GUSTAVO FIGUEIREDO ARAÚJO

CODIFICAÇÃO E CLUSTERING DE PROTEÍNAS

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

LAVRAS
MINAS GERAIS – BRASIL
2007

GUSTAVO FIGUEIREDO ARAÚJO

CODIFICAÇÃO E CLUSTERING DE PROTEÍNAS

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

Área de Concentração:

Bioinformática

Orientador(a) / Co-orientador(a):

Thiago de Souza Rodrigues

LAVRAS
MINAS GERAIS – BRASIL
2007

Ficha Catalográfica

Araújo, Gustavo Figueiredo

Codificação e Clustering de Proteínas / Gustavo Figueiredo Araújo. Lavras – Minas Gerais, 2007. 46p : il.

Monografia de Graduação – Universidade Federal de Lavras. Departamento de Ciência da Computação.

1. Introdução. 2. Referencial Teórico. 3. Metodologia. 4. Resultados e Discussão. 5. Conclusões e Propostas Futuras. I. ARAÚJO, G. F. II. Universidade Federal de Lavras. III. Título.

GUSTAVO FIGUEIREDO ARAÚJO

CODIFICAÇÃO E CLUSTERING DE PROTEÍNAS

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

Aprovada em 17 de Janeiro de 2008

_____
Prof. Msc. Cristiano Leite de Castro

_____
Prof. Dra. Marluce Rodrigues Pereira

_____
Prof. Dr. Thiago de Souza Rodrigues
(Orientador)

LAVRAS
MINAS GERAIS – BRASIL

CODIFICAÇÃO E CLUSTERING DE PROTEÍNAS

RESUMO

Este trabalho visa analisar o método de codificação de seqüências *Sequence Coding by Sliding Window (SCSW)* aplicado na codificação de seqüências de aminoácidos. Para realizar a avaliação do método, foi utilizado o algoritmo de clusterização K-Means, a fim de verificar a eficácia do método de codificação na comparação de similaridades entre seqüências. Apesar de levantados alguns problemas no método SCSW, os resultados mostraram que esse método pode ser útil na determinação de similaridade entre seqüências, entretanto estes problemas devem ser minimizados a fim de se obter melhores resultados.

Palavras-chave: codificação de proteínas, comparação de sequências, clustering de proteínas.

CODING AND CLUSTERING OF PROTEINS

ABSTRACT

This work aims to analyze the method of coding sequences called here Coding Sequence by Sliding Window (SCSW) applied in amino acids sequences. The K-Means clustering algorithm was used to verify the SCSW effectiveness comparing the similarities between the original sequences. The results showed that this method can be useful in determining the similarity between sequences, however some problems should be solved or minimized in order to obtain better results.

Keywords: *coding of protein, comparing sequences, clustering of proteins.*

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

1 INTRODUÇÃO

Os métodos computacionais são atualmente imprescindíveis na biologia molecular. Entretanto, para se utilizar métodos de inteligência computacional em uma determinada aplicação, os dados de entrada devem possuir sempre a mesma dimensão e valores, necessariamente, numéricos. Sendo assim algum tipo de codificação deve ser aplicado às seqüências de nucleotídeos e aminoácidos. Neste trabalho será analisado a aplicação do método de codificação de seqüências *Sequence Coding by Sliding Window (SCSW)* na codificação de seqüências de aminoácidos. Para realizar a avaliação deste método, será utilizado o algoritmo de clusterização K-Means, a fim de verificar a eficácia do método SCSW na determinação de similaridade entre seqüências. A metodologia e a discussão dos resultados serão apresentados nas sessões 3 e 4 respectivamente.

Neste capítulo é apresentado alguns conceitos sobre a importância dos métodos computacionais na biologia molecular, principalmente quando estes métodos são utilizados adicionalmente aos bancos de dados de seqüências de nucleotídeos ou aminoácidos, além de apresentar os problemas encontrados na classificação de proteínas. Os objetivos, geral e específicos, e a organização geral do texto são mostrados no final do capítulo.

1.1 Considerações Iniciais

O avanço tecnológico ocorrido principalmente após os anos 80, deu início ao grande crescimento de dados referente à seqüências (nucleotídeos ou aminoácidos). Período este em que pesquisas de seqüênciamento de DNA tornavam-se largamente difundidas como, por exemplo, o projeto do genoma humano, publicado pela *International Human Genome Sequence Consortium* em 2001.

Estas seqüências estão armazenadas em diversos bancos de dados públicos tais como CDD (*Conserved Domain Database*)[1], COG (*Clusters of Orthologous Groups*)[2], DDBJ (*DNA Data Bank of Japan*)[3], EMBL (*European Molecular Biology Laboratory*)[4], GenBank[5],

1 http://www.ncbi.nlm.nih.gov/Structure/cdd/cdd.shtml
2 http://www.ncbi.nlm.nih.gov/COG/
3 http://www.ddbj.nig.ac.jp/
4 http://www.ebi.ac.uk/embl/
5 http://www.ncbi.nlm.nih.gov/Genbank/

Pfam (*Protein Family*)[6], PIR (*Protein Information Research*)[7], Smart (*Simple Modular Architecture Research Tool*)[8], Swiss-Prot (*Protein knowledgebase*)[9], dentre outros.

Adicionalmente aos bancos de dados de seqüências, os métodos computacionais são decisivos e imprescindíveis nos estudos da biologia molecular, como: na busca de similaridade, predição de estrutura, predição de função dentre outros (FUCHS, 2002).

Assim que o seqüênciamento de genomas é finalizado, os pesquisadores voltam-se aos seus produtos: as proteínas. Decorrente do conhecimento de um genoma, existe o interesse em determinar a função de todas as proteínas do organismo, porque através da função de uma proteína entende-se o papel que esta desenvolve, como é sua participação em uma rota metabólica, a transmissão de sinais no interior da célula ou a regulação da função de outras proteínas.

Duas estratégias podem ser utilizadas para atribuição de função a uma dada proteína: pesquisas em laboratório ou utilização de métodos computacionais. A primeira alternativa é a mais adequada do ponto de vista de confiabilidade, entretanto demanda mais tempo e recursos. A segunda alternativa é indicada para tratamento de grandes quantidades de seqüências devido à rapidez em que os resultados são obtidos, apesar de possuir soluções com certa margem de erro. Entretanto na biologia, resultados aproximados também podem ser utilizados em pesquisas e estudos na área, justificando assim, a possibilidade de se utilizar algoritmos que encontrem soluções sub-ótimas em um tempo computacional aceitável. Desta forma, métodos de inteligência computacional tais como: redes neurais artificiais, lógica *fuzzy*, *clustering*, programação genética, dentre outras, vêm sendo estudadas na tentativa de encontrar soluções ao problema da classificação de proteínas (McGARRAH & JUDSON, 1993; HERING et al, 2002; PICCOLBONI & MAURI, 1997; KING et al, 2000; WEINERT & LOPES, 2004).

6 http://www.sanger.ac.uk/Software/Pfam/
7 http://pir.georgetown.edu/
8 http://smart.embl-heidelberg.de/
9 http://ca.expasy.org/sprot/

1.2 Motivação

Quando é observada a seqüência de um genoma vê-se um pouco mais que uma ou várias longas cadeias de caracteres. Ali está a informação que se busca, mas de forma que não se sabe interpretar. A anotação genômica se refere a tarefa de entender o que diz a seqüência de um genoma: basicamente, quais genes contém, onde se encontram e quais funções realizam as proteínas que são codificadas por eles (RUST et al, 2002).

Devido à elevada quantidade de seqüências a anotação não é repetida com freqüência, uma vez que esta repetição levaria muito tempo. Conseqüentemente, algumas seqüências depositadas que não possuem classificação podem ter similaridade com alguma seqüência classificada recentemente, necessitando serem re-anotadas. Existe atualmente um grande número de proteínas depositadas que ainda não puderam ser classificadas devido às deficiências dos métodos que são utilizados. Também existem seqüências anotadas em uma classe que podem ter sua classificação modificada pelo fato de um novo domínio, presente na proteína, ter sido identificado recentemente.

Para os casos onde os dados, a serem utilizados por métodos de inteligência computacional possuem valores nominais, como seqüências de nucleotídeos (alfabeto de 4 letras) e aminoácidos (alfabeto de 20 letras), cada elemento deve ser convertido em um valor numérico já que os dados de entrada devem ser, necessariamente, numéricos. Além disso, é importante perceber que para se utilizar métodos de inteligência computacional em uma determinada aplicação, os dados de entrada devem possuir sempre a mesma dimensão. Entretanto a Figura 1.1 mostra que quantidade de aminoácidos entre um conjunto de seqüências pertencentes ao COG é variável. O eixo horizontal representa um conjunto de proteínas que pertencem ao COG, e, o eixo vertical representa a quantidade de aminoácidos encontrada nestas proteínas.

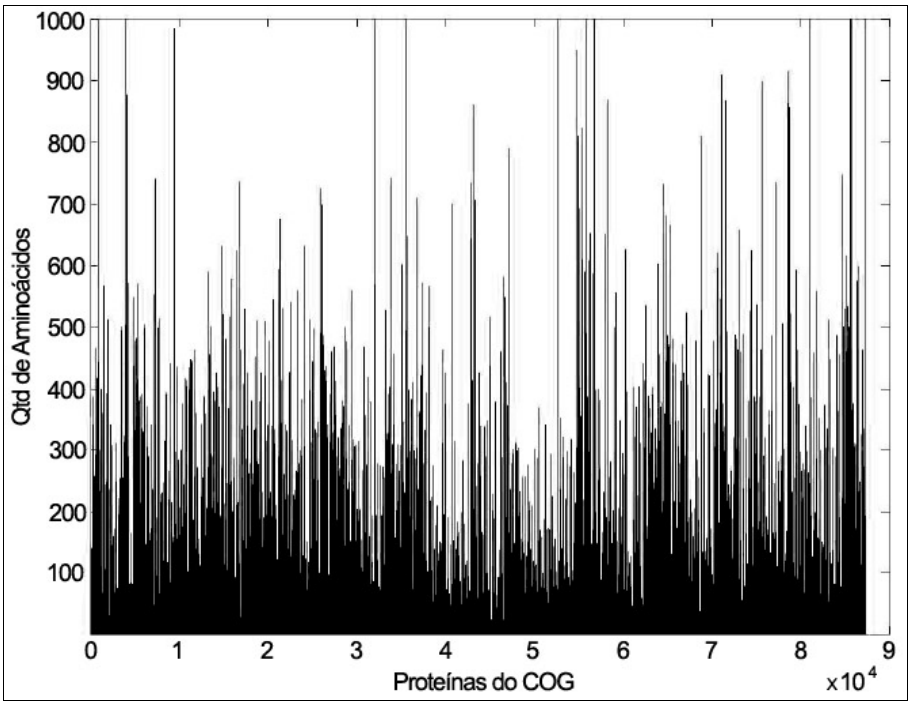


Figura 1.1: Quantidade de aminoácidos de um conjunto de proteínas armazenadas no banco de dados público de proteínas COG, onde pode ser observada a diferença de dimensionalidade entre os dados. (Fonte: (RODRIGUES, 2007)

Uma forma de se utilizar um conjunto de seqüências de nucleotídeos ou aminoácidos é selecionar somente uma faixa das seqüências sempre de mesma dimensão e aplicar a codificação direta. Esta metodologia é útil em aplicações que utilizam subseqüências como dados de entrada. Entretanto, em operações onde todos os resíduos de aminoácidos são relevantes, tal como uma classificação funcional, a seleção de uma faixa da seqüência original se torna inapropriada, pois se algum domínio importante para a função desta proteína não for selecionado, o conjunto de dados resultante não terá representatividade.

