

DETECÇÃO DE HOMOLOGIAS DISTANTES UTILIZANDO HMMs
E INFORMAÇÕES ESTRUTURAIS

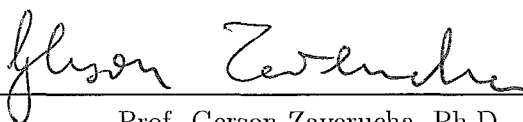
Juliana Silva Bernardes

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO
DOS PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA
UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS
REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE
EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Aprovada por:



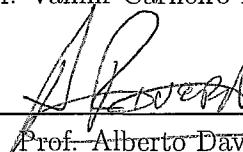
Prof. Vitor Manuel de Moraes Santos Costa, Ph.D.



Prof. Gerson Zaverucha, Ph.D.



Prof. Valmir Carneiro Barbosa, Ph.D.



Prof. Alberto Davila, D. Sc.

RIO DE JANEIRO, RJ - BRASIL

DEZEMBRO DE 2005

BERNARDES, JULIANA SILVA

Detecção de Homologias Distantes Utilizando HMMs e Informações Estruturais [Rio de Janeiro] 2005

XIV, 116p. 29,7 cm (COPPE/UFRJ, M.Sc., Engenharia de Sistemas e Computação, 2005)

Dissertação - Universidade Federal do Rio de Janeiro, COPPE

1. Aprendizado de Máquina
2. Hidden Markov Models
3. Biologia Computacional
4. Detecção de Homologias

I. COPPE/UFRJ II. Título(série)

Agradecimentos

Primeiramente agradeço a Deus pela vida, por me dar coragem para enfrentar os desafios e pela oportunidade de desenvolver esse trabalho.

Aos meus pais, por todo apoio e compreensão, aos meus irmãos pelo carinho e ao meu sobrinho e seu olhar simples sobre a vida, achando que tudo é uma grande brincadeira. Agradeço especialmente a minha sogra, pelas sessões de relaxamento e por me fazer acreditar que tudo iria dar certo.

Agradeço intensamente a meu esposo, por estar sempre ao meu lado, por compreender meus ataques de *stress* e por sempre me tratar com imensa paciência. Por todo incentivo e amor. Por me fazer feliz e principalmente por sempre acreditar em mim.

A todos os meus amigos e toda a equipe do Dr. Alberto Davila, especialmente ao Glauber, por toda ajuda com os *scripts* do Bioperl, a Kary pelos longas conversas no café, debatendo HMMER e SAM. *kary já no puedo mas*. Agradeço também a Sil pela ajuda com biologia molecular e por ter sido minha revisora.

Gostaria de agradecer aos professores Pedro Cabello, da Fiocruz e ao professor Marco Ferreira, do instituto de matemática da UFRJ, pelos esclarecimentos envolvendo probabilidade e estatística. Ao professor Alberto Davila e ao pesquisador Julian Gough pela idéia inicial do projeto. Principalmente ao professor Alberto

pelo espaço físico cedido na Fiocruz, onde realizei grande parte deste trabalho, e principalmente pela coorientação extra-oficial. Por toda a confiança e atenção direcionadas a mim. Por tudo, muito obrigada.

Aos meu orientador Vítor Costa e ao meu coorientador Gerson Zaverucha por toda atenção e incentivo e principalmente por acreditarem em meu trabalho.

Ao CNPQ pelo apoio financeiro e a todas as pessoas que direta ou indiretamente possibilitaram a realização deste trabalho, MUITO OBRIGADA!

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

DETECÇÃO DE HOMOLOGIAS DISTANTES UTILIZANDO HMMs E INFORMAÇÕES ESTRUTURAIS

Juliana Silva Bernardes

Dezembro/2005

Orientadores: Vítor Santos Costa
Gerson Zaverucha

Programa: Engenharia de Sistemas e Computação

Detecção de homologias distantes, entre seqüências de proteínas, tem se tornado um componente central na análise de dados genômicos. Para detectar homologias distantes, *profiles hidden Markov models* (pHMMs) produzem melhores resultados do que métodos baseados em similaridades de seqüências. Para aumentar a sensibilidade dos métodos voltados à detecção de homólogos distantes, informações sobre a estrutura tridimensional de proteínas tem sido amplamente empregadas.

Inicialmente, comparamos o uso de alinhamento estrutural com alinhamento primário nos sistemas de pHMMs, HMMER e SAM. O uso de alinhamento estrutural produziu resultados significativos, o que nos levou a modificar o algoritmo de atribuição de pesos à seqüências, da fase de treinamento do HMMER. Nossa abordagem constrói um conjunto de pHMMs, atribuindo mais pesos aos aminoácidos, considerando cada uma das seguintes propriedades estruturais: estrutura primária, secundária e terciária, abordadas em trabalhos prévios, e acessibilidade e empacotamento de aminoácidos, usadas pela primeira vez no treinamento de pHMMs. A classificação de novas seqüências combina a classificação dos diferentes pHMMs. Uma das principais vantagens do nosso trabalho é que embora informações estruturais sejam usadas no treinamento de pHMMs, inferências continuam sendo a nível de seqüências primárias. Nosso método foi implementado estendendo o pacote HMMER, e os resultados mostraram melhorias significativas sobre outros métodos comumente usados.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for degree of Master of Science (M.Sc.)

REMOTE HOMOLOGY DETECTION WITH HMMs AND
STRUCTURAL ISSUES

Juliana Silva Bernardes

December/2005

Advisors: Vitor Santos Costa
Gerson Zaverucha

Program: Systems Engineering and Computer Science

The detection of remote homologies between protein sequences has become a central problem in genome analysis. Profile hidden Markov Models (pHMMs) are probabilistic models that have been widely used in tackling this problem. pHMMs construct models of protein families based on sequence information. Recent work has shown that remote homology detection can be further improved by considering their three-dimensional structure.

Initially, we compared the use of structural alignments versus the use of primary alignments to train the two pHMMs system, HMMER and SAM. We show that the use of structural alignments can produce significantly better results. Next, we modify the sequence weighting algorithm in the HMMER training phase to consider structural information. Our approach builds different pHMMs, and each pHMM weights is based on structural properties. We consider primary, secondary, and tertiary structure, as used in previous methods. Further, we used solvent accessibility and residue packing properties, that have not been used before to train pHMMs. The classification of a new sequence combines the classification from the several pHMMs. The main advantage of our method is that structural information is only used to train the pHMMs, search is still performed using sequence data. Our method has been implemented by extending the HMMER package, and showed a significant improvement over other commonly used methods.

SUMÁRIO

Agradecimentos	iii
Lista de Figuras	x
Lista de Tabelas	xiii
1 Introdução	1
1.1 Motivação	1
1.2 Objetivos	6
1.3 Organização do Trabalho	7
2 Conceitos Básicos	9
2.1 Biologia Molecular	9
2.1.1 As Células	10
2.1.2 DNA, RNA e Proteínas	10
2.2 Evolução	16
2.2.1 Princípios da Evolução	17
2.2.2 Homologia e Similaridade	18
2.2.3 Homologia Remota ou Distante	21
2.3 Proteínas e suas Estruturas	21
2.3.1 Estrutura Primária	23
2.3.2 Estrutura Secundária	25
2.3.3 Estrutura Terciária, Motivos e Domínios	28
2.3.4 Interações Protéicas, Estrutura Quaternária	29
3 Hidden Markov Models Aplicados à Detecção de Homologias	30
3.1 Cadeias de Markov e <i>Hidden Markov Models</i>	31

3.2	<i>Profile Hidden Markov Models</i>	37
3.3	Arquitetura de <i>Profiles</i> HMMs	37
3.4	Aprendizado da Arquitetura e dos Parâmetros	42
3.4.1	Aprendendo a Arquitetura e os Parâmetros Separadamente	42
3.4.2	<i>Maximum a Posteriori Construction</i>	44
3.5	Priors e Informações Evolucionárias	46
3.5.1	Matrizes de Substituição	47
3.5.2	Pseudo Contadores	50
3.5.3	Misturas de Dirichlet	51
3.6	Associando Pesos às Seqüências	53
3.6.1	Método da Simple Pontuação	53
3.6.2	Método de Gerstein, Sonnhammer e Chothia	54
3.6.3	Voronoi	55
3.6.4	Máxima Entropia	56
3.7	Buscando com <i>Profile</i> HMM	57
3.7.1	Verossimilhança	57
3.7.2	Algoritmo Forward	58
3.7.3	Algoritmo Viterbi	59
3.7.4	Calculando <i>E-values</i>	59
3.8	Análise dos Resultados	61
3.8.1	Curvas ROC	61
3.8.2	Curvas <i>Precision</i> e <i>Recall</i>	62
4	Avaliação de <i>Profiles</i> HMMs	64
4.1	<i>Profiles</i> HMMs, HMMER e SAM	65
4.1.1	HMMER	65
4.1.2	SAM	66
4.2	Classificação Estrutural de Proteínas (SCOP)	67
4.3	Alinhamentos Múltiplos	69
4.4	Metodologia Experimental	70
4.5	Testes e Resultados	72
5	Adicionando Informações Estruturais à <i>Profiles</i> HMM	82
5.1	Modificação do Algoritmo de Atribuição de Pesos às Seqüências	84
5.1.1	Metodologia para Atribuição de Pesos Estruturais	84
5.1.2	Estruturas Secundárias	86
5.1.3	Acessibilidade dos Aminoácidos	87
5.1.4	Empacotamento dos Aminoácidos	88

5.1.5	Estruturas Terciárias	89
5.2	Biblioteca de Modelos	90
5.3	Resultados	91
6	Conclusão	96
6.1	Contribuições	96
6.2	Trabalhos Futuros	98
	Referências Bibliográficas	100

LISTA DE FIGURAS

1.1	Problema alvo.	2
1.2	Solução dos métodos baseados em similaridade.	3
1.3	Solução dos métodos baseados em pHMM.	4
1.4	Inclusão de informações estruturais no aprendizado de pHMMs.	5
2.1	Esquema de um cromossomo.	11
2.2	Esquema de um nucleotídeo.	12
2.3	Correspondência entre códons e aminoácidos.	13
2.4	Relação entre DNA, gene e proteína.	14
2.5	Esquema RNA transportador.	15
2.6	Síntese protéica.	16
2.7	Genes homólogos.	18
2.8	Alinhamento entre duas seqüências.	20
2.9	Fases da estrutura das proteínas.	23
2.10	Fórmula geral de um aminoácido.	24
2.11	Estrutura primária.	24
2.12	Classificação dos aminoácidos	25
2.13	Estrutura secundária regular alpha-hélice.	26
2.14	Estrutura secundária regular folhas-Betas.	27
2.15	Estrutura quaternária.	29
3.1	Fases de um pHMM	31
3.2	Cadeias de Markov para reconhecimento de seqüência de DNA	32
3.3	Esquema de um HMM.	34
3.4	HMM representado como um modelo temporal probabilístico.	35
3.5	Cadeias de Markov de primeira e segunda ordem	35
3.6	Representação Rabiner e Russel	36

3.7	Alinhamento parcial de proteínas da família das globinas.	38
3.8	Arquitetura básica para pHMMs	39
3.9	Arquitetura para HMMs incluindo estados de <i>insert</i>	40
3.10	Arquitetura completa para pHMMs	41
3.11	Exemplo de aprendizado da arquitetura de pHMMs	43
3.12	Exemplo do algoritmo MAP.	46
3.13	Árvore filogenética	54
3.14	Exemplo de curvas ROC.	62
3.15	Exemplo de curvas <i>Precision</i> e <i>Recall</i>	63
4.1	Proteínas da mesma super-família que apresentam menos de 16% de identidade.	70
4.2	Divisão da base de dados e metodologia experimental.	71
4.3	Qualidade dos alinhamentos.	73
4.4	Análise de desempenho para o HMMER, através de curvas ROC, considerando todas as ferramentas de alinhamento.	74
4.5	Análise de desempenho para o HMMER, através de <i>Precision</i> e <i>Recall</i> , considerando todas as ferramentas de alinhamento.	74
4.6	Análise de desempenho para o SAM, através de curvas ROC, considerando todas as ferramentas de alinhamento.	75
4.7	Análise de desempenho para o SAM, através de <i>Precision</i> e <i>Recall</i> , considerando todas as ferramentas de alinhamento.	75
4.8	Análise de desempenho para o HMMER e SAM, através de curvas ROC, considerando apenas MAMMOTH e CLUSTALW.	77
4.9	Análise de desempenho para o HMMER e SAM, através de curvas <i>precision</i> e <i>recall</i> , considerando apenas MAMMOTH e CLUSTALW.	77
4.10	Análise de desempenho para o HMMER e SAM, através de curvas ROC, considerando as modificações propostas por (WISTRAND, SONNHAMMER, 2005) e os alinhamentos produzidos por MAMMOTH.	79
4.11	Análise de desempenho para o HMMER e SAM, através de curvas <i>precision</i> e <i>recall</i> , considerando as modificações propostas por (WISTRAND, SONNHAMMER, 2005) e os alinhamentos produzidos por MAMMOTH.	79
4.12	Análise de desempenho para o HMMER e SAM, através de curvas ROC, considerando as modificações propostas por (WISTRAND, SONNHAMMER, 2005) e os alinhamentos produzidos por CLUSTALW.	80

4.13	Análise de desempenho para o HMMER e SAM, através de curvas <i>precision</i> e <i>recall</i> , considerando as modificações propostas por (WISTRAND, SONNHAMMER, 2005) e os alinhamentos produzidos por CLUSTALW.	80
5.1	Esquema HMMER-STRUCT.	83
5.2	Matriz de pesos a partir de informações de estrutura secundária.	87
5.3	Valores de <i>Ooi number</i> para a proteína Dehaloperoxidase.	88
5.4	Comparação entre os modelos do HMMER-STRUCT, através de curvas ROC, considerando os alinhamentos produzidos por MAMMOTH.	91
5.5	Comparação entre os modelos do HMMER-STRUCT, através de curvas <i>precision</i> e <i>recall</i> , considerando os alinhamentos produzidos por MAMMOTH.	92
5.6	Avaliação do desempenho do HMMER-STRUCT, através de curvas ROC, variando o número de classificadores e considerando os alinhamentos produzidos por MAMMOTH.	93
5.7	Avaliação do desempenho do HMMER-STRUCT, através de curvas <i>precision</i> e <i>recall</i> , variando o número de classificadores e considerando os alinhamentos produzidos por MAMMOTH.	94
5.8	Comparação entre HMMER, SAM e HMMER-STRUCT, através de curvas ROC, considerando os alinhamentos produzidos por MAMMOTH.	94
5.9	Comparação entre HMMER, SAM e HMMER-STRUCT, através de curvas <i>precision</i> e <i>recall</i> , considerando os alinhamentos produzidos por MAMMOTH.	95

LISTA DE TABELAS

3.1	Esquema de Voronoi.	56
4.1	Experimentos realizados e nossos experimentos.	65
4.2	Lista de super-famílias.	71
4.3	Resultado do <i>paired t-test</i> e significância estatística entre os testes realizados para o pacote HMMER, considerando todas as ferramentas de alinhamento.	74
4.4	Resultado do <i>paired t-test</i> e significância estatística entre os testes realizados para o pacote SAM, considerando todas as ferramentas de alinhamento.	76
4.5	Resultado do <i>paired t-test</i> e significância estatística entre os testes realizados para o pacote HMMER e SAM, considerando os alinhamentos produzidos por MAMMOTH e CLUSTALW.	77
4.6	Resultado do <i>paired t-test</i> e significância estatística entre os testes realizados para o pacote HMMER e SAM, considerando as modificações propostas por (WISTRAND, SONNHAMMER, 2005) e os alinhamentos produzidos por MAMMOTH.	80
4.7	Resultado do <i>paired t-test</i> e significância estatística entre os testes realizados para o pacote HMMER e SAM, considerando as modificações propostas por (WISTRAND, SONNHAMMER, 2005) e os alinhamentos produzidos por CLUSTALW.	81
5.1	Resultado do <i>paired t-test</i> e significância estatística dos testes realizados entre os modelos do HMMER-STRUCT, considerando os alinhamentos produzidos por MAMMOTH.	92

5.2	Resultado do <i>paired t-test</i> e significância estatística dos testes realizados entre HMMER, SAM e HMMER-STRUCT-3, considerando os alinhamentos produzidos por MAMMOTH.	95
-----	---	----

CAPÍTULO 1

Introdução

1.1 Motivação

As pesquisas genéticas da última década, motivaram os projetos genoma. Esses projetos são destinados a explorar e elucidar a biologia dos mais diversos organismos. Com o aprimoramento das tecnologias envolvidas nos processos de extração das informações genéticas (DNA e RNA), tornou-se possível a obtenção dessas informações em tempos cada vez mais curtos.

Entretanto todas as informações genéticas geradas nos projetos genoma necessitam de uma cuidadosa análise. Dessa forma, tornou-se necessário o desenvolvimento de ferramentas computacionais com o objetivo de auxiliar a interpretação dos dados genômicos. A *anotação* de seqüências genômicas (genes e proteínas) é o processo de identificação de suas funções. *Seqüências homólogas* são seqüências que compartilham o mesmo ancestral, e provavelmente compartilham a mesma função. Essas seqüências podem ser classificadas dentro de uma mesma família. A medida que os dados genômicos são processados, eles são anotados e depositados em bancos de dados públicos, tais como GENBANK (DENNIS, LIPMAN, et al., 2004), TREMBL (BOECKMANN, BAIROCH, et al., 2003) e SWISS-PROT (BAIROCH, BOECKMANN, et al., 2005). O processo de anotação de novas seqüências é auxiliado pela comparação dessas seqüências com bancos de dados público. O objetivo dessa comparação é descobrir relacionamentos evolutivos (detectando homologia), e com isso

diminuir a quantidade de testes laboratoriais necessários para concluir a anotação. A figura 1.1 ilustra o problema, dado um banco de dados que contem proteínas da família das hemoglobinas, deseja-se saber quais as chances de uma nova proteína pertencer a esta família.

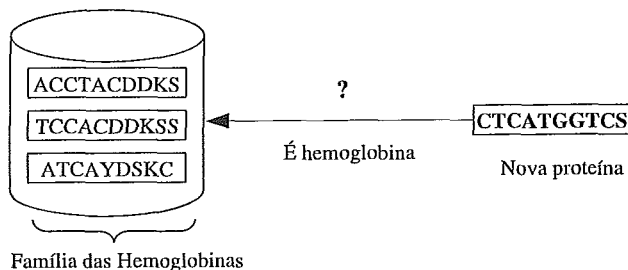


Figura 1.1: Problema alvo.

A comparação por similaridades tem sido o método mais utilizado com o propósito de detectar homologias. Esses algoritmos baseiam-se principalmente em técnicas de programação dinâmica, como descrito por Needleman Wunsch (NEEDLEMAN, WUNSCH, 1970) e Smith Waterman (SMITH, WATERMAN, 1981). Ferramentas populares, como BLAST (ALTSCHUL, GISH, et al., 1990) e FASTA (PEARSON, 1985) são implementações desses algoritmos. Um exemplo da solução desses métodos é mostrado na figura 1.2. O primeiro procedimento é alinhar a nova proteína com cada membro da família das hemoglobinas. A cada alinhamento i é associado um valor de significância S_i , que expressa o quanto as seqüências alinhadas são similares. Para determinar as chances da nova proteína pertencer a família das hemoglobinas, é necessário saber se $S_1 > T$ ou $S_2 > T$ ou $S_3 > T$, onde T é um limite pré-estabelecido. Esses métodos funcionam bem, quando a similaridade, entre as seqüências homólogas analisadas, é alta ou média. O mesmo não ocorre quando a similaridade é baixa, e geralmente essas seqüências são classificadas como homólogas distantes.

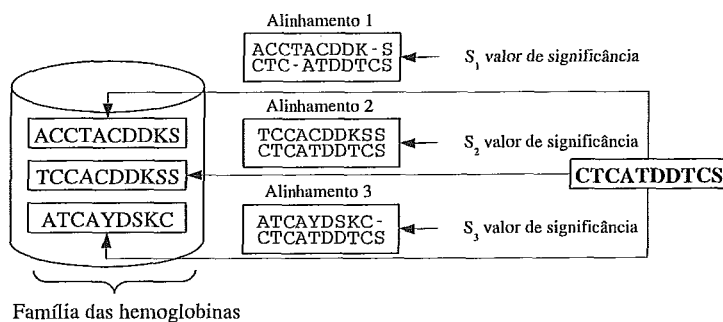


Figura 1.2: Solução dos métodos baseados em similaridade.

A maioria dos avanços, no sentido de detectar homólogos distantes, envolve técnicas que fazem uso de um conjunto de seqüências relacionadas. Desse conjunto são extraídos padrões, que são comparados com novas seqüências, na tentativa de encontrar algo que as identifique como um potencial membro do conjunto.

Alguns métodos na tentativa de aumentar a sensibilidade expandem métodos convencionais empregando técnicas como, matrizes de substituição e modelos de penalidades (HUANG, ZHANG, 1996; GERSTEIN, LEVITT, 1996; HUANG, MILLER, 1991). Porém, a maioria dos métodos, tais como PROSITE (HULO, SIGRIST, et al., 2004), PSI-BLAST (ALTSCHUL, MADDEN, et al., 2000) e métodos baseados em *hidden Markov models* (HMMs), como HMMER (EDDY, 1998), SAM (HUGHEY, KROGH, 1996a) e THMM (QIAN, GOLDSTEIN, 2004) criam modelos, a partir de um conjunto de homólogos conhecidos e avaliam o quanto outras seqüências se adaptam ao modelo. Particularmente, os métodos baseados em HMMs utilizam um tipo especial de modelo probabilístico, denominado *profile hidden Markov models* (pHMMs), e tem mostrado eficiência na detecção de homologias distantes (WISTRAND, SONNHAMMER, 2005; MADERA, GOUGH, 2002; GOUGH, KARPLUS, et al., 2001; KARPLUS, BARRET, et al., 1998; PARK, KARPLUS, et al., 1998). Um exemplo da solução adotada por pHMMs é mostrado na figura 1.3, a partir de um alinhamento múltiplo das proteínas da família das hemoglobinas, é construído um modelo probabilístico, que passa por uma etapa de aprendizado e pode ser usado para realizar inferência, ou seja, identificar novos membros da família das hemoglobinas.

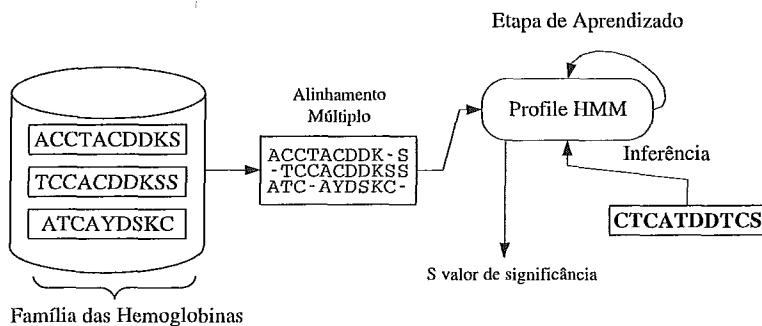


Figura 1.3: Solução dos métodos baseados em pHMM.

Todavia, os principais programas de detecção de homologies distantes, HMMER e SAM constroem modelos probabilísticos extraíndo apenas informações dos alinhamentos múltiplos de seqüências. No entanto, no universo das proteínas a estrutura tridimensional tende a ser mais conservada do que as seqüências de aminoácidos. Sendo assim, modelos que fazem uso apenas das informações contidas nas seqüências sofrem uma perda de sensibilidade na detecção de homologies distantes.

