e distância Euclidiana com o resultado gerado por um programa padrão de alinhamento de sequências, quando da classificação de uma sequência desconhecida. Quanto à interpretação biológica para a SVD, pode-se afirmar que os valores singulares visualizados como *scree plots* revelam os principais componentes, o número de processos escondidos num banco de dados de sequências protéicas. Ao se aliar a SVD com técnicas de otimização, foi possível a visualização multidimensional de genomas e de outros dados multivariados em 2D ou 3D. Já a combinação de modelos de regressão logística com SVD permitiu a seleção de atributos importantes para a classificação de seqüências protéicas. A principal contribuição desta tese refere-se à validade biológica do uso da decomposição em valores singulares (SVD) para análise de similaridade e extração de padrões em sequências protéicas. Antes da realização deste trabalho, persistiam muitas dúvidas em relação à significância biológica de se considerar uma proteína como um vetor no espaço multidimensional e, principalmente, quanto à validade da análise de similaridade por meio de técnicas de Álgebra Linear. Mesmo sem se trabalhar com matrizes de substituição nem com algoritmos de alinhamentos de sequências, foram obtidos resultados biologicamente válidos. Descrever uma proteína na forma de um vetor permite que não só a SVD possa ser usada na sua análise, mas todas as outras ferramentas utilizadas para a manipulação de vetores e matrizes, da Álgebra Linear, Física, Estatística, Geometria, Computação, também poderão ser usadas na busca por similaridades e na extração de padrões em sequências protéicas.

Abstract

Extracting patterns from protein sequence data is one of the challenges of Computational Biology. Here we use linear algebra methods and logistic regression models to analyze sequences without the requirement of multiples alignments. Firstly, we consider a biomolecular sequence as a complex written language that is recoded as *p*-peptide frequency vector using all possible overlapping *p*-peptides window. With 20 amino acids is generated a 20^p high-dimensional vector, where *p* is the word-size. After that, singular value decomposition (SVD) and/or logistic regression models are applied on data to extract patterns or to allow visualizing of high dimensional data. Spearman correlation (r) was used to evaluate the association between statistics used by BLAST and similarity metrics used by SVD. Euclidean distance was negatively correlated with bit score (r>-0.6) and positively correlated with E value (r>+0.7). Cosine had negative correlation with E value (r>-0.7) and positive correlation with bit score (r>+0.8). In addition, we compared edit distance between each pair of sequences with respective cosines and Euclidean distances from SVD. Correlation between cosine and edit distance was -0.32 ($P < 0.01$) and between Euclidean distance and edit distance was +0.70 ($P < 0.01$). Besides, the ability of SVD in classifying sequences according to their categories was evaluated. With a 3-peptide frequency matrix, all queries were correctly classified (accuracy = 100%). We proposed a biological significance of the SVD: the singular value spectrum visualized as scree plots unreveals the main components, the process that exists hidden in the protein database. A feature selection for protein sequence classification was made by using logistic regression models and SVD. In addition to the feature selection, combining logistic regression models with SVD allowed better classification of unknown sequences than using SVD alone. We also presented a method that utilizes information from known protein databases to build logistic regression models that allow prediction of a new amino acids sequence. We successfully tested the method in ten instances, which generated models for predicting insulin, globin, keratin, cytochrome, albumin, collagen, fibrinogen and proteins related with cystic fibrosis, Alzheimer disease and schizophrenia. SVD, followed by optimization allows visualization of high dimensional genomes by mapping multivariate data from their high dimensional representation into 2D or 3D space. All results found in this work and the characteristics described are important because SVD can be a solution for the potential problems with alignment algorithms and can be a substitute for those methods, for example, in whole genome analysis.

Introdução

Muitas ferramentas de Bioinformática têm como objetivo a detecção de padrões em sequências de proteínas ou de DNA, por meio de pesquisa de similaridades (Figura 1). Estes padrões, quando detectados, podem estar associados com a função ou a estabilidade estrutural da proteína, podem trazer informações sobre uma família de genes ou serem usados para descrever o relacionamento evolutivo de um grupo de sequências (GIBAS e JAMBECK, 2001; HUNTER, 1993). Atualmente, a pesquisa de similaridade entre sequências é o método mais poderoso para predizer a função de um gene desconhecido, sendo a principal técnica usada na Biologia Computacional (HOLM e SANDER, 1998; LIU *et al.*, 2008; PERTSEMLIDIS e FONDON III, 2001).

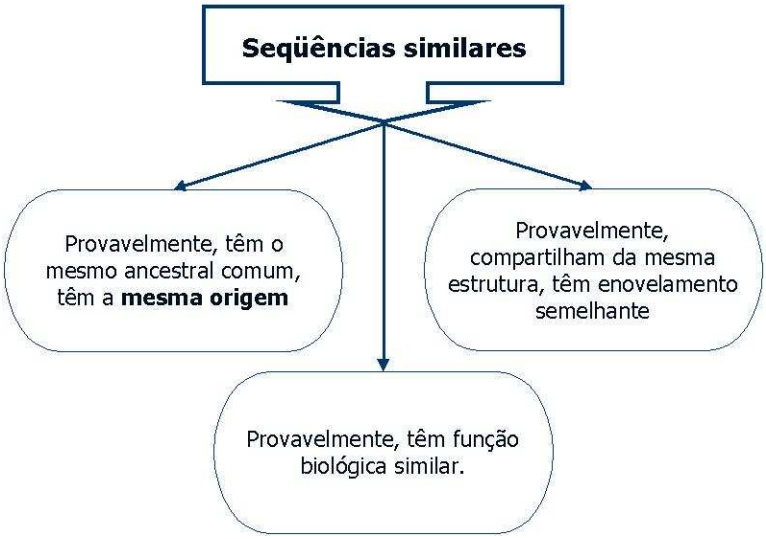


Figura 1: Importância da análise de similaridades.

A maioria das análises moleculares, inclusive a inferência filogenética, é baseada em alinhamentos múltiplos de sequências, com comparações caractere-a-caractere (KRAWETZ e WOMBLE, 2003). Os alinhamentos par-a-par são realizados usando algoritmos altamente sensíveis (mas computacionalmente intensivos) como Needleman & Wunsch (NEEDLEMAN, 1970) e Smith-Waterman (1981) ou suas aproximações, baseadas em heurísticas mais rápidas e menos sensíveis (HOCHREITER *et al.*, 2006): FASTA (PEARSON e LIPMAN, 1988) e BLAST, que se tornou, isoladamente, a peça mais importante no campo da Bioinformática (KORF *et al.*, 2003). Até o final de 2008, de acordo

com o Google (www.scholar.google.com), o artigo original descrevendo o programa BLAST (ALTSCHUL *et al.*, 1990) já foi citado 22.350 vezes.

Apesar da predominância das análises usando alinhamento de sequências, a intenção deste trabalho é usar uma alternativa para a detecção de padrões em proteínas. Foram implementados e validados algoritmos baseados na decomposição em valores singulares – SVD (DEERWESTER *et al.*, 1990), um método que não requer alinhamentos múltiplos de sequências e permite a estimação da relação entre biomoléculas. O objetivo é apresentar uma metodologia de análise que seja computacionalmente eficiente e biologicamente válida, possibilitando a representação de proteínas, a extração de padrões, a caracterização e a classificação de genes usando representação vetorial de sequências. Além de usar Álgebra Linear, por meio da representação vetorial de proteínas e aplicação da SVD, modelos de regressão logística e técnicas de otimização também foram utilizadas em algumas etapas do trabalho.

Justificativa

Avaliar o quanto duas sequências protéicas são similares é uma questão complexa. O primeiro problema refere-se aos diferentes termos (mal) usados quando esta questão é analisada: identidade, similaridade e homologia.

Identidade de sequências existe quando exatamente o mesmo aminoácido ocorre nas mesmas posições das duas sequências. Similaridade leva em conta pareamentos aproximados e é significativa somente quando as substituições de aminoácidos ocorrem entre aqueles com alta probabilidade de serem intercambiáveis (em função de semelhanças de propriedades físico-químicas e da frequência de observação da própria substituição na natureza). A homologia ou o termo "sequências homólogas" é mais importante dos três já que refere-se ao fato das duas sequências compartilharem um ancestral comum no passado. Quando duas sequências são homólogas, além delas serem muito semelhantes uma com a outra, elas têm um relacionamento evolucionário, com ancestrais parecidos e derivando de um mesmo ancestral (PERTSEMLIDIS e FONDON III, 2001). É importante ressaltar que os dois primeiros termos, identidade e similaridade, são quantitativos e têm diferentes formas de serem validados. Já a homologia é qualitativa, sendo muito vulnerável a questionamento (KOSKI, 2001).

Os métodos padronizados para quantificar a similaridade entre duas proteínas utilizam alinhamentos globais ou locais entre suas sequências primárias. O objetivo é encontrar o alinhamento ótimo, quantificando-o por meio de alguma métrica. O algoritmo de Needleman & Wunsch (NEEDLEMAN, 1970) usa programação dinâmica para encontrar o alinhamento global ótimo, já o algoritmo de Smith-Waterman (1981) usa a mesma técnica computacional para achar o alinhamento ótimo local entre duas sequências. Já os programas FASTA (PEARSON e LIPMAN, 1988) e BLAST (ALTSCHUL *et al.*, 1990) utilizam heurísticas que não garantem, com certeza, o alinhamento local ótimo, mas são rápidos (quando comparados aos métodos exatos) e quase sempre atingem a otimalidade (KORF *et al.*, 2003; PERTSEMLIDIS e FONDON III, 2001). Na verdade, apesar das semelhanças de desempenho com o FASTA, atualmente o BLAST tornou-se ubíquo e *de facto* o programa padrão para a comparação de sequências (KANTOROVITZ, 2007; VINGA e ALMEIDA, 2003). O termo BLAST já se tornou até "verbo" dentro da Biologia Computacional.

Mesmo com bons resultados, métodos de detecção de padrões de sequências protéicas baseados em alinhamentos ainda apresentam problemas, tanto no algoritmo em si, devido à complexidade computacional e outras questões, quanto no sistema de escores usados para quantificar as possíveis substiuições de aminoácidos durante o alinhamento (Vinga e Almeida, 2003).

A complexidade envolvida no processo de estimação do relacionamento de várias biomoléculas de grande porte é enorme, já que depende do tamanho das sequências comparadas, o que dificulta a sua utilização nos grandes bancos de dados (HOCHREITER, 2007). Por exemplo, para classificar as proteínas identificadas num genoma recentemente sequenciado, os alinhamentos mais rápidos, feitos com o BLAST, levarão aproximadamente um mês para classificar os genes pertencentes a uma única classe (HOCHREITER, 2007). Em suma, métodos baseados em comparações caractere-a-caractere, para produzirem alinhamentos em larga escala, tornaram-se impraticáveis, muito além da capacidade computacional atualmente disponível (STUART *et al.*, 2002a; STUART e BERRY, 2004). Com a geração de sequências completas de genoma em bancos de dados públicos, contendo bilhões de sequências de caracteres, torna-se crucial o desenvolvimento de métodos efetivos para comparação e categorização de genes, preferencialmente que não tenham tempo de processamento limitado ao tamanho da base de dados (WU *et al.*, 1992). Por exemplo, já estão disponíveis mais de 50 genomas completos de procariotos, 5 genomas de eucariotos *(yeast, roundworm, fruit fly, human* e *A. thaliana*) e mais de 160 genomas mitocondriais de vertebrados. O uso destes dados, com todas as sequências do genoma, é muito difícil de ser feito por alinhamentos.

Uma outra consideração crítica refere-se aos escores usados por algoritmos de alinhamento: matrizes PAM (DAYHOFF *et al.*, 1978) e BLOSUM (HENIKOFF e HENIKOFF, 1992). Estas soluções heurísticas refletem incompletudes metodológicas na abordagem da divergência de sequências e também refletem a suposição de conservação da contiguidade entre seguimentos homólogos. Isto torna difícil aos alinhamentos lidar com recombinação genética e *genetic shuffling* (VINGA e ALMEIDA, 2003).

Além disto, os algoritmos de alinhamentos são intrinsecamente subjetivos e altamente sensíveis à matriz de substituição usada, além de utilizarem pontos de corte (*cut-off*) e penalidades de *gap* difíceis de serem definidos e que, quando alterados, podem produzir resultados discordantes (KRAWETZ e WOMBLE, 2003; STUART *et al.*, 2002a). De acordo com Thorne (2000), o erro mais significante em filogenias moleculares deve-se a alinhamentos incorretos. A confiabilidade nos resultados de múltiplos alinhamentos é

questionável, tanto que os programas que fazem esse processo automaticamente devem ser considerados somente como um ponto de partida, necessitando de melhorias feitas manualmente através de edições (KRAWETZ e WOMBLE, 2003). Estas edições levam ao descarte de parte da sequência original, fazendo com que a homologia postulada seja restrita a poucos domínios selecionados (STUART *et al.*, 2002a ; THORNE, 2000). Um outro ponto refere-se ao fato de que alinhamentos também podem ignorar que sequências dissimilares podem ter funções similares. "*Proteínas com sequências diferentes (< 20% de identidade) podem ter enovelamento muito similares, como exemplificado pelas globinas carreadoras de oxigênio dos mamíferos, insetos e plantas*" (STRYER, 1996): Hemoglobina humana, Eritrocruorina de insetos e Leghemoglobina de nódulo de raiz . A detecção de homologia remota, com baixo nível de similaridade também é muito difícil de ser obtida usando análise por alinhamentos (DONG *et al.*, 2006).

Uma outra análise que também deve ser considerada nesta discussão é a filogenia, que é a reconstrução da história evolucionária de uma coleção de organismos ou o processo de se desenvolver hipóteses sobre a relação evolutiva de organismos com base nas suas características observáveis. A análise filogenética tenta descrever o relacionamento evolutivo de um grupo de sequências de genes, proteínas ou genomas completos (GIBAS, 2001). Os estudos filogenéticos partem do pressuposto que todas as formas vivas da terra (tanto as existentes hoje quanto as já extintas) compartilham de uma origem comum, uma provável molécula replicadora. Conseqüentemente, todos os organismos vivos podem ser relacionados através de padrões de descendência, tendo um ancestral comum mais recente ou antigo. Organismos proximamente relacionados descendem de ancestrais comuns mais recentes, enquanto organismos mais distantemente relacionados possuem ancestrais comuns mais antigos. Já a filogenia de genes e proteínas não trata da evolução do organismo inteiro, mas de mudanças evolutivas em regiões codificantes específicas. Neste caso, procura-se identificar qual a relação evolutiva entre uma família de sequências dentro de um único organismo ou entre diferentes organismos.

A filogenia de genes pode ser baseada em alinhamentos de sequências, entretanto, métodos para produção de filogenias baseados em múltiplos alinhamentos de sequências completas de genoma são impraticáveis pois demandam um enorme esforço computacional. É interessante observar que, mesmo se fosse viabilizada tal análise (comparação caractere-a-caractere de genomas completos), haveria problema pois, muitas das sequências

disponíveis contêm uma alta fração de falsa similaridade ou homoplasia[1] resultante de uma evolução estocástica neutra, evolução convergente ou transferência horizontal de genes. Determinar quais caracteres são homologias verdadeiras e quais são fruto de homoplasia é um problema difícil e geralmente é decidido somente quando as relações de ancestralidade já estão estabelecidas (STUART *et al.*, 2002a). A solução para esta questão é a construção de filogenias baseadas em dados de genomas completos e procedimentos para se medir a similaridade de sequências sem a necessidade de alinhamentos.

Como alternativa aos alinhamentos, vários métodos para comparação de sequências e de genomas completos, que não utilizam explicitamente comparações de caracteres par-a-par, têm sido propostos e praticados com sucesso (DONG *et al.*, 2006; LIU *et al.*, 2008; RODRIGUES *et al.*, 2004; SANDBERG, 1997; STUART e BERRY, 2003; STUART e BERRY, 2004; STUART *et al.*, 2002a; STUART *et al.*, 2002b; TEICHERT *et. al*, 2007; VINGA e ALMEIDA, 2003; WU *et al.*, 1992; WU *et al*., 2007; YUAN *et al.*, 2005). Na verdade, a maioria dos métodos de análise de similaridade e detecção de homologia podem ser divididos em 3 grupos (DONG *et al.*, 2006): algoritmos de comparação par-a-par de sequências (método padrão); modelos generativos para famílias de proteínas, usando cadeias de Markov; classificadores discriminativos, usando exemplos positivos e negativos de similaridade.

[1] Enquanto na homologia a similaridade é devida a um ancestral comum, na homoplasia, a similaridade ocorre devido a evolução paralela, evolução convergente ou perda secundária.

Objetivos

Objetivo geral

Apresentar uma metodologia de análise de sequências primárias de proteínas que seja computacionalmente eficiente e biologicamente válida; representar proteínas por meio de vetores e aplicar técnicas de Álgebra Linear, Estatística e Otimização para a extração de padrões, a caracterização e a classificação de genes.

Objetivos específicos

1. Representar sequências primárias de proteínas como vetores de frequência de peptídeos.
2. Propor uma interpretação biológica para a decomposição em valores singulares (SVD), no contexto da análise de sequências protéicas.
3. Avaliar se medidas de similaridade da SVD, distância Euclidiana e cosseno, estão associadas com a distância global de edição e com medidas de similaridade usadas por um programa de alinhamento de sequências.
4. Estimar modelos de regressão que utilizem, como variáveis explicativas, as métricas de similaridade da SVD (distância Euclidiana e cosseno), e, como variáveis resposta, distância global de edição e medidas de similaridade usadas por um programa de alinhamento de sequências.
5. Estimar o grau de concordância entre cosseno e distância Euclidiana com o resultado gerado por um programa padrão de alinhamento de sequências, quando da classificação de uma sequência desconhecida.
6. Identificar aminoácidos importantes para a classificação de uma determinada categoria de proteína por meio dos vetores de frequência de aminoácidos.
7. Identificar bipeptídeos importantes para a classificação de uma determinada categoria de proteína por meio dos vetores de frequência de bipeptídeos.
8. Mapear a relação multidimensional de genomas e outros dados multivariados para o espaço bi e tridimensional (2D e 3D), desenvolvendo mecanismos que permitam identificar visualmente relações entre os elementos na representação proposta.

Resultados

Os resultados desta tese estão apresentados em sete capítulos, cada um deles com artigos que tratam da representação vetorial de proteínas, que foram analisadas sem a necessidade de múltiplos alinhamentos. Inicialmente, cada sequência protéica foi transformada num vetor de frequências de peptídeos de tamanho "p", considerando todas as combinações possíveis de aminoácidos para formarem um *p*-peptídeo. Com 20 aminoácidos, o modelo de espaço vetorial é formado por vetores de tamanho 20^p. Decomposição em valores singulares (SVD) e/ou modelos de regressão logística são aplicados aos dados para extrair padrões ou para permitir a visualização de dados multidimensionais.

O primeiro capítulo "***Singular value decomposition (SVD) and BLAST: quite different methods achieving similar results***" (COUTO *et al*., 2011a), apresenta uma análise cujo objetivo é mostrar como sequências primárias de proteínas podem ser codificadas como vetores de frequência de peptídeos, avaliando o significado biológico desta codificação. No capítulo, medidas de similaridade da SVD, distância Euclidiana e cosseno, são comparadas com medidas de similaridade usadas por um programa de alinhamento de sequências (BLAST). Correlação de Spearman (r) é usada para avaliar a associação entre estatísticas usadas pelo BLAST e métricas da SVD. A distância euclidiana foi negativamente correlacionada com *bit score* (r>-0,6) e positivamente correlacionado com *E value* (r>+0,7). Já o cosseno apresentou correlação negativa com *E value* (r>-0,7) e correlação positiva com *bit score* (r>+0,8). Neste mesmo capítulo, é feita uma estimava para o grau de concordância entre cosseno e distância Euclidiana com o resultado gerado por um programa padrão de alinhamento de sequências, quando da classificação de uma sequência desconhecida.

O capítulo 2, "***Application of latent semantic indexing (LSI) to evaluate the similarity of sets of sequences without multiples alignments character-by-character***" (COUTO *et al*., 2007) apresenta uma visão geral do método. Sequências foram comparadas usando a distância de edição entre cada par de sequências e respectivos cosseno e distância Euclidiana. A correlação entre cosseno e distância de edição foi de -0.32 e entre distância Euclidiana e distância de edição foi de +0.70. Além disto, a habilidade da SVD na classificação de uma sequência de acordo com sua categoria também foi avaliada. Com matrizes de tripeptídeos todas as consultas foram corretamente classificadas.

O capítulo 3, "***Unrevealing biological process with linear algebra: extracting patterns from noisy data***" (COUTO *et al*., 2011b), propõe uma interpretação biológica para a

decomposição em valores singulares (SVD): os valores singulares visualizados como *scree plots* revelam os principais componentes, o número de processos escondidos num banco de dados de sequências protéicas.

No quarto capítulo "***Feature selection for protein sequence classification by using logistic regression models and singular value decomposition***" (COUTO *et al*., 2010a), modelos de regressão logística e SVD foram usados para a seleção de atributos importantes para a classificação de seqüências protéicas. Além da identificação de atributos, a combinação de modelos de regressão logística com SVD permitiu uma melhor classificação de seqüências desconhecidas do que quando isto era feito somente pela SVD.

O quinto capítulo, "***Protein sequence retrieval system based on logistic regression models***" (COUTO *et al*., 2010b),apresenta um método que gera modelos de regressão logística que permitem a previsão de uma nova seqüência de ácidos aminados. Testamos com sucesso o método em dez casos: insulina, hemoglobina, queratina, citocromo, albumina, colágeno, fibrinogênio e proteínas relacionadas com fibrose cística, doença de Alzheimer e esquizofrenia.

O sexto capítulo "***Genome Visualization in Space***" (MARCOLINO *et al*., 2010), usa SVD e Otimização para a visualização multidimensional de genomas em 2D ou 3D. O último capítulo "***Visualizing high dimensional and multivariate data applying singular value decomposition followed by optimization***" (COUTO *et al*., 2010c), apresenta um artigo similar ao sexto, no qual é feita uma abordagem que usa SVD e otimização para mapear dados multivariados de proteínas para representação multidimensional em espaço 2D e 3D. Tanto no sexto quanto no sétimo capítulos, foram desenvolvidos mecanismos cujo objetivo é permitir que relações entre os elementos vetoriais multidimensionais, de conjuntos de proteínas ou de outros dados, possam ser visualmente identificadas.

Capítulo 1 – Decomposição em valores singulares (SVD) e BLAST: diferentes métodos produzindo resultados semelhantes

SINGULAR VALUE DECOMPOSITION (SVD) AND BLAST: QUITE DIFFERENT METHODS ACHIEVING SIMILAR RESULTS

Bráulio Roberto Gonçalves Marinho Couto
Centro Universitário de Belo Horizonte / UNI-BH, Av. Professor Mário Werneck 1685, Belo Horizonte, Brazil
braulio.couto@unibh.br

Macelo Matos Santoro
Departamento de Bioquímica e Imunologia, UFMG, Av. Antonio Carlos 6627, Belo Horizonte, Brazil
santoro@icb.ufmg.br

Marcos Augusto dos Santos
Departamento de Ciência da Computação, UFMG, Av. Antonio Carlos 6627, Belo Horizonte, Brazil
marcos@dcc.ufmg.br

Keywords: genomics; matrix analysis; BLAST; SVD.

Abstract: The dominant methods to search for relevant patterns in protein sequences are based on character-by-character matching, performed by software known as BLAST. In this paper, sequences are recoded as *p*-peptide frequency matrix that is reduced by singular value decomposition (SVD). The objective is to evaluate the association between statistics used by BLAST and similarity metrics used by SVD (Euclidean distance and cosine). We chose BLAST as a standard because this string-matching program is widely used for nucleotide searching and protein databases. Three datasets were used: mitochondrial-gene sequences, non-identical PDB sequences and a Swiss-Prot protein collection. We built scatter graphs and calculated Spearman correlation (ρ) with metrics produced by BLAST and SVD. Euclidean distance was negatively correlated with bit score (ρ>-0.6) and positively correlated with E value (ρ>+0.7). Cosine had negative correlation with E value (ρ>-0.7) and positive correlation with bit score (ρ>+0.8). Besides, we made agreement tests between SVD and BLAST in classifying protein families. For the mitochondrial gene database, we achieved a kappa coefficient of 1.0. For the Swiss-Prot sample there is an agreement higher than 80%. The fact that SVD has a strong correlation to BLAST results may represent a possible core technique within a broader algorithm.

1 INTRODUCTION

Comparison of protein sequences is one of the most fundamental issues in Bioinformatics. The dominant methods of such analysis are based on character- by-character matching, made by rapid but not very sensitive algorithms with heuristics, known as BLAST – the basic local alignment search tool (Altschul *et al.*, 1990). Even with good performance, these methods still have difficulties, due to computational complexity and other issues, as problems with genetic recombination and genetic shuffling (Vinga and Almeida, 2003). BLAST, for example, is inherently subjective and highly sensitive to the substitution matrix used in cut-off points and applied gap penalties, that are difficult to define and when altered, can produce conflicting results (Krawetz and Womble, 2003) and even "BLASTphemy" when users are unable to interpret its results (Pertsemlidis and Fondon III, 2001). Database redundancy, very common in a large protein sequence collection, is another problem for BLAST, slowing down searches and reducing the significance of an alignment because of the linear dependency of BLAST E value and the database size (Holm and Sander, 1998).

Several methods for comparing sequences and complete genomes, which do not explicitly use comparisons of character-by-character, have been proposed and successfully applied as alternative to alignments approaches (Wu *et al.*, 1992; Stuart *et al.*, 2002; Stuart & Berry, 2004; Yuan *et al.*, 2005; Dong *et al.*, 2006; Teichert *et. al*, 2007; Liu *et al.*, 2008; Jun, S.R. *et al.*, 2010). In this paper, proteins are recoded as p-peptide frequency matrix that is reduced by singular value decomposition (SVD), in a latent semantic indexing information retrieval system as described by Stuart (Stuart *et al.*, 2002) and adapted by Couto (Couto *et al.*, 2007). We first represented proteins as vectors and then calculated sequences similarities using linear algebra methods.

Figure 1 shows the simplest case where proteins are represented as three-dimensional vectors (3D): frequencies of Cystein, Alanine and Isoleucine are used to recode mitochondrial genes for four species. It is interesting to notice that protein vectors from the same family (COX3 and COX2) point to the same direction, which can be measured by the cosine among the vector angles (Eldén, 2006).

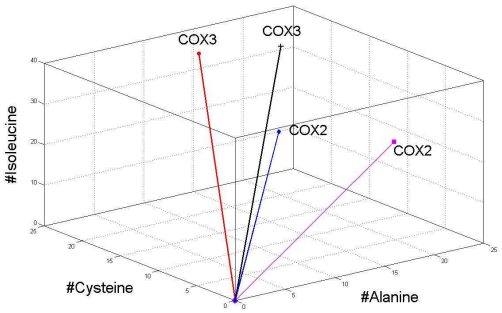


Figure 1: Representation of proteins as three-dimensional vectors.

The first objective here is to assess the relationship among similarity metrics from SVD, cosine and Euclidean distance, bit score and E value, statistics used by BLAST. We applied a scatter graph analysis and Spearman's rank correlations technique to do so (ρ). The second objective is to verify if there is an agreement, when an unknown sequence is classified or identified, among SVD results and the "gold standard", defined by the most similar BLAST hit. This was made by analysis of percent agreement, kappa coefficient, sensitivity, specificity and ROC curve (Altman, 1991). We chose BLAST as a standard because this string-matching program "has become the single most important piece of software in the field of bioinformatics" and it is widely used for nucleotide searching and protein databases (Korf *et al.*, 2003). According to Google, the first paper describing BLAST (Altschul *et al.*, 1990) was cited over 23,000 times (www.scholar.google.com).

2 SYSTEM AND METHODS

2.1 Programs and datasets

Programs implemented for this analysis were written in MATLAB (The Mathworks, 1996), using its inbuilt functions (SVD, sparse matrix manipulation subroutines etc). Three datasets were used in this paper. The first evaluated database had 64 vertebrate mitochondrial genomes composed of 832 proteins from 13 known gene families (ATP6, ATP8, COX1, COX2, COX3, CYTB, ND1, ND2, ND3, ND4, ND4L, ND5 and ND6). This curated protein database was downloaded from the online information by Stuart *et al.* paper (Stuart *et al.*, 2002). The file "pdb_seqres.txt.gz", located in http://bioserv.rpbs.jussieu.fr/PDB/, was the second database. This file has 121,556 redundant protein sequences from PDB (Protein Data Bank), which was reduced to 37,561 non-identical sequences. A randomly sample of 40,000 sequences from the Swiss-Prot section of the Universal Protein Resource (UniProt) was the third protein collection (http://www.uniprot.org/downloads).

2.2 Vector representation of proteins

Before one can apply the linear algebra methods used here, it is necessary to represent proteins as vectors in a high-dimensional Euclidean space.

Firstly, we consider a bio molecular sequence as a complex written language, so its analysis can be very similar to that used by Information Retrieval Systems, where large amounts of textual information are organized, compared and categorized. In this case, individual protein sequences correspond to 'passage' of text, whereas peptides of a given size (p) serve as 'words' (Stuart *et al.*, 2002). Hence, sequences are recoded as p-peptide frequency values using all possible overlapping p-peptides window. With 20 amino-acids it is generated a 20^p x n matrix, where p is the word-size and n is the number of proteins to be analyzed. In these matrices, proteins are treated as documents and the p-peptides as terms, which allow the problem to be solved by linear algebra methods (Eldén, 2006).

The amino-acid word-size p that can be used to build the p-peptide frequency matrix varies from one to four. The utility of larger peptides is yet to be explored, but to use 5 or more amino-acids can be

result in computational problems. With five amino-acids the frequency matrix will be 3,200,000 rows, most of that with zero. This structure is huge and hard to handle. Besides computational issues, larger peptides will lead to problem during the similarity search step. According to Stuart (Stuart *et al.*, 2002), tripeptides may prove useful with highly diverged sequences and tetrapeptides with highly related proteins. On the other hand, larger peptides will remain real undetected similarity, even between very highly related proteins.

Representing proteins as frequency vectors of p-peptides has the limitation that it does not consider the occurrences order of p-peptides in the sequence. Despite this possible ambiguity, several studies have shown that this approach is surprisingly effective in discriminatory analysis of protein sequences (Vinga and Almeida, 2003). Anyway, before using this protein vector representation, we made an analysis of its ambiguity rate according to the number of amino-acids (p) in the matrix of frequency protein-peptide. We compared 26,675 non-identical proteins longer than 100 amino-acids and selected from the PDB dataset. To identify ambiguities during vector recoding, we compared 355,764,475 sequences-pairs. The percentage of ambiguity felt from about 4%, when used only one amino-acid in the matrix of frequencies (p=1) to less than 0.5% in proteins with two or more amino-acids. The percentage of uncertainty was calculated considering the number of different sequences with the coding for all sequences that were compared pair-to-pair (26,675). It is noteworthy that in all pairs with identical vector coding, even among the 1,267 pairs with p=1, the protein involved was exactly the same, with minor changes of amino-acids in some positions. This happened because, before analysis, we removed from the PDB database only sequences with 100% identity. We can say that the ambiguity is a theoretical possibility in principle but not in practice.