Um método de codificação de seqüências conhecido como *Sequence Coding by Sliding Window – (SCSW)* (BLAISDELL, 1986 e RODRIGUES et al, 2003), pode ser utilizado para extrair a informação de uma seqüência completa e representá-las em vetores de mesma dimensão. Entretanto é necessário verificar se as seqüências codificadas não perderam informações significativas, principalmente aquelas que influenciam em sua classificação funcional.

1.3 Objetivo Geral

O objetivo geral deste trabalho é avaliar um esquema de codificação para proteínas que gere vetores de mesma dimensão, independente do tamanho das seqüências, de modo que estes vetores possam ser utilizados pelos diversos métodos, citados na seção 1.1, para a classificação de proteínas.

1.4 Objetivos Específicos

O presente trabalho apresenta os seguintes objetivos específicos:

- Selecionar dois conjuntos de seqüências de aminoácidos, sendo cada conjunto originado de proteínas de um mesmo grupo funcional.
- Aplicar a metodologia de codificação de seqüências *Sequence Coding by Sliding Window* à todas seqüências que foram selecionadas.
- Avaliar se a codificação SCSW foi capaz de reter informações relevantes sobre a seqüência. Para isso será utilizado um algoritmo de clusterização conhecido como K-Means.
- Analisar e discutir os resultados encontrados assim como o método apresentado.

1.5 Organização do Texto

Este trabalho está organizado da seguinte maneira:

- O Capítulo 2 apresenta conceitos sobre a comparação de seqüências, o método de codificação de seqüências *Sequence Coding by Sliding Window (SCSW)* e algumas aplicações para medir similaridade e dissimilaridade entre seqüências.
- O Capítulo 3 contém o desenvolvimento do trabalho propriamente dito. Neste capítulo é apresentado o tipo da pesquisa, a metodologia utilizada na seleção dos dados, os métodos utilizados no processo de codificação, e, por fim, a avaliação das seqüências codificadas.

- O Capítulo 4 apresenta os resultados deste trabalho, onde foram realizados testes com o esquema de codificação *Sequence Coding by Sliding Window*, a fim de verificar sua eficácia em reter as informações relevantes sobre a seqüência. Também será apresentada a discussão dos resultados encontrados.

- Finalizando, o Capítulo 5 apresenta a conclusão deste trabalho e propostas de continuidade.

2 REFERENCIAL TEÓRICO

Nesta sessão encontra-se a base teórica necessária à compreensão do trabalho. Primeiramente serão levantados alguns conceitos sobre a comparação de seqüências. Em seguida será abordado o método de codificação *Sequence Coding by Sliding Window (SCSW)* e o método de clusterização K-Means.

2.1 Comparação de Seqüências

Quando se utilizam métodos computacionais, a comparação de seqüências é a operação mais intuitiva e direta na análise de proteínas, embora uma proteína seja descrita sobre quatro aspectos relacionados à estrutura (VOET, 2000):

• estrutura primária: está relacionada ao número, à espécie e à seqüência (ordem) dos aminoácidos que compõem a proteína. É o nível estrutural mais simples e mais importante, pois dele deriva todo o arranjo espacial da molécula e, conseqüentemente, sua especificidade.

• estrutura secundária: é dada pelo arranjo espacial de aminoácidos próximos entre si na seqüência primária da proteína. Os dois arranjos locais mais comuns nas proteínas são a α-hélice e a folha-β.

• estrutura terciária: diz respeito à forma tridimensional específica assumida pela proteína como resultado do enovelamento global de toda a cadeia. É dada pelo arranjo espacial de aminoácidos distantes entre si na seqüência polipeptídica.

• estrutura quaternária: algumas proteínas são multiméricas, ou seja, compostas por duas ou mais cadeias polipeptídicas. A forma como estas cadeias se agrupam e se ajustam para formar a estrutura total da proteína é descrita pela estrutura quaternária.

A Figura 2.1 ilustra os quatro aspectos relacionados à estrutura.

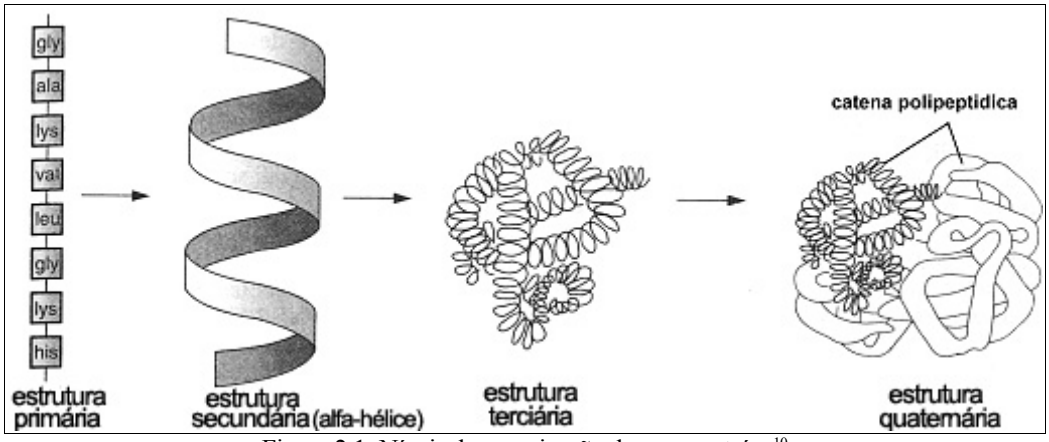


Figura 2.1: Níveis de organização de uma proteína[10]

Normalmente os métodos computacionais realizam comparação de proteínas através de suas estruturas primárias que, na maioria dos bancos de dados, são disponibilizadas em formato FASTA. Uma seqüência no formato FASTA inicia com uma linha de identificação começando pelo símbolo ">", seguido por outras linhas contendo a seqüência propriamente dita, como exemplificado na Figura 2.2:[10]

```
>gi|149245188|ref|XP_001527128.1| 60S ribosomal protein L17 [Lodderomyces
elongisporus NRRL YB-4239]
MVRYAAKASNPAKSASARGSYLRVSFKNTRETVQAINGWKLTKAQKYLDQVLDHERAIPFRRFNHSIGRT
AQGKEFGVTKARWPAKSVNFVKDLLRNAQSNAEAKGLDVEKLTISHIQVNQAPKQRRRTYRAHGRINAYQ
SSPSHIELTLTEEDEVVEKATDKKVGRLNARQRGRLASQKRLTAA
```

Figura 2.2: Seqüência de uma proteína ribossômica no formato FASTA.
(Fonte: http://www.ncbi.nlm.nih.gov/)

Quando a comparação entre duas proteínas indica um alto grau de similaridade, pode sugerir relações envolvendo estrutura, função e evolução, pois essas proteínas provavelmente pertencem a um ancestral comum. Essa similaridade pode oferecer muitas informações sobre como ocorreu a evolução dos organismos e também sobre as próprias proteínas, uma vez que algumas mudanças e mutações interferem na estrutura ou na função da proteína, acarretando mudanças no modo de vida dos organismos. O grau de certeza que estas características podem ser associadas depende de quão similar as duas proteínas são. Mesmo se a similaridade das seqüências for relativamente distante, é possível que assumam estruturas secundárias e terciárias semelhantes, sugerindo uma classificação funcional que pode servir como base para a realização de experimentos com esta nova proteína (EIDHAMMER et al., 2004).

10 Adaptado de: http://www.summagallicana.it/Volume2/001fig005.jpg

Segundo Korf et al. (2003) e Pearson (1990), as seqüências biológicas (DNAs, RNAs e proteínas) são comparadas principalmente em pares por um processo denominado alinhamento par-a-par de seqüências, o qual consiste em posicionar as seqüências lado-a-lado de tal forma que o número de posições idênticas entre as duas seja maximizado. O alinhamento entre duas seqüências de caracteres pode ser visto como essas seqüências dispostas em uma matriz 2 × n, onde n indica o número de caracteres alinhados. Cada seqüência está disposta em uma linha da matriz e cada um de seus caracteres em uma coluna, sempre mantendo a mesma ordem (Figura 2.3).

```
SEQ1: SIGRTAQGKE-GVTKARWPAKSVNFVKDLLRNAQSNAEAKGLDVEKLTISHIQVNQA
      SIGRTAQGKE GVTKARWPAKSV FV+ LL+NA +NAEAKGLD  KL +SHIQVNQA
SEQ2: SIGRTAQGKEFGVTKARWPAKSVKFVQGLLQNAAANAEAKGLDATKLYVSHIQVNQA
```

Figura 2.3: Trecho de um alinhamento entre duas seqüências de proteínas. A linha central mostra os pares idênticos. A presença do sinal "+" indica pares diferentes que possuem similares (R e K – polares e positivos, V e I – hidrofóbicos, D e N - polares) e o espaço vazio " " representa tanto aminoácidos diferentes que não possuem similaridades quanto presença de Gaps[11] (representado nas seqüências pelo sinal "-").