Alguns trabalhos recentes (ESPADALER, ARAGUES, et al., 2005; WANG, SAMUDRALA, 2005; HOU, HSU, et al., 2004a; CHAKRABARTI, SOWDHAMINI, 2004; ALEXANDROV, GERSTEIN, 2004) têm mostrado que a similaridade estrutural, entre proteínas, pode melhorar a sensibilidade do processo de detecção de homologies distantes. Embora duas proteínas relacionadas possam ter sofrido mutações, a ponto de uma completa alteração a nível de aminoácidos, as estruturas tridimensionais dessas proteínas podem reter uma estrutura mínima, que sugere a existência de uma origem comum entre as proteínas. Através de alinhamentos estruturais ou tridimensionais é possível determinar essa estrutura mínima. Esses alinhamentos determinam a sobreposição dos átomos de duas ou mais proteínas. A principal motivação deste trabalho é, a partir de alinhamentos estruturais, incluir informações sobre as estruturas das proteínas no aprendizado de pHMMs, como mostra a figura 1.4.

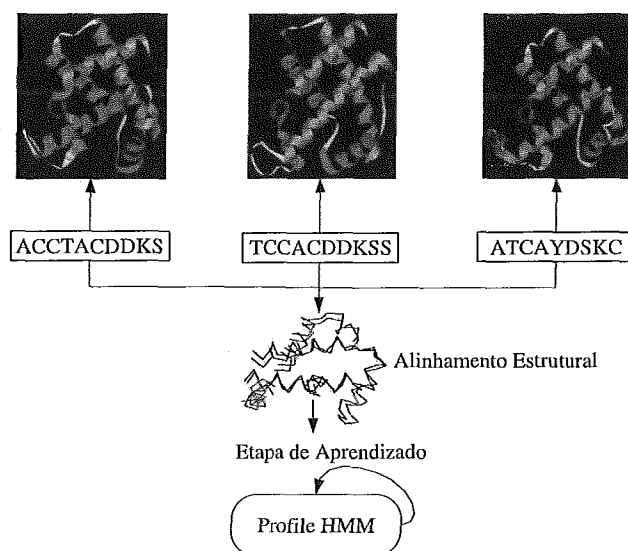


Figura 1.4: Inclusão de informações estruturais no aprendizado de pHMMs.

Entretanto, o número de estruturas de proteínas resolvidas e depositadas em bases de dados públicas, tais como *Protein Data Bank* (PDB) (HELEN, WESTBROOK, et al., 2000), é muito inferior ao número de proteínas anotadas. Isso porque as estruturas tridimensionais são determinadas precisamente, apenas por processos laboratoriais que demandam tempo e recursos. Porém, muitos avanços estão sendo obtidos na tentativa de prever estruturas de proteínas. Os métodos que apresentam os melhores resultados são baseados na previsão por homologia (TAKEDASHITAKA, TAKAYA, et al., 2004; PETREY, XIANG, et al., 2003; YANG, WANG, 2003; LAZIKANI, JUNG, et al., 2001).

Além de bancos de dados formados puramente por seqüências genômicas é crescente o número de bancos que representam famílias e domínios de proteínas, tais como PIRSF (CATHY, WU, et al., 2004), SMART (LETUNIC, COPLEY, et al., 2004), PRINTS (MOULTON, NORDLE, et al., 2003) e PROCLASS (HONGZHAN, WINONA, et al., 2003). Essas bases de dados prometem uma maior precisão, no processo de anotação, pois são baseadas em perfis que agrupam características relevantes sobre um conjunto de seqüências homólogas. Muitas dessas bases de dados são compostas por modelos produzidos pelos pacotes HMMER (BATEMAN, COIN, et al., 2004; HAFT, SELENGUT, et al., 2003) e SAM (MADERA, VOGEL, et al.,

2004). Dessa forma, o aprimoramento das técnicas para detecção de homologias, baseadas em pHMMs, implicará diretamente no desempenho das bases de dados que utilizam essa técnica.

1.2 Objetivos

O trabalho (JONES, BATEMAN, 2002) realizou experimentos demonstrando que o uso de alinhamentos estruturais não aumentavam a detecção de homólogos distantes. Um dos objetivos desse trabalho é investigar o impacto do uso dos alinhamentos estruturais no desempenho de pHMMs, tendo como motivação o surgimento de novas ferramentas de alinhamento estrutural e o fato de que o trabalho citado não aplicou os testes aos dois principais programas que implementam pHMMs, HMMER e SAM. Além disso, nós introduzimos uma nova metodologia para a realização dos experimentos, que utiliza validação cruzada (HAYKIN, 2001) e mede a significância dos resultados através de *paired t-test* (MITCHELL, 1997).

É notável o crescimento de trabalhos baseados em pHMMs, que utilizam propriedades estruturais, na tentativa de aumentar a sensibilidade de métodos voltados à detecção de homólogos distantes, (ALEXANDROV, GERSTEIN, 2004; GOYON, TUFFÉRY, 2004; BYSTROFF, BAKER, 2000). No entanto, esses métodos utilizam apenas informações contidas nas coordenadas tridimensionais das proteínas, baseando-se no fato de que proteínas homólogas possuem estrutura tridimensional similar. Porém, o estudo (CHAKRABARTI, SOWDHAMIMI, 2004) demonstrou que outras propriedades estruturais também devem ser usadas na análise de homologias distantes.

Nesse contexto, nós desenvolvemos um novo método para treinar pHMMs considerando alinhamento tridimensionais e diferentes propriedades estruturais sobre um conjunto de proteínas homólogas. Nosso método combina diferentes pHMMs, um para cada propriedade estrutural. As propriedades tratadas neste trabalho foram estruturas primárias, secundárias e terciárias, acessibilidade e empacotamento de aminoácidos. Trabalhos anteriores já exploraram o uso de informações de estrutu-

ras secundárias e terciárias aplicadas à pHMMs, porém seguindo outra abordagem. No entanto, pHMMs baseados em propriedades como acessibilidade e empacotamento de aminoácidos, estão sendo usados pela primeira vez. Nosso método trabalha aplicando diferentes pesos a cada aminoácido do conjunto de proteínas homólogas alinhadas. O peso é atribuído de acordo com a importância estrutural de cada aminoácido. Essa importância foi baseada em estudos, tais como (CHAKRABARTI, SOWDHAMIMI, 2004; DEANE, PERDERSEN, et al., 2003; NISHIKAWA, OOI, 1986), que determinaram a relevância de cada propriedade estrutural avaliada neste trabalho. O que difere a nossa abordagem das abordagens mencionadas é o fato de que as distribuições de probabilidades de emissão em pHMMs, continuam sendo sobre o alfabeto de aminoácidos, ou seja, outras abordagens utilizam alfabetos especiais para representar elementos estruturais de proteínas. As abordagens que adotam esses alfabetos são obrigadas a trabalhar com previsão de estruturas na fase de busca por novos homólogos, pois a imensa maioria das proteínas anotadas não possui estrutura tridimensional definida. É importante ressaltar que os métodos para previsão de estruturas tridimensionais, ainda apresentam uma taxa de erro elevada (MOULT, FIDELIS, et al., 2003).

1.3 Organização do Trabalho

Os próximos capítulos dessa dissertação estão estruturados da seguinte forma:

- No capítulo 2 são definidos os conceitos biológicos que serão utilizados no decorrer do trabalho. Inicialmente são descritos os conceitos básicos da biologia molecular. Em seguida são abordados alguns princípios da evolução. E finalmente são descritos os princípios que governam a formação das estruturas das proteínas.
- O capítulo 3 também trata de conceitos preliminares, abordando conceitos sobre a técnica HMMs e sobre a arquitetura, aprendizado e inferências de modelos probabilísticos (pHMMs), aplicados ao problema da detecção de homologias distantes.

- O capítulo 4 realiza uma comparação entre HMMER e SAM com objetivo de avaliar o desempenho desses pacotes, considerando alinhamentos primários e alinhamentos estruturais. A base de dados SCOP (ANDREEVA, HOWORTH, et al., 2004) foi adotada como a base de dados utilizada para a realização dos testes. Os alinhamentos estruturais foram providos pelos programas MAMMOTH (ATTWOOD, BRADLEY, et al., 2005) e 3DCOFFEE (SULLIVAN, SUHRE, et al., 2004), e os alinhamentos primários pelos programas CLUSTALW (THOMPSON, GIBSON, 1994a) e TCOFFEE (NOTREDAME, HIGGINS, et al., 2000). O desempenho dos pacotes foi medido através de curvas ROC (FAWCETT, 2004; BECK, SHULTZ, 1986; METZ, 1978) e curvas *Precision* e *Recall* (BILENKO, MOONEY, 2003; CRAVEN, SLATTERY, 2001; BUCKLAND, GEY, 1994).
- O capítulo 5 explica como propriedades estruturais foram incluídas na fase de aprendizado de pHMMs gerados pelo pacote HMMER. O pacote JOY (MIZUGUCHI, DEANE, et al., 1998) foi utilizado para prover informações sobre estruturas secundárias, acessibilidade e empacotamento de aminoácidos. As informações sobre estruturas terciárias foram obtidas a partir do alinhamento estrutural produzido pela ferramenta MAMMOTH. Os testes foram realizados seguindo os mesmos procedimentos apresentados no capítulo 4.
- O capítulo 6 conclui o trabalho, fazendo uma discussão e destacando possíveis trabalhos futuros.

CAPÍTULO 2

Conceitos Básicos

O objetivo deste capítulo é prover conceitos básicos de biologia, que são importante para o entendimento desta dissertação. A homologia entre proteínas, classificadas como homólogas distantes, pode ser detectada através de semelhanças em suas estruturas. Para compreender como propriedades estruturais auxiliam o processo de detecção de homólogos, descrito no capítulo 5 página 82, é essencial entender as leis de formação das estruturas das proteínas. Para prover esse conhecimento teórico as sessões deste capítulo foram divididas da seguinte forma: a sessão 2.1 faz uma breve revisão de conceitos de biologia molecular explicando a origem e formação das moléculas de DNA e como os genes, que são segmentos de moléculas de DNA, produzem proteínas, a sessão 2.2 aborda os princípios básicos da evolução explicando o que é homologia e como ela pode ser detectada, além de explicar o problema relacionado à detecção de homólogos distantes, a sessão 2.3 aborda a importância das proteínas e como suas estruturas são formadas.

2.1 Biologia Molecular

Biologia Molecular é a ciência que estuda a biologia a nível molecular. A *produção* ou *síntese* de proteínas é um dos processos celulares mais importantes, pois as proteínas são indispensáveis a manutenção da vida de todos os organismos. O objetivo desta sessão é explicar como as proteínas são produzidas a partir dos genes. Para isso, a sessão 2.1.1 descreve a organização das células, onde o processo de síntese

protéica ocorre e a sessão 2.1.2 descreve propriamente a produção de proteínas.

2.1.1 As Células

As *células* são as unidades vitais dos organismos vivos. Dentro de cada célula estão presentes *informações genéticas*, que são indispensáveis a manutenção da vida celular. Embora exista uma grande diversidade de espécies, todas compartilham uma estrutura celular mínima e componentes moleculares. O conteúdo interno da célula, chamado de *citoplasma*, é separado do meio extra-celular pela membrana plasmática. O citoplasma é composto por diversas estruturas indispensável à manutenção da vida celular, denominadas *organelas* (ALBERTS, BRAY, et al., 2002). As organelas assumem o papel de órgãos dentro da célula, e cada uma possui uma função específica, por exemplos os *ribossomos* participam da síntese de proteínas.

Todas as células possuem a capacidade de *auto-replicação*, gerando descendentes que guardam uma cópia de seu material genético. Além disso, é no interior das células que ocorrem as principais atividades químicas necessárias a sobrevivência dos organismos, tais como a *produção de energia* e a síntese protéica.

Os seres vivos estão divididos em dois grupos de acordo com sua organização celular: *procariontes* e *eucariontes*. Os eucariontes possuem células bem divididas, com organelas e membrana nuclear. Essa membrana separa o citoplasma do núcleo, onde o material genético reside. Nesse grupo encontram-se todos os membros dos reinos animal, vegetal e os fungos. Por outro lado, os procariontes são organismos unicelulares, não possuem organelas e o material genético se encontra espalhado no interior da célula. Nesse grupo estão presentes todos os tipos de bactérias, inclusive arque-bactérias ou bactérias primitivas e ciano-bactérias ou algas azuis.

2.1.2 DNA, RNA e Proteínas

Todas as células possuem macro-moléculas denominadas *cromossomos* (SCOTT, MATSUDAIRA, et al., 2000). Nos organismos eucariontes essas macro-moléculas estão presente no núcleo da célula, enquanto que nos organismos procariontes são encontrados no citoplasma. Os cromossomos são formados por moléculas de DNA,

ácido desoxirribonucléico. A figura 2.1, mostra uma célula de um organismo eucariote e um cromossomo em destaque, se o cromossomo for desempacotado é possível chegar a molécula de DNA. Segmentos do DNA são denominados *genes*, que são responsáveis pela produção de proteínas. O conjunto de todos os genes de um organismo compreende seu *genoma*.

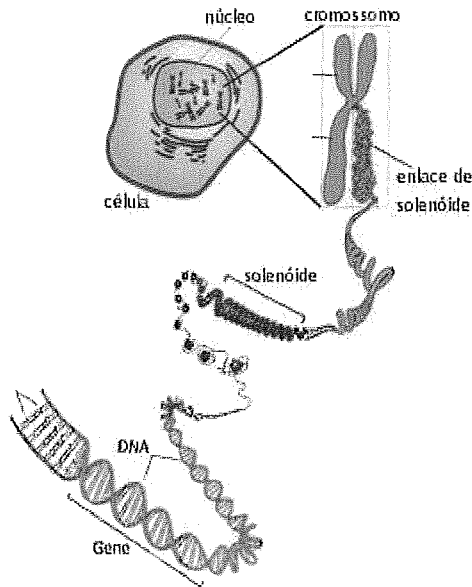


Figura 2.1: Esquema de um cromossomo.

As moléculas de DNA são responsáveis diretamente por sua própria *replicação* ou *duplicação* e pela produção de moléculas de RNA, *ácido ribonucléico*. O DNA produz três tipos de RNA: RNA mensageiro (RNAm), RNA transportador (RNAt) e RNA ribossomal (RNAr) (ALBERTS, BRAY, et al., 2002), e todos eles participam diretamente da síntese de proteínas.

O DNA e RNA são compostos por unidades chamadas *nucleotídeos*. Um nucleotídeo é um composto químico dividido em três partes: um grupo fosfato, uma pentose (molécula de açúcar com cinco carbonos) e uma *base orgânica*, conforme figura 2.2. A base orgânica ou nitrogenada é o que identifica cada um dos nucleotídeos, portanto pode-se falar de nucleotídeo apenas referindo-se as bases orgânicas, já que os outros compostos são constantes em todos os nucleotídeos. As bases Adenina (A),

Citosina (C) e Guanina (G) são encontradas nas moléculas de DNA e RNA. A base Timina (T) é encontrada particularmente nas moléculas de DNA, enquanto a base Uracila (U) ocorre apenas nas moléculas de RNA.

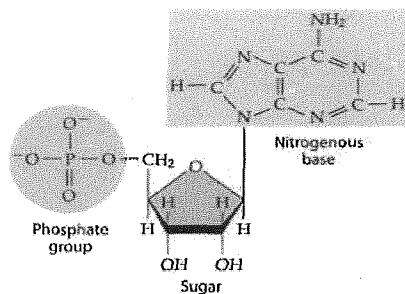


Figura 2.2: Esquema de um nucleotídeo.

Nas moléculas de DNA, os nucleotídeos ligam-se formando duas fitas. A molécula possui uma estrutura helicoidal, conhecida como dupla hélice. Essa estrutura foi descoberta por James Watson e Francis Crick em 1953 e está presente em grande parte dos organismos vivos. As duas fitas unem-se através de *ligações químicas específicas*, entre as bases. A base Adenina liga-se sempre à base Timina e a base Citosina à base Guanina. A base Adenina também pode se ligar a base Uracila, porém essa ligação só ocorre durante a produção de moléculas de RNA. Todas essas regras de ligação entre bases são chamadas de *princípio da complementaridade ou paridade*.

Síntese de Proteínas

Para a produção de proteínas é necessário que a informação contida na molécula de DNA seja transmitida até o citoplasma. Um segmento da molécula de DNA que codifica ou produz uma proteína é denominado *gene*. Para que a informação contida no gene chegue ao citoplasma é necessário que essa informação seja copiada ou *transcrita*. O processo de transcrição é o processo pelo qual a molécula de DNA produz moléculas de RNA. O RNAm é responsável por transcrever as informações contidas no DNA, que serão utilizadas na produção da proteína. Para o processo de transcrição é necessário que esteja presente uma enzima denominada RNA *polimerase*. Nessas condições as pontes de hidrogênio, que mantêm unidas as bases

orgânicas, se rompem fazendo com que as duas fitas de DNA se afastem. Nucleotídeos livres, presentes na célula encaixam-se em apenas uma das fitas, chamada de *fita ativa*. Essa fita destaca-se da molécula de DNA, dando origem a molécula de RNAm (fita única), que migra para o citoplasma. Após esse processo as duas fitas de DNA tornam a parear reconstituindo a molécula original.

Os *ribossomos* são organelas responsáveis por *ler* ou *traduzir* as informações contidas no RNAm. A leitura do RNAm está baseada em triplas de nucleotídeos, chamadas de *códons*, que são usados para identificar os *aminoácidos*. Os aminoácidos são moléculas que formam as proteínas. Combinando os quatro nucleotídeos em triplas são obtidas $4^3 = 64$ combinações. Esse número é bem superior aos 20 aminoácidos existentes, por isso, mais de um códon pode identificar um mesmo aminoácido. A figura 2.3 mostra a relação entre códons e aminoácidos, por exemplo, o códon que possui um *U* na primeira posição, outro *U* na segunda posição e outro *U* na terceira posição, ou seja, o códon *UUU* é responsável por reconhecer o aminoácido fenilalanina (*phe*). A correspondência, mostrada pela figura 2.3, entre códons e aminoácidos e chamada de *código genético*.

1ª posição term. 5'	2ª posição				3ª posição term. 3'
	U	C	A	G	
U	Phe Phe Leu Leu	Ser Ser Ser Ser	Tyr Tyr STOP STOP	Cys Cys STOP Trp	U C A G
C	Leu Leu Leu Leu	Pro Pro Pro Pro	His His Gln Glu	Arg Arg Arg Arg	U C A G
A	Ile Ile Ile Met	Thr Thr Thr Thr	Asn Asn Lys Lys	Ser Ser Arg Arg	U C A G
G	Val Val Val Val	Ala Ala Ala Ala	Asp Asp Glu Glu	Gly Gly Gly Gly	U C A G

Figura 2.3: Correspondência entre códons e aminoácidos.

A síntese de proteínas é iniciada quando o ribossomo, ao ler o RNAm, encontra um códon especial, denominado *códon de início* ou *start codon* e de modo geral é terminada quando é lido um *códon de parada* ou *stop codon*. Na figura 2.3 o códon *AUG* corresponde ao códon de início e os códons *UAA*, *UGA* e *UAG* correspondem aos códons de parada. O segmento do DNA que foi transcrito em RNAm e está entre um códon de início e um códon de parada, corresponde ao gene. Dessa forma, um gene é um conjunto de códons de tamanho arbitrário e de modo geral, o tamanho do gene determinará o tamanho da proteína, por exemplo, de forma bem simples, se um gene possui 30 nucleotídeos (10 códons) a proteína produzida será formada por 10 aminoácidos. A figura 2.4 mostra um trecho de uma seqüência de DNA, destacando os genes *a*, *b*, *c* e *d*. Para o gene *c* a figura mostra o códon de início e fim. As regiões intergênicas não produzem proteínas e são denominadas regiões não codificantes, porém essas regiões possuem *senalizadores* que guiam a síntese de proteínas (COOPER, 2000).

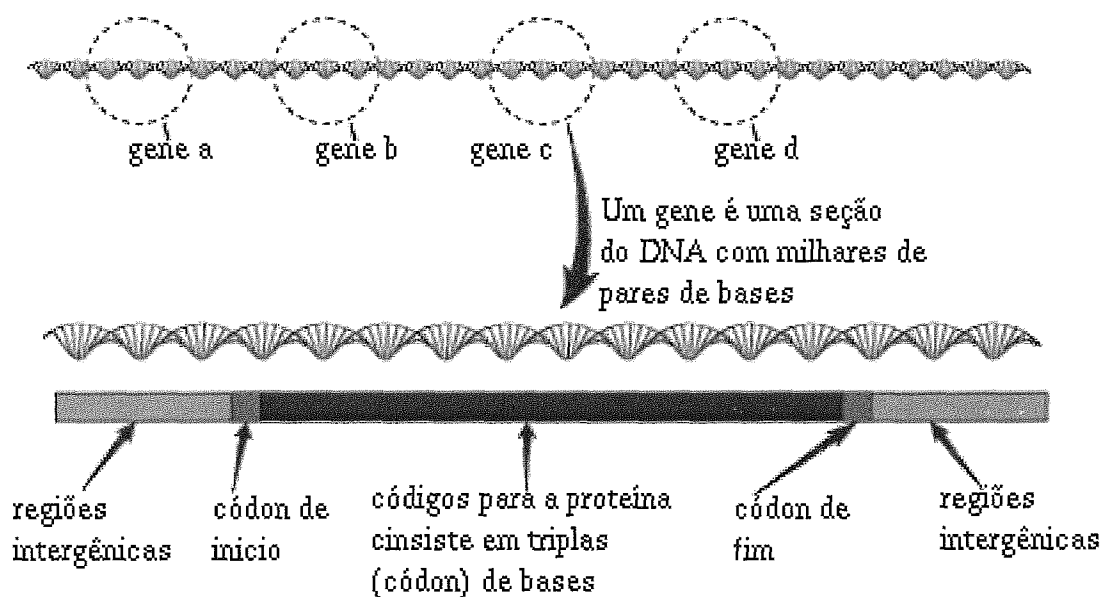


Figura 2.4: Relação entre DNA, gene e proteína.

Após a leitura do códon de início, o ribossomo lê os próximos códons, e para cada códon lido o ribossomo atrai o *anticódon* do RNAt. O anticódon é a seqüência de três bases orgânicas, que é complementar a um determinado códon, por exemplo, se

o códon lido é *UCG* o anticódon será *AGC*, pelo princípio da complementaridade de bases. O RNAt possui duas extremidades, na primeira está presente anticódon e na segunda extremidade encontra-se o aminoácido. A figura 2.5 mostra o esquema de dois RNA transportadores, o primeiro carrega em uma de suas extremidades o aminoácido *SER* e na outra extremidade o anticódon *UCA*. O anticódon *UCA* será ligado ao códon *AGU* do RNAm, liberando o aminoácido *SER*. O segundo RNAt carrega em uma extremidade o aminoácido *TYR* e na outra o anticódon *AUG*. O anticódon *AUG* será ligado ao códon *UAC* do RNAm, liberando o aminoácido *TYR*. Sendo assim, se os códons *AGU* e *UAC* forem lido pelo ribossomo, durante a síntese protéica, a proteína final conterá os aminoácidos *SER* e *TYR*.

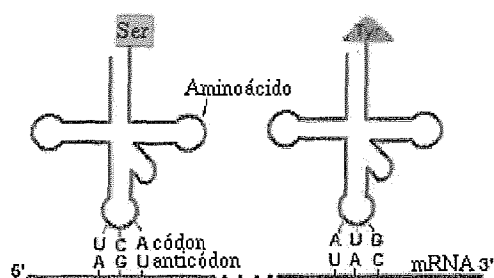


Figura 2.5: Esquema RNA transportador.