2.3 Singular value decomposition

After the generation of the p-peptide frequency matrix (M) representing each dataset with n sequences, the matrix itself is subjected to SVD (Deerwester *et al.*, 1990; Berry *et al.*, 1995) and factorized as $M = USV^T$. Where U is the p x p orthogonal matrix having the left singular vectors of M as its columns, V is the n x n orthogonal matrix having the right singular vectors of M as its columns, and S is the p x n diagonal matrix with the singular values $\sigma_1 \geq \sigma_2 \geq \sigma_3 \ldots \geq \sigma_r$ of **M** in order along its diagonal (r is the rank of M or the number of linearly independent columns or rows of M). This is performed by many software, including MATLAB (The Mathworks, 1996), used in this work. The matrix (**U**) is related to the p-peptides of the dataset, whilst (**V**) is associated with the proteins studied. The central matrix (**S**) contains the singular values of (**M**) in decreasing order. These singular values are directly related with independent characteristics within the dataset. Actually, the largest values of (**S**) provide meaning of the peptides and proteins in the matrix (**M**). On the other hand, the smaller singular values identify less significant aspects and the noisy inside the dataset (Eldén, 2006). The number of significant singular values from SVD analysis shows how many process or groups can be hidden in database.

For the sequence similarities analysis, instead of using the original matrix **M**, a rank reduction of **M** is done by using the k-largest singular values of **M**, or k-largest singular triplet $\mathbf{U_k}$, $\mathbf{S_k}$, $\mathbf{V_k}$, where $k < r$. The truncated matrix $\mathbf{M_k} = \mathbf{U_k S_k (V_k)^T}$ has two main advantages. Reduced dimensionality makes the problem computationally approachable, which is crucial in whole genome analysis. Besides, and very important, the rank reduction improve accuracy of protein matrix by discarding noise and reducing the variability in p-peptide usage for the same protein family (Couto *et al.*, 2007). The choice of k, the number of singular values that must be used in the reconstruction of the protein matrix after SVD, is critical and normally empirically decided. Ideally, the k factor or matrix dimension must be large enough to fit all the real structure in the data, and small enough not to fit the sampling error or unimportant details. In this work we used the method proposed by Everitt and Dunn, that recommends analyzing the relative variances of each singular values. Singular values which relative variance is less than 0.7/n, where n is the number of proteins in the document-term matrix, must be ignored (Everitt and Dunn, 2001).

3 RESULTS

Firstly, we analyzed 620 sequences randomly selected from the first database with mitochondrial gene families. BLAST, actually bl2seq.exe program with default parameters, were used to compare each pair of sequence, which totalling 191,890 comparisons. The same proteins were recoded as vectors in a high-dimensional space that was reduced by SVD and analyzed according to the methods described by Couto (Couto *et al.*, 2007).

Scatter plots were built and suggested that Euclidean distance is negatively related with bit score, but positively correlated with E value. For the cosine we found a negative association with E value and a positive relationship with bit score. Those results are consistent because, the higher cosine, the more similar are the two protein vector. The same happens with BLAST bit score. As the E value, the smaller Euclidean distance between the end points of two protein vectors, the more similar are the sequences. Figure 2 and 3 presents respectively scatter graphs between the bit score and cosine and between the bit score and Euclidean distance.

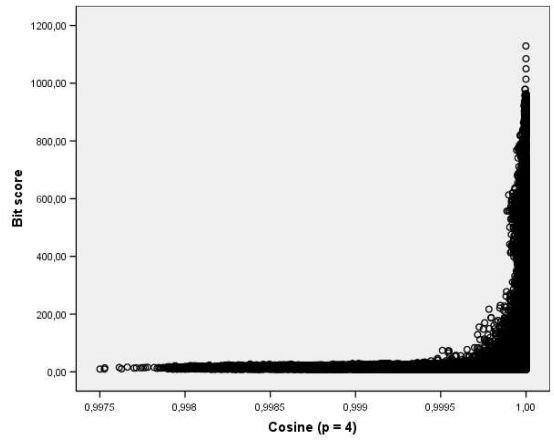


Figure 2: Scatter graph for mitochondrial gene dataset: cosine of angle between protein vectors has a positive correlation with BLAST bit score.

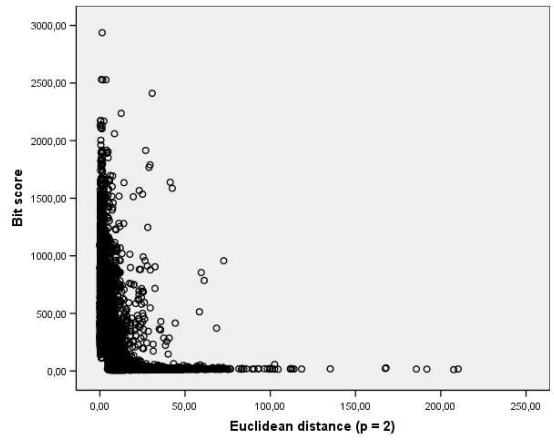


Figure 3: Scatter graph for mitochondrial gene dataset: Euclidean distance between protein vectors has a negative correlation with BLAST bit score.

For the second database, 27,361 non-identical PDB sequences longer than 100 amino-acids were compared with BLAST and SVD. In this analysis, the first protein was compared with the second, than was compared with the third and so on, which totalled 27,360 comparisons. Figure 4 shows the parameters used by bl2seq.exe program analysis.

BL2Seq	First sequence	Second sequence	Program	Substitution matrix	Cost to open a gap	Cost to extend a gap	Output format: tabular	Output file
bl2seq	–iS_A.txt	–jS_B.txt	-p blastp	-M BLOSUM62	-G 11	-E 1	-D 1	-oS_AB.txt
bl2seq	–iS_A.txt	–jS_B.txt	-p blastp	-M BLOSUM45	-G 13	-E 2	-D 1	-oS_AB.txt
bl2seq	–iS_A.txt	–jS_B.txt	-p blastp	-M BLOSUM80	-G 13	-E 2	-D 1	-oS_AB.txt
bl2seq	–iS_A.txt	–jS_B.txt	-p blastp	-M PAM30	-G 7	-E 2	-D 1	-oS_AB.txt
bl2seq	–iS_A.txt	–jS_B.txt	-p blastp	-M PAM70	-G 7	-E 2	-D 1	-oS_AB.txt
bl2seq	–iS_A.txt	–jS_B.txt	-p blastp	-M PAM250	-G 15	-E 3	-D 1	-oS_AB.txt

Obs.: S_Atxt and S_B.txt are examples of sequence files.

Figure 4: BLAST parameters used in the PDB database.

We built scatter graphs and calculated Spearman correlations (ρ) among bit score and E value from the most similar BLAST hit, respective cosine and Euclidean distance from SVD (Figure 5). All plots had the same shape that observed for the first database. For BLAST analysis we also compared the results obtained by applying different substitution matrix: BLOSUM62, BLOSUM45, BLOSUM80, PAM30, PAM70 and PAM2050. The Euclidean distance was negatively correlated with bit score ($\rho > -0.6$) and positively correlated with E value ($\rho > +0.7$). For the cosine we found a negative correlation with E value ($\rho > -0.7$) and a positive correlation with bit score ($\rho > +0.8$). It is interesting that the correlation between E value and bit score was not exactly 1.0 because of rounding errors.

Besides the correlation analysis, we made an agreement test between SVD and BLAST in classifying protein families. For the mitochondrial gene families database, we used a sample of 212 sequences from the 13 gene families as queries (test set), and the other proteins (620) were used to generate the frequency matrix (training set): the kappa coefficient between SVD and BLAST was 1.0 (agreement = 100%). If we use the first three significant singular values from the SVD analysis of the thirteen gene families' database, we can generate a three-dimensional graph showing how these genes can be visualized in space (Figure 6). It is interesting how the families are well separated in space, which facilitates classification.

In another analysis, the 27,360 pair-to-pair comparisons made by BLAST and SVD of the PDB sequences, were evaluated in order to asses the agreement of both techniques in detecting biological significance. The gold standard for a biological significant alignment was defined by an E value less than 0.05 obtained using BLOSUM62 as the substitution matrix (Pertsemlidis and Fondon III,

2001). The area under the ROC curve (AUC) was estimated for both, cosine, Euclidean distance and for the frequency matrix using one, two, three and four peptides. The eight AUCs estimated were higher than 0.80 (Figures 7 and 8), which indicates a good performance of SVD in detecting biological significant similarities (Altman, 1991).

		BLOSUM62	
		E value	Bit score
BLOSUM62	E value	1,000	
	Bit score	-0,974	1,000
Cosine	n_pep = 1	-0,631	0,635
	n_pep = 2	-0,709	0,764
	n_pep = 3	-0,740	0,772
	n_pep = 4	-0,708	0,726
Euclidean distance	n_pep = 1	0,697	-0,641
	n_pep = 2	0,734	-0,657
	n_pep = 3	0,708	-0,631
	n_pep = 4	0,639	-0,577
BLOSUM45	E value	0,942	-0,927
	Bit score	-0,899	0,942
BLOSUM80	E value	0,968	-0,941
	Bit score	-0,954	0,966
PAM30	E value	0,927	-0,911
	Bit score	-0,916	0,927
PAM70	E value	0,942	-0,923
	Bit score	-0,924	0,941
PAM250	E value	0,840	-0,849
	Bit score	-0,797	0,853

Figure 5: Correlation matrix: BLAST versus SVD.

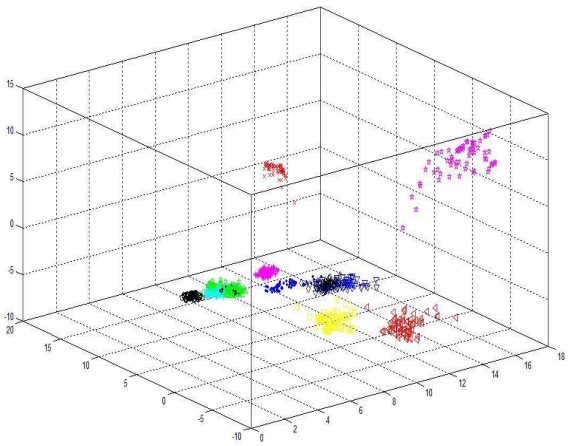


Figure 6: Visualization of mitochondrial genes using the three first singular values from SVD: the 13 gene families are well separated in space, which facilitates classification.

Table 1 summarizes the results when cosine among protein vectors is used to detect a biological significance similarity. When is used a cut-off of 0.90 for the cosine, the sensitivity and specificity for detecting biological significance were, respectively, 72% and 94%.

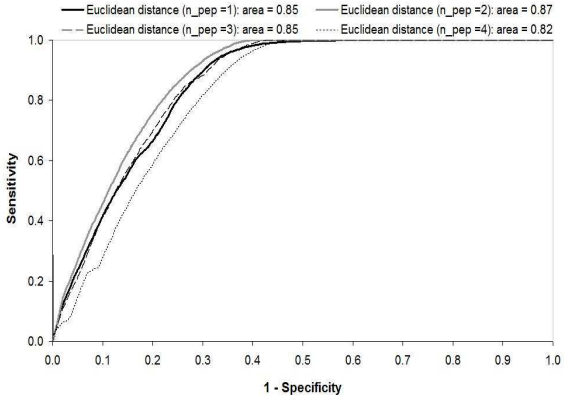


Figure 7: ROC curve built when SVD Euclidean distance is used to detect biological significant similarity.

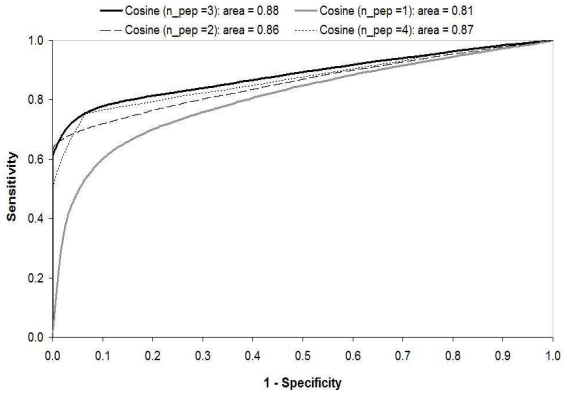


Figure 8: ROC curve built when SVD cosine is used to detect biological significant similarity.

Table 1: Two-way contingency table: cosine higher than 0.90 between protein vectors has 72% sensitivity and 94% specificity to detect biological significant similarities.

BLOSUM62 biological significance?	Cosine biological significance?		
	(+)	(−)	Total
Yes	9,678	3,843	13,521
No	808	13,031	13,839
Total	10,486	16,874	27,360

During the analysis of the third protein collection, a sample of 40,000 Swiss-Prot sequences

was randomly divided into two groups: 9,953 proteins were selected as queries (test set), and the other 30,047 sequences (training set) were used to generate the frequencies matrix of SVD and to become the BLAST database for evaluating the queries. All 9,953 unknown proteins were analyzed by SVD and BLAST (actually, *blastall* program with default parameters) and results of both methods were compared in order to detect agreement. If the Swiss-Prot mnemonic protein identification code of the most similar BLAST hit was identical as that obtained by a SVD analysis, so we had an agreement. When this happened, the matched proteins are the same, from the same or different species. Table 2 presents the percent agreement between BLAST and SVD: the results were good, except when the p-peptide matrix is built by using just one amino-acid as the word-size.

Table 2: Agreement between SVD and BLAST for classifying proteins from the Swiss-Prot dataset.

p-peptide matrix	SVD similarity metric	Percent agreement with BLAST
p=1	Cosine	20%
	Euclidean distance	30%
p=2	Cosine	79%
	Euclidean distance	82%
p=3	Cosine	80%
	Euclidean distance	82%
p=4	Cosine	69%
	Euclidean distance	72%

4 CONCLUSION

We worked with quite different techniques and we found important association among their metrics and good agreement between both methods. Despite the fact that is presumably not surprising that e.g. BLAST bit score could be positively correlated to cosine of angle, or negatively correlated to Euclidean distance, the sizes of these correlations are very interesting (Figure 5).

We achieved similar results between BLAST and SVD in several protein analyses. The findings strongly suggest that SVD can be used to protein-protein comparisons with biological significance of the similarities identified both for cosine and Euclidean distance. The fact that SVD has a strong correlation to BLAST results may represent a possible core technique within a broader algorithm.

Besides, SVD has some characteristics that could be an advantage over alignment algorithms. For example, SVD analysis can be very rapid, it does not use any heuristics to asses an unknown sequence, its metrics are exact in a sense of direction and position in a high-dimensional Euclidean space, it is not affected by database redundancy because of rank reduction, its similarity metrics do not depend on the database size, and any analyze does not need a substitution matrix nor gap penalties to produce biological significant results.

An assessment of the singular value spectrum visualized as *scree plots* (Zhu and Ghodsi, 2006) can unreveals the main components, the process that exists hidden in a database. This information can be used in many applications as clustering, gene expression analysis, immune response pattern identification, characterization of protein molecular dynamics and phylogenetic inference.

SVD can be also used to visualize the relationships between sequences and even whole genomes, which can be essential to better analyze complex systems and can be very helpful to categorize genes or species in phylogeny.

All results found in this work and the characteristics described are important because SVD can be a solution for the potential problems with alignment algorithms and can be a substitute for those methods, for example, in whole genome analysis.

ACKNOWLEDGEMENTS

We are thankful to Professor Gary W. Stuart from Indiana State University, Department of Life Sciences, who sent us helpful data. We also thank Marlon C. Souza from UNI-BH, who revised the manuscript.

REFERENCES

Altman, D.G., 1991. Practical Statistics for Medical Research. Chapman and Hall, London, UK.

Altschul, S.F. *et al*., 1990. Basic local alignment search tool. J. Mol. Biol., 215, 403-410.

Berry, M.W. *et al*., 1995. Using linear algebra for intelligent information retrieval. SIAM Review, 37, 573-595.

Couto, B.R.G.M. *et al*., 2007. Application of latent semantic indexing to evaluate the similarity of sets of sequences without multiple alignments character-by-character. GMR, 6(4), 983-999.

Deerwester, S. *et al*., 1990. Indexing by Latent Semantic Analysis. Journal of the American Society for Information Science, 41(6), 1-13.

Eldén, L., 2006. Numerical linear algebra in data mining. Acta Numerica, 327-384.

Everitt, B.S. and Dunn, G., 2001. Applied multivariate data analysis. 2nd edn. Arnold, London, UK.

Holm, L. and Sander, C., 1998. Removing near-neighbour redundancy from large protein sequence collections. Bioinformatics, 14(5), 423-429.

Jun, S.R. *et al.*, 2010. Whole-proteome phylogeny of prokaryotes by feature frequency profiles: An alignment-free method with optimal feature resolution. Proc Natl Acad Sci U S A, 107(1):133-8.

Korf, I.; Yandell, M.; Bedell, J., 2003. An essential guide to the Basic Local Align-ment Search Tool – BLAST. O'Reilly & Associates Inc., Sebastopol, USA.

Koski, L.B. and Golding, T.B., 2001. The closest BLAST hit is often not the nearest neighbor. J. Mol. Evol., 52, 540-542.

Krawetz, A.S. and Womble, D.D., 2003. Introduction to Bioinformatics: a theoretical and practical approach. Humana Press, Totowa, USA.

Liu, B. *et al*., 2008. A discriminative method for protein remote homology detection and fold recognition combining top-n-grams and latent semantic analysis. BMC Bioinformatics, 9, 510.

Pertsemlidis, A. and Fondon III, J.W., 2001. Having a BLAST with bioinformatics (and avoiding BLASTphemy). Genome Biology, 2(10), 1-10.

Stuart, G.W. *et al*., 2002. Integrated gene and species phylogenies from unaligned whole genome protein sequences. Bioinformatics, 18(1), 100-108.

Stuart, G.W. and Berry, M.W., 2004. An SVD-based comparison of nine whole eukaryotic genomes supports a coelomate rather than ecdysozoan lineage. BMC Bioinformatics, 5: 204+.

The Mathworks, 1996. MATLAB: mathematical computation, analysis, visualization, and algorithm development (version 5.0). Natick, Massachusetts, USA.

Teichert, F. *et al*., 2007. SABERTOOTH: protein structural alignment based on a vectorial structure representation. BMC Bioinformatics, 8, 425.

Vinga, S. and Almeida, J., 2003. Alignment-free sequence comparison: a review. Bioinformatics, 19(4), 513-523.

Wu, C. *et al*., 1992. Protein classification artificial neural system. Protein Science, I, 667-677.

Yuan, Y. *et al*., 2005. A Protein Classification Method Based on Latent Semantic Analysis. Conf Proc IEEE Eng. Med. Biol. Soc., 7, 7738-41.

Zhu, M. and Ghodsi, A., 2006. Automatic dimensionality selection from the scree plot via the use of profile likelihood. Computational Statistics and Data Analysis, 51, 918-930.

Capítulo 2 – Aplicando decomposição em valores singulares para a análise de similaridades de sequências sem múltiplos alinhamentos caractere-a-caractere

Online journal ISSN - 1676-5680
GMR Genetics and Molecular Research
EVOLUTION AND TECHNOLOGY

Application of latent semantic indexing to evaluate the similarity of sets of sequences without multiple alignments character-by-character

B.R.G.M. Couto[1,2], A.P. Ladeira[1,3] and M.A. Santos[4]

[1]Programa de Doutorado em Bioinformática,
Departamento de Bioquímica e Imunologia,
Universidade Federal de Minas Gerais, UFMG, Belo Horizonte,
MG, Brasil
[2]Curso de Ciência da Computação, Centro Universitário de Belo Horizonte,
UNI-BH, Belo Horizonte, MG, Brasil
[3]Escola de Ciência da Informação, Universidade Federal de Minas Gerais,
UFMG, Belo Horizonte, MG, Brasil
[4]Departamento de Ciência da Computação,
Universidade Federal de Minas Gerais, UFMG, Belo Horizonte,
MG, Brasil
Corresponding author: B.R.G.M. Couto
E-mail: bcouto@acad.unibh.br

Genet. Mol. Res. 6 (4): 983-999 (2007)
Received August 03, 2007
Accepted September 25, 2007
Published October 05, 2007

ABSTRACT. Most molecular analyses, including phylogenetic inference, are based on sequence alignments. We present an algorithm that estimates relatedness between biomolecules without the requirement of sequence alignment by using a protein frequency matrix that is reduced by singular value decomposition (SVD), in a latent seman-

tic index information retrieval system. Two databases were used: one with 832 proteins from 13 mitochondrial gene families and another composed of 1000 sequences from nine types of proteins retrieved from GenBank. Firstly, 208 sequences from the first database and 200 from the second were randomly selected and compared using edit distance between each pair of sequences and respective cosines and Euclidean distances from SVD. Correlation between cosine and edit distance was -0.32 ($P < 0.01$) and between Euclidean distance and edit distance was +0.70 ($P < 0.01$). In order to check the ability of SVD in classifying sequences according to their categories, we used a sample of 202 sequences from the 13 gene families as queries (test set), and the other proteins (630) were used to generate the frequency matrix (training set). The classification algorithm applies a voting scheme based on the five most similar sequences with each query. With a 3-peptide frequency matrix, all 202 queries were correctly classified (accuracy = 100%). This algorithm is very attractive, because sequence alignments are neither generated nor required. In order to achieve results similar to those obtained with edit distance analysis, we recommend that Euclidean distance be used as a similarity measure for protein sequences in latent semantic indexing methods.

Key words: Bioinformatics, Molecular comparisons, Sequence alignments, Latent semantic indexing

INTRODUCTION

Many molecular analyses, including phylogenetic inferences, are based on character-by-character comparisons (Krawetz and Womble, 2003). These standard methods use alignment algorithms that are intrinsically highly subjective and usually employ cut-off values and gap penalties that are difficult to define (Stuart et al., 2002a). According to Thorne (2000), the most significant error in molecular phylogenies is due to inaccurate alignments. Furthermore, once an alignment is obtained, it is necessary to discard a fraction of the original sequences compared, which restricts the postulated homology to a few selected domains (Thorne, 2000; Stuart et al., 2002a). Besides the difficulties with the alignment algorithm itself, as whole genome sequences continue to accumulate in public databases, with billions of sequence characters, effective methods for comparing and categorizing these genes are crucial. Actually, the complexity involved in estimating relatedness between large numbers of biomolecules is enormous, and methods based on character-by-character comparisons to produce large-scale alignments become impractical, far beyond the scope of currently available computational systems (Stuart et al., 2002a,b; Stuart and Berry, 2003, 2004).

In this report, we present an algorithm to compare and to categorize genes that are based on the methodology developed by Stuart et al. (2002a) for generating whole genome phylogenies using vector representations of protein sequences. The algorithm estimates relat-

edness between large numbers of biomolecules without the requirement of multiple sequence alignment. The original method (Stuart et al., 2002a) uses a tool from numerical analysis, called singular value decomposition (SVD), to process a peptide frequency matrix, a large sparse data matrix in which each protein is uniquely represented as a vector. As the comparisons among sequences are made by vector pairwise comparisons instead of sequence alignments, before applying the proposed method, we analyzed the relationship between the vector properties (cosine and Euclidean distance values) and edit distance measures, which allowed the validation of the methodology.

MATERIAL AND METHODS

A biomolecular sequence can be viewed as a complex written language, so that its analysis can be very similar to that used by information retrieval (IR) systems, where large amounts of textual information are organized, compared and categorized (Berry et al., 1999; Stuart et al., 2002a). In the IR field, commonly used models are the boolean, vector space, probabilistic model, and latent semantic indexing (LSI), which combine the vector space model with singular value decomposition (Cöster, 1999).

The method proposed by Stuart et al. (2002a) to evaluate the similarity of sequences is an LSI method, where individual protein sequences correspond to a "passage" of text, whereas peptides of a given size serve as n-gram "words". In this approach, protein sequences are recoded as p-peptide frequency values using all possible overlapping p-peptides (Stuart et al., 2002a; Rodrigues et al., 2004). With 20 amino acids, a 20^p x n matrix is generated (20^p rows and n columns or vectors, one for each n protein under analysis). For instance, by using a tripeptide, there are 20^3 = 8000 possible peptides, and if 4 amino acids are used, there are 20^4 = 160,000 possible tetrapeptides. The simplest situation, illustrated by Figure 1, occurs when only one amino acid is used for each peptide. In this case, the frequency matrix has only 20 rows and n columns, each one representing the protein vectors. These n vectors are composed of the frequency of each amino acid in the protein (f1,1 = frequency of alanine in the first protein). When all combinations of size 3 amino acids are used to build the matrix (Figure 2), each vector has the frequency of each tripeptide in the protein (f1,1 = frequency of tripeptide 1 in the first protein). In these matrices, proteins are treated as documents and peptides as terms, which allows the problem to be solved by information retrieval methods.

Programs and datasets

Programs implemented for this analysis were written in MATLAB (The Mathworks, 1996), using its built-in functions (SVD, sparse matrix manipulation subroutines, etc.). Two datasets were used in this paper. The first database evaluated had 64 vertebrate mitochondrial genomes composed of 832 proteins from 13 known gene families (ATP6, ATP8, COX1, COX2, COX3, CYTB, ND1, ND2, ND3, ND4, ND4L, ND5, and ND6). This curated protein database was downloaded from the online information at http://mama.indstate.edu/users/stuart/gaspipe/index.html from Stuart et al. (2002b).

The second database was composed of sequences from proteins retrieved from GenBank on April 19, 2006 (Figure 3). A random sample of 100 sequences was obtained of each

Terms: amino acids	Documents: proteins			
	Protein 1	Protein 2	...	Protein n
V1 = Alanine	f1,1	f1,2	...	f1,n
V2 = Arginine	f2,1	f2,2	...	f2,n
V3 = Asparagine	f3,1	f3,2	...	f3,n
V4 = Aspartic acid	f4,1	f4,2	...	f4,n
V5 = Cysteine	f5,1	f5,2	...	f5,n
V6 = Glutamine	f6,1	f6,2	...	f6,n
V7 = Glutamic acid	f7,1	f7,2	...	f7,n
V8 = Glycine	f8,1	f8,2	...	f8,n
V9 = Histidine	f9,1	f9,2	...	f9,n
V10 = Isoleucine	f10,1	f10,2	...	f10,n
V11 = Leucine	f11,1	f11,2	...	f11,n
V12 = Lysine	f12,1	f12,2	...	f12,n
V13 = Methionine	f13,1	f13,2	...	f13,n
V14 = Phenylalanine	f14,1	f14,2	...	f14,n
V15 = Proline	f15,1	f15,2	...	f15,n
V16 = Serine	f16,1	f16,2	...	f16,n
V17 = Threonine	f17,1	f17,2	...	f17,n
V18 = Tryptophan	f18,1	f18,2	...	f18,n
V19 = Tyrosine	f19,1	f19,2	...	f19,n
V20 = Valine	f20,1	f20,2	...	f20,n

Figure 1. Protein frequency matrix built with 1-letter string of amino acids.

type of protein (globin, cytochrome, histone, cyclohydrolase, pyrophosphatase, ferredoxin, keratin, and collagen) and 200 other proteins from lymphocytes and bacteriophages, totaling 1000 sequences.

Construction of the protein matrix

Terms, documents, queries, and weights are fundamental components of any IR system (Cöster, 1999). A term is an individual word or a phrase that reflects a particular concept or key word (Berry et al., 1995). Terms are extracted from either the body of a text or a surrogate text (e.g., abstract). In the context of biomolecular sequences, terms are the p-peptide strings (usually, tripeptides or tetrapeptides). Documents are the text itself, composed of terms. Here, proteins are the documents analyzed. The information needed by an IR user is called a query (Cöster, 1999). In this report, a query will be an unknown gene sequence whose category or family we need to determine. A weight is a value reflecting the importance of a term in a document or query (Cöster, 1999). For this analysis, all terms (p-peptide) have the same weight, assumed to be one. The elements of the term document or protein matrix are the occurrences of each peptide (of size p) in a particular protein.

Terms: all peptides with 3 amino acids	Documents: proteins			
	Protein 1	Protein 2	...	Protein n
ABC	f1,1	f1,2	...	f1,n
ABD	f2,1	f2,2	...	f2,n
ABE	f3,1	f3,2	...	f3,n
ABF	f4,1	f4,2	...	f4,n
ABG	f5,1	f5,2	...	f5,n
ABH	f6,1	f6,2	...	f6,n
ABI	f7,1	f7,2	...	f7,n
ABJ	f8,1	f8,2	...	f8,n
ABL	f9,1	f9,2	...	f9,n
ABM	f10,1	f10,2	...	f10,n
ABN	f11,1	f11,2	...	f11,n
ABO	f12,1	f12,2	...	f12,n
ABP	f13,1	f13,2	...	f13,n
ABQ	f14,1	f14,2	...	f14,n
ABR	f15,1	f15,2	...	f15,n
ABS	f16,1	f16,2	...	f16,n
ABT	f17,1	f17,2	...	f17,n
ABU	f18,1	f18,2	...	f18,n
ABT	f19,1	f19,2	...	f19,n
...	...	...	...	...
...	**f8.000,1**	**f8.000,2**	...	**f8.000,n**

Figure 2. Protein frequency matrix built with 3-letter string of amino acids.

Type of sequence	Number of GenBank sequences
Globin	1,958
Cytochrome	164,423
Histone	9,985
Cyclohydrolase	2,670
Pyrophosphatase	2,313
Ferredoxin	8,338
Lymphocyte	15,535
Bacteriophage	19,663
Keratin	459
Collagen	2,922

Figure 3. Number of sequences retrieved from GenBank from different types of proteins.

The document-term matrix construction is based on the protein sequences that are re-coded as p-peptide frequency values using all possible overlapping p-peptides, which generates the frequency matrix. Matrices are built using p = 1, p = 2, p = 3, and p = 4 peptides. These sparse matrices have dimensions of 20 x n, 400 x n, 8000 x n, and 160,000 x n, respectively, where *n* is the number of sequences analyzed. A larger number of peptides is not used because it will produce huge matrices, with more than 3 million rows (20^5 = 3,200,000 rows). The MATLAB codes in Figure 4A and B build the protein matrix using sequence data in a text file, for example, in a file named "mitgenes_M.stu". The first line of the file has the number of sequences to be analyzed (n), and the other lines have the string sequences of each protein in the dataset.

It is important to note that, with four amino acids in the p-peptide, there will be 160,000 possible tetrapeptides in the protein matrix, most of which will have zero frequency. Actually, the matrix produced by the algorithm 4A and B will be very sparse, which is computationally good in terms of memory requirements.

Figure 5 shows the protein frequency matrix in the simplest case (variable n_pep = 1), when the peptide is composed of only one amino acid. In this situation, we have 20 terms, and in analyzing 5 proteins, the document-term matrix has 20 rows and 5 columns. The five proteins correspond to 2 genes (COX3 and COX2) from different vertebrate mitochondrial genomes. The original amino acid frequency for each protein varies across each vector (columns of the protein matrix).