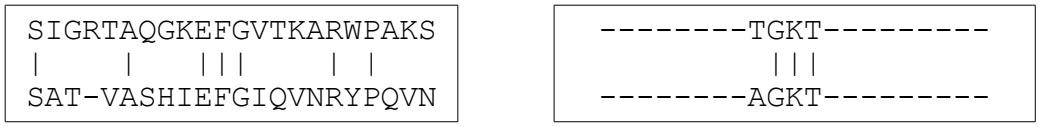
A qualidade de um alinhamento é medida pelo *score* de alinhamento que é simplesmente a soma dos *scores* de cada caractere alinhado. O alinhamento com um *gap* também possui um *score* associado, normalmente baixo. A soma dos *scores* resulta em uma pontuação para o alinhamento, que é proporcional à similaridade entre as duas seqüências em questão.

É importante ressaltar que, freqüentemente, é possível haver várias combinações de alinhamento entre duas seqüências de caracteres e algumas regiões podem ter um alinhamento muito melhor que outras regiões. Deste modo, as regiões com o melhor alinhamento sempre possuem prioridade.

Alinhamentos par-a-par são executados de forma global ou local, dependendo do contexto e do propósito do alinhamento. Os alinhamentos globais forçam um alinhamento completo das seqüências de entrada, ou seja, todos os caracteres são usados Figura 2.4(a). Este tipo de alinhamento é apropriado para comparações de seqüências que são muito similares e que possuem o mesmo tamanho (NEEDLEMAN & WUNSCH, 1970). No alinhamento local, são alinhadas os trechos das seqüências com a mais alta densidade de similaridade, gerando ilhas de sub-alinhamentos entre estas seqüências, como mostrado na Figura 2.4(b). Este alinhamento é apropriado para comparações de seqüências: similares em certas regiões e

11 Espaçamentos que são incluídos para possibilitar um melhor alinhamento.

dissimilares em outras, de diferentes tamanhos ou que conservam uma certa região ou domínio (SMITH & WATERMAN, 1981).

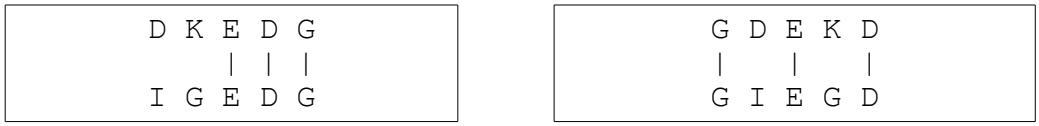


(a) (b)

Figura 2.4: Exemplo de um alinhamento global em (a) e um alinhamento local em (b).

Entretanto, os métodos de alinhamento par-a-par possuem uma limitação que deve ser destacada: quando é calculado o *score* em alinhamentos par-a-par, caracteres seqüenciais e caracteres individuais possuem o mesmo valor. Entretanto o alinhamento seqüencial de caracteres deveria ter um valor mais significativo, uma vez que a subseqüência alinhada pode representar um domínio relevante para a função das proteínas que estão sendo alinhadas (VINGA & ALMEIDA, 2003)

Na Figura 2.5(a) e Figura 2.5(b) as seqüências possuem pares de alinhamento iguais, trocando apenas a ordem em que estes aparecem na seqüência, porém em ambos os alinhamentos o *score* é o mesmo. Entretanto, o alinhamento representado na Figura 2.5(a) deveria ter um *score* total maior, pois esta seqüência tem grande probabilidade de ser um domínio que caracteriza a função das duas seqüências.



(a) (b)

Figura 2.5: Caracteres isolados em (b) e seqüência de caracteres em (a)

Portanto, a fim de superar a limitação dos métodos de alinhamento par-a-par, foi utilizado o método SCSW para medir a similaridade entre duas seqüências.

2.2 Esquema de Codificação Sequence Coding by Sliding Window

Vetores de mesma dimensão já foram utilizados em vários trabalhos a fim de medir a similaridade entre duas seqüências (BLAISDELL, 1986; WU et al, 1997; PETRILLI, 1993). A codificação denominada de *Sequence Coding by Sliding Window (SCSW)* (RODRIGUES et al, 2004 e RODRIGUES et al, 2003) baseada na codificação proposta por (BLAISDELL, 1986) e utilizada em diversos trabalhos como (PETRILLI, 1993; WU et al, 1992; WU et al, 1997; RODRIGUES et al, 2003; RODRIGUES et al, 2004; RODRIGUES et al, 2005) converte seqüências de dimensões diferentes em vetores de mesma dimensão, solucionando o problema da diferença de dimensionalidade. Segundo Rodrigues (2007) a codificação é definida da seguinte forma:

1) Considerando uma seqüência qualquer S de tamanho N definida sobre um alfabeto de tamanho α;

2) Uma janela deslizante w_n de tamanho $1 \leqslant n \leqslant N$ é posicionada na posição 1 da seqüência S e vai sendo deslocada até posição $N - n + 1$;

3) Um vetor V_n de dimensão α^n é definido, onde cada posição corresponde a uma possível ***n-tupla*** dos elementos de α;

4) A cada deslocamento de w_n em S a posição de V_n correspondente à ***n-tupla*** encontrada é incrementada de 1;

5) Após w_n atingir a posição $N - n + 1$ em S, o vetor V_n conterá a quantidade de cada ***n-tupla*** da seqüência percorrida e, independentemente do tamanho da seqüência, o vetor V_n terá dimensão α^n.

Para exemplificar, uma proteína com 274 aminoácidos (Figura 2.6) é utilizada na codificação SCSW com janela deslizante de tamanho $n = 2$ (Tabela 2.1). Para uma melhor visualização, o vetor de tamanho 400 é apresentado em forma de uma matriz 20 × 20, onde cada posição corresponde a um par de aminoácidos equivalente a concatenação do índice da linha com o índice da coluna. Por exemplo, existe somente 1 subseqüência WF indicado pela

linha W coluna F. Da mesma forma existem 25 subseqüências GA indicado pela linha G coluna A .

```
MKFLKLVIIGALFLNVLCLDGGAQPPKSGDGGAQPPKSGDGGAQPPKSGDGGAQPPKSGDGGA
QPPKSGDGGAQPPKSGDGGAQPPKSGDGGAQPPKSGDGGAQPPKSGDGGAQPPKSGDGGAQPP
KSGDGGAQPPKSGDGGAQPPKSGDGGAQPPKSGDGGAQPPKSGDGGAQPPKSGDGGAQPPKSG
DGGAQPPKSGDGGAQPPKSGDGGAQPPKSGDGGAQPPKSGDGGAQPPKSGDGGAQPPKSGAQR
PFSHWIAGWFLVPLEVKASDHF
```

Figura 2.6: Antígeno Cs44 do Clonorchis sinensis (gi:4927222)
(Fonte: http://www.ncbi.nlm.nih.gov/)

Tabela 2.1: Matriz resultante da codificação SCSW aplicada à seqüência da Figura 2.6

	M	A	C	D	E	F	G	H	I	K	L	N	P	Q	R	S	T	V	W	Y
M	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
A	0	0	0	0	0	0	1	0	0	0	1	0	0	24	0	1	0	0	0	0
C	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
D	0	0	0	0	0	0	23	1	0	0	0	0	0	0	0	0	0	0	0	0
E	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
F	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	1	0	0	0	0
G	0	25	0	22	0	0	23	0	0	0	0	0	0	0	0	0	0	0	1	0
H	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0
I	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0
K	0	1	0	0	0	1	0	0	0	0	1	0	0	0	0	23	0	0	0	0
L	0	0	1	1	1	1	0	0	0	1	0	1	0	0	0	0	0	2	0	0
N	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
P	0	0	0	0	0	1	0	0	0	23	1	0	23	0	0	0	0	0	0	0
Q	0	0	0	0	0	0	0	0	0	0	0	0	23	0	1	0	0	0	0	0
R	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
S	0	0	0	1	0	0	23	1	0	0	0	0	0	0	0	0	0	0	0	0
T	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
V	0	0	0	0	0	0	0	0	1	1	1	0	1	0	0	0	0	0	0	0
W	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
Y	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Fonte:(RODRIGUES, 2007)

Segundo Hide et al. (1994) e Blaisdell (1986), a codificação SCSW é eficiente computacionalmente quando utilizada na busca de similaridade e dissimilaridade, podendo encontrar características que não são analisadas pelos algoritmos de alinhamento par-a-par (PEARSON, 1990; NEEDLEMAN & WUNSCH, 1970; SMITH & WATERMAN, 1981), como por exemplo, as seqüências de caracteres têm maior relevância que caracteres individuais quando são comparados os vetores resultantes da codificação. Uma demonstração pode ser vista nas Tabelas 2.2 e 2.3 que representam a codificação SCSW aplicada às seqüências da Figura 2.5(a) e (b) respectivamente, com janela deslizante de tamanho n = 2. Nos vetores, as duplas de caracteres sem representatividade possuem valor 0. Se for considerada a distância

Euclidiana (Equação 2.1), a distância calculada na Tabela 2.2 será igual a 2 enquanto que na Tabela 2.3 será igual a 2,45.

$$d(x,y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \; para\; x\; e\; y\; com\; i\; dimensões \quad (2.1)$$

Tabela 2.2: SCSW aplicado à seqüência da Figura 2.5(a)

	DG	**DK**	**ED**	**GE**	**IG**	**KE**
SEQ 1	1	1	1			1
SEQ 2	1		1	1	1	

Tabela 2.3: SCSW aplicado à seqüência da Figura 2.5(b)

	DE	**EG**	**EK**	**GD**	**GI**	**IE**	**KD**
SEQ 1	1		1	1			1
SEQ 2		1		1	1	1	

Como pode ser observado, as seqüências da Tabela 2.2 são mais "próximas" que as seqüências da Tabela 2.3, portanto, no método de codificação estudado, as seqüência de caracteres possuem maior relevância em comparação à caracteres isolados.

2.3 Métodos de Clusterização

A clusterização, também conhecida por agrupamento, é um método que segmenta uma grande população em um número pequeno de subgrupos (*clusters*), não possuindo classes pré-determinadas. Esta técnica pode proporcionar o direcionamento dos dados para cada cluster, considerando as semelhanças entre os mesmos. Os métodos de clusterização utilizados para a criação de grupos de objetos baseiam-se nas semelhanças e diferenças entre os elementos. Esse agrupamento é tal que o grau de associação entre elementos do mesmo grupo é alto e entre elementos de grupos diferentes é baixo.