A medida que os códons são lidos, o ribossomo atrai o respectivo RNAt, o códon (RNAm) se liga com o anticódon (RNAt) e o aminoácido é liberado. Em seguida, os aminoácidos liberados são ligados formando a molécula de proteína. A figura 2.6 ilustra um trecho da síntese de proteínas, supondo que o códon de início já foi lido pelo ribossomo, sendo assim, o primeiro códon traduzido é *CUG*, o RNAt que contém o anticódon *GAC* e carrega o aminoácido *LEU* é atraído, e o códon liga-se ao anticódon. Em seguida o segundo códon é lido *UUU*, da mesma forma o RNAt, que contém o anticódon *AAA* e carrega o aminoácido *PHE*, é atraído e o códon liga-se ao anticódon, figura 2.6-a. O aminoácido *LEU*, correspondente ao primeiro códon, liga-se ao aminoácido *PHE* que corresponde ao segundo códon lido, nesse momento o aminoácido *LEU* desprende-se do RNAt, figura 2.6-b. A tradução continua com a leitura do terceiro códon, figura 2.6-c, e quarto códon, figura 2.6-d, os aminoácidos vão sendo ligados formando a molécula da proteína, até que o códon de parada seja lido.

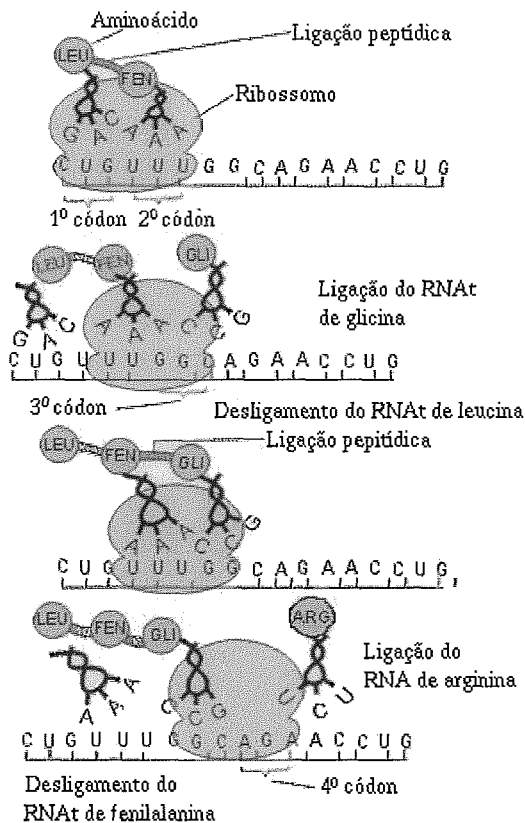


Figura 2.6: Síntese protéica.

2.2 Evolução

Acredita-se que todas as células desenvolveram-se a partir de um único ancestral. Sendo assim, as espécies compartilham semelhanças no genoma. Essas semelhanças traduzem-se em genes e proteínas homólogas. O objetivo desta sessão é explicar o que homologia em termos de evolução e abordar o problema tratado por essa dissertação, ou seja, o que são homólogos distantes. As próximas sessões estão divididas da seguinte forma: a sessão 2.2.1 descreve os princípios básicos da evolução e descreve o que é homologia, a sessão 2.2.2 explica a diferença entre homologia e similaridade, e finalmente a sessão 2.2.3 discute o problema de detecção de homólogos distantes.

2.2.1 Princípios da Evolução

Embora os organismos vivos apresentem inúmeras diferenças, todos compartilham propriedades bioquímicas (SCOTT, MATSUDAIRA, et al., 2000). Estas propriedades variam desde similaridades estruturais, como composição química celular, até semelhanças no genoma. Essas semelhanças podem ser explicadas pelo fato de que o código genético, discutido na sessão 2.1.2, possui um significado universal, ou seja, com raras exceções, ele é o mesmo para os mais diversos organismos, desde as bactérias até o homem.

Através de processos evolutivos tais como a *duplicação gênica* (COTTON, 2005; YIN, HARTEMINK, 2005) e a *mutação* (WEBSTER, SMITH, et al., 2004; SMITH, WEBSTER, et al., 2002), as espécies adquirem suas particularidades mantendo de seus ancestrais, apenas o necessário. Quando um gene específico é de vital importância para um determinado genoma, ele pode sofrer uma duplicação, gerando duas cópias idênticas. Na figura 2.7-a, o gene A sofreu duplicação gerando duas cópias A_1 e A_2 . Após a duplicação, as duas cópias seguem livres e através de processos de mutação, distanciam-se do ancestral comum, evoluindo de forma independente. Esses novos genes são chamados de *homólogos*, pois possuem uma origem comum. Por exemplo, a figura 2.7-b mostra que o gene A_1 , através de mutação, transformou-se no gene A_3 , e o mesmo ocorreu com os genes A_2 e A_4 . A espécie que possui os genes homólogos pode passar por um processo, denominado processo de *especiação* (COYNE, ORR, 2004), e dar origem a novas espécies. A figura 2.7-c mostra que a espécie I sofreu especiação e deu origem as espécies II e III, os genes A_3 e A_4 da espécie I foram herdados pelas espécies II e III. Após a especiação os genes homólogos continuam sofrendo mutação. Os genes homólogos dentro de uma mesma espécie são chamados de *parálogos*, caracterizando um tipo de homologia, e genes homólogos em diferentes espécies são chamados de *ortólogos*, o segundo tipo de homologia. Na figura 2.7-d $A_3^*A_4^*$, na espécie II e $A_3^+A_4^+$, na espécie III, são classificados como parálogos. Os genes $A_3^*A_3^+$, $A_3^*A_4^+$, $A_4^*A_3^+$ e $A_4^*A_4^+$ são classificados como ortólogos.

Conseqüentemente as proteínas que são sintetizadas por genes duplicados sofrem modificações proporcionais dando origem a proteínas diferentes, porém relacionadas, ou seja, genes homólogos produzem proteínas homólogas.

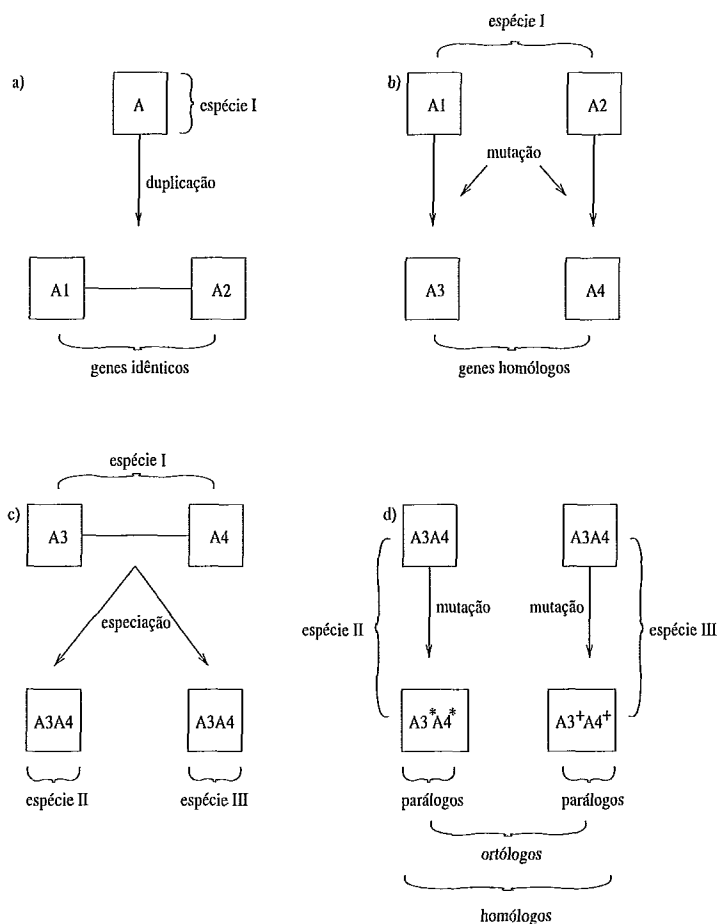


Figura 2.7: Genes homólogos.

2.2.2 Homologia e Similaridade

Genes e proteínas homólogos são classificados dentro de uma mesma *família*. Os membros dessas famílias provavelmente compartilham a mesma função. A classificação de genes e proteínas por meio de detecção de homologias é um dos principais objetivos dos projetos de *seqüenciamento de dados genômicos*. Os genes e proteínas de milhares de espécies têm sido seqüenciados, anotados e depositados em bancos de dados públicos, tais como GENBANK (DENNIS, LIPMAN, et al., 2004), TREMBL (BOECKMANN, BAIROCH, et al., 2003) e SWISS-PROT (BAIROCH, BOECK-

MANN, et al., 2005). Desta forma, é possível comparar novos genes e proteínas com as bases de dados genômicas, com o intuito de inferir homologia e classificá-los dentro de uma família. Isto poupa tempo e recursos, visto que os testes laboratoriais para a verificação da função e descoberta de relacionamentos evolutivos serão consideravelmente reduzidos.

A detecção *semi-automática* de seqüências homólogas tornou-se possível graças a *programas de alinhamento*, tais como BLAST (ALTSCHUL, GISH, et al., 1990) e FASTA (PEARSON, 1985). O termo *semi-automática* é usado, pois sempre é necessária a intervenção do biólogo (bioinformata) para comprovar homologia. Os programas de alinhamento buscam encontrar regiões repetidas, dentro de duas ou mais seqüências, de forma que estas regiões sejam o mais similar possível. Por exemplo, a figura 2.8-a mostra duas seqüências não alinhadas e a figura 2.8-b mostra o alinhamento dessas seqüências, para alinhar as seqüências da melhor maneira possível, ou seja, tendo o máximo de *resíduos* idênticos alinhados, é necessário introduzir *buracos* (representados por um traço na figura 2.8-b) no alinhamento das seqüências. O termo *resíduo* é utilizado para fazer referência a aminoácidos ou nucleotídeos. Os buracos são interpretados como *inserções* ou *exclusões* que ocorreram durante a evolução. Por exemplo, suponha que as duas seqüências da figura 2.8 são homólogas. Se a seqüência ancestral é representada pela seqüência *PAWHEE*, então pode-se dizer que a seqüência *Seq1* ganhou um resíduo na posição 1, 3 e 6, e que perdeu um resíduo na posição 9. O mesmo ocorreu com a seqüência *Seq2*, ela perdeu resíduos nas posições 1, 3 e 6, e ganhou um resíduo na posição 9. Quando as seqüências perdem resíduos é dito que ocorreu uma exclusão, e quando as seqüências ganham resíduos é dito que ocorreu uma inserção. Tanto a inserção quanto a exclusão de resíduos é um evento de mutação, discutido na sessão 2.2.1.

a) Seq1 = EAGAWGHEE
 Seq2 = PAWHEAE

b)

	1	2	3	4	5	6	7	8	9	10
Seq1	E	A	G	A	W	G	H	E	-	E
Seq2	-	P	-	A	W	-	H	E	A	E
provável ancestral		P		A	W		H	E		E

Figura 2.8: Alinhamento entre duas seqüências.

Os programas de alinhamento são divididos em dois grupos de acordo com o número de seqüências alinhadas. Programas como BLAST e FASTA alinham apenas pares de seqüências, sendo classificado como programas de alinhamento *par a par*. Por outro lado, ferramentas como CLUSTALW (THOMPSON, GIBSON, 1994a), T-COFFEE (NOTREDAME, HIGGINS, et al., 2000) e ALIGN-M (WALLE, I., et al., 2004), entre outras, alinham qualquer quantidade de seqüências e são classificadas como programas de *alinhamento múltiplo*, pois todas as seqüências são alinhadas simultaneamente.

A forma mais simples de dizer que duas seqüências podem ser homólogas é através da *similaridade* entre as seqüências. A similaridade entre duas ou mais seqüências mede, considerando o alinhamento entre elas, a conservação entre os resíduos. Para medir similaridade é considerada a identidade (resíduos idênticos alinhados) e positividade (resíduos diferentes alinhados que compartilham propriedades físico-químicas). Porém, similaridade não necessariamente reflete homologia. Similaridade entre fragmentos de seqüências pode ser resultado de processos evolutivos ou obra do acaso. Quando duas seqüências possuem similaridade alta ou média, as chances deste evento ocorrer ao acaso são remotas e as seqüências podem ser consideradas homólogas, desde que uma análise adicional seja realizada comprovando a origem comum entre essas seqüências. Por outro lado, a baixa similaridade não descarta a possibilidade de homologia. A próxima sessão explica como a homologia pode ser detectada quando a similaridade entre as seqüências envolvidas é baixa.

2.2.3 Homologia Remota ou Distante

Os programas de alinhamento par a par, discutidos na sessão 2.2.2, além de alinharem duas seqüências, também são usados para medir similaridade entre novas seqüências e bancos de dados públicos. Como foi dito na sessão 2.2.2 a homologia entre duas seqüências pode ser inferida através da similaridade entre elas, porém quando a similaridade é baixa esses programas não detectam ou até descartam a possibilidade de homologia. Duas proteínas podem apresentar seqüências de aminoácido divergentes, enquanto suas estruturas terciárias, sessão 2.3.3, podem ser similares. Essas proteínas provavelmente possuem uma origem comum. Porém, as seqüências de aminoácidos sofreram tantas modificações, que simples comparações não são capazes de comprovar o relacionamento entre elas. Sendo assim, a homologia entre essas seqüências é classificada como *distante* ou *remota*.

Programas capazes de detectar homólogos distantes precisam utilizar recursos que não estejam inteiramente baseados na similaridade entre as seqüências. Os principais programas com este propósito utilizam modelos probabilísticos e alinhamentos múltiplos de seqüências homólogas, tais como o alinhamento mostrado na figura 3.7. A detecção de homologia distante se faz necessária para auxiliar o processo de anotação, com o objetivo de diminuir o número de seqüências sem relacionamento ou *hits*, também chamadas de *genes órfãos* (DAVIDS, FUXELIUS, et al., 2003).

2.3 Proteínas e suas Estruturas

As proteínas são de vital importância para a vida celular (ALBERTS, BRAY, et al., 2002). Elas desempenham milhares de funções específicas dentro dos organismos vivos, tais como transporte (hemoglobina transporta oxigênio no sangue), estrutura (colágeno presente nos tendões estabelece uma estrutura resistente e elástica), regularização (controle de atividades celulares), catalisação (enzimas tais como pepsina responsável pela digestão de alimentos) e sinalização (percebem e reagem ao meio, tais como os anticorpos), entre outras. A partir do momento que uma proteína é sintetizada, sessão 2.1.2, ela passa por vários estágios, formando sub-estrutura que formarão sua *estrutura final*, figura 2.9. Através de métodos experimentais como

a *cristalografia* (WANG, ADAMS, et al., 2005) ou *ressonância magnética nuclear* (BONANNO, ALMO, et al., 2005), é possível determinar a estrutura tridimensional de muitas proteínas. A estrutura final de uma proteína, ou seja, a disposição espacial de seus átomos, está diretamente ligada com a função exercida por ela. Proteínas com estrutura similar, provavelmente compartilham funções similares. Porém, para interpretar estas informações e utiliza-las na detecção de homólogos distantes, é necessário compreender um conjunto de regras de formação dessas estruturas. O objetivo desta sessão é explicar como são formadas as estruturas das proteínas. Sendo assim, as próximas sessões descrevem todos os estágios na formação dessas estruturas. A sessão 2.3.1 explica a primeira forma da proteína, ou seja, sua estrutura primária. A sessão 2.3.2 descreve a formação da estrutura secundária a partir da estrutura primária. Em seguida a sessão 2.3.3 descreve a formação da estrutura terciária das proteínas e como os elementos de estrutura secundária contribuem para sua formação. Finalmente, a sessão 2.3.4 descreve como várias estruturas terciárias interagem formando a estrutura quaternária da proteína.

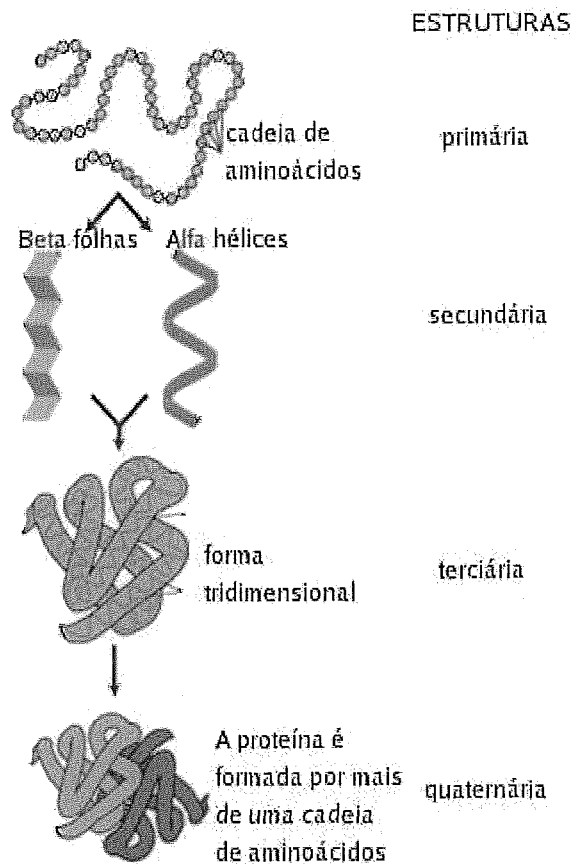


Figura 2.9: Fases da estrutura das proteínas.

2.3.1 Estrutura Primária

Todos os 20 aminoácidos presentes na natureza, possuem em comum um átomo de carbono central ($C\alpha$) ao qual está ligado um átomo de hidrogênio (H), um grupo amino (NH_2) e um grupo carboxílico ($COOH$), representado na figura 2.10. O que distingue um aminoácido de outro é a *cadeia lateral* ligada ao carbono central ($C\alpha$). Existem 20 principais cadeias laterais identificadas pelo código genético, (outras cadeias também ocorrem, porém são raras). Durante o processo de síntese protéica, o grupo carboxílico do aminoácido n reage com o grupo amino do aminoácido $n + 1$, liberando uma molécula de água, essa ligação é conhecida como *ponte peptídica*, figura 2.11. Esse processo se repete até que todos os aminoácidos estejam ligados formando a *cadeia principal* da proteína, também chamada de *estrutura primária*. As pontes peptídicas possuem uma natureza planar e bastante rígida, o grau de liberdade da cadeia principal é conferido apenas às ligações do carbono central ($C\alpha$). Sendo

assim, duas rotações são possíveis, uma em torno da ligação N-C α , cujo ângulo é chamado de phi (ϕ) e a outra em torno da ligação C α -C, cujo ângulo é chamado de psi (ψ).

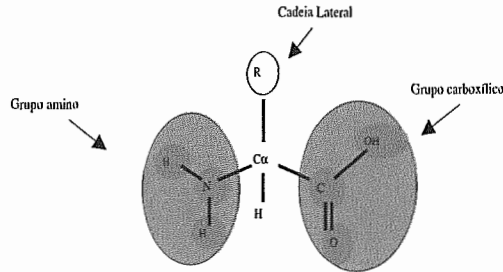


Figura 2.10: Fórmula geral de um aminoácido.

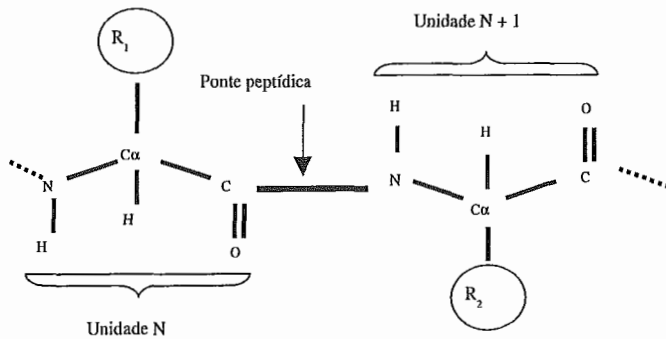


Figura 2.11: Estrutura primária.

A figura 2.12 mostra as principais cadeias laterais. Dependendo da natureza química das cadeias laterais, os aminoácidos podem ser divididos em três classes. A primeira classe compreende os aminoácidos que possuem cadeia lateral estritamente *hidrofóbica* (não são solúveis em água), a segunda compreende os aminoácidos *carregados eletricamente* (positiva ou negativamente) e a terceira é formada por aminoácidos com cadeia lateral *polar* (solúveis em água).

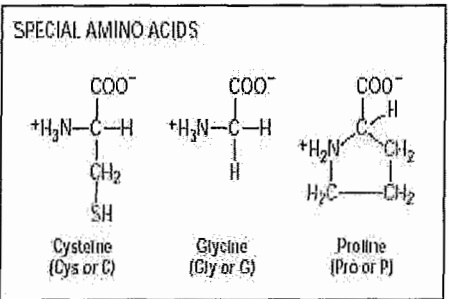
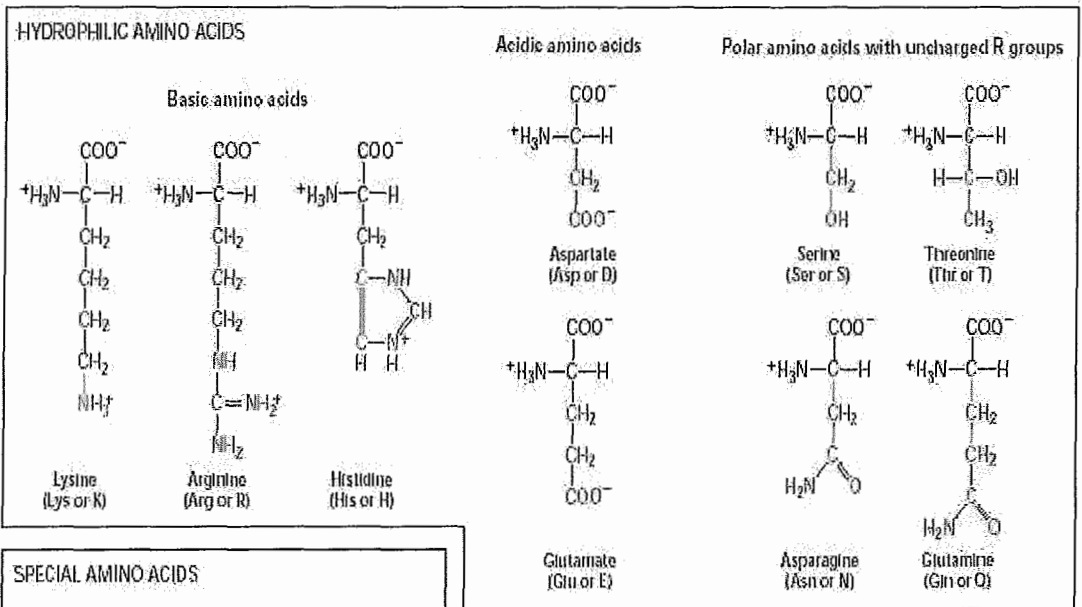
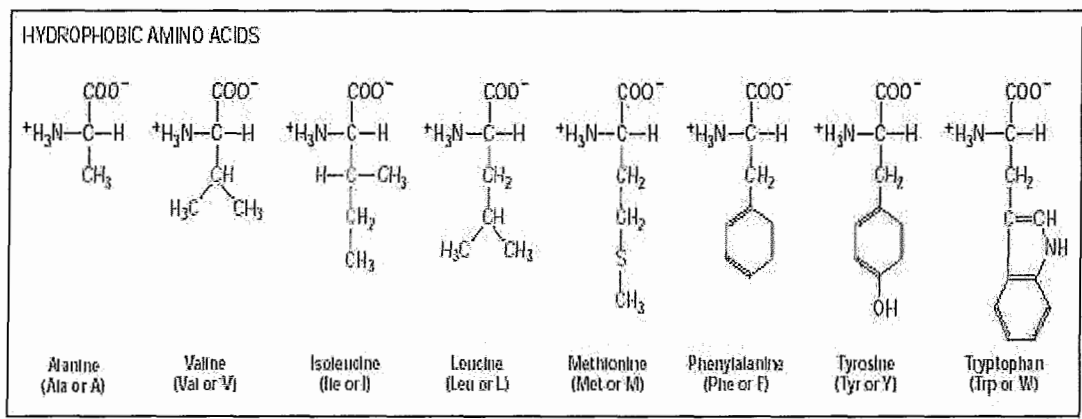


Figura 2.12: Classificação dos aminoácidos

2.3.2 Estrutura Secundária

A partir da estrutura primária da proteína, os átomos que formam as cadeias peptídicas sofrem uma disposição espacial de acordo com interações intramoleculares. Essas interações envolvem apenas os átomos da cadeia principal da proteína, ou seja, átomos das cadeias laterais dos aminoácidos não estão envolvidos. Esse fenômeno

recebe o nome de *estrutura secundária* da proteína. As estruturas secundárias são divididas em estruturas *regulares* e *irregulares*. As estruturas regulares podem assumir duas formas principais: *Alpha-hélices* (PAULING, COREY, et al., 1951) e *fitas* ou *folhas-beta* (BRANDEN, TOOZE, 1991a). Enquanto que a estrutura irregular é formada por regiões denominadas *loops* (BRANDEN, TOOZE, 1991a). Tanto as estruturas secundárias regulares quanto as irregulares são de vital importância na formação da estrutura terciária, sessão 2.3.3, chegando a envolver 60% do total dos aminoácidos presentes em uma proteína (SCOTT, MATSUDAIRA, et al., 2000). A seguir são descritas as formações e propriedades destes elementos.