Latent semantic indexing

LSI, developed by Deerwester et al. (1990), is an IR method that uses singular value decomposition and a vector space model to retrieve information (Orengo, 2004). In a vector space representation of information, vectors that form a frequency term-by-document matrix, as illustrated in Figures 1 and 2, are used to represent each document or proteins. The aim of LSI is to perform the retrieval of a query in terms of conceptual content, rather than literally matching terms (Deerwester et al., 1990; Berry et al., 1995; Orengo, 2004). Due to synonymy, where the same concept can be expressed in many different ways, and polysemy, where a word can have multiple meanings, in the traditional IR systems individual words provide unreliable evidence about the meaning of the document (Orengo, 2004). To overcome the synonymy and polysemy problems, LSI estimates the usage of terms across documents, revealing its underlying semantic structure. Terms that occur frequently together are associated, which in practice means that a query may retrieve documents which have none of the query terms (Deerwester et al., 1990).

In a mathematical way, synonymy and polysemy are solved by applying an SVD in the term-by-document matrix, followed by a rank matrix reduction. After the SVD, the matrix reduction is performed by replacing the original matrix with another that is as close as possible to the original but whose column space is only a subspace of the column space of the original matrix (Berry et al., 1999). The objective of breaking down the term-document matrix is to remove extraneous information or noise from the original database.

SVD is performed by many software, including MATLAB (The Mathworks, 1996) used in this study. Given a (m x n) term-by-document matrix *M*, the SVD of *M* is defined using Equation 1 (Deerwester et al., 1990):

$$M = USV^T \qquad \text{(Equation 1)}$$

where *U* is the m x m orthogonal matrix having the left singular vectors of *M* as its columns, *V* is the n x n orthogonal matrix having the right singular vectors of *M* as its columns, and *S* is the m x n diagonal matrix with the singular values $\sigma 1 \geq \sigma 2 \geq \sigma 2 \ldots \geq \sigma r$ of *M* in order along its diagonal (*r* is the rank of *M* or the number of linearly independent columns or rows of *M*).

A

```
n_pep = input('Number of amino acids in the p-peptide (1, 2, 3 or 4): ');
m = 20^n_pep;
MAT=sparse(m,n);

% Obs.: "mitgenes_M.stu" is an example of file with the protein sequences
archive = 'mitgenes_M.stu'
fid = fopen(archive,'rt');
n=fscanf(fid,'%d \n',1);

% building each protein vector
for i=1:n
    fprintf('\n %6i of %4i',i,n);
    [protein] = fgetl(fid);
    size      = length(protein);
    column    = i;
    [MAT]     =montaMAT(MAT, protein,column,size,n_pep);
end
fclose(fid);
```

B

```
function [MAT]=montaMAT(MAT,protein,column,size,n_pep)

amino acids = 'ACDEFGHIKLMNPQRSTVWY';
terms  = length(amino acids);

% overlapping window of size n_pep
for k=1:(size-n_pep)
   line  = 0;
   for j=1:(n_pep-1)
       str = protein(k+j-1);
       index  = findstr(amino acids,str);

       % calculating the peptide row in the protein matrix
       line  = line  + (index  - 1)*(terms^(n_pep-j));
   end
   str = protein(k+n_pep);
   index  = findstr(amino acids,str);

   % calculating the peptide row in the protein matrix
   line  = line  + index ;
   MAT(line,column) = MAT(line,column) + 1;
end
```

Figure 4. A. Protein matrix construction subroutine (part I). **B.** Protein matrix construction subroutine (part II).

Terms: amino acids	Documents: proteins				
	Protein 1	Protein 2	Protein 3	Protein 4	Protein 5
	COX3	COX3	COX3	COX2	COX2
V1 = Alanine	23	14	20	16	9
V2 = Arginine	1	2	1	3	3
V3 = Asparagine	5	4	4	13	12
V4 = Aspartic acid	8	7	7	14	12
V5 = Cysteine	24	23	23	8	6
V6 = Glutamine	19	21	18	9	8
V7 = Glutamic acid	18	16	16	10	8
V8 = Glycine	16	14	10	18	19
V9 = Histidine	4	3	2	4	5
V10 = Isoleucine	32	36	35	30	32
V11 = Leucine	9	9	11	8	16
V12 = Lysine	4	7	7	5	5
V13 = Methionine	12	11	12	14	11
V14 = Phenylalanine	6	7	7	7	6
V15 = Proline	5	5	5	5	6
V16 = Serine	18	21	15	21	21
V17 = Threonine	22	22	25	12	21
V18 = Tryptophan	14	15	18	18	11
V19 = Tyrosine	12	12	12	5	5
V20 = Valine	9	12	13	8	11

Figure 5. The 20 x 5-original protein matrix.

The rank reduction of *M* matrix is performed using the k-largest singular values of *M*, or k-largest singular triplet U_k, S_k, V_k, where k ≤ r. The truncated matrix M_k is defined in Equation 2:

$$M = USV^T \approx M_k = U_k S_k V_k^T \quad \text{(Equation 2)}$$

The dimension of the vector in U_k and V_k is equal to *k*, the number of SVD factors used. The extent of dimension reduction, i.e., the choice of *k*, will be detailed in the next sections. This choice is critical, being an open issue in the literature and normally decided via empirical testing (Deerwester et al., 1990; Berry et al., 1999). The truncated SVD has two main advantages. Reduced dimensionality makes the problem computationally approachable, which is crucial in whole genome analysis. Besides, and very importantly, rank reduction improves the accuracy of term-document or protein matrix by discarding noise or variability in term or peptide usage, which can remove possible homoplasy in the data (Stuart et al., 2002b). Another formula (Equation 3) to reconstruct the protein matrix, based on the *k* first singular values is:

$$A_k = \sum_{i=1}^{k} s_i \cdot u_i \cdot v_i^T \quad \text{(Equation 3)}$$

Another advantage of rank reduction is the possibility of graphical analysis and data visualization. Using the two first singular values (k = 2), the data can be analyzed by a 2-dimensional (2-D) plot and, with 3 factors (k = 3), data can be visualized in a 3-D graph.

In Figure 6, we have the *M* protein matrix, reconstructed by using two SVD factors. It is interesting to observe how the data variability, measured by the coefficient of variation, is reduced. The average coefficient of variation of the amino acid frequency for both genes was reduced from approximately 15% in the original matrix to only 3% in the reconstructed matrix. This reduction in variability is optimal for pattern recognition and clustering (Schalkoff, 1992).

Terms: amino acids	Documents: proteins				
	Protein 1	Protein 2	Protein 3	Protein 4	Protein 5
	COX3	COX3	COX3	COX2	COX2
V1 = Alanine	19	19	19	12	12
V2 = Arginine	1	1	1	3	3
V3 = Asparagine	4	5	4	12	13
V4 = Aspartic acid	7	8	7	12	13
V5 = Cysteine	23	23	24	8	6
V6 = Glutamine	19	19	20	9	8
V7 = Glutamic acid	17	16	17	9	9
V8 = Glycine	13	14	13	18	19
V9 = Histidine	3	3	3	4	5
V10 = Isoleucine	34	35	34	31	31
V11 = Leucine	10	10	9	12	12
V12 = Lysine	6	6	6	5	5
V13 = Methionine	12	12	12	12	13
V14 = Phenylalanine	7	7	7	6	7
V15 = Proline	5	5	5	5	6
V16 = Serine	18	18	18	21	22
V17 = Threonine	23	23	23	17	17
V18 = Tryptophan	16	16	16	14	14
V19 = Tyrosine	12	12	12	5	5
V20 = Valine	11	11	11	10	10

Figure 6. The 20 x 5-protein matrix reconstructed with two factors.

Besides homogenizing the amino acid frequency in each gene by eliminating data noise in COX3 and COX2 vectors, dimension reduction allows a data visualization of proteins in a 2-D plot (Figure 7), with two separated clusters (G1 = COX3 and G2 = COX2, from vertebrates A, B and C). The x-coordinate is obtained by multiplying the first column of the matrix *V* (from SVD) by the reduced *S* matrix, with only the two first singular values. The y-coordinate is calculated by the multiplication of the second column of *V* by the reduced *S* matrix, with two SVD factors.

Dimension reduction

As discussed before, the choice of *k*, the number of singular values that must be used in the reconstruction of the protein matrix after SVD, is critical and normally empirically decided. Ideally, the *k* factor or matrix dimension must be large enough to fit all the real structure in the data, and also small enough not to fit the sampling error or unimportant details. According to Deerwester et al. (1990), the best performance of any IR system is achieved when the maximum number of singular values is less than 300.

In this study, we used the method proposed by Everitt and Dunn (2001), who recommend the analysis of the relative variances of each of the singular values (v_i), calculated by Equation 4. Singular values whose relative variance is less than 0.7/n, where *n* is the number of proteins in the document-term matrix, must be ignored (Everitt and Dunn, 2001; Wall et al., 2003).

$$v_i = \frac{S_i^2}{\sum_{j=1}^{r} S_j^2}; i = 1,2,3\cdots r \qquad \text{(Equation 4)}$$

where v_i is the relative variance of the singular value S_i, from *r* singular values of the document-term matrix. The idea is to use only the most significant singular values when the protein matrix is reconstructed. For the 20 x 5-protein matrix in Figure 5, only two singular values are significant (Figure 8). In this case, *k* must be equal to 2, which was done when the 2-D plot was constructed (Figure 7).

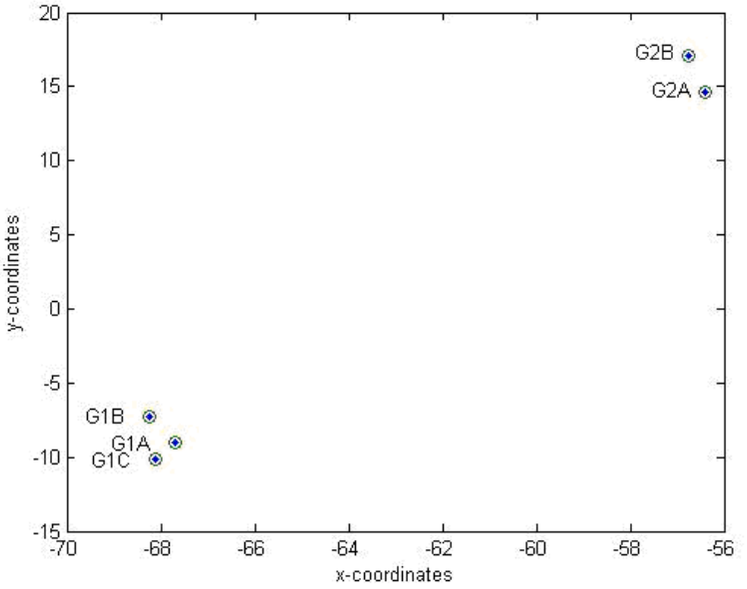


Figure 7. Two-dimensional plot of proteins for the 20 x 5 example.

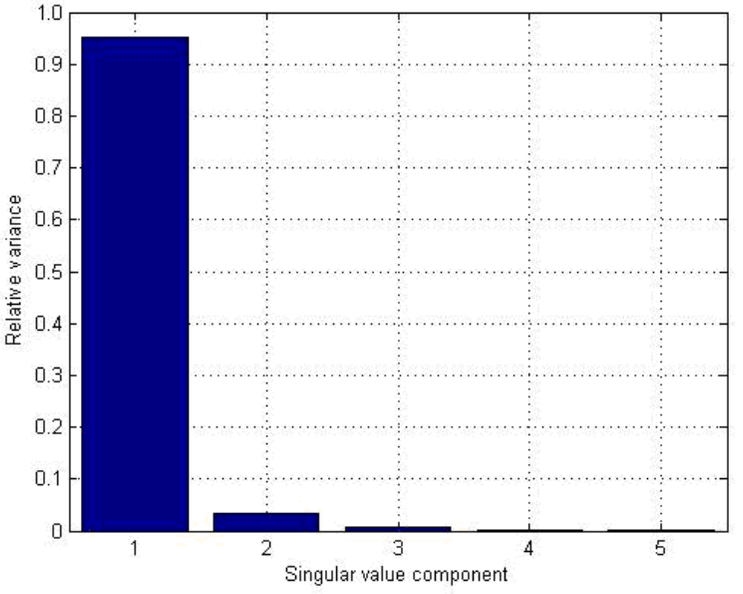


Figure 8. Relative variance plot of the 20 x 5-protein matrix of Figure 5.

Query retrieving algorithm

In the LSI information retrieval system built, it is possible to perform various comparisons: protein-by-protein, peptide-peptide, peptide-protein, and query-protein. Stuart et al. (2002a,b) and Stuart and Berry (2003, 2004) use these comparisons to build gene and species phylogenetic trees and to identify motifs.

Herein, the fundamental operation is the query-to-protein analysis, which allows the classification of the unknown gene (query) in one of the protein categories of the database. In this paper, the classification and retrieving algorithm applies a voting scheme based on the five most similar proteins with the unknown gene.

Since the query is not part of the original protein matrix (M), its vector (q) must be first generated and projected into the same form as a protein vector. The algorithms in Figure 4A and B can be used to generate the query vector q, which is modified according to Equation 5 to become another LSI protein vector:

$$q = q^T US^{-1} \qquad \text{(Equation 5)}$$

To compute similarity between the query vector and each of the protein vectors, to retrieve the most similar proteins with respect to the unknown gene, we can use many measures (Berry et al., 1995). The most often used similarity measures are the cosine of the angle and the Euclidean distance between the vectors. Despite the fact that some authors have recommended

cosine as the most effective similarity measure for text retrieval (Cöster, 1999; Kuruvilla et al., 2002; Orengo, 2004), we evaluated both measures for biomolecular sequence analysis.

The cosine of the angle between two vectors yields a value in the real range [-1.0, +1.0]. If the cosine is close to 1.0, it means that both vectors are in the same direction. A negative value close to -1.0 means that the vectors are in the opposite direction.

Two vectors define two points in the space. The Euclidean distance measures the absolute distance between the points defined by the vectors under comparison. This is a measure of neighborhood between vectors. The higher the similarity is between the two vectors, the smaller the Euclidean distance is.

The top five similar proteins with the query, by using either cosine or Euclidean distance, were used to define the category of the unknown sequence. This query is classified as a gene from a family that includes *t* most of these five sequences. For example, if the five most similar proteins with one query are from two different families A and B (Gene_A, Gene_B, Gene_B, Gene_A, and Gene_A, ordered by similarity with the query), the query is classified as a gene from family A. This method was called the voting algorithm.

The standard methods for comparisons among sequences are based on character-by-character alignments. Before applying the proposed LSI system, we analyzed the relationship between the two similarity measures with the edit distance, obtained from global sequence alignments using dynamic programming (Krawetz and Womble, 2003). In this way, it was possible to validate the method and to determine which similarity measure, cosine or Euclidean distance, is better to produce results approximately equal to the edit distance values. A correlation and a regression analysis (Neter et al., 1996) was performed to evaluate the relationship among the three similarity measures.

RESULTS AND DISCUSSION

To assess the correlation between the cosines, the Euclidean distance and a sequence alignment measure, 208 sequences from the first database and 200 from the second set were randomly selected and compared by using the global edit distance between each pair of sequences and respective cosines and Euclidean distances. The protein matrix was generated with tripeptide terms and reconstructed with 30 SVD factors (the definition of the number of SVD factors followed the relative variance criteria; Equation 4). The pairwise analysis generated 41,428 similarity measures. Despite the fact that we worked with quite different methods (LSI and global distance alignment), the correlation between the cosine and edit distance was -0.32 ($P < 0.01$) and between the Euclidean distance and edit distance was +0.70 ($P < 0.01$). These results indicate that Euclidean distance is better than the cosine in determining the similarity of sequences, when the objective is to achieve the same results as that observed with multiple alignments character-by-character (Figures 9 and 10). Actually, the square root of the Euclidean distance was better than the distance itself, with a Pearson correlation of 0.76 (Figure 10).

The negative correlation between the cosine and edit distance was expected. The higher the cosine of the angle between the two sequence vectors, higher the similarity was and, consequently, the smaller the edit distance. The Euclidean and edit distances showed the same behavior, and thus, the correlation was positive: the higher their values, the lower the similarity was between the two sequences.

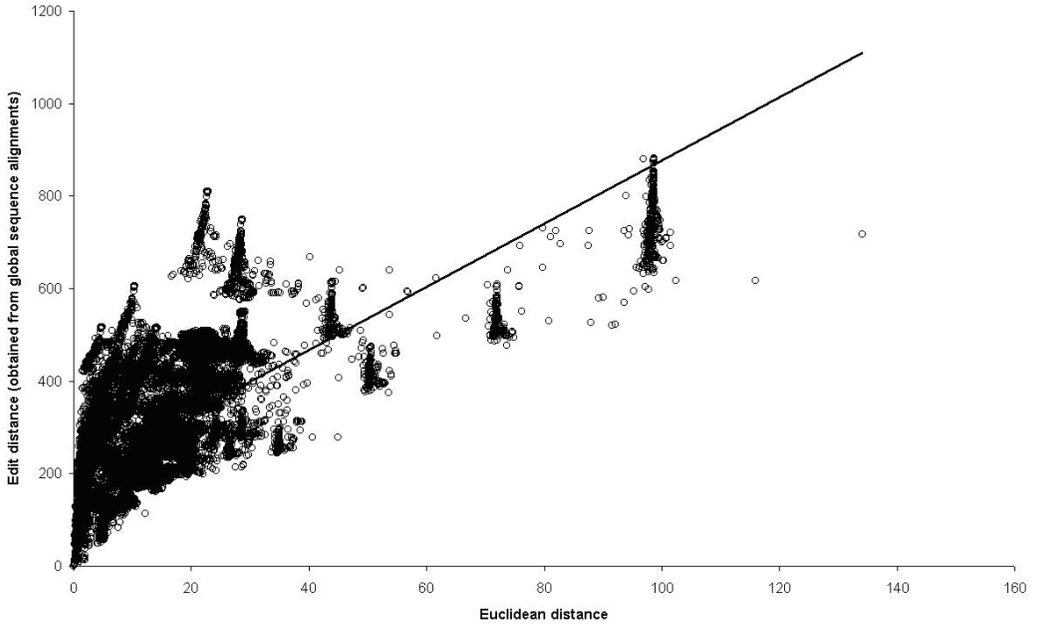


Figure 9. Scatter plot of Euclidean distance and global edit distance.

Similarity measure	Correlation coefficient with **Sij**
Sij = global edit distance of the unknown gene sequence **i** and protein sequence **j**	1.00
Cij = Cos(θ) = cosine of the angle θ between the query vector **q** and the protein vector	-0.32
Dij = Euclidean distance between the query vector **q** and the protein vector	0.70
$r_{ij} = \sqrt{D_{ij}}$	0.76
$DC_{ij} = C_{ij} \times D_{ij}$	-0.43

Figure 10. Correlation coefficient (r) between each singular value decomposition similarity measure and edit distance (Sij).

Despite the moderate correlation between Euclidean distance and edit distance (r = +0.76), it is possible to fit a linear model to estimate edit distance according to the Euclidean distance (Equation 6):

$$S_{ij} = 50 + 69 \times \sqrt{D_{ij}} \qquad \text{(Equation 6)}$$

where S_{ij} = edit distance (from a global sequence alignment), and D_{ij} = Euclidean distance.

After comparing SVD results with edit distance measure, we evaluated the ability of LSI to classify the sequences according to their categories. A sample of 202 sequences from the 13 gene families was randomly chosen as queries and the other proteins (630) were used to generate the p-peptide frequency matrix. For the second database, 735 sequences were selected to build the training set (the p-peptide frequency matrix), and 265 proteins were randomly selected as queries or test set. Figure 11 shows the file format of the original sequences from the first database. In Figure 12, we have part of the protein matrix of these data in the simplest case, where only one amino acid is used in the p-peptide term.

For both datasets, the protein frequency matrix was built by using the subroutines in Figure 4A and B, and the SVD was applied in each matrix that was reconstructed by using a number of factors defined by the relative variance analysis (Equation 4). The number of factors varied from 2 up to 56 (Figure 13). The advantage of the relative variance criteria is that

#Family	Gene and organism	Sequence
1	**COX3_Aame**	MAHQAHSYHMVDPSPWPIFGAAAALLTTSG...
2	**COX2_Aame**	MANHSQLGFQDASSPIMEELVEFHDHALIV...
3	**CYTB_Aame**	MAPNIRKSHPLLKMINNSLIDLPAPSNISA...
4	**ND4_Aame**	MLKIILPTIMLLPTTLLSPPKFLWTNTTMY...
5	**ND5_Aame**	MNATLLINSLTLLTLATLLTPIVFPLLFKN...
6	**ATP6_Aame**	MNLSFFDQFSSPYLLGIPLILLSLLFPALL...
7	**ND3_Aame**	MNMLTFMFSLSLALSAILTALNFWLAQMTP...
8	**ND2_Aame**	MNPHATPILVLSLMLGTTITISSNHWVLAW...
9	**ATP8_Aame**	MPQLNPAPWFSIMIMTWLTLALLIQPKLLT...
10	**ND1_Aame**	MPQMTMMSYLIMSLLYAIPILIAVAFLTLV...
11	**ND4L_Aame**	MSPLHLSFYSAFVLSGLGLAFHRTHLVSAL...
12	**COX1_Aame**	MTFINRWLFSTNHKDIGTLYLIFGAWAGMI...
13	**ND6_Aame**	MTYFVFFLGVCFVVGVLGVASNPSPYYGVV...
1	COX2_Ajam	MAYPFQLGLQDATSPIMEELLHFHDHTLMI...
2	COX1_Ajam	MFISRWFFSTNHKDIGTLYLLFGAWAGMVG...
3	ND4_Ajam	MLKIIIPTIMLMPLTWLSNPKMIWINSTAH...
4	ND6_Ajam	MMTYIVFVLSTIFVLSFVGFSSKPSPIYGG...
5	ATP6_Ajam	MNENLFASFITPTMMGLPIVILIIMFPTIM...
6	ND5_Ajam	MNLVSSMMLLSLMILSMPIMTTMLYPQNHP...
7	ND3_Ajam	MNMAITLLTNTFLASLLVMIAFWLPQTNSY...
8	ND2_Ajam	MNPIIFSMIMTTVILGTTIVMTSSHWLMVW...
9	ATP8_Ajam	MPQLDTSTWFITILATILTLFIIMQLKIST...
10	ND4L_Ajam	MSLTYMNMFMAFTISLLGLLMYRSHMMSSL...
11	COX3_Ajam	MTHQTHAYHMVNPSPWPLTGALSALLLTSG...
12	CYTB_Ajam	MTNIRKTHPLLKIINSSFVDLPAPSSLSSW...
13	ND1_Ajam	MYLMNLLTTIVPVLLAVAFLTLVERKILGY...
...	...	...

Figure 11. File format of the original sequence data from the first database.

p-peptide terms using one amino acid (p = 1)	Proteins																
	COX3 Aame	COX2 Aame	CYTB Aame	ND4 Aame	ND5 Aame	ATP6 Aame	ND3 Aame	ND2 Aame	ATP8 Aame	ND1 Aame	ND4L Aame	COX1 Aame	ND6 Aame	COX2 Ajam	COX1 Ajam	ND4 Ajam	...
A	23	16	30	31	60	17	14	36	3	30	7	46	18	9	42	32	...
C	1	3	5	3	7	0	1	1	0	2	3	1	2	3	1	3	...
D	5	13	7	2	6	1	3	1	0	4	1	15	3	12	14	5	...
E	8	14	7	10	14	4	6	5	0	11	3	10	3	12	10	8	...
F	24	8	31	15	34	8	9	12	2	18	6	42	14	6	43	17	...
G	19	9	23	20	33	8	5	11	0	14	6	47	26	8	47	17	...
H	18	10	11	15	15	4	0	9	0	2	6	19	0	8	18	12	...
I	16	18	30	41	49	19	6	25	3	24	4	43	1	19	36	46	...
K	4	4	10	10	23	4	1	14	3	7	0	9	0	5	9	10	...
L	32	30	63	102	107	61	30	65	8	61	18	63	26	32	60	94	...
M	9	8	9	25	31	10	4	18	3	18	7	25	6	16	29	34	...
N	4	5	21	11	25	9	2	13	2	10	3	15	1	5	17	20	...
P	12	14	25	28	31	17	7	24	10	25	4	30	4	11	29	23	...
Q	6	7	8	12	19	7	4	9	2	6	2	10	0	6	6	11	...
R	5	5	8	12	9	5	2	3	0	8	2	8	6	6	9	10	...
S	18	21	25	39	38	17	8	32	4	29	11	27	11	21	30	35	...
T	22	12	25	48	63	20	6	35	10	17	8	39	4	21	38	42	...
V	14	18	18	10	20	8	1	15	0	17	5	33	36	11	40	12	...
W	12	5	11	12	12	4	5	10	5	8	1	17	5	5	17	14	...
Y	9	8	13	13	11	4	2	8	0	14	1	17	7	11	19	14	...

Figure 12. Protein frequency matrix of the first database (p-peptide = 1 amino acid).

dimension reduction is done according to the information in the protein matrix itself, instead of using external data, as utilized by Stuart et al. (2002a). They used prior categorical information concerning family memberships, which could be difficult for unknown sequences. According to these authors, "the development of a procedure whereby optimal dimension can be approximated without reference to prior information would represent an important advancement" (Stuart et al., 2002b). This is done by using the relative variance criteria.

Number of amino acids in the p-peptide terms	1st database (630 reference sequences)		2nd database (735 reference sequences)	
	#SVD factors for the protein matrix reconstruct	Size of the original protein frequency matrix	#SVD factors for the protein matrix reconstruct	Size of the original protein frequency matrix
1	2	20 x 630	2	20 x 735
2	13	400 x 630	17	400 x 735
3	28	8,000 x 630	32	8,000 x 735
4	35	160,000 x 630	56	160,000 x 735

Figure 13. Dimension reduction according to the relative variance criteria. SVD = singular value decomposition.

In the first database, the best result was achieved with a 3-peptide frequency matrix (size of 8000 rows and 630 columns), reconstructed by SVD with 28 terms: all 202 queries were correctly classified into each of the 13 gene families, with 100% accuracy (Figure 14).

For the second database, 735 sequences were selected to build the p-peptide frequency matrices, and 265 proteins were randomly selected as queries. By using a 3-peptide frequency matrix (size of 8000 rows and 735 columns), reconstructed by SVD with 32 terms, we obtained a global accuracy of 72% in classifying the 265 queries in one of the nine protein categories.

Actual gene family	Classification of the gene query according to the voting algorithm													
	ATP6	ATP8	COX1	COX2	COX3	CYTB	ND1	ND2	ND3	ND4	ND4L	ND5	ND6	Total
ATP6	18													18
ATP8		10												10
COX1			16											16
COX2				12										12
COX3					22									22
CYTB						16								16
ND1							14							14
ND2								16						16
ND3									13					13
ND4										18				18
ND4L											16			16
ND5												15		15
ND6													16	16
Total	18	10	16	12	22	16	14	16	13	18	16	15	16	202

Figure 14. Cross classification table results of the first database.

We had 100% accuracy for cytochrome, 92% for histone, 85% for keratin, 80% for globin, 74% for collagen, 66% for cyclohydrolase, 55% for pyrophosphatase, 52% for ferredoxin, and 65% for other proteins (Figure 15).

Actual gene family	Classification of the gene query according to the voting algorithm									
	Collagen	Cyclohydrolase	Cytochrome	Ferredoxin	Globin	Histone	Keratin	Pyrophosphatase	Other	Total
Collagen	17			3	1	1		1	0	23
Cyclohydrolase	2	19			1			5	2	29
Cytochrome			21						0	21
Ferredoxin	1	1		14				3	8	27
Globin	1	1		1	16			1	0	20
Histone						23		1	1	25
Keratin					1	2	23	1	0	27
Pyrophosphatase		5		2	3	1		17	3	31
Other	3	4	0	6	2	2	0	5	40	62
Total	24	30	21	26	24	29	23	34	54	265

Figure 15. Cross classification table results of the second database.

CONCLUSIONS

The algorithm and methods presented estimate relatedness between large numbers of biomolecules without the requirement of multiple alignments. Proteins are recoded as p-peptide frequency values using all possible overlapping p-peptides, which generates a matrix, reduced by SVD.

The results show that the application of LSI to evaluate the similarity of sets of sequences is a promising method and very attractive, because sequence alignments are neither generated nor required. In order to achieve results similar to those observed using edit distance analysis, we recommend that Euclidean distance be used as a similarity measure for protein sequences in LSI methods.

In a randomly selected GenBank dataset, the results were very promising, with 72% accuracy for classifying unknown gene queries in one of the nine protein categories. However, in a curated protein database, the method was perfect in classifying the unknown genes according to their actual category. Besides using the method in classification analysis, the information retrieval system can be used to generate phylogenetic inferences by using whole genome sequences and global data analysis.

ACKNOWLEDGMENTS

We are thankful to Professor Gary W. Stuart from Indiana State University, Department of Life Sciences, who sent us helpful data.

REFERENCES

Berry MW, Dumais ST and O'Brien GW (1995). Using linear algebra for intelligent information retrieval. *SIAM Rev*. 37: 573-595.

Berry MW, Drmac Z and Jessup ER (1999). Matrices, vector spaces, and information retrieval. *SIAM Rev*. 41: 335-362.

Cöster R (1999). Learning from relevance feedback in latent semantic indexing. Master's thesis (Asker L, orienting professor). Stockholm University and Royal Institute of Technology, Stockholm.

Deerwester S, Dumais S, Furnas G, Landauer T, et al. (1990). Indexing by latent semantic analysis. *J. Am. Soc. Inf. Sci*. 41: 1-13.

Everitt BS and Dunn G (2001). Applied multivariate data analysis. 2nd edn. Arnold, London.

Krawetz AS and Womble DD (2003). Introduction to Bioinformatics: a theoretical and practical approach. Humana Press, Totowa.

Kuruvilla FG, Park PJ and Schreiber SL (2002). Vector algebra in the analysis of genome-wide expression data. *Genome Biol*. 3: RESEARCH0011.

Neter J, Kutner MH and Nachstheim C (1996). Applied linear statistical models. 4th edn. Ie-McGraw-Hill, Boston.

Orengo VM (2004). Assessing relevance using automatically translated documents for cross-language information retrieval. PhD thesis (Huyck C, orienting professor), Middlesex University, London.

Rodrigues TS, Pacífico LGG, Teixeira SMR, Oliveira SC, et al. (2004). Clustering and artificial neural networks: classification of variables lengths of *Helminth* antigens in set of domains. *Genet. Mol. Biol*. 27: 673-678.

Schalkoff RJ (1992). Pattern recognition: statistical, structural and neural approaches. 1st edn. John Wiley & Sons Inc., New York.

Stuart GW and Berry MW (2003). A comprehensive whole genome bacterial phylogeny using correlated peptide motifs defined in a high dimensional vector space. *J. Bioinform. Comput. Biol*. 1: 475-493.

Stuart GW and Berry MW (2004). An SVD-based comparison of nine whole eukaryotic genomes supports a coelomate rather than ecdysozoan lineage. *BMC Bioinformatics* 5: 204.

Stuart GW, Moffett K and Baker S (2002a). Integrated gene and species phylogenies from unaligned whole genome protein sequences. *Bioinformatics* 18: 100-108.

Stuart GW, Moffett K and Leader JJ (2002b). A comprehensive vertebrate phylogeny using vector representations of protein sequences from whole genomes. *Mol. Biol. Evol*. 19: 554-562.

The Mathworks (1996). MATLAB: mathematical computation, analysis, visualization, and algorithm development (Version 5.0). Natick, Massachussetts, USA.