2.3.1 Método de K-Means

Dentre os algoritmos de clusterização, o k-Means é um método clássico da literatura que busca os centros de agrupamentos pela minimização direta do critério de erro calculado em função da distância. Como a maioria dos métodos de agrupamentos não-supervisionados, o algoritmo k-Means necessita da definição a priori do número de agrupamentos K (do nome k-Means) (MACQUEEN, 1966).

A partir de uma estimativa inicial das coordenadas dos centros de agrupamento, o algoritmo calcula a distância de cada ponto do conjunto de treinamento às coordenadas de cada centro de agrupamento que foi estimado (Figura 2.7).

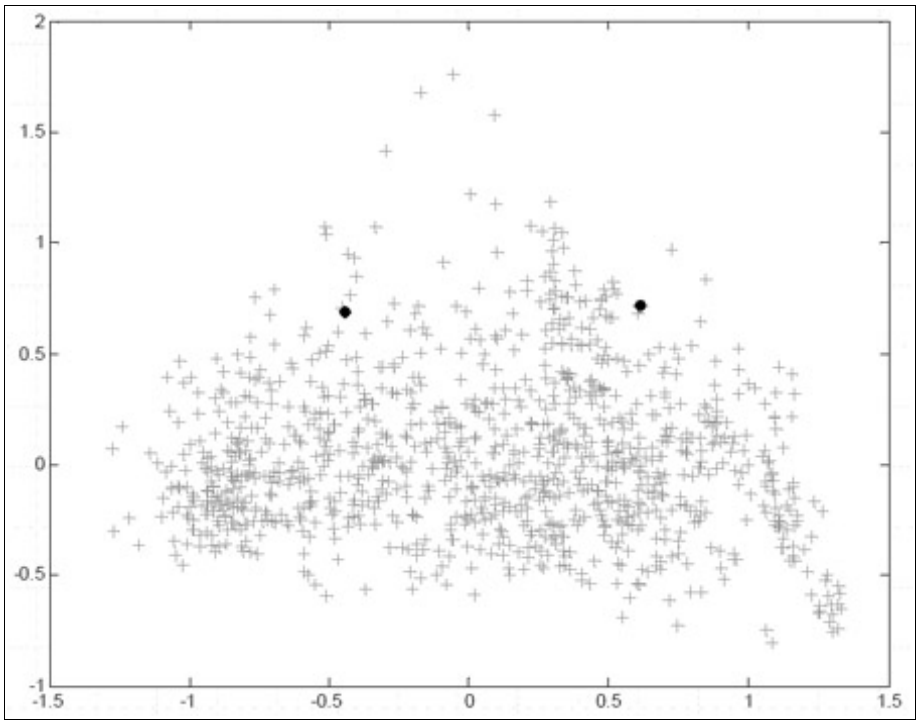


Figura 2.7: Conjunto de pontos em um plano cartesiano (duas dimensões) e dois centros de agrupamentos escolhidos aleatoriamente (k=2).

O centro de agrupamento que estiver mais próximo do ponto que está sendo analisado, alocará este ponto para o grupo correspondente a este centro (Figura 2.8).

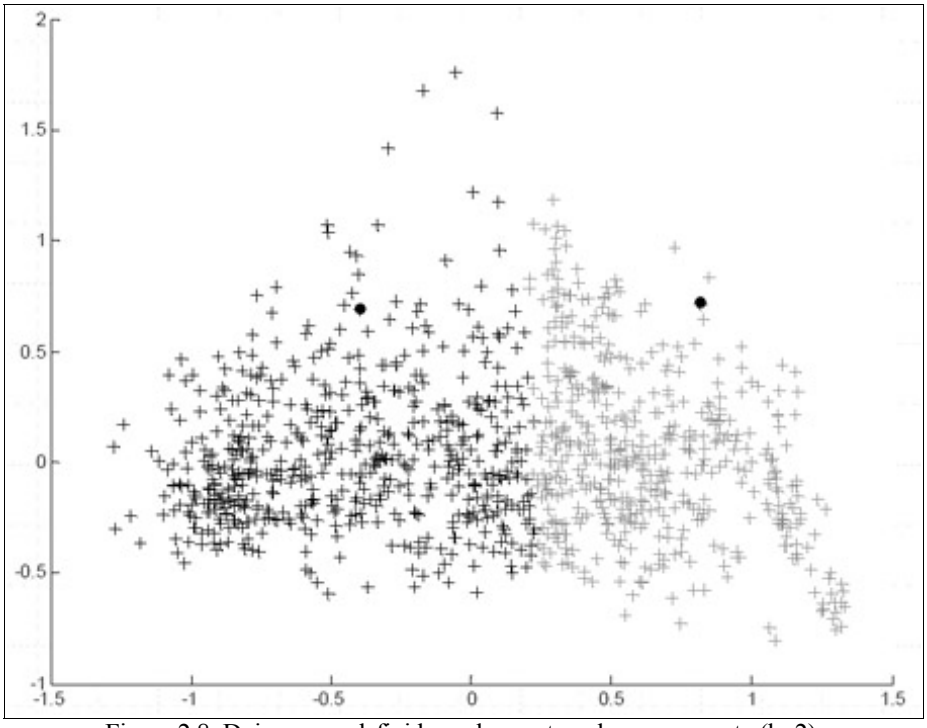


Figura 2.8: Dois grupos definidos pelos centros de agrupamento (k=2).

O novo valor do centro de agrupamento de cada grupo é calculado fazendo-se a média aritmética dos pontos que pertencem aos respectivos grupos (Figura 2.9).

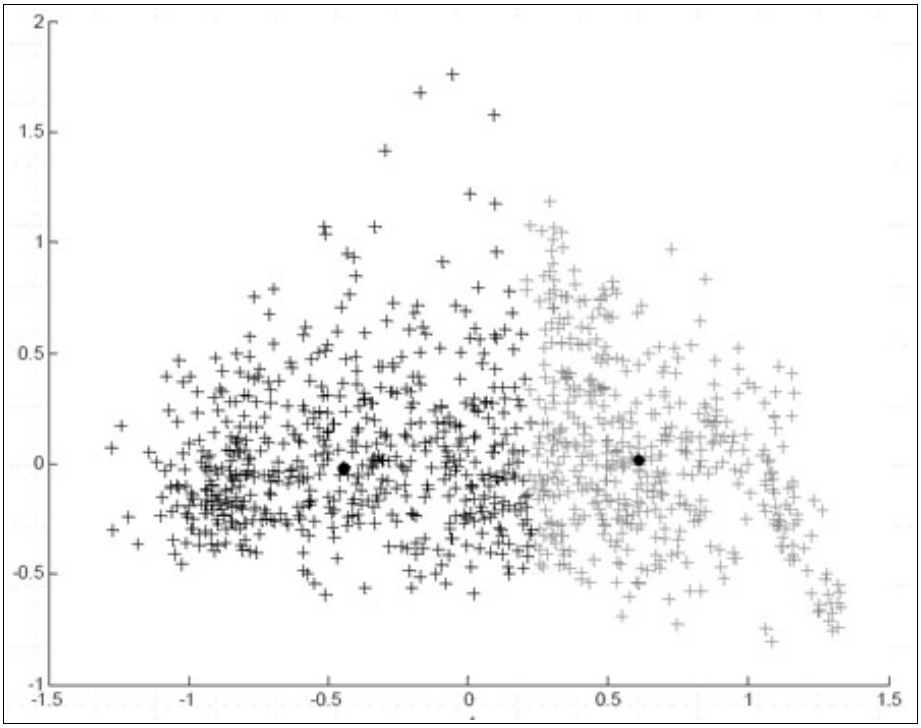


Figura 2.9: Cada grupo recalcula seu centro fazendo-se a média aritmética de seus pontos.

Desse modo, é finalizada uma iteração do algoritmo e uma nova é recomeçada, ou seja, é re-calculada a distância dos pontos do conjunto de treinamento às coordenadas de cada centro de agrupamento que foi calculado. O centro de agrupamento que estiver mais próximo do ponto que está sendo analisado, alocará este ponto para o grupo correspondente a este centro, e assim por diante, até o começo de uma nova iteração. A condição de parada pode ser determinada pelo número de iterações desejadas, ou até o momento em que as coordenadas dos centros de agrupamento não alterarem durante mais de uma iteração.

A seqüência de execução do algoritmo de K-Means pode ser resumida da seguinte forma:

1) Definir um número k de agrupamentos.

2) Escolher valores iniciais aleatórios para os centróides μ_i. (Centro de um conjunto de pontos x_j, pertencentes ao grupo S_i $(i = 1...k)$).

3) Cada ponto x_j é associado ao grupo S_i cujo centróide μ_i seja o mais próximo de x_j .

4) Os centróides de cada grupo S_i são recalculados com base nos pontos associados a eles.

5) Repete-se os passos 3 e 4 até a convergência (os centróides não mudam mais de lugar).

O objetivo do algoritmo é minimizar a variância V dos atributos dos pontos que pertencem a um determinado grupo.

$$V = \sum_{i=1}^{k} \sum_{x_j \in S_i} |x_j - \mu_i|^2 \quad (2.2)$$

O algoritmo de K-Means será utilizado para testar o esquema de codificação SCSW, com o propósito de avaliar sua eficácia na determinação de similaridade entre seqüências. Os procedimentos adotados na clusterização de seqüenciais são mostrados na sessão 3.2.

3 METODOLOGIA

Este capítulo apresenta, inicialmente, o tipo de pesquisa em que se enquadra o presente trabalho. Em seguida, são apresentados os procedimentos utilizados na obtenção dos dados. E por fim, são apresentados os métodos e materiais aplicados no teste do esquema de codificação SCSW.

3.1 Tipo de Pesquisa

De acordo com Jung (2004), a pesquisa desenvolvida é do tipo tecnológica, uma vez que se utiliza de conhecimentos e experiências adquiridos por estudiosos e profissionais na área de bioinformática, e, aplica técnicas que já existem na literatura.

Quanto ao objetivo, esta é uma pesquisa exploratória, pois segundo Jung (2004), na pesquisa exploratória estuda-se um fenômeno atual, ainda pouco examinado entre as comunidades, assim como o tema tratado neste trabalho. As investigações desta natureza objetivam aproximar o pesquisador do fenômeno para que este possa familiarizar-se com as características e peculiaridades do tema a ser explorado, para assim definir o problema com maior precisão.