Os Elementos Regulares Alpha-Hélice

Os elementos de hélice são criados por uma curvatura na cadeia principal da proteína. As hélices podem ser dobradas em duas direções, esquerda ou direita. Porém a maioria apresenta rotação para a direita. Dentre elas, a mais conhecida e comumente encontrada é a *alpha-hélice*. A estrutura desses elementos é estabilizada pela interação de pontes de hidrogênio entre o átomo de oxigênio do grupo carboxílico, de cada aminoácido, e o hidrogênio do grupo amino, do quarto aminoácido a seguir, figura 2.13, formando um intervalo de 3.6 resíduos por volta da cadeia principal.

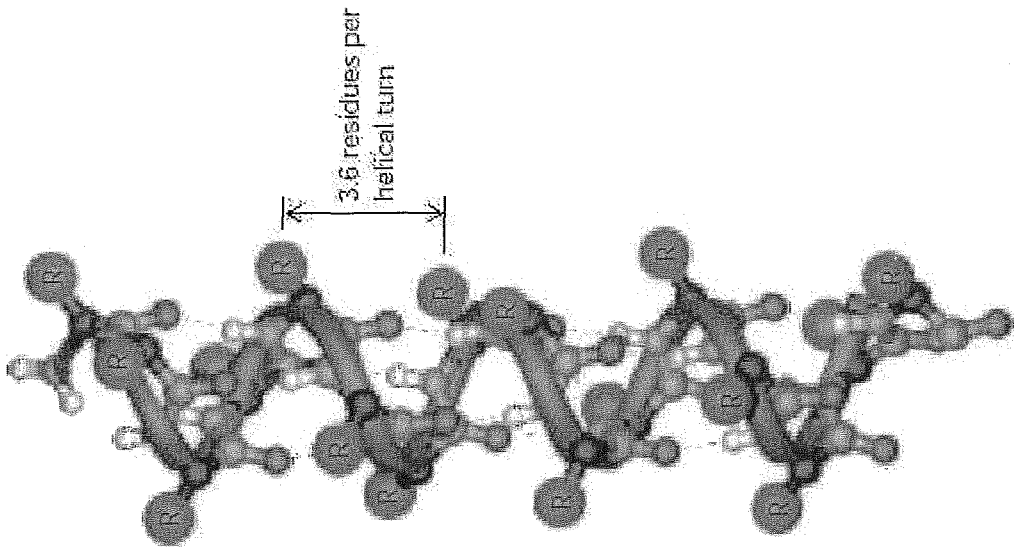


Figura 2.13: Estrutura secundária regular alpha-hélice.

Os Elementos Regulares Fitas ou Folhas-Betas

Diferente das hélices, os elementos conhecidos como fitas ou folhas-beta são formados por pontes de hidrogênio de cadeias polipeptídicas adjacentes. A figura 2.14 mostra duas visões dos elementos folhas-betas: superior 2.14-a e lateral 2.14-b. Cada trecho da cadeia polipeptídica contém, em média, de cinco a oito aminoácidos e é chamado de folha. As folhas-betas causam uma drástica conformação na cadeia polipeptídica de tal forma que os ângulos ϕ e ψ , discutidos na sessão 2.3.1, são distorcidos cerca de 180 graus, um com relação ao outro. Dois tipos de folhas-betas são encontrados: as *paralelas* e *anti-paralelas*. Nas paralelas as folhas estão dispostas na mesma direção, enquanto que nas anti-paralelas em direções opostas.

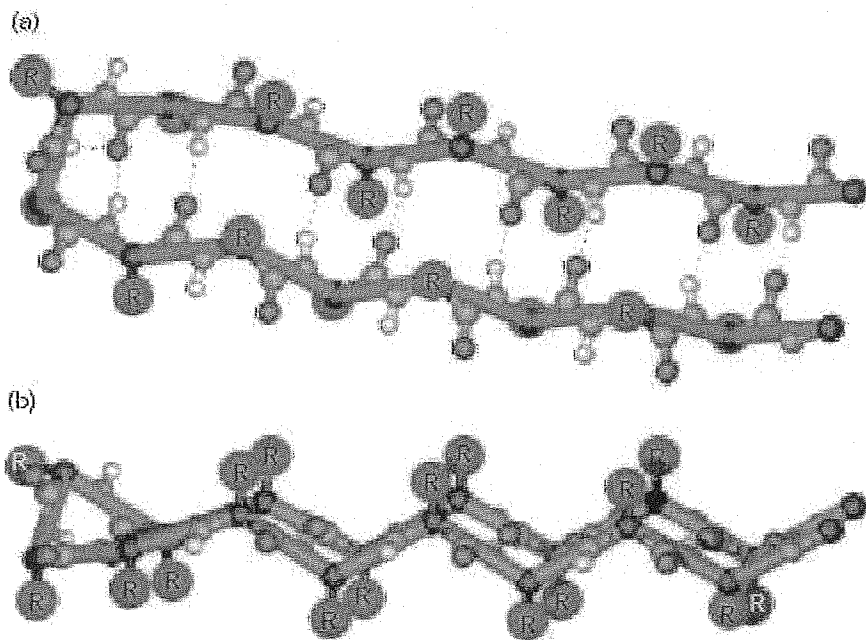


Figura 2.14: Estrutura secundária regular folhas-Betas.

Os Elementos Irregulares *Loops*

Os elementos regulares totalizam a maior parte da estrutura secundária presente em uma proteína. Entretanto essas estruturas regulares são conectadas através de regiões de *loops*, que possuem formas e tamanhos irregulares. Essas regiões são freqüentemente encontradas na superfície das proteínas. Embora sejam simples transições entre estruturas secundárias regulares, elas podem possuir um significado

estrutural de vital importância, por conter o *sítio ativo* (SCOTT, MATSUDAIRA, et al., 2000), ou seja, o conjunto de aminoácidos que determinam a função da proteína.

Quando proteínas homólogas são comparadas, as inserções e exclusões de alguns aminoácidos ocorrem quase que exclusivamente nas regiões de *loops*. Isso se explica devido ao fato de que o interior das proteínas tende a sofrer menos com as mutações causadas pela evolução, enquanto que as regiões externas, onde os *loops* predominam, são mais predispostas.

2.3.3 Estrutura Terciária, Motivos e Domínios

A *estrutura terciária* refere-se à completa conformação da cadeia polipeptídica (BOURNE, WEISSIG, 2003). Isto é, o arranjo tridimensional de todos os átomos da proteína. O efeito hidrofóbico contribui de forma decisiva para formação final da proteína. Aminoácidos com cadeia lateral hidrofóbica tendem a migrar para o interior da proteína, com o intuito de se distanciar das moléculas de água. Por outro lado, os aminoácidos com cadeia lateral polar ou carregada (positiva ou negativamente), tendem a se manter na superfície da proteína. A estabilidade da estrutura é mantida por pontes de hidrogênio entre unidades não envolvidas na estrutura secundária, por pontes de hidrogênio entre grupos das cadeias laterais dos aminoácidos, por ligações iônicas entre grupos carregados contrariamente e por ligações covalentes do tipo sulfídicas, envolvendo dois átomos de enxofre nos aminoácidos de *cysteine*. Todavia, estas forças intramoleculares não são fortes o suficiente para manter a estrutura da proteína fixa, o rompimento de algumas ligações e a formação de novas pode ocasionar uma leve flutuação na estrutura terciária de uma proteína.

As estruturas secundárias podem ser combinadas formando regiões com geometria específica que são frequentemente encontradas em estruturas tridimensionais de diversas proteínas. Essas combinações são chamadas de *motivos* (BRANDEN, TO-OZE, 1991b). Alguns destes motivos podem assumir determinado papel funcional. Por exemplo, a sub-estrutura formada por duas *alphas-hélices* unidas por uma região de *loop* é um ligante de cálcio (KRETSINGER, 1980), as proteínas que possuem

esse motivo provavelmente terão a função de se ligar a átomos de cálcio. Os motivos costumam ser regiões conservadas, mantendo a mesma seqüência de aminoácidos ou variações que respeitam as propriedades físico-químicas dos aminoácidos.

Domínios são regiões compactas da proteína que possuem representação estrutural, ou seja, caso sejam separadas da proteína coexistem independentemente. Os domínios estão diretamente relacionados com a função da proteína. Eles possuem cerca de 100 a 150 resíduos e são formados por diferentes combinações de elementos estruturais secundários e motivos.

2.3.4 Interações Protéicas, Estrutura Quaternária

As estruturas terciárias descrevem a organização estrutural de uma simples cadeia polipeptídica. Todavia, a maioria das proteínas é formada pela associação de várias sub-unidades que compõem a *estrutura quaternária* (KLOTZ, LANGERMAN, et al., 1970), conforme figura 2.15. As sub-unidades podem ser idênticas, caracterizando uma proteína *homogênea*, por exemplos as cadeias α da figura 2.15, ou podem ser diferentes dando origem a proteínas *heterogêneas*, como as cadeias α e β na figura 2.15. As ligações entre as sub-unidades são as mesmas encontradas na formação de estruturas secundárias e terciárias.

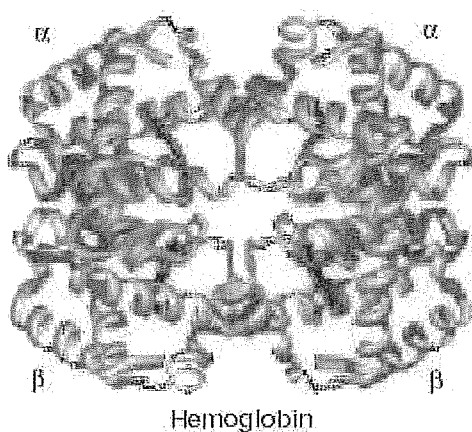


Figura 2.15: Estrutura quaternária.

CAPÍTULO 3

Hidden Markov Models Aplicados à Detecção de Homologias

Hidden Markov models (HMMs), descritos em (EDDY, 1996; HUGHEY, KROGH, 1996b; KROGH, BROWN, et al., 1994), são aplicados a diversos problemas da biologia molecular, tais como identificação de genes (BREJOVA, BROWN, et al., 2005; MAJOROS, PERTEA, et al., 2005), alinhamentos múltiplos de seqüências (MAMITSUKA, 2005; EDGAR, SJOLANDER, 2004; KNUDSEN, MIYAMOTO, 2003) e predição de estruturas de proteínas (BAE, MALLICK, et al., 2005; CAMPROUX, TUFFERY, 2005; LIN, SIMOSSIS, et al., 2005), entre outros. HMMs foram utilizados inicialmente no reconhecimento de voz (MENDEL, 1992), posteriormente foram empregados com êxito na biologia molecular. O objetivo deste capítulo é apresentar os conceitos básicos sobre HMMs, com foco em *profile hidden Markov models* (pHMMs), que são HMMs utilizados para representar famílias de genes ou proteínas. O capítulo está organizado da seguinte forma: a sessão 3.1 introduz cadeias de Markov e aborda HMMs, a sessão 3.2 discute pHMMs. O desenvolvimento e utilização de pHMMs requer três etapas, mostradas na figura 3.1. Na primeira etapa, foi proposta uma arquitetura capaz de descrever famílias de genes e proteínas, descrita na sessão 3.3. Na segunda etapa, são aprendidos os parâmetros que melhor descrevem uma família, abordada na sessão 3.4. Nessa fase existem dois problemas: o primeiro ocorre quando os exemplos, usados para o aprendizado dos parâmetros, são insuficientes e/ou incompletos, sendo necessário a inclusão de informações a priori, esse

assunto é tratado na sessão 3.5. O segundo problema esta relacionado a dependência entre os exemplos usados no aprendizado. A solução é empregar métodos que atribuem diferentes pesos às seqüências. Como o principal objetivo deste trabalho é incorporar informações de estruturas de proteínas, no aprendizado de pHMMs, como será descrito no capítulo 5, essas informações são incorporadas através de algoritmos de atribuição de pesos à seqüências. Os principais algoritmos são descrito na sessão 3.6. A terceira etapa refere-se a inferência ou busca por novos homólogos, a sessão 3.7 descreve os principais métodos utilizados. Finalmente, a sessão 3.8 aborda como os resultados obtidos durante a aplicação de pHMMs podem ser analisados.

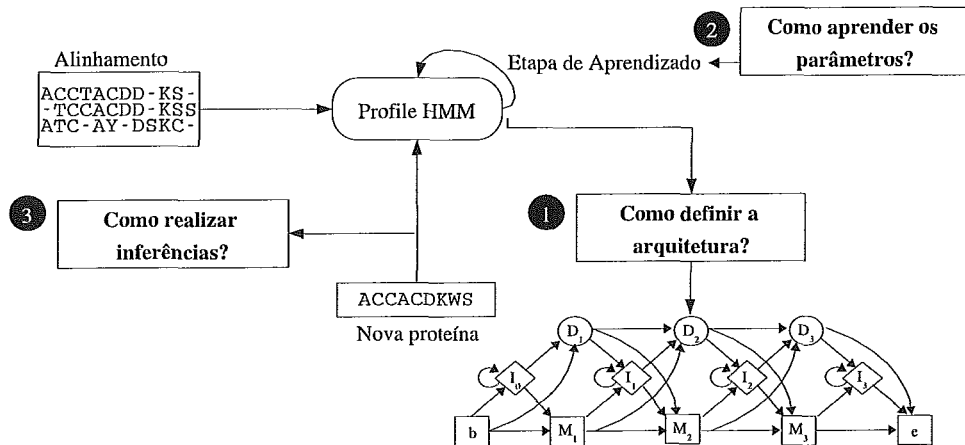


Figura 3.1: Fases de um pHMM

3.1 Cadeias de Markov e *Hidden Markov Models*

Seja M um sistema que em qualquer momento pode encontrar-se em um estado S_i , de um conjunto de estados S_1, \dots, S_N . O sistema muda de estados a intervalos regulares (possivelmente voltando ao mesmo estado), de acordo com um conjunto de probabilidades definidas para cada estado. Os momentos associados a mudança de estado serão escritos como $t = 1, 2, \dots$ e o estado no tempo t será dado como q_t . Se para um dado instante t a probabilidade de q_t assumir qualquer S_i depende apenas do estado anterior, então trata-se de uma cadeia de Markov (RABINER, 1989).

Uma cadeia de Markov pode portanto ser definida por $M = (S, T)$, onde S é o conjunto de estados, definido anteriormente e $T = (t_{ij})$ representa a *matriz de*

probabilidades de transição ou *modelo de transição*, para a qual, $\sum_j t_{ij} = 1$ e $t_{ij} = P(q_t = S_j \mid q_{t-1} = S_i)$. Cada elemento t_{ij} representa a probabilidade de mudança do estado S_i para o estado S_j . A cadeia é *estacionária*, ou seja, a probabilidade de transição t_{ij} entre quaisquer estados S_i e S_j não sofre alteração ao longo do tempo. Essa probabilidade é definida de acordo com o princípio da probabilidade condicional (DEGROOT, 1987). Dada uma seqüência observada $O = \{S_1, S_2, \dots, S_N\}$ que corresponde a $t = 1, 2, \dots, N$ é possível determinar a probabilidade de O dado o modelo, através da fórmula 3.1.

$$\begin{aligned} P(O \mid \text{Modelo}) &= P(S_1, S_2, \dots, S_N) \\ P(O \mid \text{Modelo}) &= P(S_1)P(S_2 \mid S_1) \dots P(S_N \mid S_{N-1}) \end{aligned} \tag{3.1}$$

Cadeias de Markov têm sido aplicadas ao reconhecimento de segmentos de seqüência de DNA, como regiões promotoras (VLADIMIR, TAN, et al., 2004) e ilhas CpG (WANG, HANNENHALLI, 2005; BIRD, 1982). Cada estado da cadeia de Markov da figura 3.2, representa um dos quatro nucleotídeos A , C , T , e G , sendo $S = \{S_a, S_c, S_t, S_g\}$, b e e representam os estados inicial e final, respectivamente. As setas representam as transições entre os estados. Dada uma seqüência de DNA, por exemplo, $x = ACTTG$, é possível afirmar que esta seqüência foi gerada visitando os estados $b, S_a, S_c, S_t, S_t, S_g, e$.

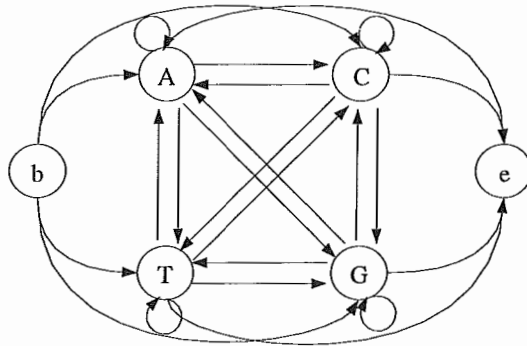


Figura 3.2: Cadeias de Makov para reconhecimento de seqüência de DNA

Os modelos apresentados até agora poderiam ser chamados de modelos de Markov observáveis, pois cada estado resulta na observação direta de um evento (por exemplo, o estado S_A na figura 3.2 resulta na geração do nucleotídeo A). Infe-

lizmente, este modelo é demasiadamente limitado para muitas aplicações. *Hidden Markov models* (RABINER, 1989) são modelos de Markov onde *cada observação é uma função probabilística do estado*. Como resultado, em um HMM o estado q_t do sistema está escondido, pois o sistema pode apenas ser observado por um conjunto de processos estocásticos que geram observações E_t .

Hidden Markov models (RABINER, 1989) são uma extensão de cadeias de Markov e podem ser definidos por $M = (S, A, T, E)$, onde S é o conjunto de estados, $A = \{\sigma_1, \dots, \sigma_R\}$ é o alfabeto de símbolos de tamanho R , $T = (t_{ij})$ a matriz de probabilidades de transição e $E = (e_i(\sigma))$ uma matriz de *probabilidades de emissão* ou *modelo de emissão*. Cada t_{ij} representa a probabilidade de transição do estado S_i para o estado S_j , como definido anteriormente e cada $e_i(\sigma)$ é a probabilidade de emissão do símbolo σ pelo estado S_i , onde $\sigma \in A$, $\sum_{k=1}^R e_i(\sigma_k) = 1$ e $0 \leq e_i(\sigma_k) \leq 1$. É assumido que $E = (e_i(\sigma))$ depende apenas do estado S_i .

Para o problema de detecção de homologias, o domínio do alfabeto A se restringe a 20 símbolos que representam os aminoácidos para seqüências de proteínas ou quatro símbolos que representam os nucleotídeos para seqüências de DNA ou RNA. Diferente das cadeias de Markov, HMMs possuem uma distribuição de probabilidades de emissão associada a cada estado e diferentes estados podem emitir o mesmo símbolo. Dessa forma, dado um conjunto de símbolos do alfabeto não é possível saber quais foram os estados responsáveis por gerar estes símbolos. Por exemplo, a figura 3.3 mostra um HMM com três estados $S = \{S_1, S_2, S_3\}$. O alfabeto A corresponde a três símbolos que representam os elementos de estrutura secundária, discutidos na sessão 2.3.2 página 25. Sendo assim, $A = \{a=\text{alpha-hélice}, b=\text{folha beta}, l=\text{loop}\}$, t_{ij} representa as transições entre os estados S_i e S_j . A figura 3.3 inclui o modelo de emissão (isto é, as probabilidades de emissão de cada símbolo em cada estado). É habitual representar apenas as transições possíveis. Diferente das cadeias de Markov, dada uma seqüência x qualquer não é possível saber qual a seqüência de estados responsável por emitir x . Por exemplo, dada a seqüência $x = abb$ e o HMM da figura 3.3, qualquer uma das seguintes seqüências de estados $S = \{(b, S_1, S_1, S_1, e), (b, S_1, S_2, S_2, e), \dots, (b, S_3, S_3, S_3, e)\}$ pode ser responsável por

gerar x . Por isso, é dito que HMMs possuem variáveis escondida ou ocultas. Observe que a cadeia de Markov mostrada na figura 3.2 não possui estados escondidos, ou seja, dada uma seqüência x sempre é possível determinar a seqüência de estados responsável por gerar x .

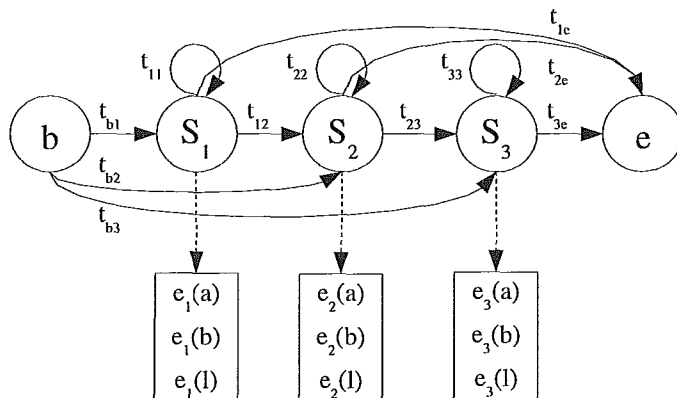


Figura 3.3: Esquema de um HMM.

A figura 3.3 apresenta um HMM como um autômato probabilístico. É importante ressaltar que esse autômato gera um processo estocástico. A figura 3.4, inspirada em Russel (RUSSEL, NORVIG, 2002), apresenta o processo estocástico para cadeias de Markov (onde a variável de estado é q_t) e para HMMs (onde a variável de estado é q_t e a variável de evidência é E_t). No caso do HMM, as probabilidades de transição entre os estados q_{t-1} e q_t são obtidos através da matriz T , e a relação entre as observações e os estados é dada pela matriz E .

A figura 3.4 mostra que um HMM é um caso particular de rede bayesiana, e que os algoritmos de inferência e de aprendizado desenvolvidos para estes modelos gráficos se aplicam a HMMs. Por outro lado, existem algoritmos que tiram vantagem das características específicas dos HMMs, tanto para inferência como para aprendizado de parâmetros, dos quais os relevantes para este trabalho são apresentados neste capítulo.