Thorne JL (2000). Models of protein sequence evolution and their applications. *Curr. Opin. Genet. Dev*. 10: 602-605.

Wall ME, Rechtsteiner A and Rocha LM (2003). Singular value decomposition and principal component analysis. In: A practical approach to microarray data analysis (Berrar DP, Dubitzky W and Granzow M, eds.). Kluwer, Norwell, 91-109.

Capítulo 3 – Revelando processos biológicos por meio de Álgebra Linear: extraindo padrões de dados com ruído

UNREVEALING BIOLOGICAL PROCESS WITH LINEAR ALGEBRA: EXTRACTING PATTERNS FROM NOISY DATA

Bráulio Roberto Gonçalves Marinho Couto
Centro Universitário de Belo Horizonte / UNI-BH, Av. Professor Mário Werneck 1685, Belo Horizonte, Brazil
braulio.couto@unibh.br

Marcelo Matos Santoro
Departamento de Bioquímica e Imunologia, UFMG, Av. Antonio Carlos 6627, Belo Horizonte, Brazil
santoro@icb.ufmg.br

Marcos Augusto dos Santos
Departamento de Ciência da Computação, UFMG, Av. Antonio Carlos 6627, Belo Horizonte, Brazil
marcos@dcc.ufmg.br

Keywords: linear algebra; data mining; information retrieval; SVD.

Abstract: Extracting patterns from protein sequence data is one of the challenges of computational biology. Here we use linear algebra to analyze sequences without the requirement of multiples alignments. In this study, the singular value decomposition (SVD) of a sparse *p*-peptide frequency matrix (M) is used to detect and extract signals from noisy protein data ($\mathbf{M = USV^T}$). The central matrix S is diagonal and contains the singular values of M in decreasing order. Here we give sense to the biological significance of the SVD: the singular value spectrum visualized as *scree* plots unreveals the main components, the process that exists hidden in the database. This information can be used in many applications as clustering, gene expression analysis, immune response pattern identification, characterization of protein molecular dynamics and phylogenetic inference. The visualization of singular value spectrum from SVD analysis shows how many processes can be hidden in database and can help biologists to detect and extract small signals from noisy data.

1 INTRODUCTION

Many bioinformatics tools are designed to detect patterns in protein or DNA sequences by using statistically based sequence similarity methods. The patterns detected can be associated with the function or structural protein stability, can predict family genes or can be used to describe the evolving relationship of group sequences (Hunter, 1993). Such bioinformatics predictions help experimental determination simpler and more efficient (King *et al*., 2001). However, to evaluate how two proteins are similar is a complex issue. The standard methods quantify the similarity between two proteins using global or local alignments with their primary sequences. The goal is to find the optimal alignment, quantifying it by some metric. In this work, instead of using alignment analysis, the approach applied is based on linear algebra algorithms, similar to that used in systems for information retrieval in large textual databases and by Google™ web search engine. The ideas and linear algebra methods applied here are important in several areas of data mining, pattern recognition (for example, classification of hand-written digits), and PageRank computations for web search engines (Eldén, 2006). Our objective is to use singular value decomposition – SVD (Berry *et al*., 1995) of a sparse tripeptide frequency matrix to detect and extract signals from noisy protein data. Such analysis, when done in micro array gene expression data, associates the number of the most significant singular values from SVD with the gene groups and the cell-cycle structure (Wall *et al*., 2003).

We will analyze the singular value spectrum to visualize them and to unreveal the main components, the number of process that exists hidden in the database. More specifically, as an application of SVD, we want to show that the number of the most significant singular values is associate with the number of protein families in a sequence database. Such prediction can be used in phylogenetic inference, data mining, clustering etc, making experimental tests more efficient, and avoiding randomly determination for possible outcomes.

2 SYSTEM AND METHODS

Programs implemented for this analysis were written in MATLAB (The Mathworks, 1996), using its inbuilt functions (SVD, sparse matrix manipulation subroutines etc). Four datasets were used in this paper. The first evaluated database had 64 vertebrate mitochondrial genomes composed of 832 proteins from 13 known gene families (ATP6, ATP8, COX1, COX2, COX3, CYTB, ND1, ND2, ND3, ND4, ND4L, ND5 and ND6). This curated protein database was downloaded from online information by Stuart *et al*. paper (Stuart *et al*., 2002). The second database was composed by sequences from proteins retrieved from GenBank in 19/04/2006. It is a random 100 sequences sample of each protein type: globin, cytochrome, histone, cyclohydrolase, pyrophosphatase, ferredoxin, keratin and collagen and 200 other proteins, totalling 1,000 sequences from ten different types of genes. The third database was the file "pdb_seqres.txt.gz", located in http://bioserv.rpbs.jussieu.fr/PDB/. This file has 121,556 redundant protein sequences from PDB (Protein Data Bank), which was reduced to 37,561 non-identical sequences. From this file we recovered all sequences related to six types of enzymes: Ligase, Isomerase, Lyase, Hydrolase, Transferase and Oxidoreductase, which totalled 10,915 proteins. We also recovered a sample of 219 globins from the PDB file that was used as another test set. Besides, we extracted 86 sequences of haemoglobin alpha-chain and a sample from the PDB file with all sequences higher than 47 amino acids (31,906 proteins from several types of genes). Each of the above sequence files was analyzed by MATLAB subroutines that generate twelve tripeptide sparse matrices as described by Stuart (Stuart *et al*., 2002) and adapted by Couto (Couto *et al*., 2007).

All sequences were recoded as 3-peptide frequency values using all possible overlapping tripeptide window. With 20 amino-acids it is generated a matrix M (8,000 x n), where n is the number of proteins to be analyzed. After the generation of the tripeptide frequency matrix (M), the matrix itself is subjected to SVD (Deerwester *et al.*, 1990; Berry *et al*., 1995) and factorized as $M = USV^T$. Where U is the p x p orthogonal matrix having the left singular vectors of M as its columns, V is the n x n orthogonal matrix having the right singular vectors of M as its columns, and S is the p x n diagonal matrix with the singular values $\sigma_1 \geq \sigma_2 \geq \sigma_3 \ldots \geq \sigma_r$ of **M** in order along its diagonal (r is the rank of M or the number of linearly independent columns or rows of M). These singular values are directly related to independent characteristics within the dataset. Actually, the largest values of (S) provide the meaning of the peptides and proteins in the matrix (M). On the other hand, the smaller singular values identify less significant aspects and the noisy inside the dataset (Eldén, 2006).

In this work our focus is only in the matrix (S) and its diagonal values (s_i) that make up the singular value spectrum. The magnitude of any singular value is indicative to its importance in explaining the data (Wall *et al*., 2003). Then, the objective here is to visualize the singular value spectrum as plots that help biologists to discover the main components, the process, and the groups hidden in the database. Two graphs were built:

a) the *scree* plot, with 25 bigger singular values for each database;
b) the cumulative relative variance (V_i) captured by the ith-singular value: $V_i = 1 - (S_i)^2/\sum_k(S_k)^2$; S_i = ith-singular value; k = 1, 2, … n.

The visual examination of the *scree* plot looks for a “gap” or an “elbow” that indicates how many significant singular values exist in database. After the “gap” there is no significant value. The second graph helps to understand how much variance is explained by each singular value. Despite the effort for automatic analysis, graphic visual inspection still is one of the most commonly used in practice for dimensionality selection (Zhu and Ghodsi, 2006).

3 RESULTS

When there is only one specific type of protein in database, as haemoglobin alpha-chain, the singular value spectrum obtained shows a “big gap” after the first eigenvalue (Figure 1.1). Such result is confirmed by the second graph (Figure 1.2) that indicates more than 90% variance is explained by

the first singular value, which is compatible with the database itself. For the globin matrix (Figures 2.1 and 2.2) is more difficult to define exactly where the "gap" or "elbow" is, because there are more than one type protein in database. However, the objective here is not to be very precise, but sufficiently accurate to help biologists in finding an interval with the number of process or groups that exists hidden in the database. Such predictions need validation by experimental determination that becomes simpler. In the globin database for example, is reasonable to define between one and three groups that explains about 60% of the variance in database (Figure 2.2). After the third singular value there is stability in the singular value spectrum (Figure 2.1).

For the database with 13 mitochondrial genes (Figures 3.1 and 3.2) it is possible to define the number of groups around 10: after this interval the singular value spectrum stabilizes and there is between 50% and 60% explained variance. When the GenBank matrix is analyzed, with ten different types of genes, it is necessary carefully combine both graphs. Despite the fact that there is a "gap" after the sixth singular value (Figure 4.1), the variance explained until this point is only about 40% (Figure 4.2). The interval between 10 and 15 singular values corresponds to about 50% of relative variance and the spectrum becomes flat.

The PDB database, with more then 31,000 proteins from several types of genes, presents a singular value spectrum where is necessary more than 20 eigenvalues to explain about 30% of variance. There is an "elbow" between the second and third singular value (Figure 5.1) that is insufficient to explain most data (Figure 5.2). Similar result is obtained with the PDB enzymes database that apparently had only 6 types of proteins. The visual analysis of the scree plot and cumulative variance graph (Figures 6.1 and 6.2) suggest more than 25 groups hidden under the six enzymes denomination. This is a clue, a possibility that should be analyzed by another bioinformatics tool.

Table 1 summarizes the visualization of all singular value spectrums for each database, plotted in the Figures 1.1 to 6.2. The suggested numbers of significant singular values for each dataset is coherent, except the enzymes database, which seems to be actually formed by several quite different sequences. SVD analysis unreveals biological motives associated with biological processes and other biological properties in each dataset.

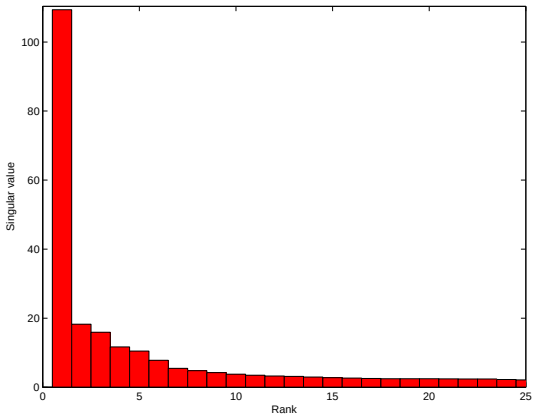


Figure 1.1: Scree plot showing singular values of haemoglobin α-chain database.

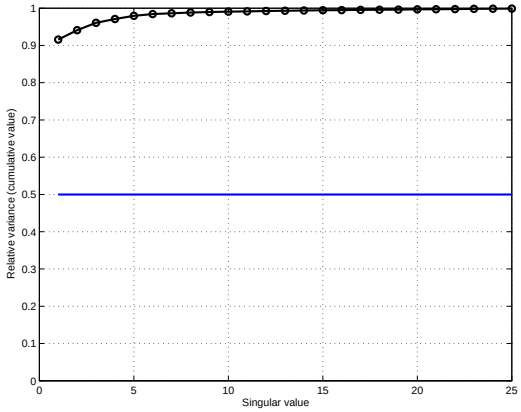


Figure 1.2: Cumulative relative variance of haemoglobin α-chain database.

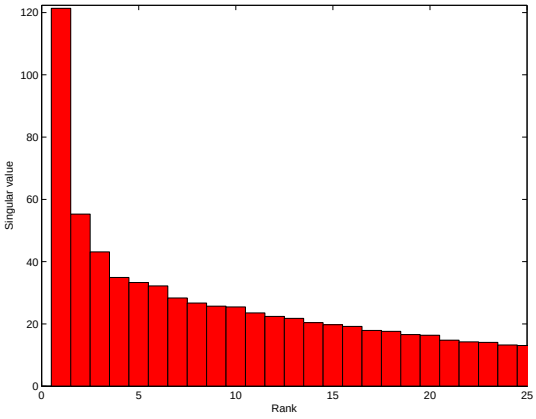


Figure 2.1: Scree plot showing singular values of globin database.

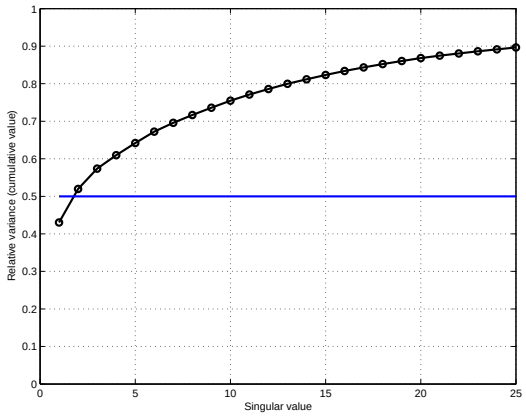


Figure 2.2: Cumulative relative variance of globin database.

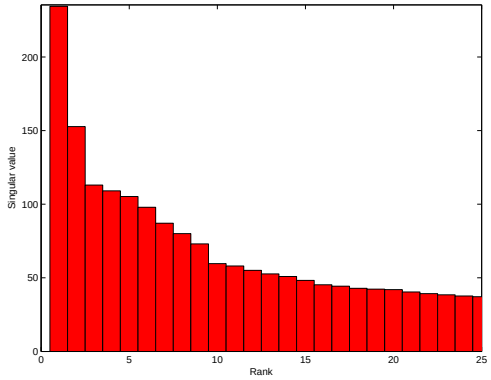


Figure 3.1: Scree plot showing singular values of mitochondrial genes database.

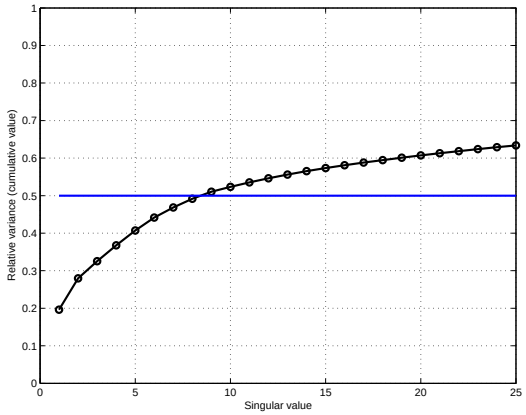


Figure 3.2: Cumulative relative variance of mitochondrial genes database.

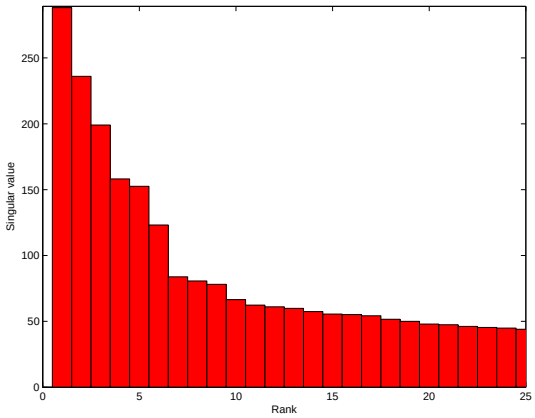


Figure 4.1: Scree plot showing singular values of sample genes from GenBank.

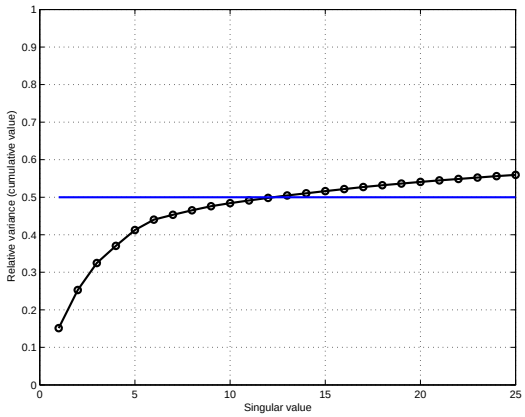


Figure 4.2: Cumulative relative variance of sample genes from GenBank.

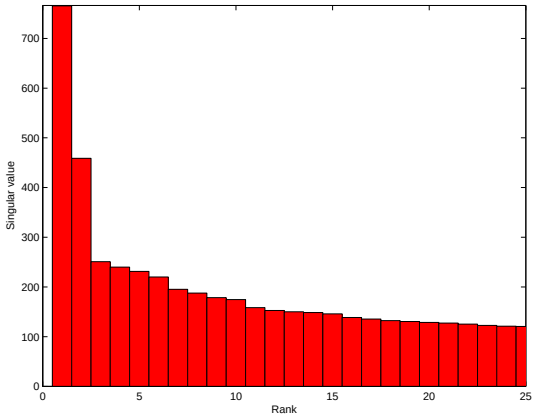


Figure 5.1: Scree plot showing singular values of random PDB sequences dataset.

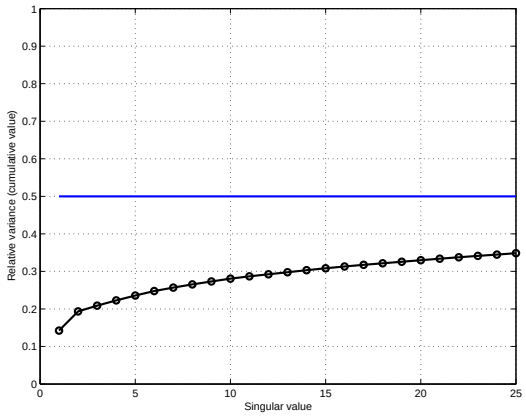


Figure 5.2: Cumulative relative variance of random PDB sequences dataset.

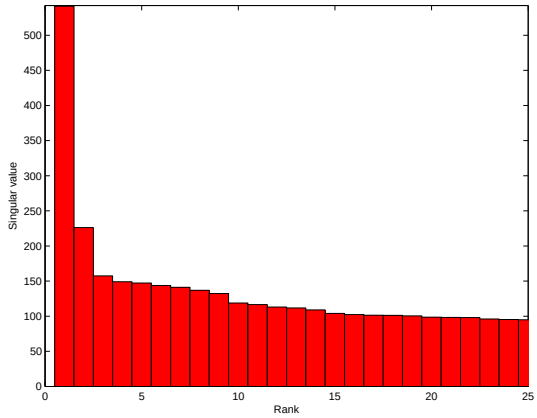


Figure 6.1: Scree plot showing singular values of PDB enzymes database.

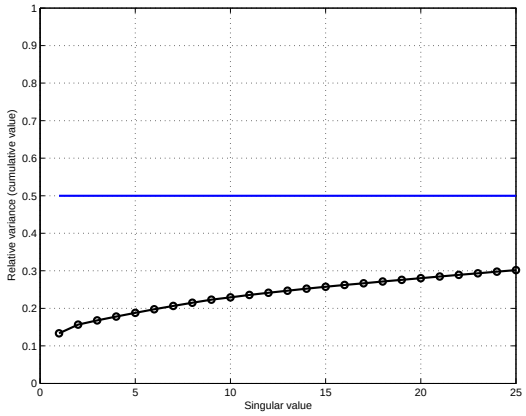


Figure 6.2: Cumulative relative variance of PDB enzymes dataset.

4 CONCLUSION

A biologist could ask: "What is the biological significance of the SVD?" We answered this question: the visualization of singular value spectrum from SVD analysis shows how many process can be hidden in database. The singular value plot is a suggestion, a clue that helps biologists to detect and extract small signals from noise data.

Table 1: Suggested number of significant singular values.

Dataset	Predefined # groups	Suggested number singular values	
		Min	Max
Haemoglobin α-chain	1	1	1
Globin	1	1	3
Mitochondrial genes	13	9	15
GenBank	10	10	15
PDB sequences	Several	> 20	
Enzymes	6	> 25	

REFERENCES

Berry, M.W. *et al*., 1995. Using linear algebra for intelligent information retrieval. SIAM Review, 37, 573-595.

Couto, B.R.G.M. *et al*., 2007. Application of latent semantic indexing to evaluate the similarity of sets of sequences without multiple alignments character-by-character. GMR, 6(4), 983-999.

Deerwester, S. *et al*., 1990. Indexing by Latent Semantic Analysis. Journal of the American Society for Information Science, 41(6), 1-13.

Eldén, L., 2006. Numerical linear algebra in data mining. Acta Numerica, 327-384.

Hunter, L., 1993. Artificial Intelligence and Molecular Biology. American Association for Artificial Intelligence, MIT Press, Cambridge.

King, R.D. *et al*., 2001. The utility of different representations of protein sequence for predicting functional class. Bioinformatics, 17(5): 445-454.

Stuart, G.W. *et al*., 2002. Integrated gene and species phylogenies from unaligned whole genome protein sequences. Bioinformatics, 18(1), 100-108.

The Mathworks, 1996. MATLAB: mathematical computation, analysis, visualization, and algorithm development (version 5.0). Natick, Massachusetts, USA.

Wall, M.E. *et al*., 2003. Singular value decomposition and principal component analysis. In: Berrar, D.P. et al. (eds.), A practical approach to microarray data analysis, Kluwer, Norwell, pp. 91-109.

Zhu, M. and Ghodsi, A, 2006. Automatic dimensionality selection from the scree plot via the use of profile likelihood. Computational Statistics and Data Analysis, 51, 918-930.

Capítulo 4 – Usando modelos de regressão logística e decomposição em valores singulares para a seleção de atributos importantes para classificação de sequências protéicas

Feature selection for protein sequence classification by using logistic regression models and singular value decomposition

Braulio RGM Couto[1,2,*§], Marcelo M Santoro[3], Amjad Ali[4], Marcos A Santos[5*]

[1]*Programa de Doutorado em Bioinformática*, Universidade Federal de Minas Gerais, UFMG, Belo Horizonte, Minas Gerais, Brasil

[2]*Departamento de Ciências Exatas e Tecnologia*, Centro Universitário de Belo Horizonte, UNI-BH, Belo Horizonte, Minas Gerais, Brasil

[3]*Departamento de Bioquímica e Imunologia*, UFMG, Belo Horizonte, Minas Gerais, Brasil

[4]*Laboratory of Molecular and Cellular Genetics (LGCM), Departamento de Biologia Geral, ICB/UFMG*, UFMG, Belo Horizonte, Minas Gerais, Brasil

[5]*Departamento de Ciência da Computação*, UFMG, Av. Antonio Carlos 6627, Belo Horizonte, Minas Gerais, 31270-010, Brasil

*These authors contributed equally to this work

[§]Corresponding author

Email addresses:

BRGMC: braulio.couto@unibh.br

MMS: santoro@icb.ufmg.br

AA: amjad_uni@yahoo.com

MAS: marcos@dcc.ufmg.br

Abstract

Background

Searching for relevant patterns in protein sequences is a critical Bioinformatics goal. In this work we will present a computational tool to support genomic research that uses logistic regression models and singular value decomposition to feature selection and protein sequence classification. Firstly, we consider a biomolecular sequence as a complex written language that is recoded as *p*-peptide frequency vector using all possible overlapping *p*-peptides window. With 20 amino acids it generates a 20^p high-dimensional vector, where p is the word-size. Each vector row is the peptide that is analyzed by logistic regression to feature selection for the protein sequence classification. If we use a word-size window (p=1) one of the features analyzed, the amino acids are important for a group of proteins. With p=2 we can identify bipeptides associated with a specific sequences group. Besides peptides we include sequence length as another feature candidate. The model-building strategy for the feature selection was an automatic forward stepwise logistic regression. After the feature selection step, proteins are recoded again only by the *p*-peptides selected as important for each sequences group. The rank of the protein frequency matrix produced for each target group is reduced by singular value decomposition (SVD) and the results are used to classify unknown sequences. A database with 516,081 sequences from the Swiss-Prot section of the Universal Protein Resource (UniProt) was the protein collection used in all analysis. We tested the method in seven target groups: insulin, globin, keratin, cytochrome and proteins related with cystic fibrosis, Alzheimer disease and schizophrenia. A case-control study was done to examine each target group. In this approach, sequences from the target group (the cases) are selected from database for comparison with series of random sequences where the protein is

absent (the controls). For all groups, available number of cases in database is fixed and restricted, much smaller than the number of controls. In order to try an optimal allocation of cases and controls during each feature selection analysis, we used a 1:4 case: control ratio. The ratio of four random controls to each case (4:1) compensates few numbers of cases, being enough to detect features related to each protein group.

Results

Combined method was able to identify the amino acids and bipeptides important to each protein group. Sensitivity to classify unknown sequences using the SVD system based on the initial matrix with 400 rows, ranged from 76% for proteins related with Alzheimer disease and more than 90% for other six groups. All specificities were over 90% for all proteins. After frequency matrix reconstruction using only bipeptides identified by the logistic regression, decomposition by SVD and subsequent rank reduction, query retrieval has a sensitivity ranging from 74% for cytochrome to more than 90% for globin, keratin and proteins related to cystic fibrosis and schizophrenia. As for the initial matrix, all specificities in this situation were over 90% for all proteins.

Conclusions

In addition to the feature selection, combining logistic regression models with singular value decomposition method allows better classification of unknown sequences than using SVD alone. Matrices used by the combined method are much smaller than the original one, which leads to optimized oracles. The tool is perfectly scalable and adaptable to huge problems because it is independent of reference database size and much less from the length of involved sequences.

Background

Searching for relevant patterns in protein sequences is a critical Bioinformatics goal. Detected patterns can be used to classify unknown sequences predicting genes family or can be used to describe evolving relationship of sequences group, instead of being associated with the function or structural proteins stability [1]. Here we present a computational tool to support genomic research using a new method based on logistic regression models to feature of protein sequence classification selection.

Firstly, we consider a bio molecular sequence as a written language that is recoded as *p*-peptide frequency vector using all possible overlapping *p*-peptides window. The methodology was developed by Stuart, Moffett and Baker, to generate whole genome phylogenies using vector representations of proteins sequence, and adapted by Couto *et al.* [2, 3]. Each row of frequency vector is a peptide that is analyzed by logistic regression to feature selection for protein sequence classification. With p=2 we can identify bipeptides associated with a specific sequences group. We also include sequence length as another feature candidate in logistic model that is built for each target protein. After feature selection, the second objective is to classify unknown protein sequences. This can be done by logistic models built in the first step and by using singular value decomposition (SVD) [4]. Third objective is to identify amino acids associated with a specific sequences group. The aim is to find amino acids which presence and absence, in terms of frequency in the sequence, is a pattern that can be used to identify a specific protein group.

It is important to observe that, instead of use any alignment analysis, the approach applied is based on SVD, a linear algebra method. This technique is similar as used in information retrieval systems in large textual databases and Google™ web search engine. Linear algebra is known as an efficient approach to deal with semantic relationships between a large numbers of elements in spaces of high dimensionality. However, before using a linear algebra method, we need to represent proteins as vectors in a high dimensional space, and then calculate similarities among them. So, protein is represented as a vector built by frequency of all possible overlapping *p*-peptides along the sequence. Before using the chosen protein vector representation, it is necessary to discuss two issues: initially, there is a problem when a protein is recoded as a frequency vector of *p*-peptides because the order of each *p*-peptide in the sequence is not considered. The second issue, there is a necessity to evaluate if Euclidean distance and cosine, similarity metrics used by linear algebra, are suitable to evaluate biological similarities among proteins.

First question was discussed in a previous work, when we concluded that the representation ambiguity is a theoretical possibility in principle but not in practice because two different proteins do not occupy the same point in the high dimensional space defined by the frequency *p*-peptides matrix [5]. Second question was also discussed in other report where the relationship among similarity metrics from SVD, cosine and Euclidean distance, and alignments statistics used by BLAST were assessed [6]. In that work, we chose to compare SVD with BLAST because this string-matching program is widely used for searching of nucleotide and protein databases [7]. We achieved similar results between BLAST and SVD in several protein analyses and concluded that SVD can be used to protein-protein comparisons

with biological significance of the similarities identified both for cosine and Euclidean distance [6]. Before these analysis, we had already evaluated in two different situations the relationship between cosine and Euclidean distance with the edit distance, obtained from global sequence alignments using dynamic programming [3,8]. In both studies, edit distance, Euclidean distance and cosine were strongly correlated. Euclidean distance was chosen here because our previous results recommend that this measure is better than cosine to evaluate similarities of proteins represented as vectors [3, 6, and 8].

Methods

Figure 1 presents a flowchart summarizing the entire method. First step is to get a reference database with sequences of known proteins. A file with 516,081 curated sequences from the Swiss-Prot section of the Universal Protein Resource (UniProt) was the protein collection used in all analysis [9]. A case-control study was done to examine each target group. We tested the method in seven instances: insulin, globin, keratin, cytochrome and proteins related with cystic fibrosis, Alzheimer disease and schizophrenia. In this approach, sequences from the target group (the cases) are selected from database for comparison with a series of random sequences where protein is absent (the controls). For all groups, available number of cases in database is fixed and restricted, much smaller than the control numbers. In order to try an optimal allocation of cases and controls during each feature selection analysis, we used a 1:4 case: control ratio. The ratio of four random controls to each case (4:1) compensates a few numbers of cases, being enough to detect features related with each protein group [10].

In all analysis, protein sequences are recoded as *p*-peptide frequency vectors using all possible overlapping *p*-peptides window, which generates sparse matrices as described by Stuart [2] and adapted by Couto *et al.* [3]. With 20 amino-acids are generated 20^p high-dimensional vectors, where *p* is the word-size. Each vector row is the peptide that is analyzed by logistic regression to feature selection for the protein sequence classification. If we use a word-size window of one (p=1) the features considered are the amino-acids alone. This needs to be done when we want to identify amino acids associated to a specific sequences group, ie, when we want to find amino acids which presence and absence, in regard to frequency in the sequence, is a pattern that can be used to identify a specific group of protein. With p=2 there are 400 bipeptides which frequency can be associated to a specific group of sequences. In both analysis, ie, when p=1 or p=2, logistic regression models are build and can be used for feature selection. We worked with only with p=1 and p=2 because, if the vector was built with a word size higher, case numbers in database will be unable to allow a logistic regression adjustment model. For example, with p=3 the number of candidate features is 8,000, too much for the sample size available.

A regression model was developed for each case-control study, allowing feature selection. Each logistic model can also be used to predict the probability ($\boldsymbol{\pi}$) of a sequence to be from a specific type of protein for any combination of the k explanatory features in the model:

$$\pi = \frac{\exp\left(\beta_0 + \sum_{i=1}^{k} \beta_i X_i\right)}{1 + \exp\left(\beta_0 + \sum_{i=1}^{k} \beta_i X_i\right)}$$

In the above equation, $\boldsymbol{\pi}$ is the probability of a sequence belongs to the case group, **k** is the number of explanatory features significantly selected for the model and β_i is the coefficient regression for each feature (i = 1, 2, 3 …k). The model-building strategy for the feature selection was an automatic forward stepwise logistic regression performed by SPSS (SPSS Inc., 2008).