Segundo Jung (2004), considerando-se os procedimentos a serem adotados, esta pesquisa é do tipo operacional, uma vez que aplica métodos científicos a problemas complexos para auxiliar no processo de tomada de decisões.

3.2 Obtenção dos Dados

As seqüências utilizadas neste trabalho são provenientes do NCBI[12] (*National Center for Biotechnology Information*), recurso do governo americano que serve como fonte de informação para a área da Biologia Molecular e Bioinformática. O NCBI cria bases de dados públicas, conduz investigação em biologia computacional, desenvolve software para análise de dados genômicos e dissemina a informação biomédica. Esta base de dados formou-se em 1988,

12 http://www.ncbi.nlm.nih.gov/

financiada pelo NIH[13] (*National Institute of Health*), foi construída com base em seqüências de nucleotídeos submetidas por laboratórios de investigação, incluindo dados trocados com outras bases de seqüências de nucleotídeos, como o EMBL e a DDBJ.

Neste trabalho, a coleta de seqüências foi dividida em duas etapas:

Na primeira etapa, as seqüências foram extraídas da base de dados do COG[14] ("*Clusters of Orthologous Groups*"). Cada COG é formado por um grupo de proteínas originadas de genes ortólogos[15], proporcionando assim, recursos na tentativa de classificação filogenética de proteínas. O COG pode ser usado para diversos fins, tais como: identificar similaridade e diferenças entre espécies, identificar famílias de proteínas e predição de novas proteínas (TATUSOV et al, 2000). Nesta etapa foram selecionadas seqüências de dois COGs diferentes: COG1187 (*16S rRNA uridine-516 pseudouridylate synthase and related pseudouridylate synthases*) do grupo funcional J (no processo de armazenamento de informação e processamento, é responsável pela tradução, estrutura ribossômica e biogênese) e o COG0372 (*Citrate synthase*) do grupo funcional C (no processo de metabolismo, é responsável pela conservação e produção de energia), sendo o primeiro cog formado por 64 proteínas e o segundo cog formado por 50 proteínas (formato FASTA).

Na segunda etapa, as seqüências foram obtidas com o auxílio da ferramenta BLAST (*Basic Local Alignment Search Tool*)[16], que segundo Korf et al. (2003), é o método mais comum para realizar buscas de seqüências similares em bancos de dados de seqüências, sendo que suas implementações mais conhecidas são a do NCBI e o da *University of Washington*, conhecido como WU-BLAST. Uma comparação entre os parâmetros das versões WU-BLAST e NCBI- BLAST pode ser vista no site do WU-BLAST[17]. Neste trabalho são analisadas apenas seqüências protéicas, sendo assim, nesta etapa é necessário um dos serviços oferecidos pelo BLAST, o Protein-blast, que é empregado na comparação de seqüências de aminoácidos em bancos de dados de proteínas. Através do Protein-blast, foram selecionados dois grupos de seqüências de aminoácidos: o primeiro grupo formado pelas 250 seqüências mais semelhantes a seqüência definida na Figura 3.1 e o segundo grupo formado pelas 250 seqüências mais semelhantes a seqüência definida na Figura 3.2.

13 http://www.nih.gov/
14 http://www.ncbi.nlm.nih.gov/COG/
15 Genes ortólogos são genes de diferentes espécies que derivam de um único gene ancestral do último descendente comum das respectivas espécies.
16 http://www.ncbi.nlm.nih.gov/blast/
17 http://blast.wustl.edu/blast/

```
>gi|16077174|ref|NP_387987.1| hypothetical protein BSU01060 [Bacillus
subtilis subsp. subtilis str. 168]
MSEHYYSEKPSVKSNKQTWSFRLRNKDFTFTSDSGVFSKKEVDFGSRLLIDSFEEPEVEGGILDVGCGYG
PIGLSLASDFKDRTIHMIDVNERAVELSNENAEQNGITNVKIYQSDLFSNVDSAQTFASILTNPPIRAGK
KVVHAIFEKSAEHLKASGELWIVIQKKQGAPSAIEKLEELFDEVSVVQKKKGYYIIKAKKV
```

Figura 3.1: Seqüência de aminoácidos (formato FASTA) utilizada no Protein-blast para formar o 1º grupo de seqüências. (Fonte: http://www.ncbi.nlm.nih.gov/)

```
>gi|11499434|ref|NP_070675.1| carbon monoxide dehydrogenase, catalytic
subunit (cooS) [Archaeoglobus fulgidus DSM 4304]
MKIEGKVSEHESINMMYERVSKEGVTNIVDRFNAQEKGRCPFCEKGLSCQLCSMGPCRISKDKPTGACGI
DAAGMVVRNFTHKNMLGTEAYTYHAIEAAKTLKATAEGKTIYEIKDVEKLKWFAKLLGIEGEDVNELAAK
VADFVISDLSSLEKSRLVEIFAPEKRKELWEKLGIFPSGVFQELLTMGSSAMTNVDSNYVSLAKKSMSMS
IATCMAAQIALETIQDILFGTPMPHESHSDLGILDPEYVNIAVNGHEPFVGIALIKLAEREEIQEKARKA
GAKGLRIIGFIETGQEILQRVDSPVFAGIVGNWIVQEYALATGCVDVFAADMNCTLPSLPEYQRYGVKIV
PVSRLVRLKGIDEGLDYEPEKAEEIAMKLIDMAIENFKQRDKSKAVKVEQKKKIVVGFSPEAILKALNGD
LNVLLDAIKKGDIKGVVALVSCTTLKNGPHDSSTVTIAKELIKRDILVLSMGCGNAALQVAGLTSMEAVE
LAGEKLKAVCKALNIPPVLSFGTCTDTGRAAYLVRLIADALGVDVPQLPVAVTAPEYMEQKATIDAVFAV
AYGLTTHVSPVPPITGSEDAVKLFTEDVEKLTGGKVVVEEDPLKAAELLEKVIEEKRKALGI
```

Figura 3.2: Seqüência de aminoácidos (formato FASTA) utilizada no protein-blast para formar o 2º grupo de seqüências. (Fonte: http://www.ncbi.nlm.nih.gov/)

A Figura 3.3 mostra todos os campos da página[18] do Protein-blast onde são passados os parâmetros necessários para executar a comparação de seqüências.

18 Esta página pode ser acessada através de: http://www.ncbi.nlm.nih.gov/blast/

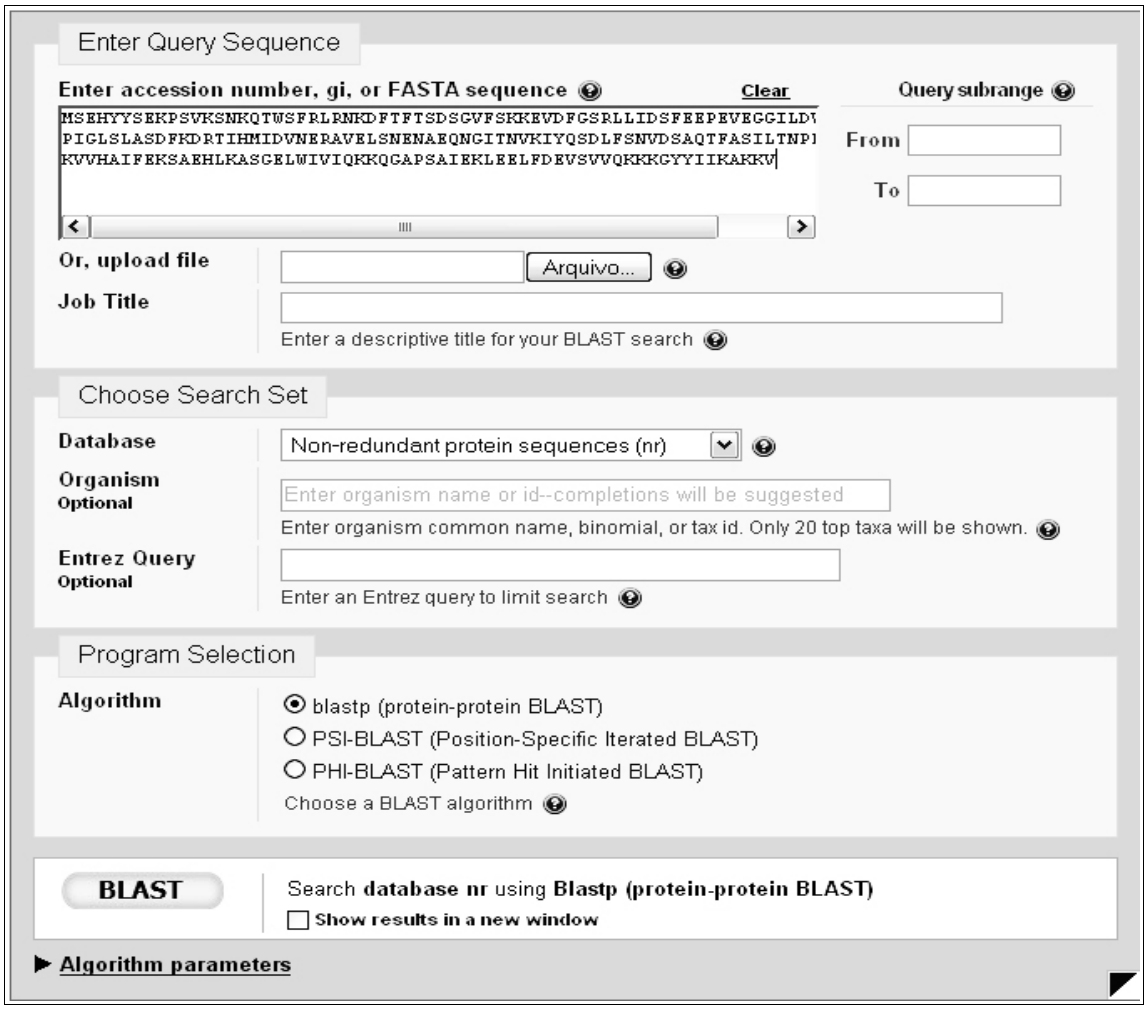


Figura 3.3: Campos da página do protein-blast onde serão passados os parâmetros de entrada.
(Fonte: http://www.ncbi.nlm.nih.gov/blast/)

Na área "*Enter Query Sequence*" tem-se o campo "*Enter accession number, gi, or FASTA sequence*" onde é passada a seqüência que vai servir como referência na busca por seqüências semelhantes. Já na área "*Choose Search Set*" tem-se o campo "*Database*" onde é escolhido um banco de dados sem seqüências redundantes (*Non-redundant protein sequences*). No campo "*Algorithm*" da área "*Program Selection*" é escolhido o algoritmo *blastp*. Por padrão, é exibido apenas as 100 seqüências mais semelhantes à seqüência de referência, sendo assim, é necessário habilitar a visão dos parâmetros do algoritmo através do link "*Algorithm parameters*", localizado no canto inferior esquerdo da Figura 3.3. Os novos campos que serão habilitados são mostrados na Figura 3.4.