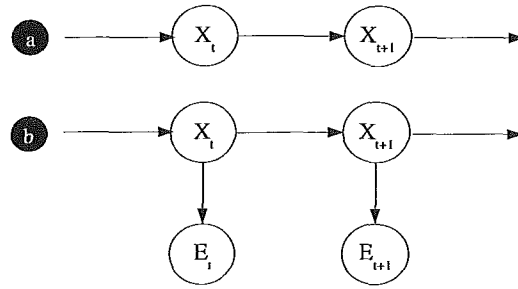


Figura 3.4: HMM representado como um modelo temporal probabilístico.

A figura 3.4 também sugere uma forma de generalizar HMMs. Em vez de uma variável de estado e uma de evidência para cada instante de tempo é possível ter várias variáveis de estado X_{1t}, X_{2t}, \dots e várias variáveis de evidência, E_{1t}, E_{2t}, \dots . Esta generalização é conhecida pelo nome de rede bayesiana dinâmica (DBNs). É fácil provar que o poder expressivo das DBNs é equivalente ao dos HMMs, porém DBNs são populares porque oferecem uma boa representação para modelos com muitos estados e matrizes de transição muito esparsas.

Alguns HMMs incluem estados silenciosos, ou seja, estados que não emitem símbolos. Estados silenciosos são convenientes para representar mudanças internas no autômato. Neste capítulo é descrito como os principais algoritmos de inferência e aprendizado de parâmetros foram generalizados para lidar com este caso.

Em todas as cadeias de Markov e modelos de Markov considerados até agora o estado no tempo t depende apenas do estado no tempo $t - 1$. Tais modelos são também conhecidos como modelos de Markov de primeira ordem. Em geral em um modelo de Markov de ordem k , o valor da variável de estado no tempo t depende das variáveis de estado para $t - k, \dots, t - 1$. Este trabalho discute apenas modelos de primeira ordem.

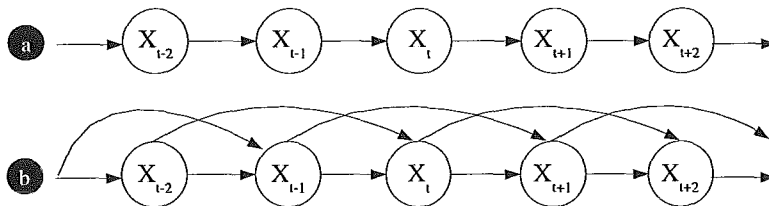


Figura 3.5: Cadeias de Markov de primeira e segunda ordem

Finalmente, a figura 3.6 apresenta um exemplo de um HMM mais complexo. O HMM é apresentado tanto como um autômato probabilístico (Rabiner), como escrevendo explicitamente as matrizes T e E (Russel). Note que a matriz de transição é bastante esparsa. Note ainda que a matriz de emissão não inclui colunas para os estados D , que são estados silenciosos. A figura 3.6-a mostra o modelo de transição. Supondo que $q_t = M_1$, q_{t+1} poderá assumir os valores I_1, M_2 e D_2 , explicitamente mostrados pelas transições do modelo Rabiner e mostrada pela segunda linha no modelo do Russel. A figura 3.6-b mostra o modelo de emissão. A figura 3.6 mostra um exemplo da arquitetura pHMM, um tipo de HMM especificamente desenvolvido para representar famílias de genes ou proteínas e que será tema da sessão seguinte.

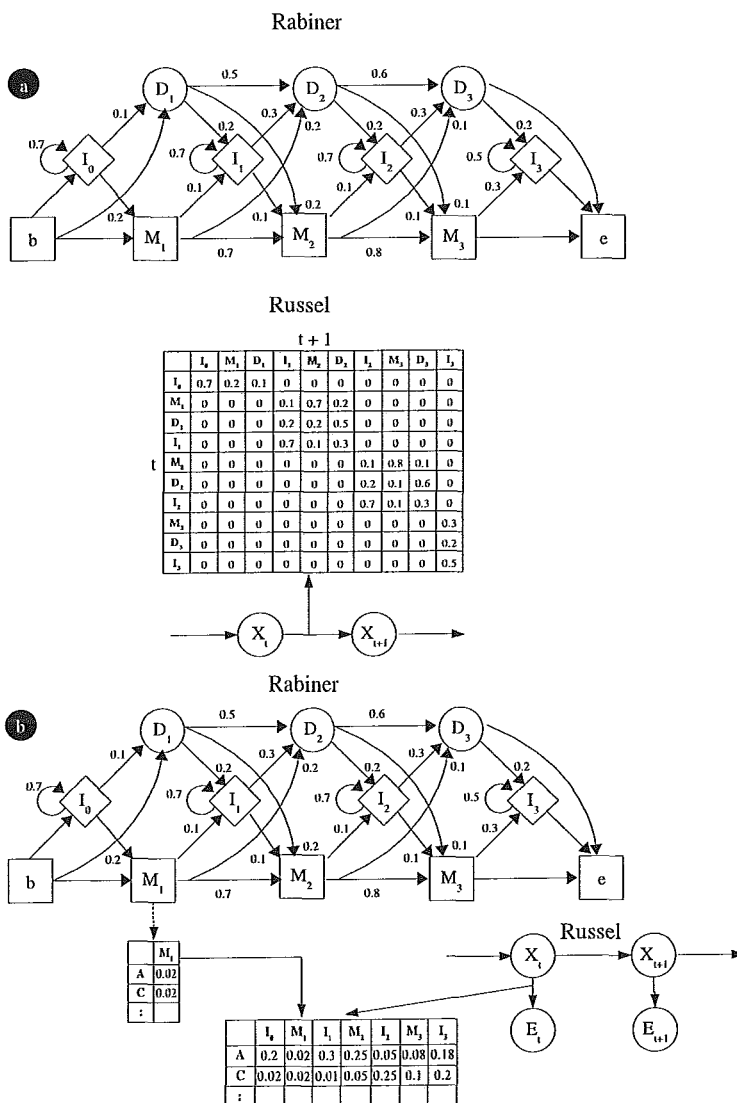


Figura 3.6: Representação Rabiner e Russel

3.2 *Profile Hidden Markov Models*

HMMs foram introduzidos em aplicações biológicas em 1989 (CHURCHILL, 1989), e alguns anos mais tarde foram adaptados ao problema de identificação de genes (KROGH, BROWN, et al., 1994). Porém, apenas em 1998 foram aplicado à detecção de homologias distantes. HMMs que representam famílias de genes ou proteínas são denominado *profiles hidden Markov models* (pHMMs). *Profile* HMMs solicitam um conjunto de seqüências (genes ou proteínas) como entrada que são geralmente seqüências alinhadas e relacionadas. A partir desse conjunto o modelo é construído e pode ser usado para identificar novos membros da família de genes ou proteínas, para a qual foi modelado. As próximas sessões apresentam cada fase necessária para que pHMMs sejam utilizados na detecção de homólogos distantes.

3.3 *Arquitetura de Profiles HMMs*

A arquitetura de pHMM é composta por um conjunto de estados, cuja a função é representar os padrões de um conjunto de seqüências relacionadas, que representam uma família de genes ou proteínas. Nesta sessão é explicado, passo a passo, como a arquitetura de pHMMs foi definida. Assuma que as seqüências da família que se deseja modelar estão alinhadas. Para construir um modelo capaz de representar as propriedades de uma família é necessário identificar padrões na formação das seqüências envolvidas. A figura 3.7 apresenta parte de um alinhamento múltiplo de algumas proteínas da família das globinas. Nesse alinhamento é possível notar um consenso ou padrão de formação. Nas colunas 11, 23, 40, 48, 54 e 57 os aminoácidos *L*, *W*, *L*, *P*, *F* e *F* (representados por letras maiúsculas) prevalecem, ou seja, são encontrados em maior número. Enquanto que em outras há uma variedade maior de aminoácidos diferentes, porém alguns são mais numerosos. Por exemplo, nas colunas 12, 19, 21, 25, 26, 36 e 44 os aminoácidos *s*, *v*, *a*, *k*, *v*, *g*, *f* (representados por letras minúsculas). Existem ainda colunas com poucos aminoácidos e a ausência de aminoácidos em algumas proteínas é representada por um traço. Todas estas informações extraídas do alinhamento refletem os processos evolutivos, pelos quais cada proteína da família das globinas passou. A seqüência consenso representa

o ancestral comum, a partir do qual cada proteína derivou. Sendo que algumas perderam aminoácidos durante a evolução, outras apenas substituíram o aminoácido original por outro e finalmente algumas proteínas ganharam novos aminoácidos. Porém, a seqüência consenso permanece caracterizando os membros da família. Para maiores detalhes sobre alinhamentos de seqüências consulte a sessão 2.2.2 página 18.

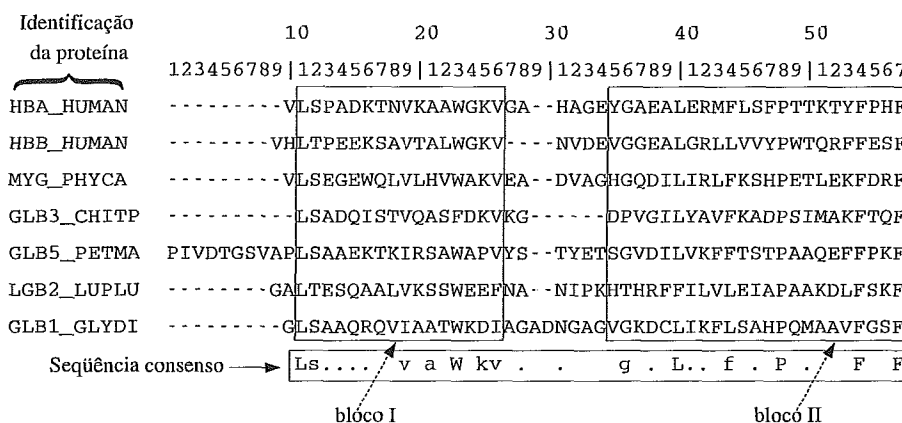


Figura 3.7: Alinhamento parcial de proteínas da família das globinas.

Para representar os padrões das seqüências através de pHMM, é necessário representar as colunas do alinhamento que refletem os consensos da família. Se as colunas com buracos são desconsideradas, observa-se um conjunto de blocos bem formados, que unidos contem a seqüência consenso capaz de representar os padrões da família, na figura 3.7 esses blocos estão representados por bloco I e II. A primeira arquitetura proposta para pHMM, figura 3.8, descrita em (HENIKOFF, HENIKOFF, 1991), considera apenas blocos de seqüências alinhadas sem buracos. Nessa arquitetura, cada estado, denominado estado de *match* M_j , representa uma coluna do alinhamento. Por exemplo, o bloco I da figura 3.7 pode ser representado pela arquitetura da figura 3.8, onde cada coluna j do bloco I está associada ao estado M_j e a constante k é igual a quantidade de colunas do Bloco I, neste exemplo $k = 16$. Cada estado M_j possui uma distribuição de probabilidades que representa as freqüências de ocorrência de cada resíduo na coluna j do alinhamento. As setas que separam os estados na figura 3.8 são denominadas transições e representam as probabilidades de mudanças de estado. O aprendizado dessas probabilidades é discutido na sessão

3.4. O estado S serve para encaixar resíduos antes de M_1 . Por exemplo, todos os resíduos que ocorrem antes do bloco I serão representados pelo estado S , observe que o estado S possui uma auto-transição, que permite que mais de um resíduo seja emitido. Da mesma forma, o estado N serve para encaixar resíduos após o estado M_k . Finalmente para completar a arquitetura dois estados *silenciosos*, que não emitem símbolos, são incluídos. Esses estados delimitam o início (b) e fim (e) do padrão representado.

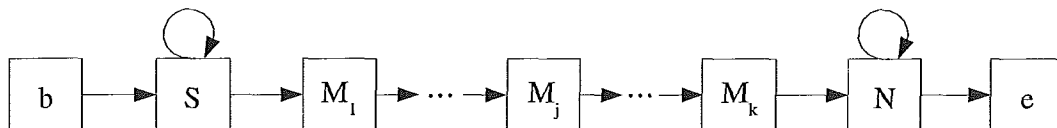


Figura 3.8: Arquitetura básica para pHMMs

Essa arquitetura foi baseada em PSSMs *Position-specific scoring matrices* (GRIBSKOV, MCLACHLAN, et al., 1987). Essa técnica emprega uma matriz de probabilidades, onde as colunas representam as posições do alinhamento, e as linhas o alfabeto de símbolos. PSSM, pode ser vista como um caso bem simples de pHMM, já que os processos de *inserções* e *exclusões* de resíduos não são tratados.

Para considerar o alinhamento completo, da figura 3.7, sem desconsiderar os buracos, dois tipos de eventos devem ser tratados: as *inserções* e as *exclusões*. Os termos *inserções* e *exclusões* foram discutidos na sessão 2.2.2 página 18. Se os blocos I e II são considerados, observa-se uma região entre os dois blocos, os resíduos nessa região são tratados como *inserções*. Para representar as *inserções*, ou seja, a porção do alinhamento de seqüências que não foi caracterizada como estado de *match*, um novo conjunto de estados é adicionado a arquitetura da figura 3.8. Esses estados são chamados de estados de *insert* I_j , onde I_j será usado para encaixar *inserções* após o j -ésimo resíduo ter sido reconhecido pelo estado M_j . Por exemplo, na figura 3.9 os estados M_1 a M_j representam as colunas do bloco I, sendo j o número de colunas desse bloco. O estado I_j representa as colunas entre os blocos I e II. O bloco II é representado pelos estados M_{j+1} a M_k , sendo k o número de colunas do bloco I mais o número de colunas do bloco II e $k - j$ o número de colunas do bloco II.

A cada estado de *match* está associada apenas uma coluna do alinhamento, enquanto que a cada estado de *insert* pode está associada mais de uma coluna do alinhamento. Os estados de *insert*, assim como os estados de *match*, possuem uma distribuição de probabilidades sobre o alfabeto de símbolos, denominada probabilidade de emissão. Os estados são separados pelas seguintes transições: M_j para I_j , uma auto-transição em I_j (permitindo que mais de um resíduo seja inserido) e I_j para M_{j+1} . O aprendizado das probabilidades de emissão e transição serão discutidos na sessão 3.4. Os estados S , N , b e e , discutidos anteriormente completam a arquitetura. Essa arquitetura é adotada pelo programa META-MEME (GRUNDY, BAKER, 1997), que faz uso de pHMM para a detecção motivos.

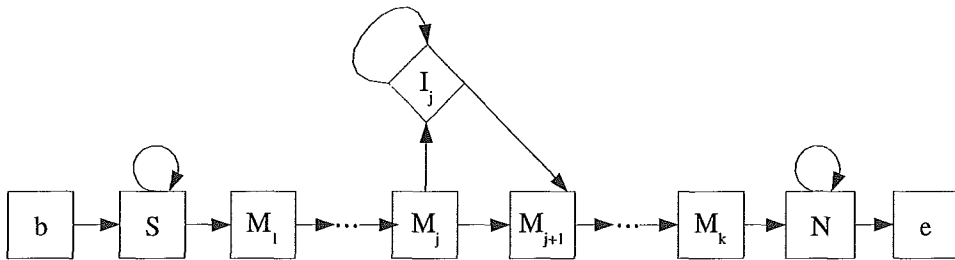


Figura 3.9: Arquitetura para HMMs incluindo estados de *insert*

Para compor a arquitetura final, é necessário incluir o tratamento de exclusões. Suponha que os estados de *match* admitam um certo nível de buracos. Por exemplo, colunas com menos de 50% de buracos são consideradas estados de *match*. Sendo assim, o bloco I será formado pelas colunas 10 a 28, e o bloco II pelas colunas 31 a 57, do alinhamento da figura 3.7. Para tratar buracos em estados de *match* foram adicionados novos estados, denominados estados de *delete* D_j . Esses estados representam a ausência de resíduos em determinados estado de *match*, ocasionando novas transições. Sendo assim, é possível que existam transições entre os estado M_j e M_{j+t} , onde t é uma constante positiva e $t < k$, sendo k o último estado do modelo. Os estados D_j são silenciosos, seu propósito é adicionar saltos à arquitetura proporcionando uma melhor adaptação do pHMM aos padrões extraídos dos alinhamentos múltiplos de seqüências. A arquitetura completa, apresentada na figura 3.10, foi introduzida por (KROGH, 1994). Os círculos representam os estados

de *delete*, os losangos os estados de *insert* e os quadrados os estados de *match*, cada conjunto M_i, I_i, D_i é chamado de nó. O estado I_0 serve para encaixar inserções antes do primeiro *match*. Essa arquitetura é fixa o que varia é o número de nós. Esse número é determinado durante a fase de aprendizado da arquitetura, descrita na sessão 3.4. A arquitetura da figura 3.10, com algumas alterações, é adotada pela maioria dos programas que implementam pHMMs na detecção de homologies, tais como HMMER (EDDY, 1998) e SAM (HUGHEY, KROGH, 1996a).

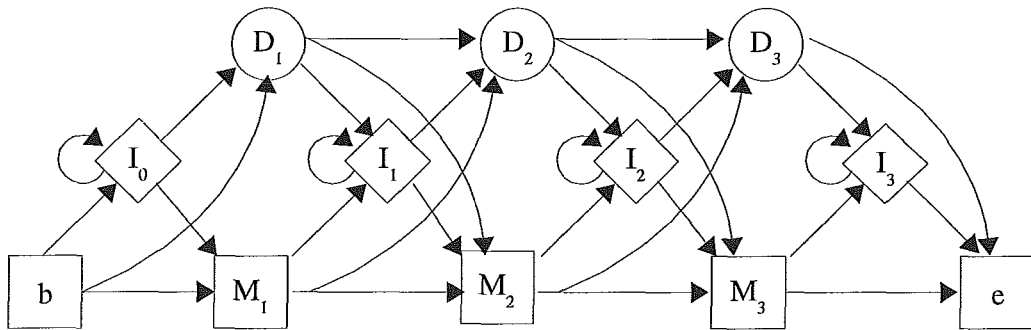


Figura 3.10: Arquitetura completa para pHMMs

3.4 Aprendizado da Arquitetura e dos Parâmetros

Tanto os parâmetros quanto a arquiteturas de pHMMs são aprendidos através do conjunto de seqüências relacionadas, ou seja, seqüências homólogas. Quando essas seqüências foram previamente alinhadas, a arquitetura e os parâmetros do pHMM são extraídos diretamente do alinhamento, pois os caminhos para as seqüências de treinamento são conhecidos. Por outro lado, se os caminhos não são conhecidos, não há uma forma direta de estimar os parâmetros e procedimentos iterativos devem ser usados. O principal algoritmo para esse propósito é conhecido por Baum-Welch (BAUM, 1972) ou EM (*Expectation Maximization*), diferentes algoritmos de otimização numérica (COLLINS, 2002; BAGOS, LIAKOPOULOS, et al., 2002) e outros tais como GEM (*generalized EM*) (BALBI, CHAUVIN, 1994) também podem ser empregados no aprendizado da arquitetura e dos parâmetros de pHMMs. Tradicionalmente pHMMs, voltados à detecção de homologias distantes, trabalham com conjunto de seqüências previamente alinhadas. Isso pode ser visto como uma vantagem da técnica, pois os alinhamentos podem ser providos por diferentes ferramentas. Aprender a arquitetura de pHMMs, a partir de um conjunto de seqüências alinhadas, consiste em determinar que colunas do alinhamento devem ser relacionadas a estados de *match* e quais a estados de *insert*, como visto na sessão 3.3. O aprendizado da arquitetura também é chamado de estratégia de seleção de estados. Existem pelo menos três estratégias diferentes, duas delas, abordadas na sessão 3.4.1, aprendem em separado a arquitetura e os parâmetros. A terceira estratégia, abordada na sessão 3.4.2, aprende simultaneamente.

3.4.1 Aprendendo a Arquitetura e os Parâmetros Separadamente

A primeira estratégia, conhecida como estratégia manual, é bem simples e passa a tarefa ao usuário que se torna responsável por decidir quais serão os estados de *match* e *insert*. A segunda estratégia faz uso de uma heurística que marca como estado de *match* colunas com menos de x% de buracos. Por exemplo, a figura 3.11 mostra parte de um alinhamento de seqüências, suponha que a heurística adotada considera como *match* colunas do alinhamento com menos de 60% de buracos. Sendo

assim, a coluna 1 corresponderá ao estado M_1 , a coluna 2 ao estado M_2 , a coluna 6 ao estado M_3 , e as colunas 3, 4, 5 corresponderão ao estado I_2 .