After the bipeptides selection, the classification of an unknown protein sequences can be done by using singular value decomposition. To classify an unknown sequence by SVD, proteins are recoded again only by the bipeptides selected as important for each target group. If m bipeptides are selected by the logistic regression, it generates a matrix m x n, where n is the number of sequences analyzed. This new matrix (**M**) produced for each target group and smaller than the first one that had size 400 x n, is decomposed by SVD ($\mathbf{M=USV^T}$)[11]. **U** is the m x m orthogonal matrix having the left singular vectors of **M** as its columns, **V** is the n x n orthogonal matrix having the right singular vectors of **M** as its columns, and **S** is the m x n diagonal matrix with the singular values $\sigma_1 \geq \sigma_2 \geq \sigma_3 \geq ... \sigma_r$ of **M** in order along (r is the rank of **M** or the number of linearly independent columns or rows of **M**). Before analyzes a query (unknown sequence) a rank reduction of the frequency matrix **M** is done by using the k-largest singular values of M and generating the matrix $\mathbf{M_k} = S_k V_k^T$. In this work we used the method proposed by Everitt and Dunn [12] that recommends analyzing the relative variances of singular values from SVD. Relative variance (V_i) captured by the ith-singular value is equal to $(S_{i,i})^2/\sum_k(S_{k,k})^2$; k = 1, 2, … r. Singular values which relative variance is less than 0.7/n, where n is the number of proteins in the matrix **M**, must be ignored. An unknown sequence is also recoded only by the m bipeptides selected by the logistic model, which generates the query vector (q). This vector query

is projected to reduced space of SVD, which produces $q^*=q^T U_k$. After that, we compute Euclidean distance among the vector query (q*) and all protein vector in the reduced space ($\mathbf{M_k}$). The query category is defined by the protein, of all sequences from the case-control study, with shortest distance to the vector query.

Sensitivity and specificity are calculated to evaluate the quality of SVD in classifying unknown sequences (queries) [13]. Sensitivity is the chance a known type sequence to be correctly identified. Specificity measures the chance of a sequence from other type than the target group to be negatively classified. All source code was implemented in MATLAB (The Mathworks, 1996).

Results

The method summarized in Figure 1 was applied in seven target proteins: insulin, globin, keratin, cytochrome and proteins related with cystic fibrosis, Alzheimer disease and schizophrenia. All cases and controls were selected from Swiss-Prot (http://www.uniprot.org/downloads), which is a section from the UniProt Knowledgebase [9]. Swiss-Prot downloaded file contains manually annotated and reviewed protein with 516,081 sequences. Tables 1 to 7 present amino acids which presence and absence, in terms of frequency in the sequence, is a pattern that can be used to identify a specific protein group. All analysis were made considering a significant level of 0.05 (α = 5%) after forward stepwise logistic regression. The odds ratio for each amino acid, calculated by exp (β_i), where β_i is the regression coefficient, summarizes the direction and frequency importance of each amino acid to characterize a gene. If odds ratio is higher than 1.0, the amino acid must be in the

sequence for a gene. If odds ratio is less than 1.0, the amino acid must be out the sequence to characterize a specific gene. For example, each cysteine in a sequence increases the chance of the sequence to be insulin in 1.84 times (table 1). On the other hand, each aspartate in a sequence reduces 0.81 times the chance of the sequence to be insulin.

Table 1 also shows that each residue of cysteine, leucine, arginine and tyrosine increases the chance of a sequence to be insulin. For globin, higher is the number of histidine, tryptophan, aspartate, phenylalanine, lysine, alanine, leucine or valine, higher is the chance of the sequence to be this type of protein (table 2). Keratin sequences (table 3) have a higher number of cysteine, serine, tyrosine, glutamate, glutamine, glycine and arginine, with a smaller length. Table 4 shows how the presence of phenylalanine, histidine and tyrosine in a sequence is important to a cytochrome. Presence of phenylalanine, isoleucine, leucine, arginine and serine increases the probability of a sequence to be associated with cystic fibrosis (table 5). Proteins associated with Alzheimer disease have a higher number of cysteine, glutamate, proline and valine in the sequence (table 6). The presence of histidine, glutamine and serine is associated with proteins related to schizophrenia (table 7). We are focusing on a residue presence in the sequence to characterize a type of protein, but in all analysis both situation, ie presence and absence, are described by the results in tables 1 to 7.

After identifying amino acids related to a protein, we made a feature selection for sequence classification. For all seven target groups, bipeptides and sequence length were analyzed by forward stepwise logistic regression. Table 8 presents, as an

example, results for insulin, showing important features to characterize this type of protein. Only eight bipeptides from 400 are important for an insulin sequence. Actually, double cysteines (CC), glycine followed by cysteine (GC) and tyrosine followed by cysteine (YC) increases the chance of a sequence to be insulin. On the other side, a presence of the following bipeptides reduces the chance of a sequence to be insulin: histidine and tyrosine (HY), methionine with cysteine (MC), arginine with cysteine (RC) and valine with aspartate (VD). Bipeptide frequency pattern of cytochrome is summarized in table 9. Only eleven bipetides are important to classify this kind of protein. Same analysis was made for all target groups and can be achieved for any other protein group.

To test the system during a query classification, unknown sequences were randomly selected from database and classified by SVD. The classification quality of this sample queries with SVD were summarized in table 10. Results for the original SVD, without the feature selection by logistic regression, the number of features selected by logistic regression, sensitivity and specificity to the SVD query retrieval system made after feature selection are presented in table 10. Sensitivity to classify unknown sequences using the SVD system based on the initial matrix with 400 rows, ranged from 76% for proteins related with Alzheimer disease and more than 90% for the other six groups. All specificities were over 90% for all proteins (table 10).

After reconstruction of the frequency matrix using only bipeptides identified by logistic regression, decomposition by SVD and subsequent rank reduction, query retrieval has a sensitivity ranging from 74% for cytochrome to more than 90% for

globin, keratin and proteins related to cystic fibrosis and schizophrenia. As an initial matrix, all specificities in this situation were over 90% for all proteins (table 10).

Discussion

Logistic regression was able to identify the amino-acids and bipeptides important to each proteins group. All bipeptides identified by the method can be associated, for example, to sequence motifs widespread over the protein. However, the whole understanding of the biological significance of these findings needs to be evaluated by another bioinformatics tools and/or by experimental assays, which will be analyzed in future. All results presented by tables 1 to 9 are examples of what is possible to do by applying the method synthesized in figure 1.

In relation to the information retrieval system based on the combined method, logistic regression with SVD, results in table 10 are very promising. Although the system build using the original matrix, without the feature selection, seems to be better, both methods have similar behavior. Except for cytochrome, all sensitivity and specificity are approximately the same. However, it is crucial to observe that the matrix produced only by the frequency of bipeptides selected as important for each sequences group by the logistic regression is much smaller than the original matrix. All initial matrices have 400 rows, while the matrix produced after logistic regression analysis has a number of rows that varies from five to schizophrenia until 51 for globin (table 10). The matrix size, in terms of rows, is defined by the number of bipeptides associated with each target protein. If we intent to use SVD for analyzing huge databases this reduction is very attractive for information retrieval. This rank pre-reduction by

logistic regression shows that method is scalable and adaptable to the problem, for example a whole genome phylogeny. When we look for the second rank reduction, based on relative variances of singular values from SVD, the number of factors needs to query retrieval in the combined method, SVD/logistic regression, is only between two and three. The number of factors used in the rank reduction by SVD of the original matrix, with 400 rows, ranged from 13 to 22. These results suggest that, while the original matrix seems to have many groups, the new matrix has few groups (two or three). Combining both methods SVD and logistic regression produces perfect oracle, where only approximately two groups are present.

Conclusions

Proposed method, that combines logistic regression models with singular value decomposition, was successfully tested in seven instances. Application of logistic regression for selecting a feature set associated to a specific gene may represent a possible technique to detect hard finding patterns in protein sequences. All evaluated cases instead of initial 400 bipeptides, logistic regression showed that only few bipeptides are necessary to characterize specific proteins groups. These peptides can be related with hidden motifs in protein and should be analyzed by another bioinformatics tools.

In addition to the feature selection, the method also allows a classification of unknown sequences in SVD/logistic regression retrieval system composed by oracles. Actually, the entire system is composed by k oracles, built for specific proteins. Each oracle is made by n cases and approximately 4n controls selected randomly from

reference database and built according to the schema in figure 1. A query is then compared k-times, one in each oracle, which generates all possibilities for an unknown sequence. The success in classifying queries was equal to the original system, without pre-rank reduction by logistic regression. However, matrices sizes used by the combined method are much smaller than the original one, which leads to optimized oracles. The tool is perfectly scalable and adaptable to huge problems because it is independent from reference database size and much less from length of the sequences involved.

Authors' contributions

BRGMC conceived the research, initiated the study, carried out the implementations, data analysis and wrote the manuscript preparation. MMS and AA gave final approval for the manuscript to be published. MAS participated in computational analysis, result interpretation and manuscript writing. All authors read and approved the final text.

Acknowledgements

We are grateful to Marlon Castro de Souza from UNI-BH, who revised the manuscript.

References

1. Hunter L: **Artificial Intelligence and Molecular Biology.** American Association for Artificial Intelligence, MIT Press, Cambridge, 1993

2. Stuart GW, Moffett K, Baker S: **Integrated gene and species phylogenies from unaligned whole genome protein sequences**. Bioinformatics 2002, **18**(1): 100-108
3. Couto BRGM, Ladeira AP, Santos MA: **Application of latent semantic indexing to evaluate the similarity of sets of sequences without multiple alignments character-by-character.** GMR 2007, **6**(4): 983-999
4. Berry MW, Dumais ST, O'Brien GW: **Using linear algebra for intelligent information retrieval**. SIAM Review 1995, **37**:573-595
5. Couto BRGM, Leão IRF, Santoro MM, Santos MA: **Vector representation of protein sequences [abstract]**. X-meeting 2009 - 5th International Conference of the Brazilian Association for Bioinformatics and Computational Biology, 2009. [http://lgmb.fmrp.usp.br/xmeeting2009/abstractbook/pages/150.pdf]
6. Couto BRGM, Campos FF, Santoro MM, Santos MA: **Association among similarity metrics of latent semantic indexing and BLAST statistics [abstract]**. X-meeting 2009 - 5th International Conference of the Brazilian Association for Bioinformatics and Computational Biology, 2009. [http://lgmb.fmrp.usp.br/xmeeting2009/abstractbook/pages/149.pdf]
7. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ: **Basic local alignment search tool**. J. Mol. Biol. 1990, **215**:403-410
8. Marcolino LS, Couto BRGN, Santos MA: **Genome visualization in space**. Advances in Soft Computing, 2010, **74**(2010):225-232
9. The UniProt Consortium: **The Universal Protein Resource (UniProt) in 2010.** Nucleic Acids Res 2010, **38**(suppl 1):D142-D148
10. Schlesselman JJ: **Case-Control Studies.** Oxford U. Press, New York, 1982

11. Deerwester S, Dumais S, Furnas G, Landauer T, Harshman R: **Indexing by Latent Semantic Analysis**. Journal of the American Society for Information Science 1990, **41**(6):1-13
12. Everitt BS, Dunn G: **Applied multivariate data analysis.** 2nd edn. Arnold, London, England, 2001
13. Altman DG: **Practical Statistics for Medical Research**. Chapman & Hall, London, 1991

Figures

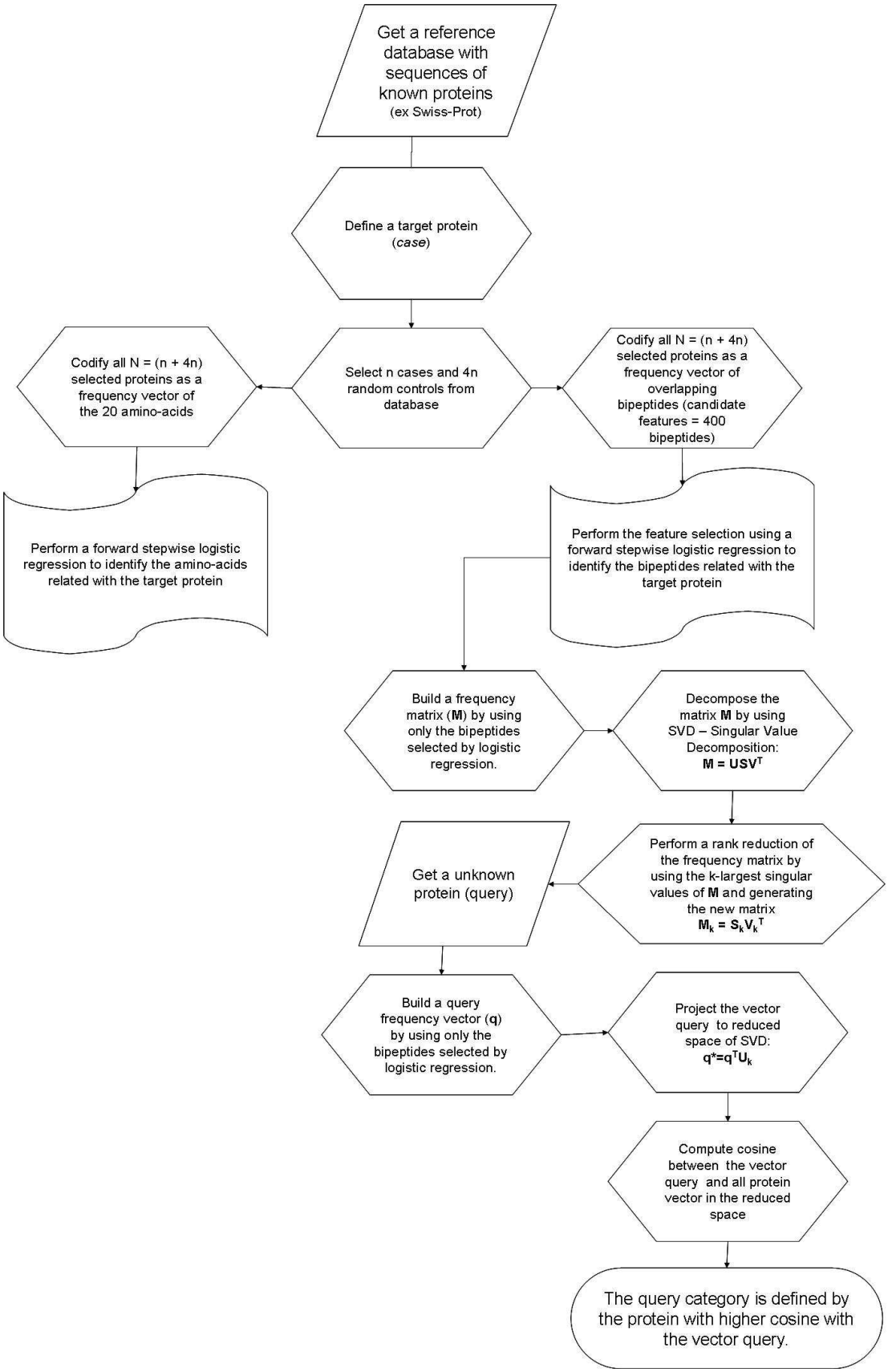


Figure 1 – Flowchart with steps for the feature selection and classification of unknown sequence by logistic regression and SVD.

Tables

Table 1 – **Amino acids which frequencies are associated with insulin.**

Amino acid	Regression coefficient	Standard error	Wald c^2 values	P values	Odds Ratio
Alanine	-0.1	0.0	4.2	0.039	0.91
Cysteine	0.6	0.1	64.3	0.000	1.84
Aspartate	-0.2	0.1	10.9	0.001	0.81
Glycine	-0.2	0.1	10.4	0.001	0.85
Isoleucine	-0.2	0.1	11.6	0.001	0.79
Leucine	0.1	0.0	10.4	0.001	1.09
Asparagine	-0.2	0.1	7.2	0.007	0.81
Arginine	0.2	0.0	20.3	0.000	1.18
Threonine	-0.2	0.1	14.1	0.000	0.79
Valine	-0.2	0.1	6.7	0.010	0.84
Tyrosine	0.2	0.1	4.0	0.046	1.20
Constant	1.9	0.4			

Table 2 – **Amino acids which frequencies are associated with globin.**

Amino acid	Regression coefficient	Standard error	Wald χ^2 values	P values	Odds Ratio
Alanine	0.1	0.0	45.0	0.000	1.1
Cysteine	-0.2	0.1	15.9	0.000	0.8
Aspartate	0.2	0.0	32.1	0.000	1.2
Glutamate	-0.3	0.0	98.2	0.000	0.7
Phenylalanine	0.2	0.0	29.1	0.000	1.2
Glycine	-0.2	0.0	38.8	0.000	0.8
Histidine	0.5	0.0	149.3	0.000	1.6
Isoleucine	-0.3	0.0	85.3	0.000	0.7
Lysine	0.2	0.0	56.0	0.000	1.2
Leucine	0.1	0.0	8.4	0.004	1.1
Methionine	-0.2	0.1	18.6	0.000	0.8
Proline	-0.4	0.0	76.7	0.000	0.7
Glutamine	-0.1	0.0	16.7	0.000	0.9
Arginine	-0.2	0.0	37.4	0.000	0.8
Threonine	-0.1	0.0	7.4	0.007	0.9
Valine	0.1	0.0	15.9	0.000	1.1
Tryptophan	0.3	0.1	26.8	0.000	1.4
Tyrosine	-0.1	0.0	4.2	0.040	0.9
Constant	0.1				

Table 3 – **Amino acids which frequencies are associated with keratin.**

Amino acid/variable	Regression coefficient	Standard error	Wald χ^2 values	P values	Odds Ratio
Sequence length	-0.03	0.0	19.1	0.000	0.97
Cysteine	0.34	0.0	74.9	0.000	1.41
Aspartate	-0.21	0.1	14.6	0.000	0.81
Glutamate	0.20	0.0	43.2	0.000	1.22
Phenylalanine	-0.32	0.1	28.9	0.000	0.72
Glycine	0.10	0.0	15.6	0.000	1.11
Histidine	-0.22	0.1	8.9	0.003	0.80
Lysine	-0.13	0.0	15.1	0.000	0.87
Proline	-0.35	0.0	53.1	0.000	0.70
Glutamine	0.13	0.0	19.4	0.000	1.14
Arginine	0.08	0.0	6.2	0.013	1.08
Serine	0.31	0.0	70.5	0.000	1.36
Tryptophan	-0.44	0.1	11.4	0.001	0.64
Tyrosine	0.23	0.1	19.7	0.000	1.26
Constant	-1.15				

Table 4 – **Amino acids which frequencies are associated with cytochrome.**

Amino acid	Regression coefficient	Standard error	Wald χ^2 values	P values	Odds Ratio
Alanine	-0.2	0.1	12.8	0.000	0.8
Cysteine	-0.3	0.1	4.4	0.036	0.8
Phenylalanine	0.3	0.1	18.4	0.000	1.4
Histidine	0.6	0.1	19.9	0.000	1.8
Lysine	-0.6	0.1	30.3	0.000	0.6
Glutamine	-0.5	0.1	22.1	0.000	0.6
Serine	-0.2	0.1	10.2	0.001	0.8
Tyrosine	0.5	0.1	17.8	0.000	1.6
Constant	1.1				

Table 5 – **Amino acids which frequencies are associated with proteins associated with cystic fibrosis.**

Amino acid	Regression coefficient	Standard error	Wald χ^2 values	P values	Odds Ratio
Alanine	-0.1	0.0	15.0	0.000	0.9
Phenylalanine	0.1	0.0	8.0	0.005	1.2
Histidine	-0.3	0.1	10.6	0.001	0.8
Isoleucine	0.1	0.0	9.5	0.002	1.1
Leucine	0.1	0.0	15.8	0.000	1.1
Asparagine	-0.3	0.1	30.1	0.000	0.7
Proline	-0.3	0.1	20.3	0.000	0.7
Arginine	0.1	0.0	10.2	0.001	1.1
Serine	0.2	0.0	19.9	0.000	1.2
Valine	-0.1	0.0	10.4	0.001	0.9
Tyrosine	-0.2	0.1	7.4	0.007	0.8
Constant	-0.4				

Table 6 – **Amino acids which frequencies are associated with proteins associated with Alzheimer disease.**

Amino acid	Regression coefficient	Standard error	Wald χ^2 values	P values	Odds Ratio
Alanine	-0.1	0.0	6.0	0.015	0.9
Cysteine	0.4	0.1	16.1	0.000	1.4
Glutamate	0.1	0.0	4.5	0.034	1.1
Phenylalanine	-0.2	0.1	13.1	0.000	0.8
Asparagine	-0.1	0.0	3.0	0.083	0.9
Proline	0.1	0.0	7.0	0.008	1.1
Arginine	-0.2	0.1	7.9	0.005	0.9
Valine	0.2	0.0	11.9	0.001	1.2
Constant	-2.0				

Table 7 – **Amino acids which frequencies are associated with proteins associated with schizophrenia.**

Amino acid	Regression coefficient	Standard error	Wald χ^2 values	P values	Odds Ratio
Glycine	-0.2	0.1	4.0	0.046	0.8
Histidine	0.6	0.2	8.1	0.005	1.7
Lysine	-0.3	0.1	9.6	0.002	0.7
Glutamine	0.4	0.1	9.5	0.002	1.6
Serine	0.3	0.1	11.4	0.001	1.3
Valine	-1.1	0.3	11.6	0.001	0.3
Constant	1.9				

Table 8 – **Stepwise regression logistic analysis for insulin.**

Bipeptide/ variable	Regression coefficient	Wald χ^2 values	P values
CC	6.5	24.7	0.000
CG	2.7	18.0	0.000
HY	-3.1	2.9	0.049
MC	-2.9	4.8	0.028
RC	-3.6	9.7	0.002
VD	-2.1	7.0	0.008
YC	5.3	18.0	0.000
YD	-3.0	5.2	0.023
Sequence length	0.01	26.3	0.000
Constant	-1.4		

Table 9 – **Stepwise regression logistic analysis for cytochrome.**

Bipeptide	Regression coefficient	Wald χ^2 values	P values
DE	-4.1	13.5	0.000
DM	-5.4	19.7	0.000
FF	1.2	14.1	0.000
FH	2.7	16.7	0.000
FM	1.9	14.3	0.000
KM	1.1	4.3	0.039
KV	-4.1	7.8	0.005
LK	-3.0	20.7	0.000
MH	3.0	24.6	0.000
SA	-1.8	12.9	0.000
VW	2.9	15.3	0.000
Constant	-2.3		

Table 10 – **Quality of the classification of sample queries with SVD.**

	SVD using the matrix with 400 rows		#bipeptides used to build the frequency matrix (k)	SVD using the matrix with k rows	
Protein	Sensitivity	Specificity		Sensitivity	Specificity
Insulin	94%	97%	8	89%	98%
Globin	96%	100%	51	92%	98%
Keratin	99%	100%	17	93%	98%
Cytochrome	91%	98%	11	74%	96%
Cystic fibrosis	91%	99%	20	95%	93%
Alzheimer	76%	94%	7	80%	91%
Schizophrenia	94%	94%	5	94%	94%

Capítulo 5 – Sistema de recuperação de sequências protéicas baseado em modelos de regressão logística

Protein sequence retrieval system based on logistic regression models

Bráulio R. G. M. Couto[1], Marcelo M. Santoro[2] and Marcos A. dos Santos[3]

Programa de Doutorado em Bioinformática, UFMG and Curso de Ciência da Computação, Centro Universitário de Belo Horizonte / UNIBH[1], Departamento de Bioquímica e Imunologia, Universidade Federal de Minas Gerais / UFMG[3], Departamento de Ciência da Computação, UFMG[3]; Av. Antonio Carlos 6627, Belo Horizonte, Minas Gerais, 31270-010, Brasil. Email addresses: BRGMC: braulio.couto@unibh.br; MMS: santoro@icb.ufmg.br; MAS: marcos@dcc.ufmg.br.

Abstract We present a method that utilizes information from known protein databases to build logistic regression models that allow prediction of a new amino acids sequence. First step is to represent a protein as *p*-peptide frequency vector using all possible overlapping *p*-peptides window. Each vector frequency row becomes a peptide that is analyzed by logistic regression to feature selection for protein sequence prediction. A file with curated sequences from the Swiss-Prot was the protein collection used in all analysis. A case-control study was done to study each target group. In this approach, sequences from the target group (the cases) are selected from database for comparison with series of random sequences where protein is absent (the controls). We tested the method in ten instances, generating ten models for predicting insulin, globin, keratin, cytochrome, albumin, collagen, fibrinogen and proteins related with cystic fibrosis, Alzheimer disease and schizophrenia. Sensitivity to classify unknown sequences ranged from 72% for collagen to 100% for keratin. Specificities were higher than 90% for all 10 groups. The method was successfully tested in all instances.

1. Introduction

To predict the protein type of a new sequence encode is one of the Bioinformatics objectives. This can be solved for example by searching for similarities among the newly sequence and previous sequences from a database, which usually is made by pair-wise alignment methods (Altschul *et al.* 1990). In this report, we present a method that utilizes information from known protein database to build logistic regression models that allow the prediction of a new amino acids sequence. After the model is built there is no more database searching or any comparison among the new sequence and known proteins.

First step is to represent a protein as *p*-peptide frequency vector using all possible overlapping *p*-peptides window. This methodology was developed by Stuart,

Moffett and Baker, for generating whole genome phylogenies using vector representations of proteins sequence (Stuart *et al*. 2002). Each vector frequency row represents a peptide that is analyzed by logistic regression to feature selection for the protein sequence prediction. Here we recode protein using p=2, that allows us to identify bipeptides which frequencies can be used to predict a specific proteins group. The sequence length is also a feature candidate in the logistic model that is built for each target protein. With the logistic regression models built we can predict the type of protein that an unknown amino acids sequence encodes.

It is important to observe that we chose to recode the protein as a vector built by the frequency of all possible overlapping bipeptides along the sequence. Before use any regression method on this (new) protein, we need to validate this kind of representation. The first problem when a protein is recoded as a frequency vector of p-peptides is a possibility of ambiguity representation because the order of each *p*-peptide in the sequence is not considered. So, two different proteins could be represented by the same vector. The second issue is the necessity to evaluate if protein vectors are suitable for meaningful biological analysis.

The first question was discussed in a previous work, when we concluded that the ambiguity representation is a theoretical possibility in principle but not in practice because two different proteins do not occupy the same point in the high dimensional space defined by frequency p-peptides matrix (Couto *et al*. 2009a). The second question was also discussed in other report where the relationship among similarity metrics calculated with protein vectors and pair-wise alignments statistics were assessed (Couto *et al*. 2009b). In that work, we achieved similar results among the metrics calculated when protein are recoded by p-peptides window and alignments statistics performed by BLAST (Altschul *et al.* 1990). In another two papers (Couto *et al*. 2007; Marcolino *et al*. 2009), we had already evaluated the relationship between the metrics calculated with protein vectors, actually cosine and Euclidean distance, with the edit distance, obtained from global sequence alignments using dynamic programming. Both studies, edit distance and Euclidean distance and edit distance and cosine were strongly correlated. These previous analysis indicate that protein vectors are suitable for meaningful biological analysis. So, we can analyze the bipeptides from the vector representation by logistic regression in order to build predictive models for proteins target. This is the objective of this work.

2. Material and methods

A flowchart of the protein sequence retrieval system is summarized in Figure 1. First step is to get a reference database with sequences of known proteins. A file with 516,081 curated sequences from the Swiss-Prot section of the Universal Protein Resource (UniProt) was the protein collection used in all analysis (The UniProt Consortium 2010). A case-control study was done to study each target group. We tested the method in ten instances: insulin, globin, keratin, cytochrome, albumin, collagen, fibrinogen and proteins related with cystic fibrosis, Alzheimer dis-

ease and schizophrenia. In this approach, sequences from the target group (the cases) are selected from database for comparison with a series of random sequences where protein is absent (the controls). For all groups, the number of available cases in database is fixed and restricted, much smaller than the number of controls. In order to try an optimal allocation of cases and controls during each feature selection analysis, we used a 1:4 case:control ratio. The ratio of four random controls to each case (4:1) compensates the few number of cases, being enough to detect the features related with each group of protein (Schlesselman 1982).

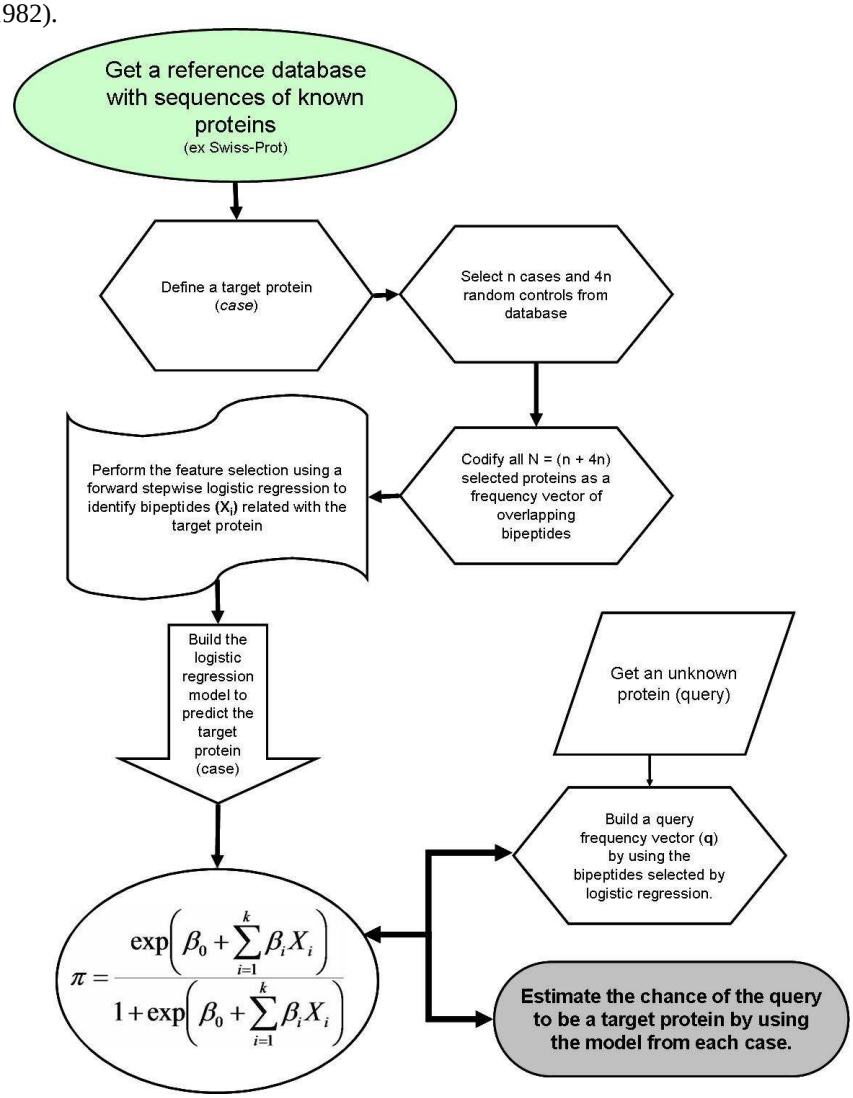


Fig. 1. Flowchart with steps for building the logistic regression models and to predict the type of protein from an unknown sequence.

In all analysis, protein sequences are recoded as *p*-peptide frequency vectors using all possible overlapping p-peptides window, that generates sparse matrices as described by Stuart *et al*. (2002). With 20 amino-acids are generated 20^p high-dimensional vectors, where *p* is the word-size. Each vector row is the peptide that is analyzed by logistic regression to feature selection for the protein sequence prediction. With *p*=2 there are 400 bipeptides that frequency can be associated to a specific sequences group. Logistic regression models are built and can be used for feature selection. We worked with only *p*=2 because, if the vector was built with a word size higher, the number of cases in database will be unable to allow a logistic regression model adjustment. For example, with p=3 the number of candidate features is 8,000, too much for the sample size available.