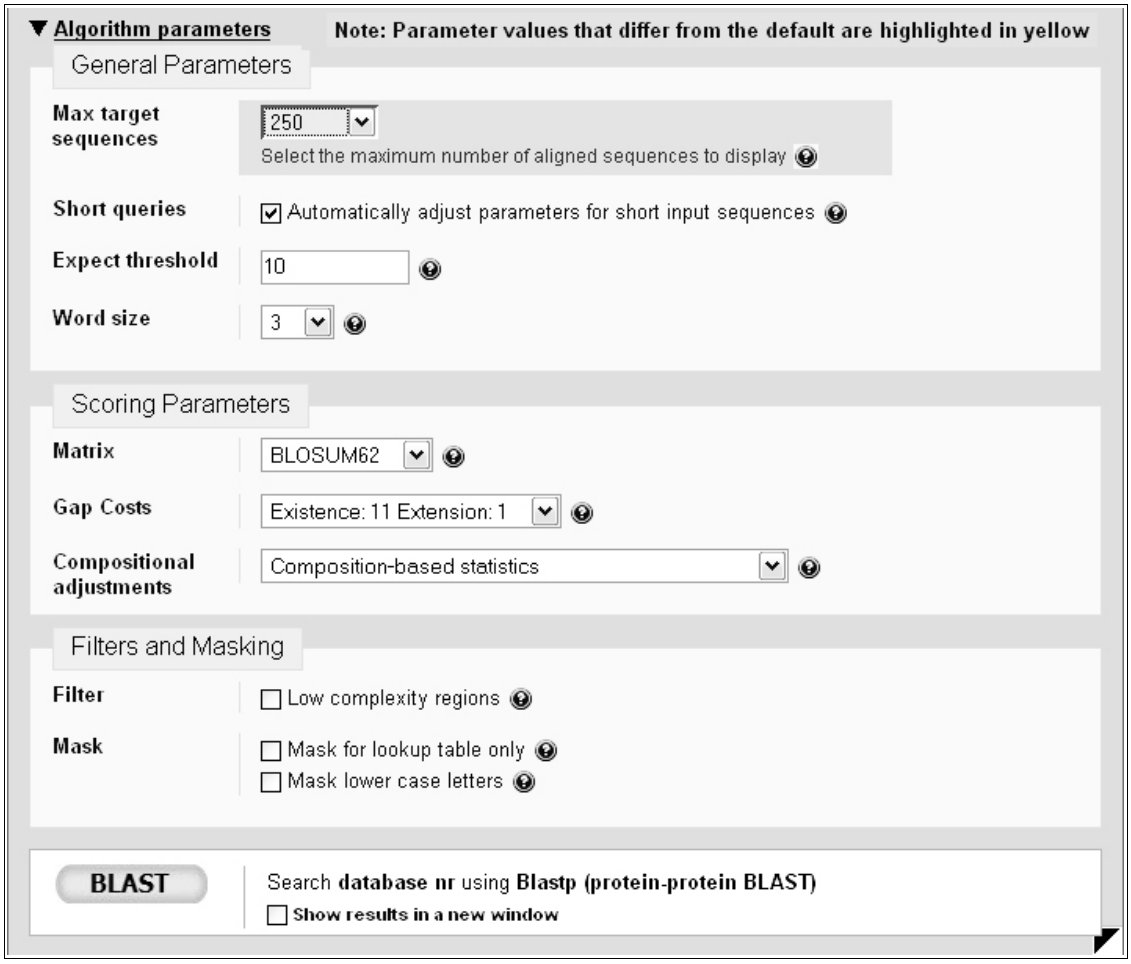


Figura 3.4: Campos onde serão passados os parâmetros de entrada do algoritmo blastp.
(Fonte: http://www.ncbi.nlm.nih.gov/blast/)

Na área "*General Parameters*" tem-se o campo "*Max target sequences*", onde é habilitada a exibição de 250 seqüências mais semelhantes à seqüência de referência. Já no restante dos campos, são usados os valores que são indicados por padrão.

Após escolher os parâmetros, basta clicar no botão "BLAST", localizado no canto inferior esquerdo da Figura 3.4, para iniciar a execução do Protein-blast e exibir as seqüências que foram encontradas (formato FASTA). Todo este processo será feito tanto para a seqüência da Figura 3.1 quanto para a seqüência da Figura 3.2.

3.3 Desenvolvimento

Para o desenvolvimento deste trabalho, foi utilizada a pesquisa bibliográfica, pois através dela é possível explicar um problema utilizando o conhecimento disponível a partir das teorias publicadas em livros, periódicos e obras congêneres. Segundo Jung (2004), na pesquisa bibliográfica o investigador levanta o conhecimento disponível na área, identificando as teorias produzidas, avaliando e analisando-as, com o objetivo de descrever, compreender ou explicar o problema objeto de investigação, portanto, torna-se um instrumento indispensável para qualquer tipo de pesquisa.

A pesquisa bibliográfica deu base para a aquisição de conhecimento sobre os temas envolvidos no projeto, principalmente no que diz respeito ao problema de classificação de proteínas. Envolveu, basicamente, consultas a livros de referência, artigos científicos e materiais disponibilizados na Internet.

Antes de avaliar o método *Sequence Coding by Sliding Window*, é necessário primeiramente codificar as seqüências (formato FASTA) obtidas na seção anterior. Portanto, foi desenvolvido um software para executar o processo de codificação SCSW. Este software, chamado neste trabalho de "Software Codificador", foi implementado utilizando a linguagem de programação C e compilado pelo GNU C Compiler (GCC) versão 3.4.5. para sistemas Windows®. O "Software Codificador" possiu complexidade $\mathbf{O}(\mathbf{n})$, onde $\mathbf{n}$ é o tamanho da seqüência a ser codificada.

O Software Codificador tem os seguintes parâmetros de entrada:

- Alfabeto: Arquivo contendo todos os caracteres pelo qual são formadas as seqüências. Como exemplo, um alfabeto de seqüências protéicas teria 20 caracteres, uma vez que estas seqüências podem ser formadas por até 20 aminoácidos diferentes e cada aminoácido é representado por um caractere.

- Tamanho da Janela: Como o próprio nome diz, é o tamanho da janela deslizante.

- Conjuntos: Um arquivo para cada conjunto de seqüências. As seqüências devem estar em formato FASTA.

Cada seqüência codificada é escrita num arquivo texto. Na primeira linha deste arquivo é armazenado um identificador para o grupo ao qual a seqüência pertence, que neste caso, é o nome do arquivo que contém o grupo da seqüência. Na segunda linha, tem-se a descrição da

proteína assim como é encontrado no formato FASTA. A seqüência codificada é encontrada na terceira linha em diante. No intuito de otimizar o processamento e reduzir o tamanho destes arquivos, cada ***n-tupla,*** onde ***n*** é o tamanho da janela, foi numerada de acordo com a ordem alfabética, de forma que na terceira linha em diante, cada linha possui o numero correspondente a ***n-tupla*** e o número de vezes que ela aparece na seqüência. As ***n*-tuplas** que não possuem representantes não são escritas. A Tabela 3.1 ilustra todo o processo apresentado neste parágrafo para um alfabeto de 3 caracteres e janela de tamanho 2.

Tabela 3.1: Esquema ilustrativo do "Software Codificador".

<table>
<tr><td>Parâmetros de entrada:</td><td>Alfabeto: ABC
Tamanho da Janela: 2
Conjuntos: grupo1, grupo2</td></tr>
<tr><td>Uma seqüência qualquer pertencente ao conjunto “grupo2”:</td><td>>gi|14000|ref|NP_075.1| ybvn protein [ATCC 1098]
AABCABCBAB</td></tr>
<tr><td>Vetor contendo a quantidade e a numeração das 2-tuplas:</td><td>
<table>
<tr><td>numeração</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td></tr>
<tr><td>2-tupla</td><td>AA</td><td>AB</td><td>AC</td><td>BA</td><td>BB</td><td>BC</td><td>CA</td><td>CB</td><td>CC</td></tr>
<tr><td>quantidade</td><td>1</td><td>3</td><td>0</td><td>1</td><td>0</td><td>2</td><td>1</td><td>1</td><td>0</td></tr>
</table>
</td></tr>
<tr><td>Arquivo texto:</td><td>grupo2
gi|14000|ref|NP_075.1| ybvn protein [ATCC 1098]
0 1
1 3
3 1
5 2
6 1
7 1</td></tr>
</table>

O próximo passo após o processo de codificação, é a avaliação do método *Sequence Coding by Sliding Window*. Esta avaliação consiste em verificar a integridade das seqüências codificadas, ou seja, mesmo depois da codificação, as seqüências deverão conservar características que possam ser usadas para identificar semelhanças entre as mesmas.

Cada grupo de proteínas é formado por seqüências que possuem certo nível de semelhança. Sendo assim, uma seqüência pertencente a um grupo x, é mais similar às seqüências deste grupo do que a um grupo y, e isso deverá ser mantido mesmo após o processo de codificação. Portanto, é necessário avaliar se as seqüências codificadas ainda são identificadas em seus respectivos grupos. Para isso foi utilizado o algoritmo de clusterização K-Means, que assim como outros métodos de clusterização, é usado para agrupar objetos baseados em suas similaridades ou diferenças. Para realizar o processo de clusterização

também foi construído um software, chamado neste trabalho de “Software de Clustering”. Este software também foi implementado utilizando a linguagem de programação C e compilado pelo GNU C Compiler (GCC) versão 3.4.5. para sistemas Windows®. O “Software de Clustering” possiu complexidade **O**(**nkt**), onde **n** é a quantidade de seqüências que serão clusterizadas, **k** é o número de grupos que serão formados e **t** é o número de iterações que o algoritmo irá realizar.