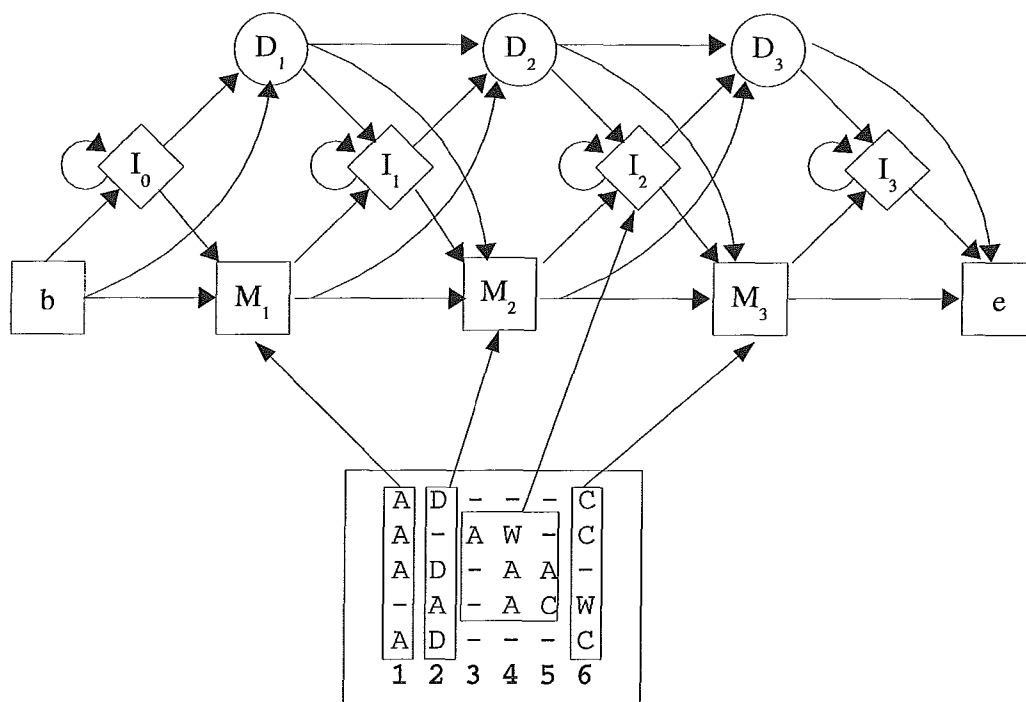


Figura 3.11: Exemplo de aprendizado da arquitetura de pHMMs

Após o aprendizado da arquitetura, os parâmetros do modelo podem ser aprendidos. Esses parâmetros são divididos em probabilidades de emissão associadas aos estados de *match* e *insert*, e probabilidades de transição entre os estados M, I, D . As probabilidades de emissão e transição são estimadas por simples contagem, pois todas as variáveis do modelo são observadas. A probabilidade de emissão é determinada pela fórmula 3.2, onde $c_i(\sigma)$ é o contador real do resíduo σ no estado i e $a(\sigma)$ é a informação a priori do resíduo σ , discutida na sessão 3.5.

$$e_i(\sigma) = \frac{c_i(\sigma) + a(\sigma)}{\sum_j c_i(\sigma_j) + a(\sigma_j)} \quad (3.2)$$

Da mesma forma as probabilidades de transição podem ser definidas pela fórmula 3.3, onde c_{ij} é o contador de transições do estado i para o estado j , a_{ij} é a probabilidade a priori, e $i, j \in \{M, I, D\}$.

$$t_{ij} = \frac{c_{ij} + a_{ij}}{\sum_j c_{ij} + a_{ij}} \quad (3.3)$$

3.4.2 *Maximum a Posteriori Construction*

A terceira estratégia faz uso de um algoritmo de programação dinâmica (BERTSEKAS, 1995) denominado *Maximum a Posteriori Construction* (MAP) (DURBIN, EDDY, et al., 1998) que ao mesmo tempo que aprende a arquitetura, determina os parâmetros do modelo. O algoritmo MAP calcula recursivamente um valor probabilístico S_j , determinado pela fórmula 3.4, para cada coluna do alinhamento. O objetivo é marcar como estado de *match* as colunas com os maiores S_j . Esse valor mostra o quanto uma coluna é mais significativa em relação as demais, ou seja, quanto maior S_j maior a probabilidade da coluna j ser um estado de *match*. S_j é calculado a partir de sub-alinhamentos que iniciam em i e terminam na coluna j , sendo $i < j$. O valor de S_i é incrementando com a soma do logaritmo das probabilidades de emissão e transição referentes a coluna i .

$$S_j = \max S_i + \tau_{ij} + M_j + l_{i+1,j-1} + \lambda \quad (3.4)$$

Onde τ_{ij} representado por 3.5, é o produto escalar entre o vetor de contadores de transição c_{xy} e o logaritmo aplicado ao vetor de probabilidades de transição t_{xy} , definido pela fórmula 3.3. M, D, I são os estados de *match*, *delete* e *insert*, respectivamente.

$$\tau_{ij} = \sum_{x,y \in M,D,I} c_{xy} + \log(t_{xy}) \quad (3.5)$$

De forma análoga M_{ij} , definido pela fórmula 3.6, é o produto escalar entre o vetor de contadores de emissão e o logaritmo aplicado ao vetor de probabilidade de emissão, definido pela fórmula 3.2, para estado de *match*. $l_{i+1,j-1}$ é similar porém refere-se a estados de *insert*, considerando as colunas entre $i + 1, \dots, j - 1$. λ é uma constante de normalização. O algoritmo completo é descrito no algoritmo 1.

$$M_j = \sum_{\sigma \in A} c_j(\sigma) + \log(e_j(\sigma)) \quad (3.6)$$

Algoritmo 1 MAP

Inicialização: $S_0 = 0, M_{L+1} = 0$
Recursão:
Para $j = 1$ até $L + 1$

$$S_j = \max_{0 \leq i < j} S_i + \tau i j + M_j + l_{i+1, j-1} + \lambda$$

$$\varphi_j = \operatorname{argmax}_{0 \leq i < j} S_i + \tau i j + M_j + l_{i+1, j-1} + \lambda$$

Fim do Para
Finalização: $j = \varphi_{L+1}$
Enquanto $j > 0$

 Marque a coluna j como estado de *match*

$$j = \varphi_j$$

Fim do Enquanto

O termo φ_j guarda o índice da coluna que deve ser considerada estado de *match*, caso j seja marcada. Após determinar todos os S_j , o algoritmo retrocede, iniciando em $j = \varphi_{L+1}$ até $j = 1$ marcando as colunas com melhores S_j , como colunas de *match*. L representa o número de colunas do alinhamento. As colunas não marcadas representarão os estados de *insert*.

Para facilitar a compreensão do algoritmo MAP é apresentado a seguir um exemplo prático. Considere o esquema de um alinhamento formado por quatro colunas, mostrado pela figura 3.12. A variável j , do algoritmo 1, assumirá os valores $j = 1$ até $j = 4$. Cada vez que j assume um determinado valor, i assumirá valores que variam de 0 até $j - 1$. Quando i assume esses valores, o algoritmo está procurando qual é a melhor configuração dos estados anteriores a j , caso j seja marcada como *match*. Por exemplo, se j aponta para a coluna 4 então existem 3 configurações possíveis para i (colunas anteriores a j). Primeiro $i = 3$ pode ser *match*, ou $i = 2$ pode ser *match* e a coluna 3 será *insert* ou $i = 1$ pode ser *match* e as colunas 2 e 3 serão *insert*. O objetivo do algoritmo MAP é encontrar a melhor configuração para cada coluna do alinhamento. Suponha que a melhor configuração para $j = 4$ seja $i = 2$, então a variável φ_4 receberá o valor 2. φ_4 guarda um ponteiro para a próxima coluna a ser marcada como *match* caso a coluna 4 seja *match*. MAP

pode ser considerado um algoritmo de aprendizado competitivo em que as colunas do alinhamentos competem entre si, pelo direito de ser marcada como *match*.

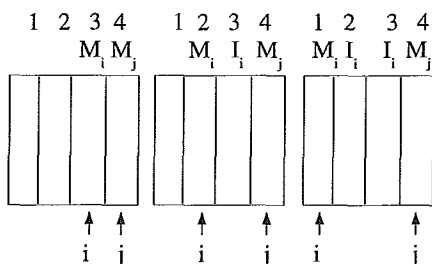


Figura 3.12: Exemplo do algoritmo MAP.

3.5 Priors e Informações Evolucionárias

Os parâmetros dos pHMMs, ou seja, as probabilidades de transição e emissão, para cada estado, devem ser extraídos diretamente do conjunto de seqüências que representa determinada família. Quando esse conjunto possui poucas seqüências ou amostras altamente relacionadas, as freqüências observadas são estimativas pobres da distribuição esperada para o modelo, ou seja, o modelo gerado não representará devidamente a família em questão, pois não possui uma quantidade de dados suficiente para estimar corretamente os parâmetros. Modelos que fazem uso apenas das freqüências observadas serão capazes de reconhecer bem somente as seqüências presentes no conjunto de treinamento, porém, não terão a capacidade de generalizar e reconhecer homólogos distantes.

Para ilustrar esse problema considere dois exemplos do trabalho (SJÖLANDER, 1997). No primeiro, existem apenas três seqüências para estimar os parâmetros do modelo. Através do alinhamento múltiplo dessas seqüências, observa-se uma coluna que contém apenas o aminoácido isoleucine. Como o número de amostras é muito pequeno, não há dados suficientes para afirmar que todas as proteínas da família analisada possuem o aminoácido isoleucine nesta posição. Portanto, se apenas as freqüências observadas forem utilizadas, a realidade não será devidamente representada. O aminoácido isoleucine é freqüentemente encontrado em ambientes que contêm folhas Beta, sessão 2.3.2 página 27, e os aminoácidos leucine e valine

são capazes de substituir isoleucine nesses ambientes. Sendo assim, a estimativa da distribuição esperada, para a coluna de isoleucine, deve ser sensível o suficiente para incluir os aminoácidos leucine e valine, além de representar outros aminoácidos com propriedades similares.

No segundo exemplo, existe um conjunto de 100 seqüência diferentes e novamente é encontrada uma coluna que contem apenas o aminoácido isoleucine. Nesse caso, existem mais evidências de que o aminoácido isoleucine está sempre conservado nessa posição, e acrescentar estimativas para aminoácido com propriedades similares, não parece conveniente.

A solução natural para resolver essas questões é introduzir informações a priori na estimativa dos parâmetros de pHMMs. Essas informações traduzem-se em probabilidades associadas a cada aminoácido, presente nas colunas do alinhamento múltiplo. Probabilidade a priori associada a um evento E é o grau de crença acordado para este evento, na ausência de quaisquer outras informações (RUSSEL, NORVIG, 2002). Alguns métodos são propostos para estimar e utilizar informações a priori. As próximas sessões abordam em detalhes os principais métodos.

3.5.1 Matrizes de Substituição

A necessidade de incorporar informações a priori, dentro de alinhamento de proteínas motivou o desenvolvimento de matrizes de substituição de aminoácidos. Elas tem sido freqüentemente empregadas em programas como BLAST (ALTSCHUL, GISH, et al., 1990), FASTA (PEARSON, 1985), PSI-BLAST (ALTSCHUL, MADDEN, et al., 2000) e CLUSTALW (THOMPSON, GIBSON, 1994a), etc.

Uma *matriz de substituição* associa a cada par de aminoácidos uma medida evolutiva. Essa medida é baseada na freqüência de substituição, observada em seqüências de proteínas depositadas em banco de dados públicos, e em geral reflete propriedades físico-químicas encontradas entre os aminoácidos. Por exemplo, aminoácidos que possuem molécula polar tem probabilidade maior de ser substituído, através de processos de mutação, por aminoácidos cuja molécula também tenha proprieda-

des polares. Novas metodologias para matrizes de substituição tem sido propostas, tais como (YU, ALTSCHUL, 2005; PORTO, BARBOSA, 2005; XU, MIRANKER, 2004). Porém, as mais utilizadas são: BLOSUM (HENIKOFF, HENIKOFF, 1992) e PAM (DAYHOFF, SCHWARTZ, et al., 1978). A seguir são descritos os principais conceitos por trás dessas abordagens.

Matrizes BLOSUM

A matriz BLOSUM foi obtida experimentalmente a partir de um conjunto de seqüências de proteínas relacionadas, que foram alinhadas sem buracos e depositadas na base de dados BLOCKS (HENIKOFF, HENIKOFF, et al., 1999). As seqüências da base BLOCKS foram selecionadas não permitindo que o grau de resíduos semelhantes entre as seqüências, superasse um valor de $x\%$. As matrizes são nomeadas de acordo com o valor x usado na extração dos dados. Dessa forma, BLOSUM62 indica que as seqüências que formam a matriz possuem menos de 62% de similaridade.

Para o conjunto de seqüências selecionadas foram calculadas as *freqüência de ocorrência* de cada aminoácido isoladamente, determinada por f_i , onde $1 \leq i \leq 20$, e a *freqüência de ocorrência* de cada dois pares de aminoácidos, determinada por f_{ij} , onde $1 \leq i, j \leq 20$. A *relação de substituição* entre os aminoácidos é dada pela razão $s(i, j) = f_{ij}/f_i f_j$. Cada elemento da matriz é obtido aplicando logaritmo a $s(i, j)$, de acordo com a fórmula 3.7.

$$s(i, j) = \log(f_{ij}/f_i f_j) \quad (3.7)$$

As matrizes BLOSUM62 e BLOSUM50 são amplamente usadas em alinhamentos múltiplos e busca por similaridade em banco de dados. Sendo BLOSUM62 mais utilizada para alinhamentos sem buracos e BLOSUM50 caso contrário.

Matrizes PAM

As matrizes PAM (*Point Accepted Mutation*) são medidas relativas de distâncias evolutivas. Nas matrizes PAM as mutações entre aminoácidos devem ser aceitáveis,

de acordo com processos evolutivos. *Mutação aceitável* é aquela que ocorreu e foi positivamente incorporada, ou seja, não causou nenhum prejuízo para as espécies envolvidas. Para construção de matrizes PAM é necessária uma lista de mutações aceitáveis, geralmente obtidas a partir de um conjunto de seqüências altamente relacionadas. Em seguida, são calculadas as probabilidades de ocorrência de cada um dos 20 aminoácidos isoladamente, representada por p_i e também as freqüências de substituição de cada par de aminoácido representada por f_{ij} , ou seja, o número de vezes que a mutação do aminoácido i para o j foi observada. É importante notar, que o processo de mutação não segue um padrão unidirecional, sendo $f_{ij} = f_{ji}$. A freqüência f_i , dada pela fórmula 3.8, mede o número de mutações em que o aminoácido i é substituído por qualquer outro aminoácido, exceto por ele mesmo.

$$f_i = \sum_{i \neq j} f_{ij} \quad (3.8)$$

A *mutabilidade relativa* m_i do aminoácido i , dada pela fórmula 3.9, mede o grau de variabilidade desse aminoácido.

$$m_i = \frac{f_i}{100fp_i} \quad (3.9)$$

Onde f é o número total de aminoácidos envolvidos em qualquer mutação e 100 é um fator normalizador que indica uma substituição a cada 100 resíduos alinhados.

A medida M_{ij} que representa a probabilidade de substituição do aminoácido i pelo aminoácido j pode ser calculada como a probabilidade condicional de i ser substituído por j , dado que i sofreu uma mutação. Essa probabilidade é estimada pela fórmula 3.10.

$$M_{ij} = P(i \rightarrow j|i) = \frac{f_{ij}m_i}{f_i} \quad (3.10)$$

Finalmente cada elemento $S(i, j)$ da matriz PAM é calculado pela taxa log-odds (DURBIN, EDDY, et al., 1998), como mostrado pela fórmula (3.11).

$$S_{ij} = \log \left(\frac{M_{ij}}{p_j} \right) \quad (3.11)$$

3.5.2 Pseudo Contadores

A cada coluna t de um alinhamento múltiplo está associada um contador de resíduos n_i . Na análise de proteínas são encontrados um total de 20 contadores por coluna do alinhamento. O somatório de todos os contadores de cada coluna é representado por $N = \sum_i n_i$. A probabilidade de cada aminoácido presente na coluna t pode ser calculada pela fórmula 3.12.

$$p_i = \frac{n_i}{N} \quad (3.12)$$

Esta probabilidade pode ser nula quando não há ocorrência do aminoácido i em dada coluna, o que é indesejável pois trata-se de sistemas probabilísticos. O método mais simples de garantir que nenhuma probabilidade seja estimada com o valor zero, é adicionar uma constante z pequena e positiva a cada contador. Este método é conhecido como *zero-offset* (TATUSOV, 1994), cada novo contador de resíduo \hat{n}_i é obtido pela fórmula 3.13.

$$\hat{n}_i = n_i + z \quad (3.13)$$

O método *pseudo contadores* (KARPLUS, 1995) é uma generalização do método *zero-offset*. A proposta é produzir distribuições mais razoáveis quando n_i tende à zero. Ao contrário de adotar uma única constante z , um conjunto de constantes z_i diferentes são adotadas, uma para cada aminoácido. O pseudo contador associado ao aminoácido i é dado por $\hat{n}_i = n_i + z_i$. Conseqüentemente a probabilidade estimada p_i é definida pela fórmula 3.14.

$$\hat{p}_i = \frac{n_i + z_i}{N + Z} \quad (3.14)$$

Onde cada constante z_i é definida experimentalmente (KARPLUS, 1995) a partir da base de dados BLOCKS, e $Z = \sum_i z_i$.

3.5.3 Misturas de Dirichlet

Misturas de Dirichlet (SJOLANDER, KARPLUS, et al., 1996) é uma generalização do método pseudo contadores. De fato, pseudo contadores podem ser visto como um caso particular de misturas de Dirichlet, com apenas um componente. Misturas de Dirichlet é um método que combina várias distribuições Dirichlet (SANTNER, DUFFY, 1989; BERGER, 1985), para estimar a probabilidade dos aminoácidos na ausência de dados reais. Cada distribuição é uma *componente da mistura*, e possui um *coeficiente* associado, representado por q . Este coeficiente é uma medida de probabilidade que expressa a importância de cada componente.

Dirichlet é uma distribuição discreta, baseada na distribuição multinomial (DEGROOT, 1987). A função densidade da distribuição Dirichlet é dada pela fórmula (3.15). Essa função é definida sobre o conjunto de todas as probabilidades de emissão, representadas pelo vetor \vec{p}_i , onde ($p_i \geq 0$ e $\sum_i p_i = 1$). Cada vetor representa uma possível distribuição de probabilidades sobre o alfabeto de aminoácidos.

$$f(\vec{p}) = \frac{\prod_i p_i^{\alpha_i - 1}}{Z} \quad (3.15)$$

A função densidade da distribuição Dirichlet possui parâmetros $\vec{\alpha} = \alpha_1, \dots, \alpha_n$, onde n é o tamanho do alfabeto e $\alpha_i \geq 0$. Z é uma constante ou fator de normalização que torna a soma de f igual a um. Este fator pode ser definido em termos da função gamma (DEGROOT, 1987).

Uma mistura de Dirichlet com k componentes é definida somando cada função de densidade da distribuição Dirichlet simples, apresentada em 3.15, multiplicada pelos coeficientes de mistura q_1, \dots, q_k . Desta forma, a função densidade da mistura de Dirichlet é dada pela fórmula 3.16.

$$f = q_1 f_1 + \dots + q_k f_k \quad (3.16)$$

Os coeficientes q_1, \dots, q_k são números positivos cuja soma é igual a um. Os parâmetros de uma mistura de Dirichlet são representados por $\Theta = \vec{\alpha}_1, \dots, \vec{\alpha}_k, q_1, \dots, q_k$.

Quando $k = 1$, $\Theta = \vec{\alpha}_1$.

Estimando Probabilidades Através de Misturas de Dirichlet

Dada uma coluna t , em um alinhamento múltiplo, pode-se combinar informações a priori com contadores de aminoácidos observados, para estimar a probabilidade \hat{p}_i de ocorrência de cada aminoácido na coluna t . Estas estimativas $\hat{p}_1 \cdots \hat{p}_{20}$ diferem das observações reais definidas pela fórmula 3.12 e deverão ser mais eficientes na ausência de dados, ou na presença de amostras pouco representativas.

Neste contexto, a probabilidade estimada \hat{p}_i do aminoácido i , dado uma função de densidade Dirichlet com parâmetros Θ e contadores reais \vec{n} é definida pela fórmula 3.17.

$$\hat{p}_i = P(i|\Theta, \vec{n}) \quad (3.17)$$

A probabilidade estimada \hat{p}_i é calculada somando todas as contribuições de cada componente k da mistura, dada pela fórmula 3.18.

$$\hat{p}_i = \sum_l^k P(l|n_i) \frac{n_i + \alpha_i^l}{\sum_j (n_j + \alpha_j^l)} \quad (3.18)$$

Onde $P(l|n_i)$ é calculado usando o teorema de Bayes, de acordo com a fórmula 3.19.

$$P(l|n_i) = \frac{q_l P(n_i|l)}{\sum_{j=1}^k q_j P(n_i|j)} \quad (3.19)$$

Onde q_l é o coeficiente da mistura para o componente l e $P(n_i|l)$ é a probabilidade a priori do símbolo i dado o componente l da mistura. Para maiores detalhes sobre como calcular $P(n_i|l)$, consulte (SJÖLANDER, 1997)

Estimando os Parâmetros da Mistura

Os parâmetros Θ das misturas de Dirichlet são divididos em dois conjuntos. Os parâmetros $\vec{\alpha}$ para cada componente e os parâmetros q , chamados de coeficiente da

mistura. Esses dois conjuntos de parâmetros precisam ser devidamente estimados, a partir de dados reais que representem o universo do problema, para garantir sua correta aplicabilidade. Qualquer algoritmo de otimização de funções contínuas pode ser aplicado à estimativa dos parâmetros da Dirichlet.

3.6 Associando Pesos às Seqüências

O desempenho de pHMMs, para a detecção de homólogos, poder ser melhorado diferenciando os pesos das seqüências do alinhamento múltiplo, dado com entrada (S. HENIKOFF, 1994; THOMPSON, GIBSON, 1994a; THOMPSON, GIBSON, 1994b). Este procedimento tem o propósito de diminuir a redundância, atribuindo pesos menos significativos à seqüências pouco representativas. O objetivo desta dissertação é propor um novo método de associação de pesos às seqüências, baseado em informações de estruturas de proteínas. Para isso é necessário compreender as estratégias existentes. A maioria pode ser dividida em dois grandes grupos. No primeiro grupo estão presentes os métodos que utilizam informações evolutivas na distribuição dos pesos entre as seqüências, descritos nas sessões 3.6.1 e 3.6.2. Os outros métodos são baseados em distâncias entre pares de seqüências, abordados pelas sessões 3.6.3 e 3.6.4.

3.6.1 Método da Simples Pontuação

Essa abordagem é baseada na construção de uma árvore evolutiva, conhecida como árvore filogenética (NEI, KUMAR, 2000), que organiza as seqüências de forma hierárquica considerando suas divergências a nível de resíduos. A figura 3.13 apresenta uma árvore filogenética ilustrativa. Nesta árvore o tamanho dos ramos indica o tempo de divergência entre as seqüências. Existem vários métodos para estimar árvores filogenéticas a partir de um conjunto de seqüências relacionadas. Entre os mais populares estão: os métodos baseados em matrizes de distâncias (BRYANT, MOULTON, 2004; DESPER, GASCUEL, 2004; KUMAR, TAMURA, et al., 2004), parcimônia (SPENCER, SUSKO, et al., 2005; CONANT, LEWIS, 2001), máxima verossimilhança (SIEPEL, HAUSSLER, 2004; CHOR, HENDY, et al., 2000) e métodos bayesianos (LARGET, SIMON, 2005; J. HUELSENBECK, 2001; YANG, RAN-

NALA, 1997).

Após a construção da árvore, por um destes métodos, uma abordagem bem simples para atribuição de pesos às seqüências foi descrita em (THOMPSON, GIBSON, 1994b). Trata-se da aplicação de um voltagem V a raiz da árvore, de forma que essa voltagem gere uma corrente I , que será distribuída a todos os nós de maneira proporcional ao comprimento de cada ramo. Este método considera a árvore filogenética como um circuito elétrico e aplica a lei Kirchhoff ou lei de conservação de corrente elétrica. Cada nó n da árvore possui uma corrente I_n e uma voltagem V_n , como mostra a figura 3.13. Considerando que a resistência do circuito é igual ao valor t dos ramos, as seguintes equações podem ser escritas $V_5 = 2I_1 = 2I_2$, $V_6 = 2I_1 + 3(I_1 + I_2) = 5I_3$, $V_7 = 8I_4 = 5I_3 + 3(I_1 + I_2 + I_3)$. Se um peso x é distribuído entre as seqüências e as correntes representam cada peso, então é correto afirmar que $I_1 + I_2 + I_3 + I_4 = x$. Com este sistema de equações é possível calcular o valor de cada corrente I_n e conseqüentemente o peso associado a cada seqüência.

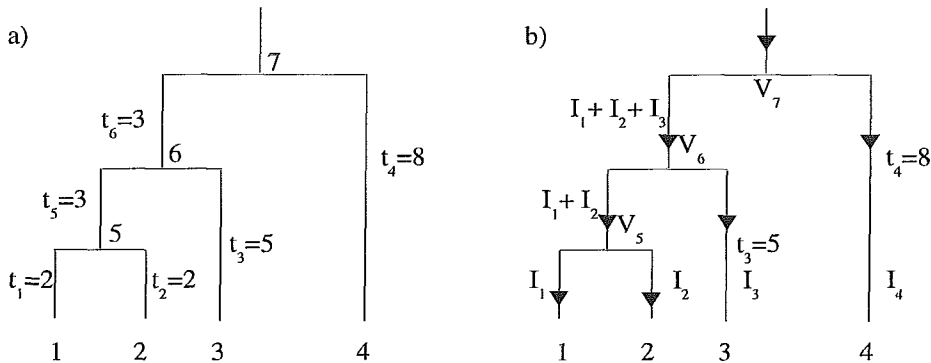


Figura 3.13: Árvore filogenética

3.6.2 Método de Gerstein, Sonnhammer e Chothia

Esse método foi proposto por (GERSTEIN, SONNHAMMER, et al., 1994). Esse algoritmo é adotado como método padrão do pacote HMMER. Trata-se de um algoritmo recursivo, também baseado em árvore filogenética, que distribuí os pesos entre as seqüências percorrendo a árvore das folhas até a raiz. A cada seqüência

é atribuído um peso inicial, cujo valor é igual ao comprimento t do ramo imediatamente acima da seqüência. Supondo que um nó n qualquer tenha sido visitado, então o comprimento t , desse nó, deve ser distribuído entre todas as seqüências dos níveis abaixo de n . Sendo assim, cada seqüência terá seu peso w_i incrementado por um fator Δw_i , dado pela fórmula 3.20.

$$\Delta w_i = t_n \frac{w_i}{\sum_k \text{folhas abaixo de } n} w_k \quad (3.20)$$

O fator de incremento é calculado recursivamente para cada seqüência, até que a raiz da árvore seja atingida. O peso de cada seqüência possuirá uma contribuição referente a cada nó hierarquicamente superior. De acordo com este método os pesos das seqüências da figura 3.13 são calculados da seguinte forma: inicialmente os pesos recebem os valores, $w_1 = w_2 = 2$, $w_3 = 5$ e $w_4 = 8$. O fator de incremento proveniente do nó cinco é calculado para as seqüências um e dois, de acordo com a fórmula 3.20, ou seja, para os pesos $w_1 = w_2$ o valor de incremento é $\Delta w_1 = \Delta w_2 = 2 + 3 * 2 / (2 + 2)$. Sendo assim $w_1 = w_2 = 3.5$. Para o nó seis que influencia as seqüências um, dois e três o fator de incremento é calculado da mesma forma, sendo $w_1 = w_2 = 3.5 + 3 * 3.5 / (3.5 + 3.5 + 5)$ e $w_3 = 5 + 3 * 5 / (3.5 + 3.5 + 5)$, esse processo é repetido até que o topo da árvore seja atingido.