A regression model was developed for each case-control study, allowing feature selection. Each logistic model can also be used to predict the probability (π) of a sequence to be from a specific type of protein for any combination of the k explanatory features in the model:

$$\pi = \frac{\exp\left(\beta_0 + \sum_{i=1}^{k} \beta_i X_i\right)}{1 + \exp\left(\beta_0 + \sum_{i=1}^{k} \beta_i X_i\right)} \qquad \textbf{(1)}$$

In the above equation, $\boldsymbol{\pi}$ is the probability of a sequence belongs to the case group, k is the number of explanatory features significantly selected for the model and β_i is the coefficient regression for each feature (i = 1, 2, 3 …k). The model-building strategy for the feature selection was an automatic forward stepwise logistic regression performed by SPSS (SPSS Inc., 200b8). The entire system, with the logistic models obtained, becomes an application developed in MATLAB (The Mathworks, 1996). This protein sequence retrieval system predicts the type of protein a new sequence encodes, considering ten groups tested here.

Sensitivity and specificity were calculates to evaluate the discriminant quality of the logistic models in classifying unknown sequences. Sensitivity is the chance a known sequence of a type to be correctly identified. Specificity measures the chance of a sequence from other type than the target group to be negatively classified. Definition of the best cut-off for the probabilities calculated by each model in order to classify a query sequence was made by ROC – 'receiver operating characteristic' curve analysis (Altman 1991).

3. Results and discussion

The method summarized in Figure 1 was applied in ten target proteins: insulin, globin, keratin, cytochrome, albumin, collagen, fibrinogen and proteins related with cystic fibrosis, Alzheimer disease and schizophrenia. All cases and controls were selected from Swiss-Prot (http://www.uniprot.org/downloads), which is a

section from the UniProt Knowledgebase. Swiss-Prot file downloaded contains manually annotated and reviewed protein with 516,081 sequences.

Tables 1 and 2 present bipeptides where presence and absence, in terms of frequency in the sequence, is a pattern that can be used to identify a specific protein group. All analysis were made considering a significant level of 0.05 (α = 5%) after forward stepwise logistic regression. The odds ratio for each bipeptide, calculated by $\exp(\beta_i)$, where β_i is the regression coefficient, summarizes the direction and importance of the frequency of each bipeptide to characterize a gene. If odds ratio is higher than 1.0, the bipeptide must be in the sequence for a gene. If odds ratio is less than 1.0, the bipeptide must be out the sequence to characterize a specific gene. Table 1 presents, as an example, the results for insulin, showing the features importance to characterize this type of protein. Only eight bipeptides from 400 are important for an insulin sequence. Actually, double cysteines (CC), glycine followed by cysteine (GC) and tyrosine followed by cysteine (YC) increase the chances of a sequence to be insulin. On the other side, presence of the following bipeptides reduces the chance of a sequence to be insulin: histidine and tyrosine (HY), methionine with cysteine (MC), arginine with cysteine (RC) and valine with aspartate (VD). The bipeptide frequency pattern of cytochrome is summarized in table 2. Only eleven bipetides are important to classify this kind of protein. The number of explanatory variables for each protein target ranged from only five bipeptides for proteins related with schizophrenia to 51 for globin. Same analysis was made for all target groups and can be achieved for any other protein group.

Figures 2 and 3 show the behavior of the logistic model for globin when the frequency of two bipeptides is changed in a sequence. When all other 50 bipeptides important to globin are kept constant, small changes in the cysteine-tyrosine frequency increases the chance of the sequence to be globin. On the other side, in the same condition, when the cysteine-tyrosine frequency increases the chance of the sequence to be globin dramatically reduces.

Tab. 1. Stepwise logistic regression analysis for insulin.

Bipeptide/ variable	Regression coefficient (β_i)	Wald χ^2 values	P values
CC	6.5	24.7	0.000
CG	2.7	18.0	0.000
HY	-3.1	2.9	0.049
MC	-2.9	4.8	0.028
RC	-3.6	9.7	0.002
VD	-2.1	7.0	0.008
YC	5.3	18.0	0.000
YD	-3.0	5.2	0.023
Sequence length	0.01	26.3	0.000
Constant	-1.4		

Tab. 2. Stepwise logistic regression analysis for cythocrome.

Bipeptide/ variable	Regression coeffi-cient (β_i)	Wald χ^2 values	P values
DE	-4.1	13.5	0.000
DM	-5.4	19.7	0.000
FF	1.2	14.1	0.000
FH	2.7	16.7	0.000
FM	1.9	14.3	0.000
KM	1.1	4.3	0.039
KV	-4.1	7.8	0.005
LK	-3.0	20.7	0.000
MH	3.0	24.6	0.000
SA	-1.8	12.9	0.000
VW	2.9	15.3	0.000
Constant	-2.3		

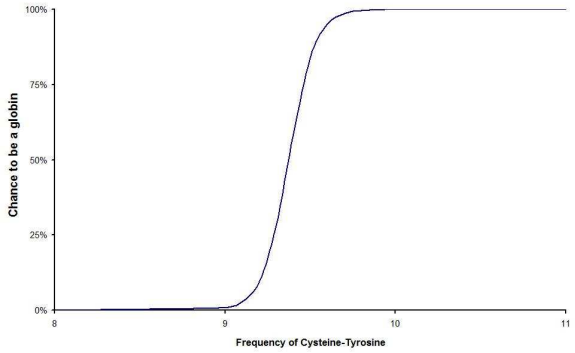


Fig. 2. Example of a bipeptide (cysteine-tyrosine) which presence increases the chance of a sequence to be globin.

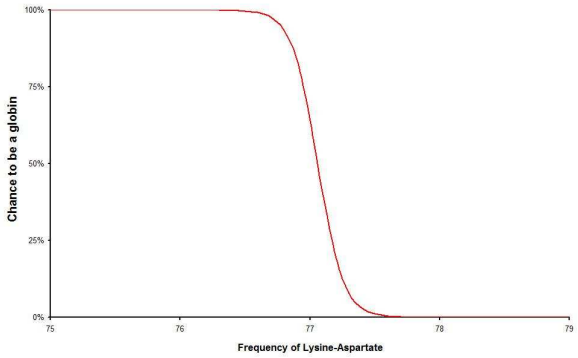


Fig. 3. Example of a bipeptide (lysine-aspartate) which presence reduces the chance of a sequence to be globin.

To test the system during a query classification, unknown sequences were randomly selected from the database and classified by each logistic model. Since the results of logistic equation (1) are a probability value ranging from 0.0 to 1.0, we need to choose a cut off to define if a sequence is the target group. Actually, logistic regression allows us to distinguish those sequences likely or unlikely to be a specific gene providing a probability value. Usually the cut off is 0.50, meaning that if the probability of the sequence to be a target group is higher than 0.50, then the sequence classified as the case modeled by the equation. ROC curves were made for all ten target groups (Fig. 4 and 5). We got a better discrimination in six sequences type: keratin, insulin, globin, cytochrome and proteins related with cystic fibrosis and Alzheimer disease (Fig. 4). The other 4 proteins had a good result but worse than the firsts (Fig. 5). The best cut off in probability is 33%, which maximizes the sum of the sensitivity and specificity, being nearest the top left-hand corner of both ROC curves.

Classification quality of sample queries is summarized in table 3. Sensitivity to classify unknown sequences ranged from 72% for collagen to 100% for keratin. Specificities were higher than 90% for all 10 groups. Since we used 33% as cut off, if the probability model for a query is higher than 0.33 so the sequence is classified as belonging to the target group.

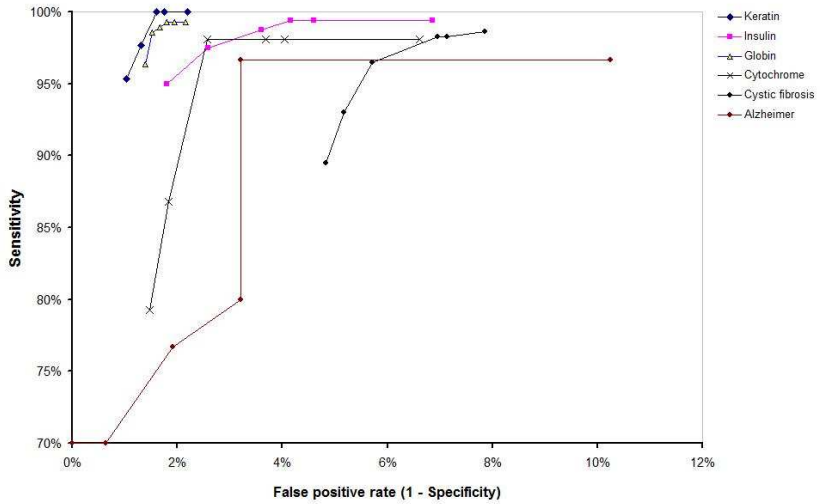


Fig. 4. ROC curve analysis for keratin, insulin, globin, cytochrome and proteins related with cystic fibrosis and Alzheimer disease.

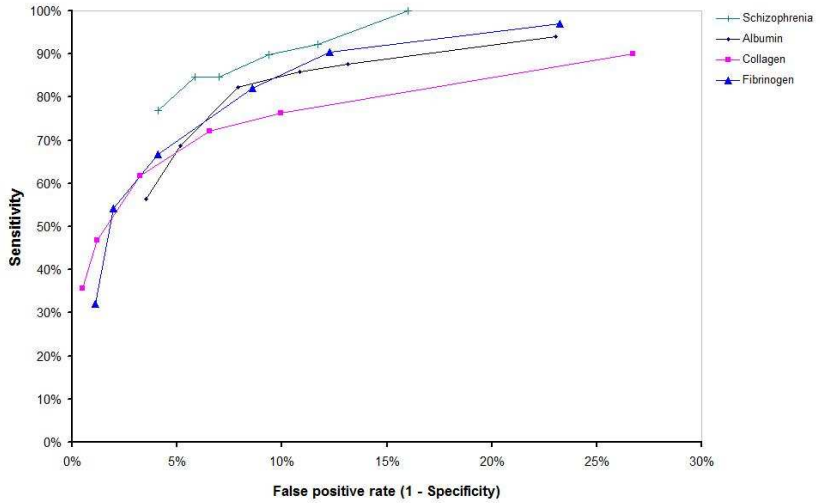


Fig. 5. ROC curve analysis for keratin, insulin, globin, cytochrome and proteins related with cystic fibrosis and Alzheimer disease.

Tab. 3. Classification quality of sample queries with logistic regression.

	Sample queries		Classification using a cut-off = 33% in logistic probability	
Protein	N (cases)	n (controls)	Sensitivity	Specificity
Insulin	160	890	99%	96%
Globin	275	1438	99%	98%
Keratin	128	683	100%	98%
Cytochrome	53	272	98%	96%
Cystic fibrosis	114	560	98%	93%
Alzheimer	30	156	80%	97%
Schizophrenia	39	171	90%	91%
Albumin	170	912	86%	89%
Collagen	999	4794	72%	93%
Fibrinogen	472	2291	82%	91%

4. Conclusion

The method was successfully tested in ten instances. After achieving the logistic models, the problem to predict the protein type of a new sequence encode became simple. In addition to good results there are two important features in the proposed method: firstly, the modeling phase is made by a case-control study that do not use all database, but only samples for each target protein. This way the

modeling problem becomes fast and adaptable to huge problems. The second and most important characteristic of this method is that, after the modeling phase, the entire system reduces to a few source code with an interface to receive queries, a subroutine to recode amino acids sequences as frequency vectors and the logistic equations to predict probabilities. After the model is built there is no more database searching or any comparison among the new sequence and known proteins.

It has not escaped our notice that the peptides pairs for each protein group suggest a possible biological meaning. All bipeptides identified by the method can be associated, for example, to sequence motifs widespread over the protein. However, the whole understanding of the biological significance of these findings needs to be evaluated by another bioinformatics tool and/or by experimental assays, which will be analyzed in future. These peptides should be analyzed by other tool that seeks for substrings that must be in sequence at the same time that other patterns must be left out of the sequence.

The tool is perfectly scalable and independently of the reference database size and much less of the length of the sequences involved. Information retrieved from reference database with known protein sequences is 'stored' in predictive models that reveal frequency bipeptides patterns from each protein type.

Acknowledgements

We are grateful to Marlon C. Souza from UNI-BH, who revised the manuscript.

5. References

Altman DG (1991) Practical Statistics for Medical Research. Chapman & Hall, London

Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ (1990) Basic local alignment search tool. J. Mol. Biol. 1990, 215:403-410

Couto BRGM, Ladeira AP, Dos Santos MA (2007) Application of latent semantic indexing (LSI) to evaluate the similarity of sets of sequences without multiples alignments character-by-character. Genetics and Molecular Research 6:983–999

Couto BRGM, Leão IRF, Santoro MM, Santos MA (2009) Vector representation of protein sequences [abstract]. X-meeting 2009 - 5th International Conference of the Brazilian Association for Bioinformatics and Computational Biology [http://lgmb.fmrp.usp.br/xmeeting2009/abstractbook/pages/150.pdf]

Couto BRGM, Campos FF, Santoro MM, Santos MA: Association among similarity metrics of latent semantic indexing and BLAST statistics [abstract]. X-meeting 2009 - 5th International Conference of the Brazilian Association for Bioinformatics and Computational Biology, 2009. [http://lgmb.fmrp.usp.br/xmeeting2009/abstractbook/pages/149.pdf]

Marcolino LS, Couto BRGN, Santos MA (2010) Genome visualization in space. Advances in Soft Computing, 74(2010):225-232

Schlesselman JJ (1982) Case-Control Studies. Oxford U. Press, New York

Stuart GW, Moffett K, Leader JJ (2002) A comprehensive vertebrate phylogeny using vector representations of protein sequences from whole genomes. Molecular Biology and Evolution 19(4): 554–562

The UniProt Consortium (2010) The Universal Protein Resource (UniProt) in 2010. Nucleic Acids Res, 38(suppl 1):D142-D148

Capítulo 6 – Visualização espacial de genomas

Genome Visualization in Space

Leandro S. Marcolino[1], Bráulio R. G. M. Couto[2] and Marcos A. dos Santos[1]

Departamento de Ciência da Computação, Universidade Federal de Minas Gerais / UFMG[1]; Programa de Doutorado em Bioinformática, UFMG and Curso de Ciência da Computação, Centro Universitário de Belo Horizonte / UNIBH[2], Av. Antonio Carlos 6627, Belo Horizonte, Minas Gerais, 31270-010, Brasil. Email addresses: LSM: soriano@dcc.ufmg.br; BRGMC: braulio.couto@unibh.br; MAS: marcos@dcc.ufmg.br.

Abstract Phylogeny is an important field to understand evolution and the organization of life. However, most methods depend highly on manual study and analysis, making the construction of phylogeny error prone. Linear Algebra methods are known to be efficient to deal with the semantic relationships between a large number of elements in spaces of high dimensionality. Therefore, they can be useful to help the construction of phylogenetic trees. The ability to visualize the relationships between genomes is crucial in this process. In this paper, a linear algebra method, followed by optimization, is used to generate a visualization of a set of complete genomes. Using the proposed method we were able to visualize the relationships of 64 complete mitochondrial genomes, organized as six different groups, and of 31 complete mitochondrial genomes of mammals, organized as nine different groups. The prespecified groups could be seen clustered together in the visualization, and similar species were represented close together. Besides, there seems to be an evolutionary influence in the organization of the graph.

1. Introduction

Phylogeny is a very important field to understand evolution and the organization of life. However, many molecular phylogenies are built using sequences sampled from only a few genes. Besides, most methods depend highly on manual study and analysis, making the construction of phylogeny based on whole genomes difficult and error prone. The problem of analyzing genomes, however, is very similar to information retrieval from a large set of documents. In both problems, it is necessary to deal with an enormous amount of information, and to find semantic links between data. Fortunately, there are very good algorithms to deal with information retrieval. Singular value decomposition (SVD), for example, is used with great success (Berry *et al*. 1994). For example, linear algebra methods are used even by Google, enabling a better comprehension of a system as complex as the Internet (Eldén 2006; Stuart *et al*. 2002) presents a method to build phylogeny trees using SVD to analyze genomes. The method is demonstrated with verte-

brate mitochondrial genomes, and is later used to analyze whole bacterial genomes and whole eukaryotic genomes (Stuart and Berry 2004). Linear algebra methods are also used to study the different genotypes in the human population (Huggins *et al.* 2007).

Visualization techniques are essential to better analyze complex systems and can be very helpful to categorize species. There are a number of visualization tools to study a single genome (Lewis *et al*. 2002; Engels *et al*. 2006; Rutherford *et al*. 2000; Stothard and Wishart 2005; Gibson and Smith 2003; Ghai *et al*. 2004). However it is desirable to visualize the relationships between a set of genomes, in order to better comprehend the species. In Xie and Schlick (2000) is presented a visualization technique using SVD to analyze chemical databases. In this paper, we used that technique as a basis to develop a method for using genomes to visualize relationships among species in space (2D and 3D). This can facilitate the construction of phylogeny trees, enabling the analyzer to quickly have insights in the similarities between the different species. We are going to show the results of our approach using 832 mitochondrial proteins obtained from 64 whole mitochondrial genomes of vertebrates.

2. Material and methods

2.1 Sequence data

We used the same set of proteins as Stuart *et al*. (2002), 64 whole mitochondrial genomes from the NCBI genome database, each one with 13 genes, totaling 832 proteins in the data set. The following species were used in this paper: *Alligator mississippiensis, Artibeus jamaicensis, Aythya americana, Balaenoptera musculus, Balaenoptera physalus, Bos taurus, Canis familiaris, Carassius auratus, Cavia porcellus, Ceratotherium simum, Chelonia mydas, Chrysemys picta, Ciconia boyciana, Ciconia ciconia, Corvus frugilegus, Crossostoma lacustre, Cyprinus carpio, Danio rerio, Dasypus novemcinctus, Didelphis virginiana, Dinodon semicarinatus, Equus asinus, Equus caballus, Erinaceus europaeus, Eumeces egregius, Falco peregrinus, Felis catus, Gadus morhua, Gallus gallus, Halichoerus grypus, Hippopotamus amphibius, Homo sapiens, Latimeria chalumnae, Loxodonta africana, Macropus robustus, Mus musculus, Mustelus manazo, Myoxus glis, Oncorhynchus mykiss, Ornithorhynchus anatinus, Orycteropus afer, Oryctolagus cuniculus, Ovis aries, Paralichthys olivaceus, Pelomedusa subrufa, Phoca vitulina, Polypterus ornatipinnis, Pongo pygmaeus abelii, Protopterus dolloi, Raja radiata, Rattus norvegicus, Rhea americana, Rhinoceros unicornis, Salmo salar, Salvelinus alpinus, Salvelinus fontinalis, Scyliorhinus canicula, Smithornis sharpei, Squalus acanthias, Struthio camelus, Sus scrofa, Sciurus vulgaris, Talpa europaea, and Vidua chalybeata*.

2.2 Representation method

In order to visualize the genomes, we must represent each one as a point in space. The distance between the points should represent the differences in the genomes as a whole. Therefore, we might expect similar species to be close together in space. The genome proteins were represented as vectors of frequencies of groups of amino acids. In this paper, a sliding window of size 3 was used to measure the frequency. To represent the genome we used the vector sum of all its proteins. We are going to evaluate the appropriateness of this representation in the sequence. Therefore, we can obtain a database of genomes, S, as a rectangular matrix, X, where each line corresponds to one of the *n* genomes:

$$X = (X_1, X_2, \ldots X_n)^T = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nm} \end{pmatrix}$$

As can be seen, the representation cannot be visualized in this high-dimensional space. With 20 amino acids, and considering that unknown amino acids are represented as a separated letter of the alphabet, each genome vector has m $= 2^{13} = 9,261$ dimensions. Therefore, to generate a suitable visualization, it is necessary to reduce the dimensionality of the space, with the minimum loss of information. When a representation in reduced space, Y, is generated for the database matrix X , we can calculate an error function E as following:

$$E = \sum_i \sum_j \left(\delta_{ij} - \gamma_{ij}\right)^2$$

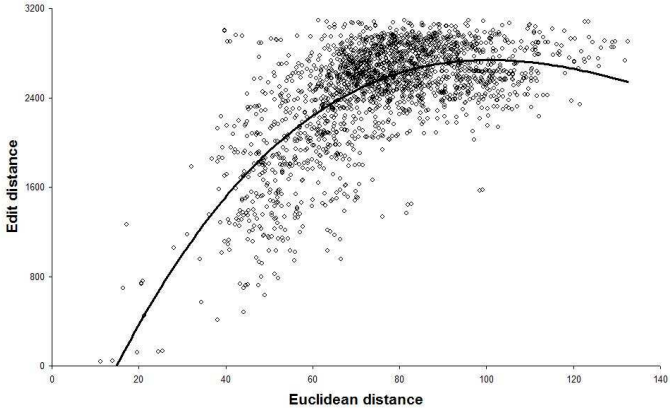
where δ_{ij} is the *euclidean distance* between genome i and j in the original space, represented in the matrix X, and γ_{ij} is the *euclidean distance* between genome i and j in the reduced space, represented in the matrix Y . The best representation of S in the reduced space will be the Y with the minimal associated error function. Therefore, we must solve an unconstrained optimization problem. Many methods can be used to solve this problem. In Xie and Schlick (2000), the truncated-newton minimization method is used. In this paper, we used a technique based on the interior-reflective Newton method. Singular value decomposition (SVD) is a popular method to reduce the dimensionality of a space, keeping the fundamental semantic association among the vectors in that space. Therefore, a good initial solution for the optimization problem can be obtained using the singular value decomposition (SVD) of X . The matrix is represented as $X = U\Sigma V^T$, where $U = [u_1\ u_2\ \ldots\ u_p]$, $\Sigma = diag(\sigma_1, \sigma_2, \ldots, \sigma_p)$, $V = [v_1\ v_2 \ldots\ v_p]$. An approximation of X in reduced space (X_k) is given by:

$$X_k = \sum_{i=1}^{p} u_i \sigma_i v_i^T ; k \leq p.$$

In this paper, we generated both two and three dimensional representations. We used a rank 2 approximation of X as the initial solution for the former, and a rank 3 approximation as the initial solution for the latter. After the optimization procedure, we have the best representation of the genomes to be visualized in a reduced space.

3. Results and discussion

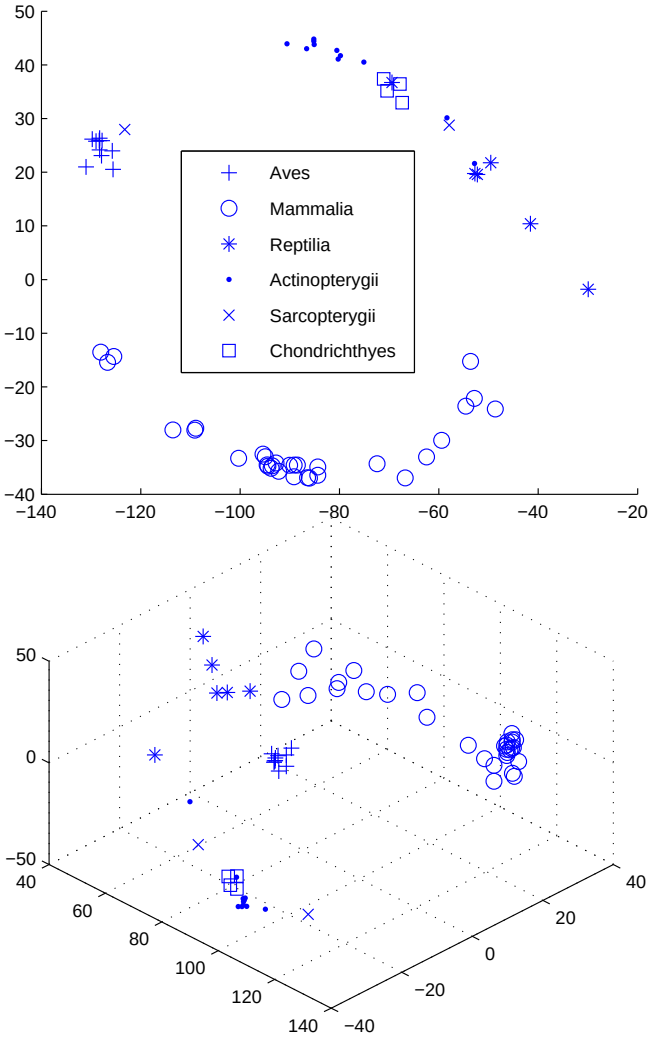
We used the proposed approach to generate two and three dimensional visualizations of 64 whole mitochondrial genomes with 832 proteins. First, we are going to evaluate if the *euclidean distance* of genomes using the chosen representation is suitable to evaluate the similarities between them. Couto *et al*. (2007) showed that the similarity of genome sequences can be measured by the *euclidean distance* in a reduced dimensional space of tripeptides descriptors. They found a correlation between the euclidean distance and global distance sequence alignment of +0.70. To perform a similar analysis we created 64 supersequences by concatenating the 13 genes from each organism. These supersequences were compared by using global edit distance between each pair of sequences and euclidean distance in the high-dimensional space. As in Couto *et al*. (2007), the correlation between the edit distance and *euclidean distance* was +0.70, but this time in a cubic model ($P < 0.01$; Figure 1). We can see, therefore, that the *euclidean distance* of genome sequences using the chosen representation can be used as a measure of similarity.



ig. 1. Scatter plot of euclidean distance and global edit distance.

We classified the species according to the class. Therefore, the following groups were used: *Aves, Mammalia, Reptilia, Actinopterygii, Sarcopterygii, Chondrichtyes*. In Figure 2 we can see the 2D and 3D results. As can be observed, the different class had a tendency to form groups in space. In the 2D graph we can see that mammals (*mammalia*) are in the bottom, birds (*aves*) are in the upper left, reptiles (*reptilia*) are generally in the middle left, and fishes (*actinopterygii,*

sarcopterygii, chondirchthyes) are in the upper right. It is notorious how the birds are close together in a single cluster. In the results in 3D the classes are even better clustered. This time, reptiles, birds and fishes are in distinctly separated groups. Only the class of the fishes are somewhat mixed.



ig. 2. Visualization of genomes in 2D and in 3D.

It is interesting to observe the relationships between the classes, as similar groups tend to be near in space. The position of the class in the graphs seems to be related to the evolutionary scale. Considering the 2D graph as an ellipse, we can see that the reptiles are between the mammals and the fishes. In 3D this can be ob-

served a second time. However, the evolutionary relationship between reptiles and birds is more clear in 3D, as there is no group between them.

Both in 2D and in 3D, mammals form a clearly distinct group from all other classes. They occupy a vast area, which might indicate more extensive diversity. We can also note that some mammals form clusters, what might be interesting to analyze. In order to better explore how the mammals are organized we separated this class in nine different groups: (i) *Prototheria*, corresponding to species in this subclass; (ii) *Marsupialia*, corresponding to species in this infraclas*s*; (iii) *Chiroptera*, corresponding to species in this *ordo*; (iv) *Cetartiodactyla*, corresponding to species in this *superordo*; (v) *Carnivora*, corresponding to species in this *ordo*; (vi) *Perissodactyla*, corresponding to species in this *ordo*; (vii) *Primates*, corresponding to individuals in this *ordo*; (viii) *Rodentia*, corresponding to individuals in this *ordo*; (ix) *Placentalia*, corresponding to all other individuals that are in this infraclass, but were not classified in any other group. In Figure 3 we can see an approximation of the region of the mammals with this new classification. Similar species appeared close together, as was expected. This shows another advantage of the proposed method: as each genome is represented as points in space, we can easily select a region to better explore, zooming in and out in the graph as appropriate for the analysis.

The proposed method, however, allows another way to visualize a selected group of genomes. We can reduce the original set and run the algorithm a second time. Therefore, in order to better visualize the mammals, we executed the algorithm with only this class in the database. The result can be seen in Figure 4. It is interesting to note that the 2D graph has a similar elliptic format as in Figure 2. Clusters that were difficult to observe in Figure 3 are very clear in this graph. Similar species are again near to each other, showing visually the proximities of the genomes. In 3D the only group that mixed with the others is the Placentalia, but this was expected, as this group is very general, holding greatly different individuals. All other groups occupy distinct positions in space. We can see, therefore, that the proposed method allows many interesting observations and analysis of a group of genomes. Prespecified groups could be seen as clusters in the resulting graphs and the positions of the species seem to be related to their evolutionary stage. We also showed how approximating a region of the graph or running the algorithm a second time with a reduced data set allows a better insight of the relationships among selected groups of genomes. The resulting graphs can be generated both in two and in three dimensions for visualization.

4. Conclusion

In this paper, we used a linear algebra method, followed by optimization, to visualize genomes in two and in three dimensional spaces. A set of complete mitochondrial genomes were used to test the algorithm. Graphs were generated to visualize the complete set and a reduced set of similar species. We noted that the

method was able to automatically cluster some of the predefined groups and biologically similar species were represented as near points in space. We also noted that the position of the genomes in space seems to be related to the evolutionary stage of the species. Our future work is directed towards using this mechanism to visualize a large set of proteins. In this way, relationships between them can be easily observed and quickly explored, facilitating new discoveries. It would also be interesting to use this technique to explore a vast number of genomes, and further explore how it can be used to gain insight in evolution and in the phylogenetic relationships between the species.

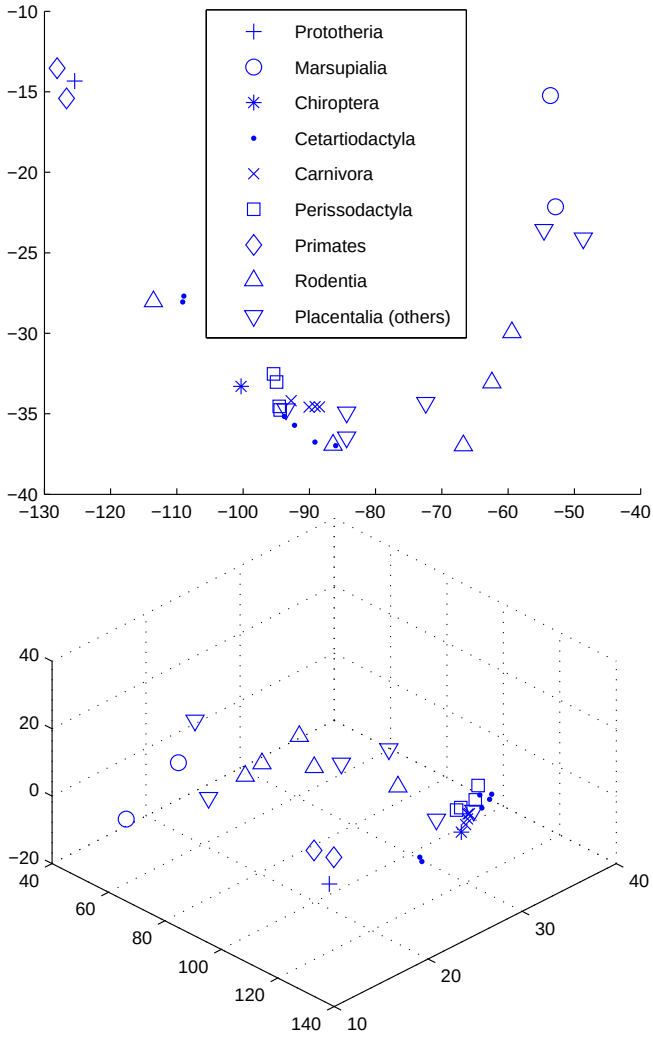


Fig. 3. Approximation of the region of the mammals in 2D and in 3D.