O Software de Clustering tem os seguintes parâmetros de entrada:

- Seqüências codificadas: local em disco onde se encontra as seqüências codificadas (arquivos texto gerado pelo Software Codificador).
- Número de Grupos: número de grupos que serão gerados pelo algoritmo de clusterização.

A seqüência de execução do Software de Clustering pode ser vista na Tabela 3.2:

Tabela 3.2: Esquema ilustrativo do "Software de Clustering.

1	**Escolha dos centros iniciais.**	No método de K-Means, cada grupo é formado por um centro de agrupamento, logo, a quantidade de centros iniciais será igual ao número de grupos passado como parâmetro. Os centros iniciais serão escolhidos aleatoriamente.
2	**Cálculo da distância entre seqüências e os centros de agrupamento.**	Será calculada a distância euclidiana de cada seqüência em relação aos centros de agrupamento. As figuras 2.5 (a) e (b) ilustram o cálculo da distância euclidiana entre duas seqüências
3	**Agrupamento.**	Cada seqüência será agrupada pelo centro mais próximo.
4	**Re-cálculo dos centros de agrupamento.**	Em cada grupo, as novas coordenadas do centro de agrupamento, serão obtidas pela média aritmética das seqüências que pertencem ao grupo.
5	**Analisar condição de parada.**	Se houver mudança de coordenada em pelo menos um centro de agrupamento, volta-se ao passo 2, senão, finaliza-se o processo de clusterização.

Após o processo clusterização, é necessário verificar se as seqüências contidas no mesmo grupo são realmente semelhantes entre si, ou seja, cada grupo deve ser formado por seqüências que pertenciam ao mesmo grupo original (antes da codificação).

Assim como na obtenção dos dados, o Software de Clustering foi executado em duas etapas. Na primeira etapa foram clusterizados as seqüências obtidos pelo COG. Na segunda etapa foram clusterizadas as seqüências obtidas pelo BLAST. Em ambas etapas, foram utilizadas janelas de tamanho de 1 a 6 e clusterização em 2 grupos.

4 RESULTADOS E DISCUSSÃO

Neste capítulo são mostrados os resultados do teste realizado com o esquema SCSW a fim de verificar sua aplicabilidade na busca de similaridade entre proteínas.

4.1 Teste do Esquema de Codificação SCSW

O teste foi realizado em duas etapas. Na primeira etapa foram clusterizadas as seqüências obtidas pelo COG. Na segunda etapa foram clusterizadas as seqüências obtidas pelo BLAST.

Na primeira etapa, 64 seqüências de aminoácidos do COG1187 (grupo funcional J) e 50 seqüências de aminoácidos do COG0372 (grupo funcional C) foram codificadas pelo método SCSW. Após a codificação, os dois grupos de proteínas foram clusterizados em 2 grupos pelo método de K-Means com janela deslizante de tamanho de 1 a 6. Os resultados são mostrados na Tabela 4.1. Nesta tabela, os grupos clusterizados foram representados conforme o COG dominante encontrado em cada grupo. Por exemplo, com janela deslizante de tamanho 1, foram formados dois grupos: um grupo contendo 62 seqüências (58 provenientes do COG1187 e 4 do COG0372) e um segundo grupo contendo 52 seqüências (6 provenientes do COG1187 e 46 do COG0372). No primeiro grupo, o COG1187 tem maior representatividade, logo, entende-se que este grupo é candidato a reconstruir o COG1187. Enquanto que o segundo grupo é candidato a reconstruir o COG0372, uma vez que as seqüências deste COG predominam neste grupo.

Tabela 4.1: Resultados da clusterização do COG0373 e do COG1187 através do método de K-Means.

Tamanho da janela deslizante	**Grupo candidato a COG1187**		**Grupo candidato a COG0372**	
	Quantidade	***Porcentagem %***	***Quantidade***	***Porcentagem %***
1	62	93,5%	52	88,5%
2	65	93,8%	49	93,9%
3	65	98,5%	49	100,0%
4	90	71,1%	24	100,0%
5	105	61,0%	9	100,0%
6	111	57,6%	3	100,0%

Com janela de tamanho 1, houve 93,5% de acerto no grupo candidato a COG1187 e 88,5% de acerto no grupo candidato a COG0372. Com janela de tamanho 2, estes valores passam para 93,8% e 93,9% respectivamente. Já com janela de tamanho 3, o grupo candidato a COG1187 obteve 98,5% de acerto e o grupo candidato a COG0372 obteve 100% de acerto, sendo estes os melhores resultados encontrados. Com janelas 4, 5 e 6 o grupo candidato a COG0372 ainda manteve 100% de acerto, porém a taxa de acerto no grupo candidato a COG1187 diminuiu.

Na segunda etapa, foram codificadas 500 seqüências obtidas pelo protein-BLAST. Sendo 250 semelhantes à seqüência da Figura 3.1 e 250 semelhantes a seqüência da Figura 3.2. Após a codificação, os dois grupos de proteínas foram clusterizados em 2 grupos pelo método de K-Means com janela deslizante de tamanho de 1 a 6. Os resultados são mostrados na Tabela 4.2. Esta tabela segue os mesmos princípios da Tabela 4.1, porém os grupos não são formados por seqüências do COG, e, sim, por seqüências obtidas através do protein-BLAST. Para facilitar a referência, o grupo clusterizado candidato ao grupo formado pela seqüência da Figura 3.1 será chamado de F3.1 e o grupo clusterizado candidato ao grupo formado pela seqüência da Figura 3.2 será chamado de F3.2.

Tabela 4.2: Resultados da clusterização dos grupos formados pelo protein-BLAST através do método de K-Means.

Tamanho da janela deslizante	**F3.1**		**F3.2**	
	Quantidade	***Porcentagem %***	***Quantidade***	***Porcentagem %***
1	249	99,6%	251	99,2%
2	248	100,0%	252	99,2%
3	131	82,4%	369	61,5%
4	74	68,9%	426	53,3%
5	30	100,0%	470	53,2%
6	1	100,0%	499	50,1%

Com janela de tamanho 1, houve 99,6% de acerto no grupo candidato a F3.1 e 99.2% de acerto no grupo candidato a F3.2. Já com janela de tamanho 2, o grupo candidato a F3.1 obteve 100% de acerto e o grupo candidato a F3.2 obteve 99,2% de acerto, sendo estes os melhores resultados encontrados. Com janelas 3 e 4 os resultados pioraram, e com as janelas 5 e 6, apesar dos grupos candidatos a F3.1 possuirem 100% de acerto, os grupos candidatos a F3.2 possuem uma taxa de acerto em torno de 50,0%.

4.2 Discussão dos Resultados

Os testes realizados na primeira e segunda etapa, mostraram que a cada acréscimo do tamanho da janela deslizante, os resultados obtidos melhoravam, até o momento em que o tamanho da janela deslizante, influenciava negativamente nos resultados. Segundo Rodrigues (2007), a redução da ambigüidade entre as seqüências codificadas, pode ser considerada como um dos motivos para melhoria dos resultados conforme o aumento da janela deslizante.

A ambigüidade de seqüências faz com que diferentes seqüências resultem em vetores idênticos ou muito semelhantes. Para exemplificar, considere as Figuras 4.1 e 4.2. As seqüências da Figura 4.1 possuem a mesma codificação quando é utilizada uma janela deslizante de tamanho 2. A Figura 4.2 mostra as seqüências da Figura 4.1 no vetor resultante (somente valores não nulos) da codificação SCSW.

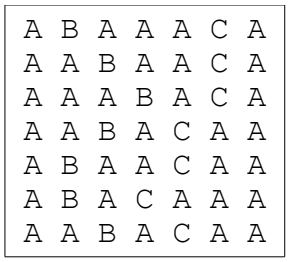


Figura 4.1: Seqüências que geram vetores idênticos quando utilizada janela deslizante de tamanho 2.

AA	AB	BA	AC	CA
2	1	1	1	1

Figura 4.2: Vetor resultante da codificação das seqüências da figura 4.1 com janela deslizante de tamanho 2

O problema de ambigüidade pode ser solucionado aumentando-se o tamanho da janela deslizante. Para as seqüências da Figura 4.1, a utilização de uma janela deslizante de tamanho 3 resultará em vetores diferentes para cada seqüência (Tabela 4.3).

Tabela 4.3: Vetores resultantes da codificação das seqüências da figura 4.1 com janela deslizante de tamanho 3.

	ABA	BAA	AAA	AAC	ACA	AAB	BAC	CAA
Seq - 1	1	1	1	1	1	0	0	0
Seq - 2	1	1	0	1	1	1	0	0
Seq - 3	1	0	1	0	1	1	1	0
Seq - 4	1	0	0	0	1	1	1	1
Seq - 5	1	1	0	1	1	0	0	1
Seq - 6	1	0	1	0	1	0	1	1
Seq - 7	1	0	0	0	1	1	1	1

Porém, com o aumento do tamanho da janela deslizante, a similaridade entre subseqüências menores que o tamanho da janela são ignoradas, conseqüentemente, pequenas regiões de similaridade não são avaliadas. Esta não avaliação de subseqüências pode ser mostrada considerando as três seqüências hipotéticas da Figura 4.3.

Figura 4.3: Similaridade desconsiderada entre subseqüências.

Numa janela deslizante de tamanho 3, a distância entre os vetores resultantes da codificação será a mesma, embora as seqüências ACE e ACH tenham claramente um maior grau de similaridade devido à subseqüência AC.

Na clusterização das seqüências do COG, a janela deslizante de tamanho 3 obteve os melhores resultados, enquanto que na clusterização das seqüências obtidas pelo protein-BLAST, os melhores resultados foram obtidos com janela de tamanho 2. Isto ocorre porque cada classe funcional do COG é formada por grupos de seqüências de aminoácidos que possuem a mesma função, porém as proteínas contidas em um COG não são rigorosamente similares entre si. Já as seqüências obtidas pelo protein-BLAST, possuem entre si, um grau de similaridade muito elevado.