3.6.3 Voronoi

A filosofia do método Voronoi (SIBBALD, ARGOS, 1990) assume a existência de um universo de pseudo seqüências, extraídas de variações de resíduos de seqüências verdadeiras. Essa abordagem é baseada nas distâncias entre pares de seqüências (VINGRON, ARGOS, 1900). Cada seqüência verdadeira é comparada com todos os membros do espaço, que abrange tanto seqüências verdadeiras como falsas. Essas comparações par a par geram uma matriz de distâncias, onde cada elemento M_{ij} contem o número de diferenças entre os pares de seqüências i e j . O domínio de i é representado por todas as seqüências do espaço e o domínio de j apenas por seqüências verdadeiras. O peso de cada seqüência real é obtido através de um sistema de votação. A coluna da matriz M que contem a menor distância em relação

as outras recebe um voto, quando há empate o voto é dividido igualmente, o peso final da seqüência j é calculado somando todos os votos obtidos por ela. A tabela 3.1 ilustra o esquema de Voronoi apresentando quatro seqüências reais, das quais foram obtidas 14 pseudo seqüências. A última linha da tabela mostra os pesos normalizados atribuídos a cada seqüência verdadeira.

Seqüência Reais	GYVGS	GFDGF	GYDGF	GYQGG
GYVGS	0 (1)	3	2	2
GFDGF	3	0 (1)	1	3
GYDGF	2	1	0 (1)	2
GYQGG	2	3	2	0 (1)
Pseudo seqüências				
GYVGF	1 (1/2)	2	1 (1/2)	2
GYVGG	1 (1/2)	3	2	1 (1/2)
GYDGS	1 (1/2)	2	1 (1/2)	2
GYDGG	2	2	1 (1/2)	1 (1/2)
GYQGS	1 (1/2)	3	2	1 (1/2)
GYQGF	2	2	1 (1/2)	1 (1/2)
GFVGS	1 (1)	2	3	3
GFVGF	2	1 (1)	2	3
GFVGG	2 (1/3)	2 (1/3)	3	2 (1/3)
GFDGS	2	1 (1)	2	3
GFDGG	3	1 (1)	2	2
GFQGS	2 (1/3)	2 (1/3)	3	2 (1/3)
GFQGF	3	1 (1)	2	2
GFQGG	3	2	3	1 (1)
Total votos	(14/3)	(17/3)	(9/3)	(14/3)
Pesos Normalizados	0.259	0.315	0.167	0.259

Tabela 3.1: Esquema de Voronoi.

O método de Voronoi é um dos métodos mais eficientes no processo de atribuição de pesos às seqüências, pois é baseado na menor distância entre seqüências reais e todas as suas possíveis variações. No entanto, quando o espaço de seqüências possui uma alta dimensionalidade, a aplicação desse método torna-se computacionalmente cara, pois o tempo de processamento é exponencial (THOMPSON, GIBSON, 1994b).

3.6.4 Máxima Entropia

A medida de uniformidade de uma distribuição pode ser medida através da *entropia* (DURBIN, EDDY, et al., 1998). Sob este conceito, Krogh e Mitchison (KROGH,

MITCHISON, 1995) propuseram um método que determina os pesos das seqüências, maximizando a entropia da distribuição dos resíduos para cada coluna do alinhamento.

$$\sum_i \sum_a p_{ia} \log(p_{ia}) + \lambda \sum_k w_k \quad (3.21)$$

A equação 3.21 apresenta a função que deve ser maximizada na seleção dos pesos. Onde $p_{ia} \log(p_{ia})$ é a medida de entropia relacionada a distribuição de cada resíduo, sendo p_{ia} a probabilidade de ocorrência do resíduo a na posição i . O fator $\lambda \sum_k w_k$ é o multiplicador de Lagrange, que torna a soma dos pesos igual a um.

3.7 Buscando com *Profile* HMM

Após a definição da arquitetura e o aprendizado dos parâmetros, o pHMM é utilizado para realizar inferências sobre seqüências genômicas. Uma nova seqüência ou um banco de dados de seqüências é comparado com um pHMM de determinada família, com o intuito de identificar potenciais membros dessa família. O mecanismo de identificação consiste em determinar a probabilidade de cada amostra ter sido gerada a partir do modelo. Essa sessão discute como a essa probabilidade é determinada através máxima verossimilhança, abordada na sessão 3.7.1, e descreve os dois principais algoritmos utilizado na inferência de pHMMs: o algoritmos *Forward*, discutido na sessão 3.7.2, e o algoritmo *Viterbi*, descrito na sessão 3.7.3.

3.7.1 Verossimilhança

Seja uma seqüência qualquer $S_N = (x_1, \dots, x_N)$ e um pHMM $M(w)$, com parâmetros w . Um caminho π em $M(w)$ será uma seqüência de consecutivos estados, iniciando pelo estado inicial e terminando no estado final, tendo produzido a seqüência S_N . A *verossimilhança* (KARLIN, ALTSCHUL, 1990) de uma seqüência S_N descreve a probabilidade de S_N ter sido gerada pelo modelo $M(w)$ através do caminho π . A verossimilhança de uma seqüência é expressada através da fórmula 3.22.

$$P(S_N|M(w)) = \sum_{\pi} P(S_N, \pi|M(w)) \quad (3.22)$$

A máxima verossimilhança de S_N pode ser encontrada desde que o melhor caminho π^* seja determinado. Porém, o número de caminhos na arquitetura, definida na sessão 3.3, torna-se exponencial na presença de muitos estados, tornando impraticável encontrar o melhor caminho π^* exaustivamente. Todavia, existem meios mais eficientes para calcular a máxima verossimilhança das seqüências avaliadas pelo modelo. Métodos iterativos, de propagação através da arquitetura, evitam a necessidade de busca por todos os caminhos existentes.

3.7.2 Algoritmo Forward

O algoritmo forward (BALDI, BRUNAK, 2001) é um método de programação dinâmica (BERTSEKAS, 1995), que visa determinar a máxima verossimilhança de uma seqüência S_N através do modelo $M(w)$, baseando-se nas probabilidades parciais, considerando sub-seqüências $S_i = (x_1, \dots, x_i)$. Essas probabilidades são denotadas pela variável $f_j(i)$, fórmula 3.23, que expressa a soma das probabilidades de todos os caminhos até o estado j tendo observado S_i .

$$f_j(i) = P(x_1, \dots, x_i, \pi_i = j \mid M(w)) \quad (3.23)$$

Dessa forma, a máxima verossimilhança pode ser calculada recursivamente através da fórmula 3.24, onde $e_j(x_{i+1})$ é a probabilidade de emissão do símbolo $(i+1)$ pelo estado j e a_{kj} é a probabilidade de transição de cada estado k (antecessor) para j .

$$f_j(i+1) = e_j(x_{i+1}) \sum_k f_k(i) a_{kj} \quad (3.24)$$

O algoritmo completo é descrito no algoritmo 2. Para a arquitetura do pHMM, figura 3.10, k varia entre os estados *match*, *insert* e *delete*. Para os estados de *delete* a probabilidade de emissão $e_j(x_{i+1})$ é omitida por se tratar de um estado silencioso.

Algoritmo 2 Forward

Inicialização: $f_0(0) = 1, f_k(0) = 0$ para $k < 0$

Para $i = 1$ até N

$$f_j(i) = e_j(x_i) \sum_k f_k(i-1) a_{kj}$$

Fim do Para

Finalização: $P(S) = \sum_k f_k(N) a_{k0}$

3.7.3 Algoritmo Viterbi

O algoritmo *Viterbi* (BALDI, BRUNAK, 2001) determina a máxima verossimilhança de uma sub-sequência S_i , encontrando o caminho mais provável π^* capaz de gerar S_i , sendo $\pi^* = \operatorname{argmax}_{\pi} P(x, \pi \mid M(w))$. A variável $v_j(i)$, dada pela fórmula 3.25, representa a probabilidade parcial do caminho π^* iniciando no estado inicial e terminando no estado j tendo observado S_i .

$$v_j(i) = e_j(x_i) \max_k (v_k(i-1) a_{kj}) \quad (3.25)$$

O caminho mais provável π^* pode ser encontrado recursivamente, como mostrado no algoritmo 3. Para isso, no tempo t é necessário manter um ponteiro para o estado que produziu a maior probabilidade no tempo $t-1$. Dessa forma, o algoritmo Viterbi além de calcular a máxima verossimilhança de uma sequência S_N , também determina o conjunto de estados que geraram a sequência S_N .

Algoritmo 3 Viterbi

Inicialização: $v_0(0) = 0, v_k(0) = 0$ para $k < 0$

Para $i = 1$ até N

$$v_j(i) = e_j(x_i) \max_k (v_k(i-1) a_{kj})$$

$$ptr_i(j) = \operatorname{argmax}_k (v_k(i-1) a_{kj})$$

Fim do Para

Finalização:

$$P(S, \pi) = \max_k (v_k(N) a_{k0})$$

$$\pi_N^* = \operatorname{argmax}_k (v_k(N) a_{k0})$$

3.7.4 Calculando *E-values*

Inferências sobre pHMMs podem ser realizadas através dos algoritmos *forward* e *Viterbi*, que relacionam uma pontuação (*score*) a cada sequência S_N a partir do modelo $M(w)$. Para evitar problemas de *underflow*, geralmente trabalha-se com o logaritmo das probabilidades. Nesse caso, o *score* pode ser calculado aplicando logaritmo a verossimilhança, sendo chamada de *log-likelihood-score* ou (*LL-score*). No entanto a *LL-score* é altamente dependente do tamanho das sequências. Sendo assim, a uma pequena sequência aleatória pode ser atribuído um melhor *score* em relação a uma longa sequência que pertence a família modelada por $M(w)$. Para

solucionar esse problema o *score* é calculado através da razão entre *LL-score* e a probabilidade da seqüências S_N ter sido gerada aleatoriamente, por um modelo randômico ou nulo, os parâmetros do modelo nulo são calculados a priori. Essa razão é denominada *log-odds* (DURBIN, EDDY, et al., 1998), fórmula 3.26, onde R representa o modelo nulo.

$$score(S) = \log_2 \frac{P(S_N|M(w))}{P(S_N|R)} \quad (3.26)$$

Entretanto, o cálculo do *score* através de *log-odds* não é significativo quando diversos pHMMs são avaliados. Por exemplo, sejam dois modelos $M_1(w_1)$ e $M_2(w_2)$, representando duas famílias de seqüências genômicas diferentes, S_N uma seqüência qualquer e s_1 o *log-odds* de S_N pelo modelo $M_1(w_1)$ e s_2 o *log-odds* de S_N pelo modelo $M_2(w_2)$. Se $M_1(w_1)$ possui menos estados do que $M_2(w_2)$ e $s_1 > s_2$, não é correto afirmar que S_N pertence a família modelada por $M_1(w_1)$. Pois como o modelo nulo é o mesmo, o cálculo de s_1 pode ter sido favorecido pelo tamanho do modelo $M_1(w_1)$. Para lidar com esse problema, um outro método de medida de significância é empregado. Esse método é conhecido por *P-value* ou (*probabilistic value*). O *P-value* para uma seqüência S_N com *log-odds* s é medido como a probabilidade de s ter sido obtido ao acaso, ou seja, a probabilidade da seqüência S ser considerada falso positiva. Para determinar o *P-value* para uma seqüência S_N é necessário determinar o número de vezes que s é menor que s_1, s_2, \dots, s_k para k seqüências aleatórias. Esse número pode ser encontrado analiticamente (BARRETT, HUGHEY, et al., 1997) ou empiricamente (GOLDSTEIN, WATERMAN, 1997). O método empirico é mais eficiente, e consiste em determinar dois parâmetros μ e λ através do cálculo do *log-odds-score* para k seqüências aleatórias. Esses parâmetros fazem parte de uma distribuição *extreme*. A função densidade dessa distribuição fornece os valores *P-value* para cada seqüência S_N . Para maiores detalhes sobre distribuição *extreme* e como calcular os seus parâmetros, consulte (EDDY, 1997).

Quando um pHMM é comparado com um banco de dados de dados de seqüências genômicas, a medida ideal de significância relacionada aos scores é o *E-value* (EDDY, 1997) ou (*extreme value*). O *E-value* relacionado a uma seqüência S_N de *score* s e

a um banco de dados de t seqüências é calculado através do número esperados de seqüências no banco de dados, que consistem de seqüências aleatórias, e possuem um score maior do que s .

3.8 Análise dos Resultados

Após a realização dos experimentos do capítulo 4 e do capítulo 5, os resultados foram analisados através de curvas ROC (FAWCETT, 2004; BECK, SHULTZ, 1986; METZ, 1978) e *Precision* e *Recall* (BILENKO, MOONEY, 2003; CRAVEN, SLATTERY, 2001; BUCKLAND, GEY, 1994). O objetivo desta sessão é explicar o significado das curvas utilizadas para análise de desempenho dos programas testados. A sessão 3.8.1 e 3.8.2 explicam o que são curvas ROC e *Precision* e *Recall*, respectivamente, e como essas curvas são interpretadas.

3.8.1 Curvas ROC

As curvas ROC são uma medida de desempenho baseada na relação de *falsos positivos* por *verdadeiros positivos*, variando os parâmetros que afetam essas taxas. Verdadeiros positivos são amostra classificadas corretamente, enquanto que falsos positivos são erros de classificação. As curvas ROC, deste trabalho, foram construídas variando o parâmetro *e-value*.

A figura 3.14 mostra um exemplo de duas curvas, cada uma representa o desempenho de um algoritmo diferente. O ideal seria que a curva permanecesse paralela ao eixo Y , pois nesse caso o método analisado não detectaria falsos positivos. O método com o melhor desempenho é aquele que em média detecta mais verdadeiros positivos considerando a mesma taxa de falsos positivos. Sendo assim, na figura 3.14 o melhor método é o algoritmo 1.

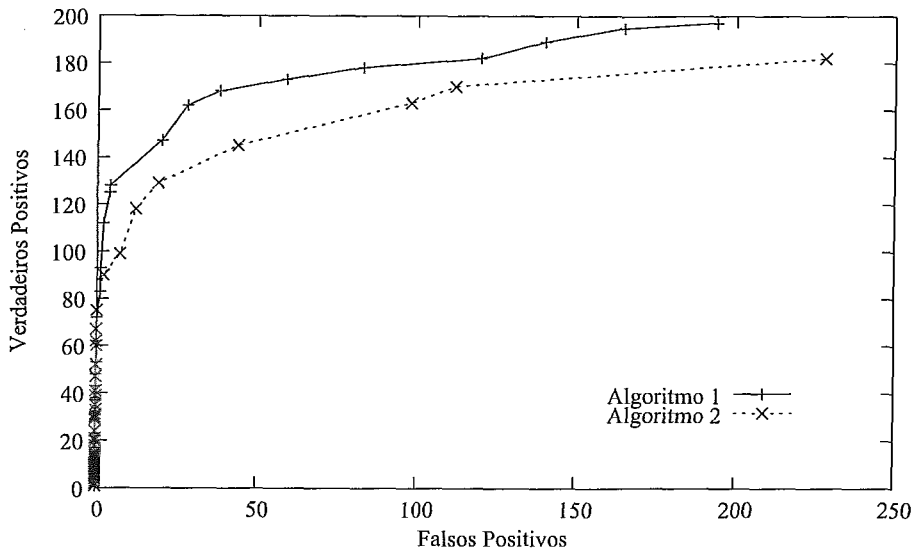


Figura 3.14: Exemplo de curvas ROC.

3.8.2 Curvas *Precision* e *Recall*

As curvas *precision* e *recall* também são curvas que avaliam o desempenho dos métodos, através da variação de um parâmetro. As curvas são determinadas através das fórmulas 3.27 e 3.28, respectivamente.

$$P = \frac{VP}{VP + FP} \quad (3.27)$$

$$R = \frac{VP}{VP + FN} \quad (3.28)$$

Onde VP representa o número de verdadeiros positivos, FP o número de falsos positivos e FN o número de *falsos negativos*. Os falsos negativos são amostras verdadeiras que não foram detectadas.

A figura 3.15 mostra um exemplo de duas curvas, cada uma representa o desempenho de um algoritmo diferente. A curva ideal é a curva em que todos os pontos coincidem com o ponto (1,1), pois nesse caso o método avaliado não detectou falsos positivos e não detectou falsos negativos. O método com o melhor desempenho é aquele em que a maioria dos pontos se aproximam do ponto (1,1). Sendo assim, na figura 3.14 o melhor método é o algoritmo 1.

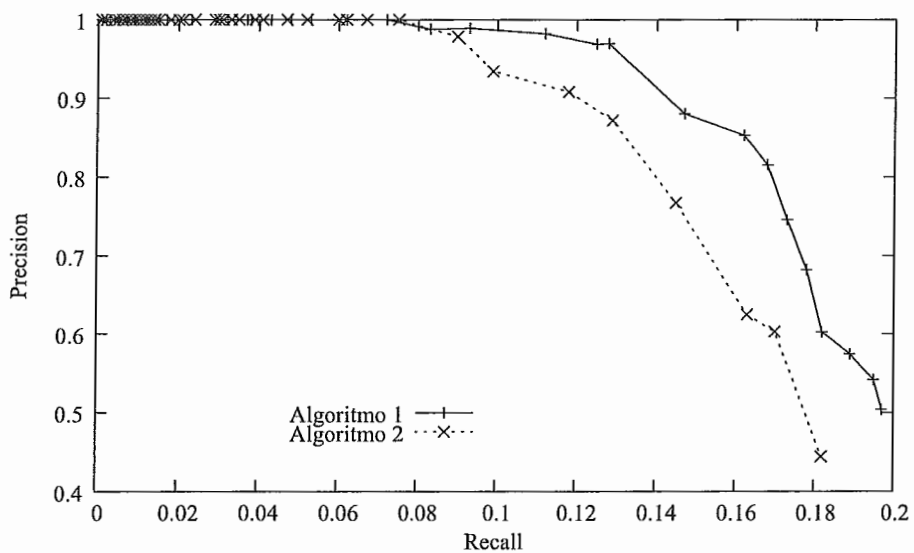


Figura 3.15: Exemplo de curvas *Precision e Recall*.

CAPÍTULO 4

Avaliação de *Profiles* HMMs

Os pacotes HMMER (EDDY, 1998) e SAM (HUGHEY, KROGH, 1996a) estão entre os principais programas empregados à detecção de homologies distantes. Trabalhos prévios realizaram testes e comparações com objetivo de determinar qual dos pacotes é mais eficiente em termos de sensibilidade, precisão e custo computacional. As comparações dos trabalhos (MADERA, GOUGH, 2002) e (GOUGH, 2002) testaram os pacotes apenas utilizando os parâmetros recomendados pelos autores, e aplicaram os testes apenas a duas famílias de proteínas. Por outro lado, o trabalho (WISTRAND, SONNHAMMER, 2005) abordou as principais diferenças entre as técnicas empregadas nas duas ferramentas, e implementaram duas melhorias nos algoritmos do pacote HMMER. A base de dados utilizada para os teste considerou amostras da base SCOP (ANDREEVA, HOWORTH, et al., 2004) e PFAM (BATEMAN, COIN, et al., 2004). O objetivo deste capítulo é avaliar os pacotes HMMER e SAM considerando alinhamentos primários e estruturais. O trabalho (JONES, BATEMAN, 2002) já realizou esse tipo de experimento e concluiu que o uso de alinhamentos estruturais não aumentavam a detecção de homólogos distantes. No entanto, esse trabalho não aplicou os testes aos dois principais programas de detecção de homólogos distantes. Além disso, nenhum dos trabalhos citados realizou experimentos empregando validação cruzada (HAYKIN, 2001) e testes de significância dos resultados através de *paired t-test* (MITCHELL, 1997), a tabela 4.1 mostra as diferenças entre os experimentos realizados e os novos experimentos propostos

por este trabalho.

	MADERA, GOUGH, 2002	JONES, BATEMAN, 2002	WISTRAND, SONNHAMMER, 2005	Nossos Experimentos
Comparação HMMER e SAM	X		X	X
SCOP		X	X	X
Melhorias HMMER			X	X
Alinhamentos estruturais		X		X
Validação cruzada				X
Paired t-test				X

Tabela 4.1: Experimentos realizados e nossos experimentos.

As próximas sessões deste capítulo estão divididas da seguinte forma: a sessão 4.1 descreve o funcionamento dos pacotes HMMER e SAM, sessão 4.2 explica como a base de dados SCOP, adotada nos experimentos, está organizada, a sessão 4.3 aborda as ferramentas de alinhamento utilizadas, a sessão 4.4 descreve a metodologia experimental empregada e a sessão 4.5 discute os resultados.

4.1 *Profiles* HMMs, HMMER e SAM

4.1.1 HMMER

O pacote HMMER foi desenvolvido pela universidade de Washington sob a responsabilidade de Sean Eddy, e provê um ambiente para análise de seqüências genômicas. Os programas são gratuitos regidos pela GNU (*General Public Licence*). Este trabalho utiliza a versão 2.3.2, atualizada em 2003. Como visto no capítulo 3, são necessárias três etapas para concluir a tarefa de detecção de homologias através de pHMMs: definição da arquitetura, sessão 3.3 página 37, aprendizado dos parâmetros do modelo, 3.4 página 42 e a busca por novos homólogos, 3.7 página 57. A arquitetura usada para a construção dos modelos é denominada *plan7* (EDDY, 2003), e possui sete transições entre os estados, sendo omitidas as transições entre os estados *insert* e *delete* ($I \rightarrow D, D \rightarrow I$). Os parâmetros do modelo são estimados

a partir de um conjunto de seqüências alinhadas, sendo necessário distinguir quais colunas do alinhamento formarão os estados de *match* e *insert*. HMMER permite uma seleção manual, heurística ou através do algoritmo MAP, que simultaneamente determina os estados de *match* e *insert* e maximiza a verossimilhança do modelo. Todas as estratégias para seleção de estados são descritas na sessão 3.4 página 42.

Os parâmetros de um pHMM são estimados combinando dados observados, providos pelo alinhamento, e informações a priori, tais como matrizes de substituição e misturas de Dirichlet, como discutido na sessão 3.5 página 46. HMMER adota, por padrão, misturas de Dirichlet com nove componentes para as probabilidades de emissão e apenas um componente para probabilidades de transição.