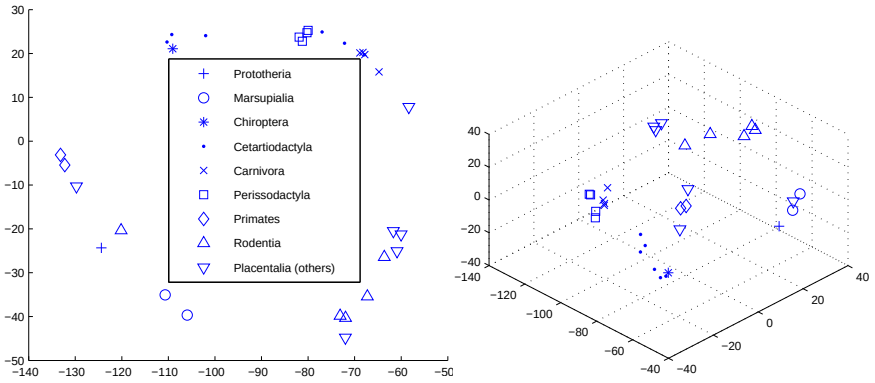


Fig. 4. Visualization of a reduced set in 2D and 3D.

5. References

Berry MW, Dumais ST, O'Brien GW (1994) Using linear algebra for intelligent information retrieval. Technical Report UT-CS-94-270, University of Tennessee, Knoxville, TN, USA

Couto BRGM, Ladeira AP, Dos Santos MA (2007) Application of latent semantic indexing (LSI) to evaluate the similarity of sets of sequences without multiples alignments character-by-character. Genetics and Molecular Research 6:983–999

Eldén L (2006) Numerical linear algebra in data mining. Acta Numerica 15:327–384

Engels R, Yu T, Burge C *et al* (2006) Combo: a whole genome comparative browser. Bioinformatics 22(14):1782–1783.

Ghai R, Hain T, Chakraborty T (2004) Genomeviz: visualizing microbial genomes. BMC Bioinformatics 5:198

Gibson R, Smith DR (2003) Genome visualization made fast and simple. Bioinformatics, 19(11):1449–1450

Huggins P, Pachter L, Sturmfels B (2007) Toward the human genotope. Bulletin of mathematical biology 69(8):2723–2735

Lewis S, Searle S, Harris N *et al* (2002) Apollo: a sequence annotation editor. Genome Biology. doi:10.1186/gb-2002-3-12-research0082

Rutherford K, Parkhill J, Crook J *et al* (2000) Artemis: sequence visualization and annotation. Bioinformatics 16(10):944–945

Stothard P, Wishart DS (2005) Circular genome visualization and exploration using cgview. Bioinformatics 21(4):537–539

Stuart GW, Berry MW (2004) An svd-based comparison of nine whole eukaryotic genomes supports a coelomate rather than ecdysozoan lineage. BMC Bioinformatics 5: 204

Stuart GW, Moffett K, Leader JJ (2002) A comprehensive vertebrate phylogeny using vector representations of protein sequences from whole genomes. Molecular Biology and Evolution 19(4): 554–562

Xie D, Schlick T (2000) Visualization of chemical databases using the singular value decomposition and truncated-Newton minimization. In: Floudas CA, Pardalos PM (eds) Optimization in Computational Chemistry and Molecular Biology: Local and Global Approaches, vol. 40, Kluwer Academic Publishers, Dororecht/Boston/London

Capítulo 7 – Visualizando dados multivariados e multidimensionais por meio da decomposição em valores singulares seguida de otimização

Visualizing high dimensional and multivariate data applying singular value decomposition followed by optimization

Braulio RGM Couto[1,2,*§], Michel AC Boaventura[3*], Leandro S Marcolino[3], Marcos A Santos[3*]

[1]*Programa de Doutorado em Bioinformática*, Universidade Federal de Minas Gerais, UFMG, Belo Horizonte, Minas Gerais, Brasil

[2]*Departamento de Ciências Exatas e Tecnologia*, Centro Universitário de Belo Horizonte, UNI-BH, Belo Horizonte, Minas Gerais, Brasil

[3]*Departamento de Ciência da Computação*, UFMG, Av. Antonio Carlos 6627, Belo Horizonte, Minas Gerais, 31270-010, Brasil

*These authors contributed equally to this work

[§]Corresponding author

Email addresses:

BRGMC: braulio.couto@unibh.br

MACB: michel@dcc.ufmg.br

LSM: soriano@dcc.ufmg.br

MAS: marcos@dcc.ufmg.br

Abstract

Background

Genomics experiments have produced massive amounts of multivariate data that are being collected into public databases. In this scenario, visualizing the non-visual high dimensional data plays an important role. This paper presents an approach, the SVD/optimization method, to map multivariate data as proteins sequence from their high dimensional representation into 2D or 3D space. The high-dimensional visualization problem in $\Re^m$ is formulated as a distance-geometry problem, i.e., to find n points in low space (2D or 3D) so that their interpoint distances match the corresponding values from $\Re^m$ as closely as possible. Firstly, protein sequences are recoded as tripeptide frequency vector using all possible overlapping tripeptides window. After to describe protein sequences as vectors in a high-dimensional space, we applied a rank reduction by using singular value decomposition (SVD) that is followed by optimization for visualizing proteins and genomes in low-dimensional space. To validate the SVD/optimization method we compared all results with PCA – Principal Components Analysis.

Results

Proposed method was successfully tested in three instances: a set geographic coordinates, other with whole mitochondrial genomes and a third database with proteins from five families. The SVD/optimization method had better visualization results than PCA.

Conclusions

The method was able to correctly visualize high dimensional and multivariate data in low space. Predefined groups of protein and biologically similar species were represented as near points in space and correctly discriminated.

Background

Genomics experiments have produced massive amounts of multivariate data that are being collected into public databases. Mining these data to understand relationships between different clustering results, to generate hypotheses about gene function or to build phylogenetic trees became critical. In this scenario, visualizing the non-visual high dimensional data plays an important role.

The importance of visualization when hidden information is extracted from a data set is far beyond any metric. For example, a correlation coefficient without its scatter plot could lead to a misleading result. Pictures, graphics and even a photograph can be very captivating and much more informative than any tables of numbers [1]. Actually, human beings are highly connected to graphs, images and visual information [2, 3]. Scientific visualization, an advancing branch of information visualization, maps physical phenomena onto 2D or 3D representations. Pictures are constructed from data that represent the underlying phenomena as a colorful image of the pattern of peak and valleys on the ocean floor [1]. However, visualization of inherently abstract information as in protein databases is much more challenging.

The objective of this paper is to present an approach to map multivariate data as proteins sequence from their high dimensional representation into 2D or 3D space. The high-dimensional visualization problem in $\Re^m$ can be formulated as a distance-geometry problem, i.e., to find n points in low space (2D or 3D) so that their interpoint distances match the corresponding values from $\Re^m$ as closely as possible [4, 5]. The idea is to preserve the inherent data structure, the geometric relationships

among all protein vectors in high-dimensional space. The visualization tool proposed here tries to preserve the original similarity relationships in the data set. Samples that are near each other in high-dimensional space will be visualized in the same neighborhood.

Firstly, we consider a bio molecular sequence as a written language that is recoded as *p*-peptide frequency vector using all possible overlapping *p*-peptides window. The methodology was developed by Stuart, Moffett and Baker, to generate whole genome phylogenies using vector representations of proteins sequence, and adapted by Couto *et al.* [6, 7]. With *p*=3 and 20 amino acids, the space protein vector has a dimension of 20^3 = 8,000 rows. After to describe protein sequences as vectors in a high-dimensional space of tri-peptides descriptors, we applied a rank reduction by using singular value decomposition (SVD) [8] that is followed by optimization for visualizing proteins and genomes in low-dimensional space (2D or 3D).

It is important to observe that, instead of use any alignment analysis, the approach applied is based on SVD, a linear algebra method. This technique is similar as used in information retrieval systems in large textual databases and Google™ web search engine. Linear algebra is known as an efficient approach to deal with semantic relationships between a large numbers of elements in spaces of high dimensionality. However, before using a linear algebra method, we need to represent proteins as vectors in a high dimensional space, and then calculate similarities among them. So, protein is represented as a vector built by frequency of all possible overlapping *p*-peptides along the sequence. Before using the chosen protein vector representation, it

is necessary to discuss two issues: initially, there is a problem when a protein is recoded as a frequency vector of *p*-peptides because the order of each *p*-peptide in the sequence is not considered. The second issue, there is a necessity to evaluate if Euclidean distance and cosine, similarity metrics used by linear algebra, are suitable to evaluate biological similarities among proteins.

First question was discussed in a previous work, when we concluded that the representation ambiguity is a theoretical possibility in principle but not in practice because two different proteins do not occupy the same point in the high dimensional space defined by the frequency *p*-peptides matrix [9]. Second question was also discussed in other report where the relationship among similarity metrics from SVD, cosine and Euclidean distance, and alignments statistics used by BLAST were assessed [10]. In that work, we chose to compare SVD with BLAST because this string-matching program is widely used for searching of nucleotide and protein databases [11]. We achieved similar results between BLAST and SVD in several protein analyses and concluded that SVD can be used to protein-protein comparisons with biological significance of the similarities identified both for cosine and Euclidean distance [12]. Before these analyses, we had already evaluated in two different situations the relationship between cosine and Euclidean distance with the edit distance, obtained from global sequence alignments using dynamic programming [7, 12]. In both studies, edit distance, Euclidean distance and cosine were strongly correlated. Euclidean distance was chosen here because our previous results recommend that this measure is better than cosine to evaluate similarities of proteins represented as vectors.

Methods

Figure 1 presents a flowchart summarizing the entire technique, called SVD/optimization method. In order to visualize the databases, we must represent each element as a point in space. The distance between the points should represent the differences between each element as a whole. Therefore, we might expect similar elements to be close together in space. In all analysis of protein sequences, they can recoded as *p*-peptide frequency vectors using all possible overlapping *p*-peptides window, which generates sparse matrices as described by Stuart [7] and adapted by Couto *et al.* [8]. With 20 amino-acids are generated 20^p high-dimensional vectors, where *p* is the word-size. Each vector row is a peptide that describes the protein sequence. In this paper, a sliding window of size three (*p*=3) was used to build the frequency matrix **M**, with dimension mxn, i.e., m=8,000 rows and **n** is the number of proteins in data set:

$$M = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn1} \end{pmatrix}$$

where $\mathbf{x_{ij}}$ is the frequency of the tripeptide **i** onto the protein sequence **j**. As can be seen, the representation cannot be visualized in this high-dimensional space. Therefore, to generate a suitable visualization, it is necessary to reduce the dimensionality of the space, with the minimum loss of information. When a representation in reduced space, **Y**, is generated for the database matrix **M**, we can calculate an error function **E** as following:

$$E = \sum_{i=1}^{n} \sum_{j=1}^{n} \left(\delta_{ij} - \gamma_{ij} \right)^2$$

where δ_{ij} is the *Euclidean distance* between genome i and j in the original space, represented in the matrix **M**, and γ_{ij} is the *Euclidean distance* between genome i and j in the reduced space, represented in the matrix **Y** . The best representation of **M** in the reduced space will be the **Y** with the minimal associated error function. Therefore, we must solve an unconstrained optimization problem. Many methods can be used to solve this problem. In Xie and Schlick [5], the truncated-newton minimization method is used. In this paper, we used a technique based on the interior-reflective Newton method, actually the conjugated gradient Newton's method. Before to minimize the objective function **E**, original matrix **M** is decomposed by SVD such that $\mathbf{M=USV^T}$ [8]. **U** is the m x m orthogonal matrix having the left singular vectors of **M** as its columns, **V** is the n x n orthogonal matrix having the right singular vectors of **M** as its columns, and **S** is the m x n diagonal matrix with the singular values $\sigma_1 \geq \sigma_2 \geq \sigma_3 \geq ... \sigma_r$ of **M** in order along (r is the rank of **M** or the number of linearly independent columns or rows of **M**).

Before the visualization be possible, after finding the **n** points in low space (2D or 3D), a rank reduction of the frequency matrix **M** is done by using the k-largest singular values of **M** which generates the new matrix $\mathbf{M_k} = U_k S_k V_k^T$. In this work we analyze the relative variances of singular values from SVD in order to identify the k-largest singular values of **M**. Relative variance (V_i) captured by the ith-singular value is equal to $(S_{i,i})^2/\sum_k (S_{k,k})^2$; k = 1, 2, … r. This previous rank reduction by SVD is done in order to eliminate noises and redundancies that can exist in database. It is important to observe that, if there is not any noisy or redundancy inside the database, all singular values of **M** will be used in the optimization phase. This happen because the number

of significant singular values in matrix **M** is related to the number of independent features of the matrix, i.e., the real matrix rank.

To validate the SVD/optimization method described in Figure 1, we compared all results with a PCA – Principal Components Analysis [2] that was used independently to visualize original matrices **M**.

Previously to protein analysis, we applied the proposed method to a set of geographic coordinates of all the Brazilian capitals, plus federal district. Actually, in order to create a two dimensional map of Brazil, we analyzed a 26x26 matrix with the distances among the twenty six Brazilian capitals plus federal district.

The second database used was the same set of proteins as Stuart *et al*. [6], 64 whole mitochondrial genomes from the NCBI genome database, each one with 13 genes, totaling 832 proteins in the data set. The following species were used in this paper: *Alligator mississippiensis, Artibeus jamaicensis, Aythya americana, Balaenoptera musculus, Balaenoptera physalus, Bos taurus, Canis familiaris, Carassius auratus, Cavia porcellus, Ceratotherium simum, Chelonia mydas, Chrysemys picta, Ciconia boyciana, Ciconia ciconia, Corvus frugilegus, Crossostoma lacustre, Cyprinus carpio, Danio rerio, Dasypus novemcinctus, Didelphis virginiana, Dinodon semicarinatus, Equus asinus, Equus caballus, Erinaceus europaeus, Eumeces egregius, Falco peregrinus, Felis catus, Gadus morhua, Gallus gallus, Halichoerus grypus, Hippopotamus amphibius, Homo sapiens, Latimeria chalumnae, Loxodonta africana, Macropus robustus, Mus musculus, Mustelus manazo, Myoxus glis, Oncorhynchus mykiss, Ornithorhynchus*

anatinus, Orycteropus afer, Oryctolagus cuniculus, Ovis aries, Paralichthys olivaceus, Pelomedusa subrufa, Phoca vitulina, Polypterus ornatipinnis, Pongo pygmaeus abelii, Protopterus dolloi, Raja radiata, Rattus norvegicus, Rhea americana, Rhinoceros unicornis, Salmo salar, Salvelinus alpinus, Salvelinus fontinalis, Scyliorhinus canicula, Smithornis sharpei, Squalus acanthias, Struthio camelus, Sus scrofa, Sciurus vulgaris, Talpa europaea, and Vidua chalybeata.

For the last test, we utilized a set of 4,888 proteins, belonging to five families: amidohydrolase, crotonase, enolase, haloacid dehalogenase and vicinal oxygen chelate (VOC).

Results

On the first experiment, PCA and the method describe in Figure 1 received a 26x26 matrix which represents the distances among the 26 Brazilian capitals plus federal district. The objective is to visualize a two dimensional draft map of Brazil by using those distances. Figures 2 and 3 present original map overlapped with the PCA results and SVD/optimization draft. We can see that although both methods have given good results, PCA is less accurate. PCA failed to find farthest points of the center, whereas our method was equally effective in all cases. By using successfully SVD/optimization method to map physical phenomena onto 2D representation, which was validated by PCA, we want to show that our method is really valid. The importance of this test is that, unlike most of the data discussed in Bioinformatics, there is an optimal solution, and it is easily verified. Thus it is easier to see the quality of solutions.

For the second test, we used the proposed approach and PCA to generate three dimensional visualizations of 64 whole mitochondrial genomes. To perform this analysis we created 64 supersequences by concatenating the 13 genes from each organism of the dataset. A sliding window of size three (p=3) was used to build the frequency matrix **M**, with dimension 8,000 x 64. This frequency matrix was analyzed by PCA and SVD/optimization method. It is interesting to observe that, as describe in Figure 1, the optimization is not made in the original matrix **M**, but in the new matrix produced after rank reduction using the k-largest singular values of **M**. In figure 4 we can see that a large part of the data is composed by noise and redundancies: after the twentieth singular value relative variance is minimal. So the optimization is applied in the matrix $\mathbf{M_k}$, built by using only the 20-largest singular value of the original matrix **M**. This previous SVD rank reduction is a interesting feature of our method, unlike PCA, it makes a noise suppression early in order to eliminate any interference in data visualization. Figures 5 and 6 show the 3D visualization of the 64 whole mitochondrial genomes. The objective function **E**, minimized during the optimization phase, i.e., the square of sum of differences from the real and calculated Euclidean distances among all genomes, was calculated using the three dimensional coordinates produced by both methods, PCA and SVD/optimization. We reached an impressive result: an improvement of 119 times on precision of our method, compared to PCA. That result was achieved only because of the previous SVD rank reduction, which is a important contribution of the proposed method.

On the last experiment, we utilized a set of 4,888 proteins, belongs to five families. Through figure 7, we can realize that the number of significant singular values it is no

bigger than five, which is compatible with the number of families in dataset. . Therefore, it is possible to reduce the original matrix's rank from 4,888 to only 5, which means a big elimination of noises and, thus, a better visualization of the data. One more time, it is possible to realize that our method was capable of better group the elements, as can be seen on Figures 8 and 9. Visually, picture produced by the SVD/optimization method was able to better discriminate all groups than the figure obtained by PCA.

Conclusions

Proposed method, that combines singular value decomposition with optimization to visualize high dimensional and multivariate data, was successfully tested in three instances. A set geographic coordinates, other with whole mitochondrial genomes and a third database with proteins from five families were used to test the method, which was compared with PCA. Graphs were generated to visualize the complete set of all databases. We noted that the method was able to correctly locate points on space and to automatically cluster some of the predefined protein groups and biologically similar species were represented as near points in space. We also noted that the position of the genomes in space seems to be related to the evolutionary stage of the species. Our future work is directed towards using this technique to explore a vast number of genomes, and further explore how it can be used to gain insight in evolution and in the phylogenetic relationships among species.

Authors' contributions

BRGMC initiated the study and wrote the manuscript preparation. MACB initiated the study, carried out the implementations, data analysis and wrote the manuscript preparation. MAS conceived the research, participated in computational analysis, result interpretation and manuscript writing. LSM gave final approval for the manuscript to be published. All authors read and approved the final text.

Acknowledgements

We are grateful to Marlon Castro de Souza from UNI-BH, who revised the manuscript.

References

1. Larkin JH and Simona HA: **Why a diagram is (sometimes) worth ten thousand words**. Cognitive Science 1987, **11**(1):65-100
2. Baeza-Yates R and Ribeiro-Neto B: **Modern information retrieval**. Addison-Wesley, Harlow, England, 1999
3. Tufte ER: **The visual display of quantitative information**. Graphics Press, Chelshire, CT, 1983
4. Sammon JW: **A nonlinear mapping for data structure analysis**. IEEE Transactions on Computer 1969, **18**(5):401-409
5. Xie D, Schlick T: **Visualization of chemical databases using the singular value decomposition and truncated-Newton minimization**. In: Floudas CA, Pardalos PM (eds) Optimization in Computational Chemistry and Molecular Biology: Local and Global Approaches, vol. 40, Kluwer Academic Publishers, Dororecht/Boston/London, 2000

6. Stuart GW, Moffett K, Baker S: **Integrated gene and species phylogenies from unaligned whole genome protein sequences**. Bioinformatics 2002, **18**(1): 100-108
7. Couto BRGM, Ladeira AP, Santos MA: **Application of latent semantic indexing to evaluate the similarity of sets of sequences without multiple alignments character-by-character.** GMR 2007, **6**(4): 983-999
8. Berry MW, Dumais ST, O'Brien GW: **Using linear algebra for intelligent information retrieval**. SIAM Review 1995, **37**:573-595
9. Couto BRGM, Leão IRF, Santoro MM, Santos MA: **Vector representation of protein sequences [abstract]**. X-meeting 2009 - 5th International Conference of the Brazilian Association for Bioinformatics and Computational Biology, 2009. [http://lgmb.fmrp.usp.br/xmeeting2009/abstractbook/pages/150.pdf]
10. Couto BRGM, Campos FF, Santoro MM, Santos MA: **Association among similarity metrics of latent semantic indexing and BLAST statistics [abstract]**. X-meeting 2009 - 5th International Conference of the Brazilian Association for Bioinformatics and Computational Biology, 2009. [http://lgmb.fmrp.usp.br/xmeeting2009/abstractbook/pages/149.pdf]
11. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ: **Basic local alignment search tool**. J. Mol. Biol. 1990, **215**:403-410
12. Marcolino LS, Couto BRGN, Santos MA: **Genome visualization in space**. Advances in Soft Computing, 2010, **74**(2010):225-232

Figures

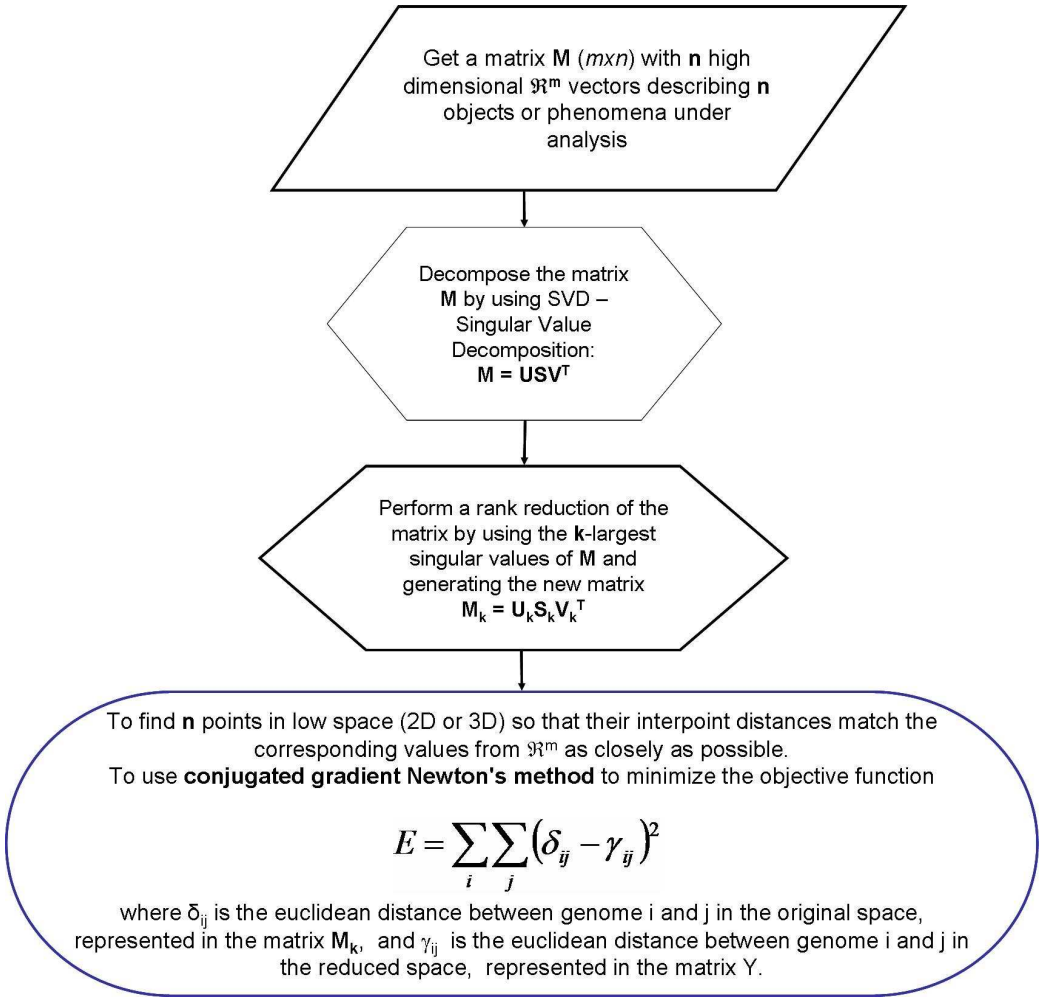


Figure 1 – Flowchart with steps for visualizing high dimensional and multivariate data applying singular value decomposition followed by optimization.

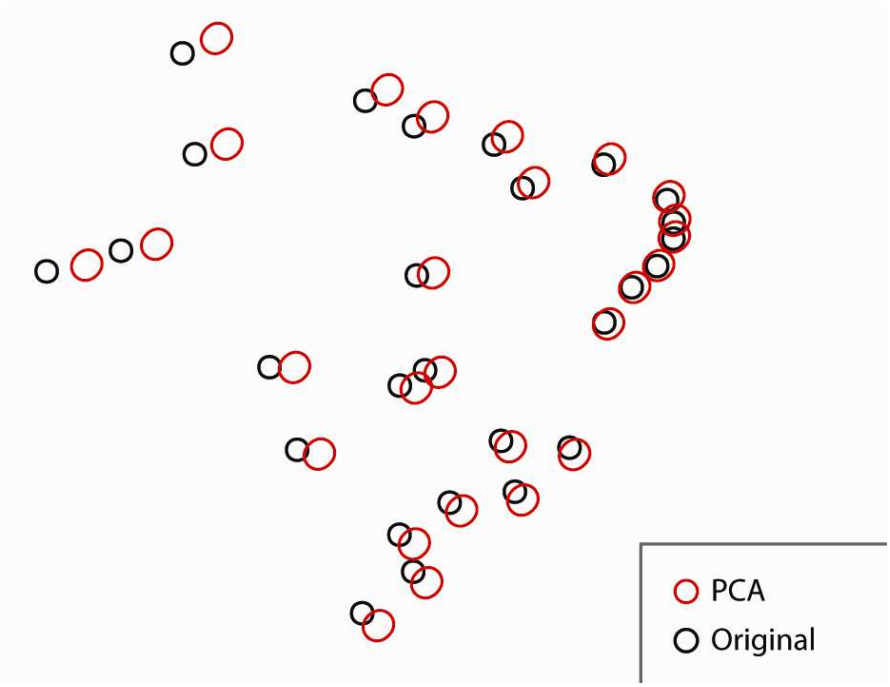


Figure 2 – Brazilian map: PCA results overlapped with the original map.

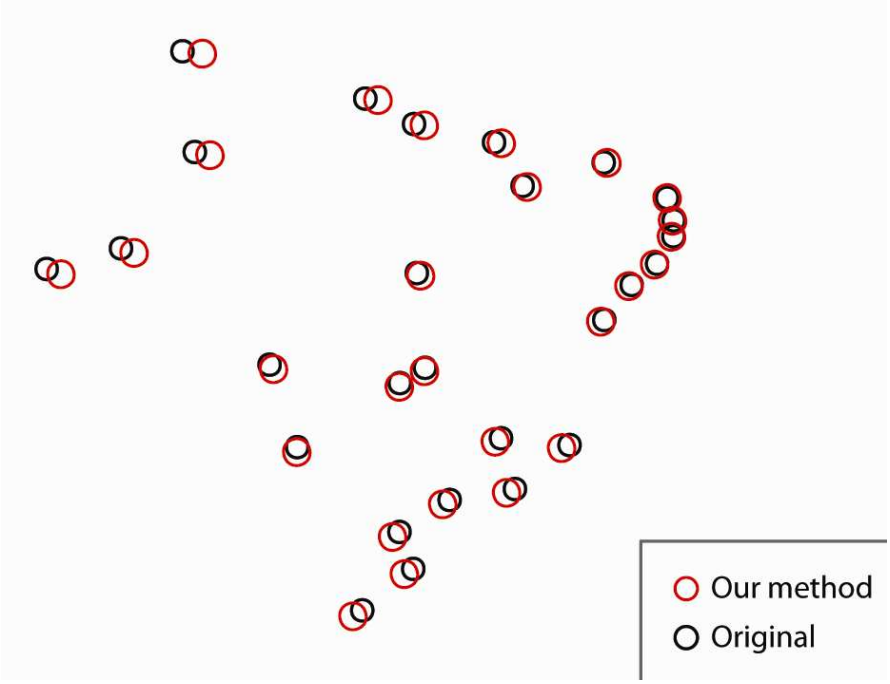


Figure 3 – Draft map of Brazil: SVD/optimization overlapped with original map.

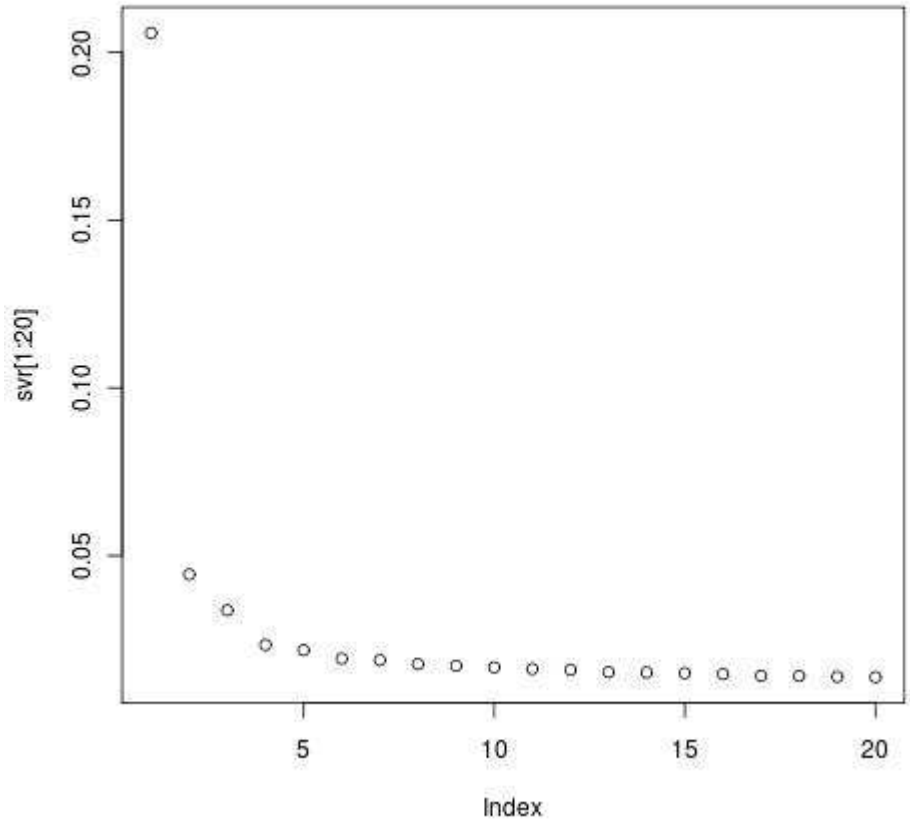


Figure 4 – Relative variances of singular values from SVD of matrix M, with 64 whole mitochondrial genomes.