Analisando a Tabela 4.3, percebe-se que quanto maior o tamanho da janela deslizante, menor a probabilidade de ocorrer o problema de ambigüidade. Entretanto, a melhoria dos resultados ocorre até um certo tamanho da janela deslizante. Quando o tamanho das janelas deslizantes aumentam, a quantidade de ocorrências de um mesmo segmento diminui. Por exemplo, o SCSW com um alfabeto de 20 caracteres e uma janela de tamanho 4, pode criar um vetor com 160.000 segmentos diferentes. A probabilidade de uma seqüência de aminoácidos possuir mais de uma quantidade de um mesmo segmento, é muito baixa. Até a probabilidade de seqüências de grupos diferentes possuírem segmentos em comum, é pequena. A Tabela 4.4 mostra três seqüências que não possuem segmentos de tamanho 4 em comum. A Tabela 4.5 mostra os seguimentos e a quantidade destes na seqüência.

Tabela 4.4: Seqüências que não possuem segmentos em comum, quando o tamanho da janela deslizante é 4.

Seq - X	A C B C C B A B C A
Seq - Y	B A C A B B C A B C
Seq - Z	C A A B C B A C B A

Tabela 4.5: Vetores resultantes da codificação das seqüências da tabela 4.4 com janela deslizante de tamanho 4.

Seq - X	**Segmentos**	ABCA	ACBC	BABC	BCCB	CBAB	CBCC	CCBA
	Quantidade	1	1	1	1	1	1	1
Seq - Y	**Segmentos**	ABBC	ACAB	BACA	BBCA	BCAB	CABB	CABC
	Quantidade	1	1	1	1	1	1	1
Seq - Z	**Segmentos**	AABC	ABCB	ACBA	BACB	BCBA	CAAB	CBAC
	Quantidade	1	1	1	1	1	1	1

Se numa situação hipotética, os vetores resultantes das seqüências X e Y forem considerados centros de agrupamento, e a seqüência Z, uma seqüência a ser clusterizada, a distância euclidiana (Equação 2.1) entre as seqüências Z-X e Z-Y seria igual, mesmo tratando-se de centros diferentes. Isto interferiu negativamente nos resultados, pois no Software de Clustering, se as distâncias entre uma seqüência e os centros de agrupamento forem iguais, a seqüência será agrupada pelo primeiro centro utilizado na comparação, sem a certeza que este grupo realmente é o mais indicado para tal seqüência.

Ainda pode existir uma outra situação caso a seqüência X não possuísse último caractere. Nesta circunstância, a seqüência Z seria agrupada pelo centro X, uma vez que o vetor resultante da seqüência X teria um segmento a menos, e, simplesmente por isso, teria um menor valor na distância euclidiana. Novamente, a similaridade entre as seqüências e os centros de agrupamento não seria levada em consideração .

Apesar do aumento da janela deslizante minimizar o efeito da ambigüidade, percebe-se que a probabilidade de ocorrer os problemas mencionados é maior, assim como pode visto nos resultados do Software de Clustering (Tabelas 4.1 e 4.2).

5 CONCLUSÕES E PROPOSTAS FUTURAS

5.1 Conclusões

Neste trabalho foi analisado a aplicação do método de codificação de seqüências *Sequence Coding by Sliding Window (SCSW)* na codificação de seqüências de aminoácidos completas, de tamanhos diferentes, em vetores de mesma dimensão. Para realizar a avaliação deste método, foi utilizado o algoritmo de clusterização K-Means, a fim de verificar a eficácia do método SCSW na determinação de similaridade entre seqüências.

Até um certo tamanho da janela deslizante, o método de codificação de seqüências Sequence Coding by Sliding Window mostrou-se apropriado para representar seqüências completas em vetores de mesma dimensão. Entretanto, algumas melhorias devem ser realizadas para que o SCSW possua a mesma acurácia dos métodos tradicionais de alinhamento par-a-par, FASTA (PEARSON, 1990) e BLAST (ALTSCHUL et al, 1990), como destacado em (WU et al, 1992).

Para aumentar a acurácia do método SCSW, a similaridade entre aminoácidos também deve ser levada em consideração, pois, proteínas com a mesma função não necessariamente possuem a mesma seqüência de aminoácidos, podem ter aminoácidos similares em posições específicas que caracterizam os domínios da seqüência.

O tamanho das janelas deslizantes pode ser determinante na eficácia do método SCSW, portanto, deve-se evitar tamanhos de janelas deslizantes que gerem ambigüidades entre as seqüências codificadas, e, evitar janelas que não consideram as pequenas regiões de similaridade.

Apesar de ser necessário solucionar ou minimizar os problemas encontrados no método SCSW, os resultados mostraram que esse método pode ser útil na determinação de similaridade entre seqüências, como mostrado também em Petrilli (1993) e Blaisdell (1986).

5.2 Propostas Futuras

Sugere-se como propostas para continuação deste trabalho, investir nos seguintes problemas relacionados ao tema:

- Criar um esquema de pontuação conforme o tamanho da janelas deslizantes;
- Após a codificação, verificar a existência de seqüências ambíguas;
- Com o auxílio de uma matriz de uma substituição, levar em consideração não só a identidade, mas também a similaridade entre os aminoácidos;
- Aplicar a codificação SCSW na classificação de proteínas com Redes Neurais Artificiais.

REFERÊNCIAS BIBLIOGRÁFICAS

ALTSCHUL, S. F.; GISH, W.; MILLER, W.; MYERS, E.W.; LIPMAN, D.J.. Basic local alignment search tool. **Journal of Computional Biology**, v. 3, n. 215, p. 403-410, 1990.

BLAISDELL, B.E.. A measure of the similarity of sets of sequences not requiring sequence alignment. **National Academy of Sciences**, USA, v. 83, n. 14, p. 5155–5159, Julho 1986.

EIDHAMMER, I.; JONASSEN, I.; AND TAYLOR, W. R. . **Protein Bioinformatics: An Algorithmic Approach to Sequence and Structure Analysis**. . USA: John Willey, 2004. 376 p.

FUCHS, R.. From sequence to biology: the impact on bioinformatics. **Bioinformatics**, v. 18, n. 4, p. 505-506, Abril 2002.

HERING, J. A.; INNOCENT P. R.; HARIS, P. I.. Neuro-fuzzy structural classification of proteins for improved protein secondary structure prediction. **PROTEOMICS**, v. 3, n. 8, p. 1464-1475, 2002.

HIDE, W.; BURKE, J.; DAVISON, D. B.. Biological evaluation of d2, an algorithm for high-performance sequence comparison. **Journal of Computional Biology**, v. 1, n. 3, p. 199–215, 1994.

JUNG, C. F.. **Metodologia Para Pesquisa & Desenvolvimento**. 1ª ed.. Rio de Janeiro: Axcel Books do Brasil, 2004. 312 p.

KING, R.; KARNATH, A.; CLARE, A.; DEHASPE, L. . Genome scale prediction of proteinfunctional class from sequence using data mining. In: **Proceedings of The Sixth ACMSIGKDD International Conference on Knowledge Discovery and Data Mining**. Boston, USA: 2000. 384-389.

KORF, I.; YANDELL, M.; BEDELL, J. . **Blast**. 1ª ed.. : O'Reilly, 2003. 339 p.

MACQUEEN, J. B.. Some Methods for classification and Analysis of Multivariate Observations. **Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability**, California, USA, v. 1, p. 281-297, 1966.

MCGARRAH, D. B.; JUDSON, R. S. . An analysis of the genetic algorithm method of molecular conformation determination. **Journal of Computer Chemistry**, v. 14, n. 11, p. 1385-1395, 1993.

NEEDLEMAN, S. B.; WUNSCH, C. D. A general method applicable to the search for similarities in the amino acid sequence of two proteins. **Journal of Molecular Biology**, v. 48, n. 3, p. 443-453, 1970.

PEARSON, W. R.. Rapid and sensitive sequence comparison with FASTP and FASTA. **Methods Enzymol**, v. 183, p. 63-98p, 1990.

PETRILLI, P.. Classification of protein sequences by their dipeptide composition. **Bioinformatics**, v. 9, p. 205-209, 1993.

PICCOLBONI, A.; MAURI, G. . Application of Evolutionary Algorithms to Protein Folding Prediction. **Lecture Notes In Computer Science**, v. 1363, p. 123-136, 1997.

RODRIGUES, T. S.. **Codificação de Seqüências de Aminoácidos e sua Aplicação na Classificação de Proteínas com Redes Neurais Artificiais**. 2007. 127 p. (Doutorado em Bioinformática) - Universidade Federal de Minas Gerais, Belo Horizonte.

RODRIGUES, T. S.; BRAGA, A. P.; PACÍFICO, L. G.; TEIXEIRA, S. M. R.; OLIVEIRA,S. C.. Clustering and artificial neural networks: Classification of variable lengths of helminth antigens in set of domains. **Genetics and Molecular Biology**, v. 27, n. 4, p. 673–678, 2004.

RODRIGUES, T. S.; BRAGA, A. P.; TEIXEIRA, S. M. R.; OLIVEIRA, S. C.. Protein classification with extended sequence coding by sliding window. In: **IX Annual International Conference on Research in Computacional Molecular Biology**. Boston, USA: 2005. .

RODRIGUES, T. S.; PACÍFICO, L. G.; TEIXEIRA, S. M. R.;OLIVEIRA, S. C.; BRAGA, A. P.. Amino Acid Coding with SlidingWindow Technique. In: **Segundo Workshop Brasileiro de Bioinformática**. Rio de Janeiro: 2003. .

RUST, A. G.; MONGIN, E.; BIRNEY, E. . Genome annotation techniques: new approaches and challenges. **Drug Discovery Today**, v. 7, n. 11, p. 70-76, 2002.

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

TATUSOV, R. L.; GALPERIN, M. Y.; NATALE, D. A.; KOONIN, E. V.. The COG database: a tool for genome-scale analysis of protein functions and evolution. **Nucleic Acids Res.**, v. 28, n. 1, p. 33-36, 2000.

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

VOET, D.. **Fundamentos de Bioquímica.**. 1ª ed.. Porto Alegre: Editora Artes Médicas Sul, 2000. 931 p.

WEINERT, W. R.; LOPES, H. S. . Neural networks for protein classification. **Applied Bioinformatics**, New Zealand, v. 3, n. 1, p. 41-48, 2004.

WU, C.; WHITSON, G.; MCLARTY, J.; ERMONGKONCHAI, A.; CHANG, T.. Protein classification artificial neural system. **Protein Science**, v. 1, n. , p. 667–677, 1992.

WU, T. J.; BURKE, J. P.; DAVISON, D. B.. A measure of dna sequence dissimilarity based on mahalanobis distance between frequencies of words. **Biometrics**, v. 53, p. 1431-1439, 1997.