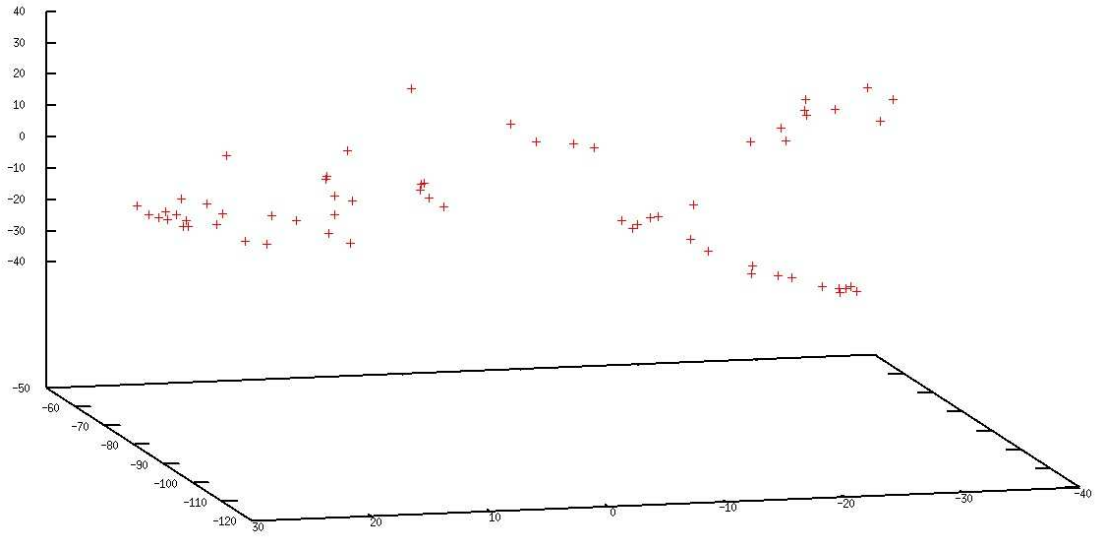


Figure 5 – 3D visualization of whole mitochondrial genomes: results obtained by SVD/optimization method.

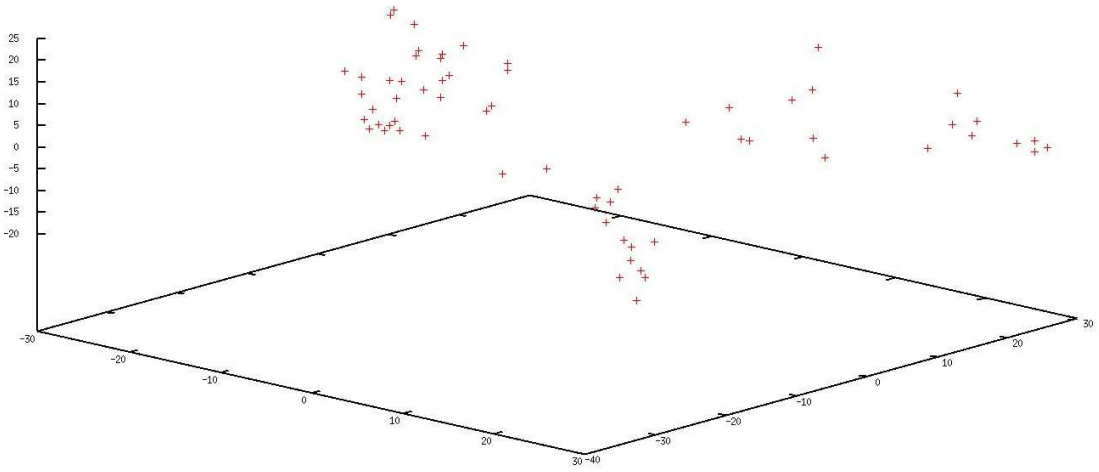


Figure 6 – 3D visualization of whole mitochondrial genomes: results obtained by PCA.

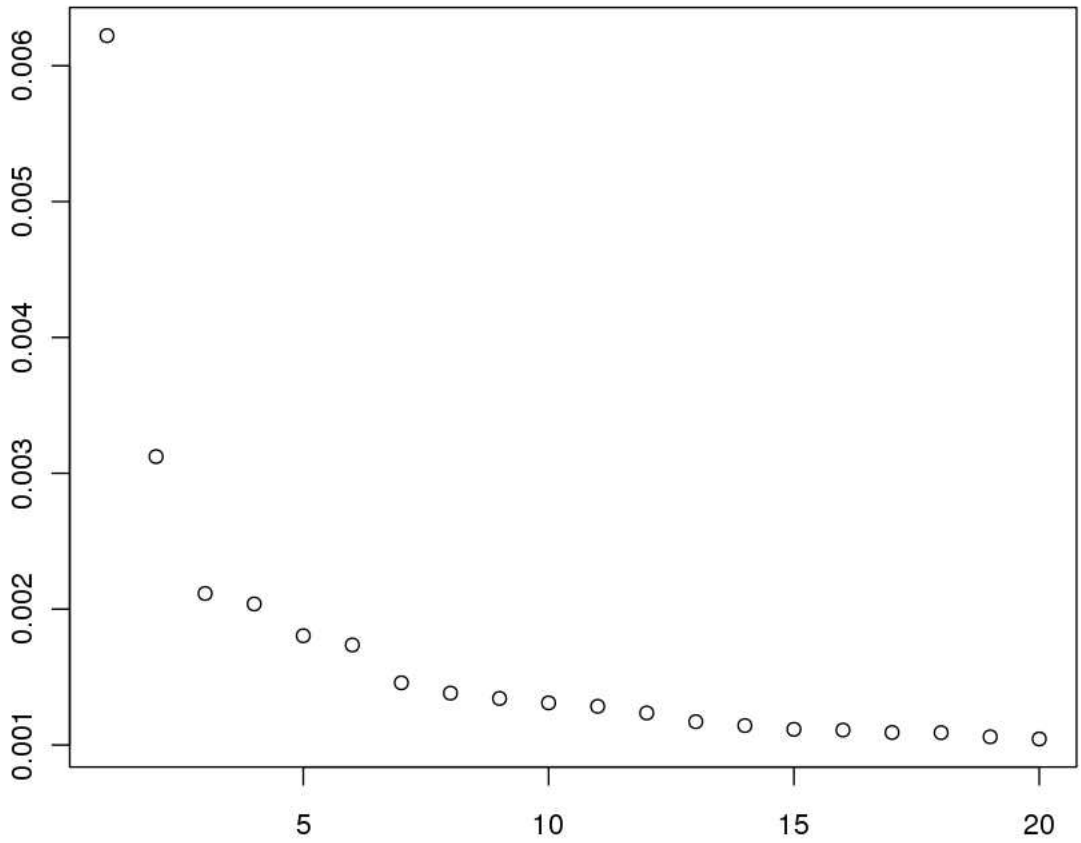


Figure 7 – Relative variances of singular values from SVD of matrix M, with 4,888 protein sequences belonging to five families.

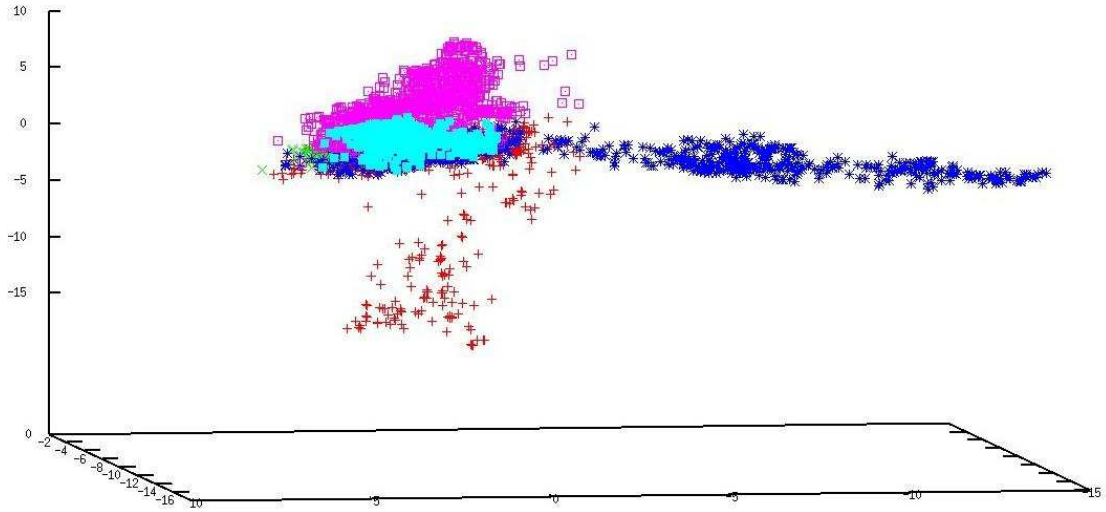


Figure 8 – 3D visualization of 4,888 proteins with five families: results obtained by SVD/optimization method.

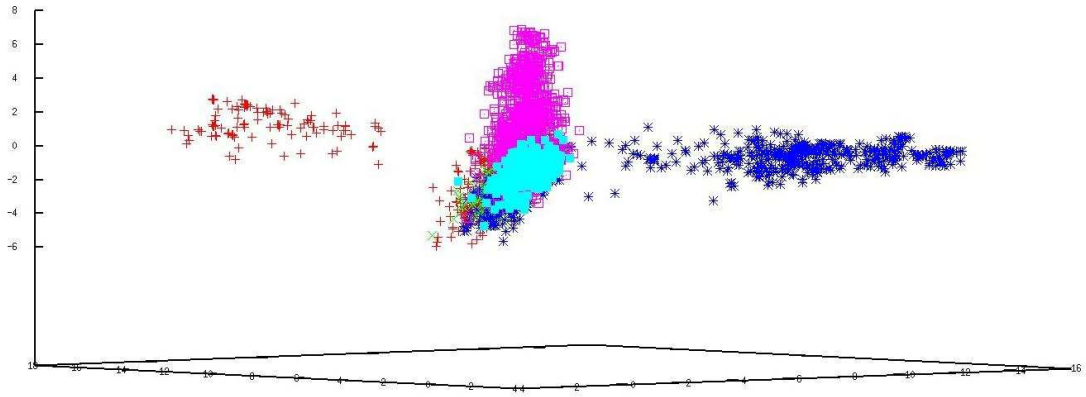


Figure 9 – 3D visualization of 4,888 proteins with five families: results obtained by PCA.

Discussão

Observando os sete artigos colocados nos capítulos 1 a 7 desta tese, surgem algumas perguntas cujas respostas podem ser extraídas do próprio trabalho:

a) Sequências primárias de proteínas podem ser representadas como vetores de frequência de *p*-peptídeos?

   Sim, sequências primárias de proteínas podem ser representadas como vetores de frequência de *p*-peptídeos.

   As figuras 1, 2 e 5 do capítulo 2 referem-se a exemplos genéricos e numéricos da construção dos vetores de frequência de *p*-peptídeos. Já a figura 4 do capítulo 2 apresenta o código fonte em MATLAB (The Mathworks, 1996) para a construção dos vetores e da respectiva matriz de freqüência de *p*-peptídeos. É interessante observar que, após a redução de posto da matriz de freqüência de aminoácidos (figura 6 do capítulo 2), há uma conseqüente redução da variabilidade na freqüência de cada aminoácido que caracterizam uma família de genes. Esta redução na variabilidade nos atributos que caracterizam famílias protéicas após a redução de posto é uma conseqüência da SVD que não foi totalmente explorada na tese, podendo ser fonte de trabalho futuro.

   De qualquer forma, os resultados mostram que é possível codificar proteínas como vetores de frequência de *p*-peptídeos em $\Re^m$, onde $m = 20^p$ e *p* é o número de aminoácidos no peptídeo (p=1, 2, 3, 4). Por meio da redução de posto da SVD pode-se ainda visualizar proteínas em espaço bi e tridimensional (figura 7 do capítulo 2, figuras 1 e 6 do capítulo 1). Além disto, torna-se viável a análise de similaridades por meio de métricas da Álgebra Linear, como distância Euclidiana e cosseno. Em relação a uma possível "ambigüidade" na codificação vetorial proposta, que não leva em consideração a ordem com que o *p*-peptídeo aparece na sequência, este aspecto foi comentado nos capítulos 1, 4, 5 e 7. Apesar da possibilidade de duas proteínas diferentes poderem ser codificadas pelo mesmo vetor de frequências, na prática esta "ambiguidade" não foi constatada. Na verdade, duas sequências diferentes não

ocupam o mesmo ponto do espaço multidimensional definido pela matriz de freqüência de *p*-peptídeos (COUTO *et al.*, 2009).

b) Qual é o significado biológico da decomposição em valores singulares (SVD)?

Resultados discutidos no capítulo 3 mostram que a visualização dos valores singulares obtidos pela SVD ajuda a identificar os principais componentes, os processos escondidos num banco de dados. A idéia não é apresentar um valor específico, mas uma faixa de possíveis valores para, por exemplo, o número de grupos de proteínas numa base de dados. Esta mesma idéia já foi aplicada a dados de microarray, no qual valores singulares mais significantes podem estar associados com grupos de genes ou com a estrutura de ciclos celulares (Wall *et al*., 2003).

c) Do ponto de vista biológico, tem significado a análise de similaridade de proteínas por meio da comparação de vetores de frequência de *p*-peptídeos?
   - Medidas de similaridade da SVD, distância Euclidiana e cosseno, estão associadas com a distância global de edição?
   - Medidas de similaridade da SVD, distância Euclidiana e cosseno, estão associadas com medidas de similaridade usadas pelo BLAST, *E value* e *bit score*?
   - Se houver associação entre as métricas de similaridade da SVD, do alinhamento global de sequências e do BLAST, é possível encontrar modelos de previsão entre as métricas envolvidas? Ou seja, é possível prever uma medida de similaridade por meio de outra?
   - Quando da classificação de uma sequência protéica desconhecida, qual é o grau de concordância entre cosseno e distância Euclidiana com o resultado gerado pelo BLAST?

A análise de similaridade de sequências protéicas, por meio da comparação de vetores de frequência de *p*-peptídeos, tem significado biológico similar àquele da análise de similaridade obtida com alinhamentos de sequências que envolvem comparações caractere-a-caractere, ponderadas por matrizes de substituição.

Técnicas de correlação e regressão linear (NETER et al., 1996) foram usadas na análise da associação entre a distância de edição, obtida por algoritmos de programação dinâmica em alinhamentos múltiplos, as métricas de similaridade da SVD, cosseno e distância Euclidiana, e estatísticas geradas pelo programa BLAST.

As figuras 9 e 10 do capítulo 2 e a figura 1 do capítulo 6 mostram que há uma forte associação entre as medidas de similaridade da SVD, distância Euclidiana e cosseno, com a distância global de edição. A equação 6 do capítulo 2 sugere uma relação matemática que pode ser usada para se prever a distância global de edição, com base na distância Euclidiana entre dois vetores de proteínas.

As figuras 2, 3 e 5 do capítulo 1 mostram que há uma forte associação entre as métricas de similaridade da SVD com aquelas usadas pelo BLAST. A obtenção de um modelo de predição, por exemplo, do *bit score* em função da distância Euclidiana entre dois vetores protéicos, é um trabalho futuro que pode ser feito a partir dos resultados observados no capítulo 1 da tese.

Em relação à capacidade discriminante de sistemas baseados em vetores de frequência de *p*-peptídeos e SVD, as figuras 14 e 15 do capítulo 2 mostram que bons resultados podem obtidos quando da classificação de sequências teste. Nestes exemplos, o padrão ouro usado foi a própria definição da categoria da proteína, conforme a sua descrição no banco de dados de origem. Na comparação da classificação da SVD com aquela obtida pelo BLAST (figuras 7 e 8 e tabelas 1 e 2 do capítulo 1) houve boa concordância de resultados entre os dois métodos.

Em suma, os resultados apresentados nos testes dos capítulos 1 e 2 mostram que as métricas da SVD não somente correlacionam-se com as estatísticas do BLAST como apresentam bom grau de concordância quando da classificação das mesmas sequências.

d) Qual o impacto de se usar bases de dados com sequências redundantes na pesquisa por homologia?

As chances de descobertas biológicas são maximizadas se a pesquisa por homologia for feita em bases de dados mais atualizadas, que crescem rapidamente não só em tamanho, mas também em redundância. Nas bases públicas, há uma super representatividade de sequências idênticas, de mesmo tamanho e com resíduos na mesma posição, e outras muito próximas, com mais de 90% de identidade. Esta distribuição desigual no espaço de sequências é causada por um vício na própria pesquisa genômica e na existência de agrupamentos (*clusteres*) de sequências muito próximas, de famílias de genes de diferentes espécies e organismos que existem naturalmente devido à duplicação gênica (PARK *et al*.,

2000; HOLM e SANDER, 1998). A busca por similaridades feita por meio de algoritmos baseados em alinhamentos de sequências é fortemente afetada pelo grau de redundância da base de dados. Por exemplo, o vício devido à super abundância de certas famílias protéicas pode afetar os métodos de escore dos programas de pesquisa (PARK *et al*., 2000). No BLAST, a significância estatística de um alinhamento depende do tamanho da base de dados. Pela equação de Karlin e Altschul (ALTSCHUL *et al.*, 1990), o número de alinhamentos esperados de ocorrer devido ao acaso (**E**) é uma função linear ao tamanho do espaço de pesquisa (**m*n**) e é exponencialmente dependente do escore normalizado de similaridade (**λS**): $E = kmne^{-\lambda S}$.

Assim, se o tamanho do espaço de pesquisa dobra, o número de alinhamentos aleatórios com um particular escore também dobra. Cada sequência redundante aumenta artificialmente a base de dados e, consequentemente reduz a significância estatística de um alinhamento (Korf *et al.*, 2003).

Quando se usa SVD, a redução de posto descrita nos capítulos 1, 2 e 4, faz com as redundâncias nas bases de dados não tenham qualquer efeito nas análises de similaridade. A redução de posto da SVD elimina automaticamente qualquer influência do grau de redundância de uma base na comparação entre vetores protéicos.

e) É possível identificar aminoácidos importantes para a classificação de uma determinada categoria de proteína por meio dos vetores de frequência de aminoácidos?

Estudos baseados nas amostragens tipo caso-controle descritas nos capítulos 4 e 5 e análises por meio de regressão logística dos vetores de freqüência de aminoácidos, descritos na figura 1 do capítulo 4, permitem identificar aminoácidos importantes para a classificação de uma determinada categoria de proteína.

f) É possível identificar bipeptídeos importantes para a classificação de uma determinada categoria de proteína por meio dos vetores de frequência de bipeptídeos?

Os esquemas descritos na figura 1 do capítulo 4 e na figura 1 do capítulo 5 permitem identificar bipeptídeos importantes para a classificação de uma determinada categoria de proteína. Investigar as razões biológicas que poderiam justificar a importância tanto dos aminoácidos quanto dos bipeptídeos identificados nos modelos de regressão logística estimados nos capítulos 4 e 5 são temas de trabalhos futuros.

g) Como mapear a relação multidimensional de genomas e outros dados multivariados para o espaço bi e tridimensional (2D e 3D)?

O mapeamento multidimensional em $\Re^m$ de genomas e outros dados multivariados para o espaço bi e tridimensional (2D e 3D), tratado nos capítulos 6 e 7, foi formulado como um problema geométrico: encontrar n pontos no espaço 2D ou 3D de tal forma que suas distâncias inter-pontos no espaço original se mantenham mais próximas quanto possível no espaço reduzido.

Nos resultados apresentados no capítulo 6, a SVD, com redução de posto 2 ou 3, foi usada somente como valor inicial para o método de minimização usado para encontrar as coordenadas de cada vetor no espaço reduzido.

Já no capítulo 7 a SVD teve um papel fundamental, pois o espaço multidimensional original não foi considerado e sim o espaço multidimensional obtido com a redução de posto da matriz em $\Re^m$ (figura 1 do capítulo 7). As novas coordenadas em $\Re^2$ e $\Re^3$ foram obtidas de tal forma que as distâncias inter-pontos no espaço reduzido da SVD fossem mais próximas quanto possível no espaço reduzido.

Conclusões

Pode-se afirmar que a principal conclusão desta tese refere-se à validade biológica do uso da decomposição em valores singulares (SVD) para análise de similaridade e extração de padrões em sequências protéicas. Antes da realização deste trabalho, persistiam muitas dúvidas em relação à significância biológica de se considerar uma proteína como um vetor no espaço multidimensional e, principalmente, quanto à validade da análise de similaridade por meio de técnicas de Álgebra Linear. Mesmo sem se trabalhar com matrizes de substituição nem com algoritmos de alinhamentos de sequências, foram obtidos resultados biologicamente válidos.

Em relação a trabalhos futuros, como era esperado, várias frentes de trabalho se abriram, entre elas:

a) Investigação sobre os efeitos da diminuição na variabilidade da freqüência de *p*-peptídeos quando se faz redução de posto com a SVD.

b) Estimação de modelos de previsão das métricas de alinhamento de sequências em função da distância Euclidiana e/ou do cosseno entre dois vetores protéicos.

c) Ánálise da viabilidade de se fazer uma nova decomposição em valores singulares de tal forma que a reconstrução da matriz usando posto reduzido produza somente valores positivos; os termos desta nova matriz poderão ser usados num modelo de regressão logística em que as variáveis explicativas sejam as freqüências de *p*-peptídeos, recalculadas com o posto reduzido. Na verdade, foi feita uma tentativa de se fazer uma modelagem após a redução de posto, mas alguns valores das novas freqüências de bipeptídeos, recalculadas após a redução de posto da SVD, ficaram negativos e perderam sentido físico.

d) Construção de um sistema *web based* de recuperação de informação de sequências protéicas desconhecidas por meio de modelos de regressão logística. Na verdade, a idéia é expandir os resultados do capítulo 5 para o maior número possível de grupos de proteínas, usando outras bases de dados de referência.

e) Investigar as razões biológicas que poderiam justificar a importância dos aminoácidos e bipeptídeos identificados nos modelos de regressão logística estimados nos capítulos 4 e 5.

f) Uso dos bipeptídeos, definidos nos capítulos 4 e 5 como importantes para cada um dos grupos de proteínas analisadas, para se encontrar motivos protéicos. Na verdade, seria uma análise de motivos especiais (*motifs*), definidos não só pela presença de determinados bipeptídeos, mas também pela simultânea ausência de outros.

Para finalizar, a mais relevante contribuição desta tese foi demonstrar a viabilidade de se codificar vetorialmente uma proteína. Descrever uma proteína na forma de um vetor permite que não só a SVD possa ser usada na sua análise, mas todas as ferramentas da Matemática, da Física, da Estatística, da Geometria e da própria Álgebra Linear, utilizadas na manipulação de vetores e matrizes, também poderão ser usadas na busca por similaridades e na extração de padrões em sequências protéicas.

Referências bibliográficas

ALTSCHUL, S.F. *et al*. Basic local alignment search tool. *J. Mol. Biol*., v. 215, p. 403-410, 1990.

COUTO, B.R.G.M. *et al*. Vector representation of protein sequences. In: X-MEETING 2009 - INTERNATIONAL CONFERENCE OF THE BRAZILIAN ASSOCIATION FOR BIOINFORMATICS AND COMPUTATIONAL BIOLOGY, 5., 2009, Angra dos Reis. *Abstracts* … Angra dos Reis: [s.n.], 2009. p. 150.

COUTO, B. R. G. M.; LADEIRA, A. P.; SANTOS, M. A. Application of latent semantic indexing (LSI) to evaluate the similarity of sets of sequences without multiples alignments character-by-character. Genetics and Molecular Research. 2007; 6: 983-999.

COUTO, B.R.G.M.; SANTORO, M.M.; SANTOS, M.A. Singular value decomposition (SVD) and BLAST: quite different methods achieving similar results. Proceedings of BIOINFORMATICS 2011 - International Conference on Bioinformatics Models, Methods and Algorithms. Rome, Italy, 26-29, jan/2011. Paper #63.

COUTO, B.R.G.M.; SANTORO, M.M.; SANTOS, M.A. Unrevealing biological process with linear algebra: extracting patterns from noisy data. Proceedings of BIOINFORMATICS 2011 - International Conference on Bioinformatics Models, Methods and Algorithms. Rome, Italy, 26-29, jan/2011. Paper #65.

COUTO, B.R.G.M.; SANTORO, M.M.; ALI, A.; SANTOS, M.A. Feature selection for protein sequence classification by using logistic regression models and singular value decomposition. Submitted to BMC Bioinformatics special issue from X-meeting 2010 - 6th International Conference of the Brazilian Association for Bioinformatics and Computational Biology. Ouro Preto, Brazil, 15-18, oct/2010.

COUTO, B.R.G.M.; SANTORO, M.M.; SANTOS, M.A. Protein sequence retrieval system based on logistic regression models. Submitted to Special volume of Advances in Soft Computing for the 15th Online World Conference on Soft Computing in Industrial Applications (WSC15). WSC15 will be held on the Internet from 15th to 27th November 2010.

COUTO, B.R.G.M.; BOAVENTURA, M.A.C; MARCOLINO, L. S.; SANTOS, M.A. Visualizing high dimensional and multivariate data applying singular value decomposition followed by optimization. Submitted to BMC Bioinformatics special issue from X-meeting 2010 - 6th International Conference of the Brazilian Association for Bioinformatics and Computational Biology. Ouro Preto, Brazil, 15-18, oct/2010.

DAYHOFF, M.O.; SCHWARTZ, R.M.; ORCUTT, B.C. A model of evolutionary change in proteins. In: DAYHOFF, M.O. (Ed.) Atlas of protein sequence and structure. *Natl. Biomed. Res. Found*., v. 5, p. 345-352, 1978.

DEERWESTER, S. *et al*. Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science*, v. 41, n. 6, p. 1-13, 1990.

DONG, Q.; WANG, X.; LIN, L. Application of latent semantic analysis to protein remote homology detection. *Bioinformatics*, v. 22, n. 3, p 285-290, 2006.

GIBAS, C.; JAMBECK, P. *Desenvolvendo Bioinformática*. Rio de Janeiro: Campus, 2001. 464 p.

HENIKOFF, S.; HENIKOFF, J.G. Amino acid substitution matrices from protein blocks. *Proc. Natl Acad. Sci*., v. 89, p. 10915-10919, 1992.

HOCHREITER, S.; HEUSEL, M.; OBERMAYER, K. Fast model-based protein homology detection without alignment. *Bioinformatics*, v. 23, n. 14, p. 1728-1736, 2007.

HOLM, L.; SANDER, C. Removing near-neighbour redundancy from large protein sequence collections. *Bioinformatics,* v. 14, n. 5, p. 423-429, 1998.

HUNTER, L. *Artificial Intelligence and Molecular Biology*. Cambridge, USA: American Association for Artificial Intelligence, MIT Press, 1993. 160 p.

KANTOROVITZ, M.R.; ROBINSON, G.E.; SINHA, S. A statistical method for alignment-free comparison of regulatory sequences. *Bioinformatics,* v. 23, p. i249-i255, 2007.

KORF, I.; YANDELL, M.; BEDELL, J. *An essential guide to the Basic Local Alignment Search Tool – BLAST.* Sebastopol: O'Reilly & Associates Inc., 2003. 368 p.

KOSKI, L.B.; GOLDING, T.B. The closest BLAST hit is often not the nearest neighbor. *J Mol Evolm* v. 52, p. 540-542, 2001.

KRAWETZ, A.S.; WOMBLE, D.D. *Introduction to Bioinformatics: a theoretical and practical approach.* Totowa: Humana Press, 2003. 746 p.

LIU, B. *et al*. A discriminative method for protein remote homology detection and fold recognition combining top-n-grams and latent semantic analysis. *BMC Bioinformatics, v.* 9, p. 510, 2008.

MARCOLINO, L. S.; COUTO, B. R. G. M.; SANTOS, M. A. Genome Visualization in Space. Advances in Soft Computing. 2010; 74:225-232.

NEEDLEMAN, S.B.; WUNSCH, C.D. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol. Biol.*, v. 48, p. 443-453, 1970.

NETER, J.; KUTNER, M.H.; NACHSTHEIM, C. *Applied linear statistical models*. 4rd ed. Boston: Mcgraw-Hill, 1996. 1408 p.

PEARSON, W.; LIPMAN, D. Improved tools for biological sequence comparison. *Proc Natl Acad Sci USA*, v. 85, p. 2444-2448, 1988.

PARK, J. *et al*. RSDB: representative protein sequence databases have high information content. *Bioinformatics,* v. 16, n. 5, p. 458-464, 2000.

PERTSEMLIDIS, A. ; FONDON III, J.W. Having a BLAST with bioinformatics (and avoiding BLASTphemy). *Genome Biology*, v. 2, n. 10, p. :reviews2002.1–2002.10, 2001.

RODRIGUES, T.S. *Codificação de Sequências de Aminoácidos e sua Aplicação na Classificação de Proteínas com Redes Neurais Artificiais*. Orientador: Antônio P. Braga. 2007. 127 f. Tese (Doutorado em Bioinformática) - Universidade Federal de Minas Gerais, Belo Horizonte, 2007.

SANDBERG, M. *Deciphering sequence data, a multivariate approach*. Orienting professor: Michael Sjöström. 1997. 78 f. Thesis (PhD) - Umeå University, Umeå, Sweden, 1997.

SMITH, T.F.; WATERMAN, M.S. Identification of Common Molecular Subsequences. *J Mol Biol*, v. 147, n. 1, p. 195-197, 1981.

STRYER, L. *Bioquímica*. 4. ed. Rio de Janeiro: Guanabara Koogan, 1996. 1114 p.

STUART, G.W.; BERRY, M.W. A comprehensive whole genome bacterial phylogeny using correlated peptide motifs defined in a high dimensional vector space. *Journal of Bioinformatics and Computational Biology,* v. 1, n. 3, p. 475-493, 2003.

STUART, G.W.; BERRY, M.W. An SVD-based comparison of nine whole eukaryotic genomes supports a coelomate rather ecdysozoan lineage. *BMC Bioinformatics,* v. 5, p. 204, 2004.

STUART, G.W.; MOFFETT, K.; BAKER, S. Integrated gene and species phylogenies from unaligned whole genome protein sequences. *Bioinformatics,* v. 18, n. 1, p. 100-108, 2002.

STUART, G.W.; MOFFETT, K. LEADER, J.J. A Comprehensive Vertebrate Phylogeny Using Vector Representations of Protein Sequences from Whole Genomes. *Mol. Biol. Evol.,* v. 19, n. 4, p. 554–562, 2002.

TEICHERT, F.; BASTOLLA, U.; PORTO, M. SABERTOOTH: protein structural alignment based on a vectorial structure representation. *BMC Bioinformatics*, v. 8, p. 425, 2007.

THORNE, J.L. Models of Protein Sequence Evolution and their Applications. *Curr Opin Genet Dev*, v. 10, p. 602-605, 2000.

VINGA, S.; ALMEIDA, J. Alignment-free sequence comparison - a review. *Bioinformatics , v.* 19, n. 4, p. 513-523, 2003.

WU, X. *et al*. Nucleotide composition string selection in HIV-1 subtyping using whole genomes. *Bioinformatics,* v. 23, n. 14, p. 1744-1752, 2007.

WALL, M.E. *et al*. Singular value decomposition and principal component analysis. In: BERRAR, D.P. *et al*. (eds.), A practical approach to microarray data analysis. Norwell: Kluwer, 2003. 384 p.

YUAN, Y. *et al.* A Protein Classification Method Based on Latent Semantic Analysis. *Conf Proc IEEE Eng Med Biol Soc.*, v. 7, p. 7738-7741, 2005.