Algoritmos para pontuação de seqüências, como previamente discutido na sessão 3.6 página 53, são tradicionalmente utilizados para evitar a criação de modelos específicos, sem poder de generalização. O pacote HMMER utiliza, por padrão, o método de Gerstein, Sonnhammer e Chothia (GERSTEIN, SONNHAMMER, et al., 1994), descrito na sessão 3.6.2 página 54.

Após a criação do pHMM, uma etapa opcional, porém recomendada é a etapa de calibração dos *e-values*. Para isso HMMER utiliza um conjunto de seqüências aleatórias, obtidas a partir do modelo nulo, pontua essas seqüências através do algoritmo Viterbi, abordados na sessão 3.7.3 página 59, e determina os parâmetros da distribuição *extreme*. O mesmo algoritmo pode ser utilizado para inferir homologia a partir de um banco de dados de novas seqüências. Os *e-values* para as seqüências classificadas são calculados utilizando a distribuição *extreme*, sessão 3.7.4 página 59.

4.1.2 SAM

O pacote SAM foi desenvolvido na universidade da Califórnia, Santa Cruz. A distribuição é gratuita para uso acadêmico, porém os autores retêm o direito sobre os modelos produzido. O código fonte é privado, não sendo permitido modificações. A versão do pacote SAM adotada por esse trabalho foi a versão 3.1, atualizada em 2002.

O conjunto de programas do pacote SAM possuem algumas vantagens em relação ao pacote HMMER. Entre as principais encontra-se o script *target2k* (HUGHEY, KARPLUS, et al., 2003), que permite a criação de alinhamentos a partir de uma seqüência alvo e um banco de dados com possíveis homólogos. Os alinhamentos criados por esse script incorporam informações que auxiliam a distinção entre estados de *match* e *insert*. Na ausência dessas informações, SAM relaciona cada coluna do alinhamento a um estado de *match*, as probabilidades dos estados de *insert* são obtidas apenas através de informações a priori. Neste trabalho foram utilizados apenas alinhamentos providos por ferramentas externas, para não favorecer nenhum dos pacotes analisados.

As seqüências do conjunto de treinamento recebem um peso, de acordo com um algoritmo baseado na maximização da entropia, não publicado pelos autores. Por padrão, SAM faz uso de informações a priori, extraídas de uma mistura de Dirichlet com 20 componentes para as probabilidades de emissão e um componente para probabilidades de transição. Todas as distribuições Dirichlet, utilizadas pelos pacotes HMMER e SAM, estão disponíveis através do site indicado na referência (SJOLANDER, KARPLUS, et al., 1996).

A etapa de calibração dos *e-values* pode utilizar seqüências aleatórias, como no HMMER, geradas a partir de um modelo nulo, chamado *reverse null model* (KARPLUS, KARCHIN, et al., 2005), ou através de um banco de dados de seqüências, selecionado pelo usuário. A etapa de busca, por padrão faz uso do algoritmo *forward*, sessão 3.7.2 página 58, para determinar a probabilidade de novas seqüências serem geradas pelo modelo.

4.2 Classificação Estrutural de Proteínas (SCOP)

A base de dados SCOP (ANDREEVA, HOWORTH, et al., 2004) contem todas as proteínas de estrutura conhecida, cujas coordenadas foram depositadas na base PDB (HELEN, WESTBROOK, et al., 2000). É notável o número de estudos que utilizam a base de dados SCOP para avaliar o desempenho de métodos voltados para a de-

teção de homologias, (ESPADALER, 2005; WISTRAND, SONNHAMMER, 2005; SÖDING, 2005; ALEXANDROV, GERSTEIN, 2004; HOU, HSU, et al., 2004b; WISTRAND, SONNHAMMER, 2005). SCOP é uma base de dados que classifica todos os domínios de proteínas de estrutura conhecida, dentro de uma ordem hierárquica obedecendo a quatro níveis: família, super-família, dobramentos ou conformação e classe.

- *Família*, agrupa proteínas seguindo dois critérios que sugerem a mesma origem evolutiva. Primeiro, a identidade entre as proteínas deve ser igual ou superior a 30%. Segundo, as proteínas podem possuir baixa identidade (inferior a 30%), desde que suas funções e estruturas sejam similares.
- *Super-família*, agrupa famílias com baixa identidade, porém as coordenadas espaciais ou características estruturais sugerem uma provável origem ancestral comum.
- Super-famílias são agrupadas dentro do mesmo nível de *dobrimento*, quando a maior parte das estruturas secundárias, que formam as moléculas das proteínas, estão arranjadas da mesma forma, compartilhando ligações químicas e conexões topológicas.
- *Classe*, agrupa dobramentos. A maioria dos dobramentos são classificados em uma das cinco classes estruturais, de acordo com a predominância dos elementos de estrutura secundária. As classes são:
 1. *Alpha*, as estruturas são essencialmente formadas por elementos alpha-hélices;
 2. *Beta*, a maioria dos elementos estruturais são compostos por beta-sheets;
 3. *Alpha* e *Beta*, os elementos alpha-hélices e folhas betas são predominantes na estrutura da proteína e ocorrem de forma intercalada;
 4. *Alpha* mais *Beta*, similar a classe (3), porém os elementos ocorrem separadamente na estrutura;
 5. *Múltiplos domínios*, essa classe agrupa proteínas com diferentes dobramentos.

4.3 Alinhamentos Múltiplos

Alinhamento múltiplo de seqüências genômicas é uma das mais antigas aplicações da bioinformática. O alinhamento entre duas ou mais seqüências expressa o grau de conservação e os relacionamentos evolutivos entre as seqüências envolvidas. A qualidade dos alinhamentos está diretamente relacionada ao desempenho de pHMMs. Dessa forma, a escolha de um programa de alinhamento torna-se crucial, para o sucesso dos pacotes HMMs. Ferramentas como CLUSTALW (THOMPSON, GIBSON, 1994a), TCOFFEE (NOTREDAME, HIGGINS, et al., 2000), MAFFT (KAZUTAKA, KAZUHARU, et al., 2002), ALIGN-M (WALLE, I., et al., 2004) e MUSCLE (EDGAR, 2004) realizam alinhamentos primários, que fazem uso apenas das informações contidas nas seqüências genômicas. No entanto, através de processos de mutação as seqüências podem divergir, dificultando a tarefa desse tipo de alinhamento.

Por outro lado, ferramentas de alinhamento tais como CE-MC (GUDA, LU, et al., 2004), 3DCOFFEE (SULLIVAN, SUHRE, et al., 2004), MULTIPROT (LUPYAN, LEO-MACIAS, 2005) e MAMMOTH (ATTWOOD, BRADLEY, et al., 2005), entre outras, fazem uso de informações estruturais, principalmente das coordenadas tridimensionais de cada átomo que compõem uma proteína, como discutido na sessão 2.3.3 página 28. A estrutura terciária das proteínas possuem uma forte tendência a ser mais conservada do que a seqüência de aminoácidos. Por exemplo, a figura 4.1 mostra as estrutura de duas proteínas da mesma super-família, que apesar de possuírem menos de 16% de identidade seqüencial, compartilham muitas características estruturais, refletindo uma origem ancestral comum.

Alinhamentos estruturais identificam regiões com similaridade espacial e alinham os resíduos dessas regiões, estendendo o alinhamento para o restante dos resíduos. O objetivo dos alinhamentos estruturais é prover a melhor sobreposição entre os átomos das proteínas. Para tanto, a maioria dos programas consideram apenas a cadeia principal, também chamada de *backbone*, desprezando os átomos das cadeias laterais dos aminoácidos.

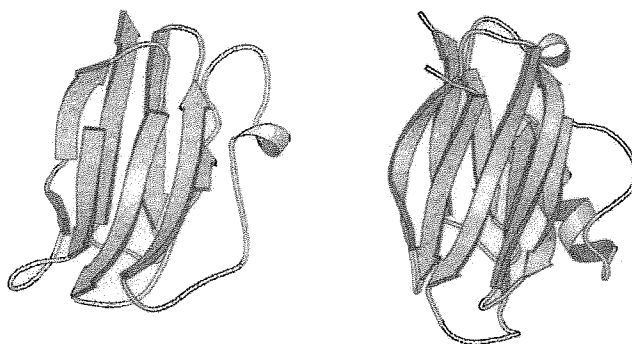


Figura 4.1: Proteínas da mesma super-família que apresentam menos de 16% de identidade.

Para a realização dos experimentos deste trabalho foram adotadas duas ferramentas para alinhamentos primários, CLUSTALW e T-COFFEE, e duas ferramentas para alinhamentos estruturais 3DCOFFE e MAMMOTH, que alinham estruturalmente utilizando apenas os carbonos alpha. Os programas CLUSTALW e T-COFFEE foram selecionados por serem muito estáveis, e por serem comumente utilizados. MAMMOTH por ser uma das mais recentes ferramentas publicadas, e 3DCOFFE por ser uma ferramenta mista, ou seja, os alinhamentos produzidos são baseados em similaridades estruturais, obtidas a partir da sobreposição de pares de estruturas.

4.4 Metodologia Experimental

Os modelos foram construídos utilizando o programa *hmmbuild*, para HMMER e *modelfromalign*, para SAM. Todos os modelos foram calibrados utilizando os parâmetros padrões. O processo de busca por novos homólogos foi realizado através dos programas *hmmsearch* e *hmmscore*.

Os procedimentos dos testes são similares aos adotados pelo trabalho THMM (QIAN, GOLDSTEIN, 2004). A base de dados SCOP foi particionada, por nível de super-família, figura 4.2-a. Foram selecionadas as super-famílias com mais de 20 seqüências e pelo menos duas famílias, totalizando 48 super-famílias, mostradas pela tabela 4.2, e 291 famílias. Para uma dada super-família x com n famílias, foram construídos n pHMMs, tomando $n - 1$ famílias para o conjunto de treina-

mento, figura 4.2-b. Para testar cada pHMM o conjunto de teste é formado pelas seqüências da super-família que não participaram do treinamento e todas as demais seqüências da base dados, ou seja, as seqüências de outras super-famílias, figura 4.2-c. A família que não participa do treinamento, mede a capacidade de detecção de homólogos distantes. A validação cruzada foi utilizada permitindo que cada família fosse excluída do treinamento uma vez (*leave one family out*). Cada teste baseou-se em alinhamentos produzidos pelas ferramentas mencionadas na sessão 4.3. As coordenadas tridimensionais necessárias aos alinhamentos estruturais foram providas pela base de dados PDB.

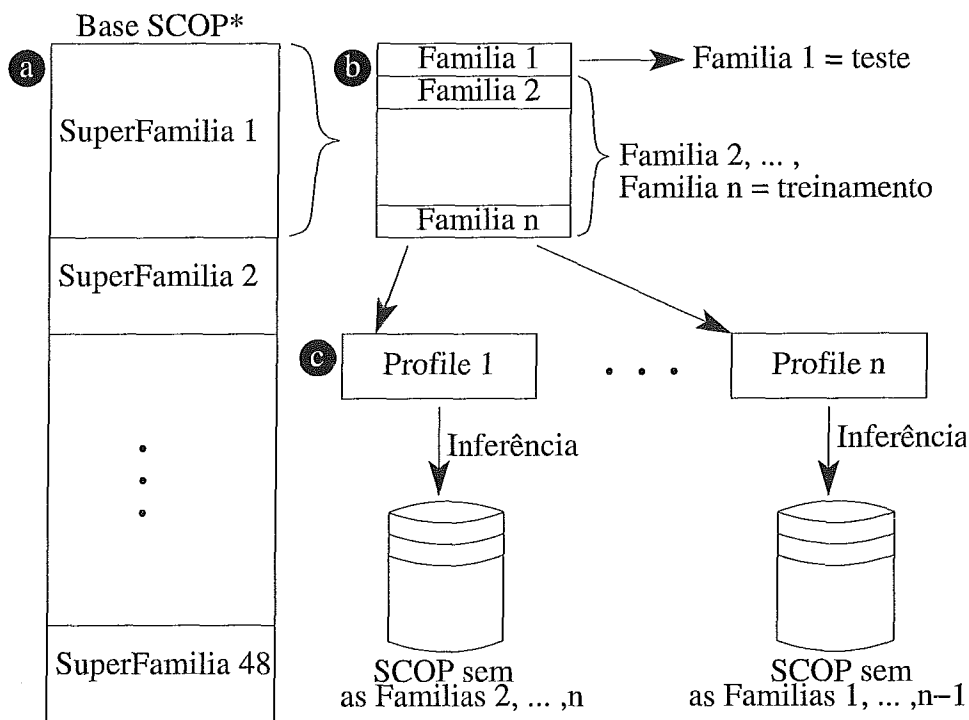


Figura 4.2: Divisão da base de dados e metodologia experimental.

a.1.1.	a.3.1	a.4.5.	a.25.1	a.26.1.	a.39.1.	b.1.1.	b.1.18.
b.6.1.	b.121.4	b.18.1.	b.29.1.	b.36.1.	b.40.4.	b.47.1.	b.55.1.
b.60.1.	b.71.1.	b.82.1.	c.1.8.	c.1.10.	c.2.1.	c.3.1.	c.23.1.
c.26.1.	c.36.1.	c.37.1.	c.47.1.	c.52.1.	c.55.1.	c.55.3.	c.66.1.
c.67.1.	c.69.1.	c.94.1.	d.3.1.	d.14.1.	d.58.7.	d.92.1.	d.108.1.
d.144.1.	d.153.1.	d.169.1.	g.3.6.	g.3.7.	g.3.11.	g.37.1.	g.39.1.

Tabela 4.2: Lista de super-famílias.

A base de dados SCOP, versão 1.67 (fevereiro de 2005), possui 6600 seqüências. Nenhuma das seqüências da base possui mais de 40% de identidade em relação as demais. O conjunto de exemplos é formado por todas as 6600 seqüências. Dessas seqüências, 1160 participaram do treinamento de pHMMs.

Para testar cada *profile*, o conjunto de teste foi formado por todas as seqüências da base SCOP, exceto as seqüências utilizadas no treinamento do *profile*. Os resultados foram analisados através de curvas ROC, discutidas na sessão 3.8 página 61 e *Precision* e *Recall*, abordadas na sessão 3.8.2 página 62. Para a construção das curvas o parâmetro *e-value* sofreu uma variação entre o intervalo [0, 10]. Embora o valor um seja habitualmente adotado como valor limite para *e-value*, para confiança nos dados, o valor dez foi escolhido para verificar o comportamentos dos pacotes em condições extremas. Para avaliar a significância dos resultados foi utilizado *paired t-test* considerando os resultados como significativos para *p* menor que 0,02.

4.5 Testes e Resultados

Em um primeiro passo, HMMER e SAM foram comparados utilizando os parâmetros recomendados pelos autores, sessão 4.1. Uma análise da qualidade dos alinhamentos tornou-se necessária para justificar o comportamento dos pHMMs, mediante aos alinhamentos produzidos pelas diferentes ferramentas.

A qualidade foi baseada na medida de significância entre pares de resíduos alinhados e penalidades para buracos. A cada coluna do alinhamento foi atribuído uma pontuação S_j , calculada através da fórmula 4.1, onde $P_{i,k}$ representa o peso atribuído ao alinhamento dos resíduos i, k . Se os resíduos são iguais, $P_{i,k} = 2$, se são diferentes, $P_{i,k} = 1$, se apenas um dos resíduos é um buraco, $P_{i,k} = 0$ e se os dois resíduos são buracos $P_{i,k} = -1$. Esses pesos foram retirados do livro (SETUBAL, MEIDANIS, 1997).

$$S_j = \sum_{i=1}^{N-1} \frac{\sum_{k=i}^{N-1} P_{i,k+1}}{N-i} \quad (4.1)$$

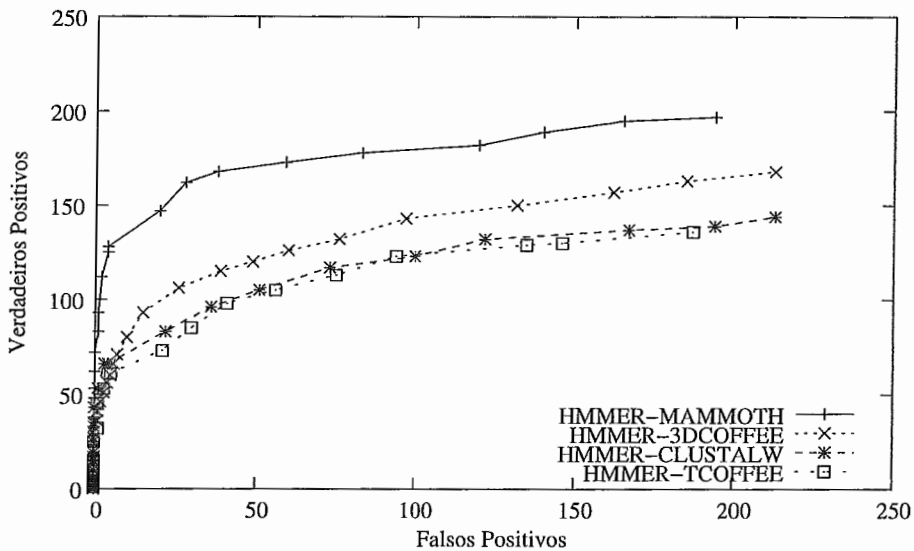


Figura 4.4: Análise de desempenho para o HMMER, através de curvas ROC, considerando todas as ferramentas de alinhamento.

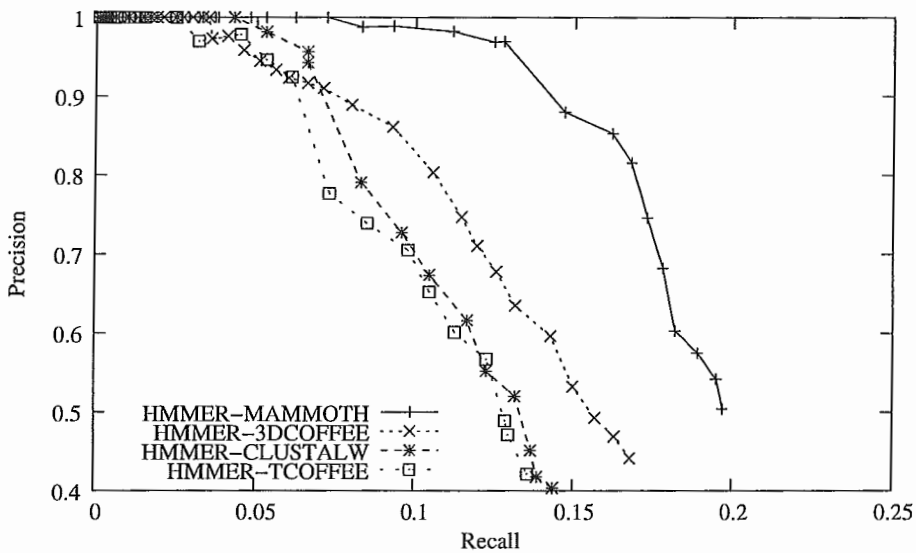


Figura 4.5: Análise de desempenho para o HMMER, através de *Precision* e *Recall*, considerando todas as ferramentas de alinhamento.

	MAMMOTH	3DCOFFEE	CLUSTALW
TCOFFEE	0,0002128 (sim)	0,0353362 (não)	0,0673345 (não)
CLUSTALW	0,0004271 (sim)	0,0329817 (não)	
3DCOFFEE	0,0000004 (sim)		

Tabela 4.3: Resultado do *paired t-test* e significância estatística entre os testes realizados para o pacote HMMER, considerando todas as ferramentas de alinhamento.

A figura 4.6 e 4.7 apresentam os resultados para o SAM, utilizando como método de análise curvas ROC e *Precision* e *Recall*, respectivamente. De acordo com os resultados, os modelos criados a partir dos alinhamentos estruturais alcançaram maiores taxas de verdadeiros positivos para a mesma taxa de falsos positivos, quando comparados aos modelos produzidos pelos alinhamentos primários, porém os resultados não são significativos, como mostra a tabela 4.4.

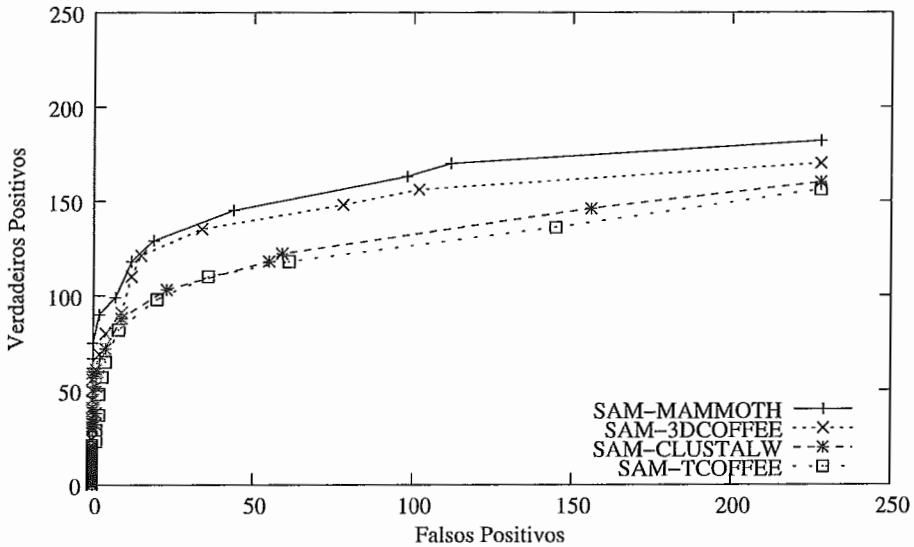


Figura 4.6: Análise de desempenho para o SAM, através de curvas ROC, considerando todas as ferramentas de alinhamento.

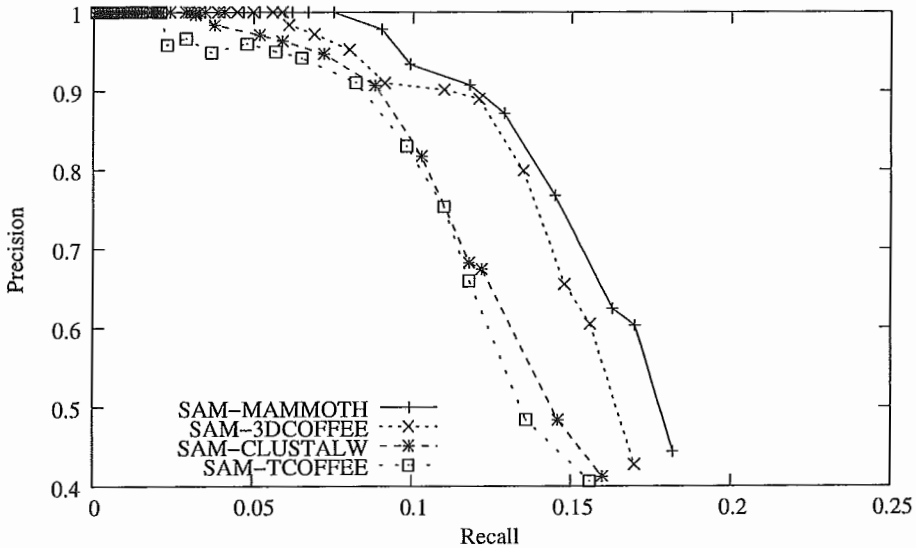


Figura 4.7: Análise de desempenho para o SAM, através de *Precision* e *Recall*, considerando todas as ferramentas de alinhamento.

	MAMMOTH	3DCOFFEE	CLUSTALW
TCOFFEE	0,024416 (não)	0,041404 (não)	0,495871 (não)
CLUSTALW	0,089783 (não)	0,180068 (não)	
3DCOFFEE	0,623762 (não)		

Tabela 4.4: Resultado do *paired t-test* e significância estatística entre os testes realizados para o pacote SAM, considerando todas as ferramentas de alinhamento.

Os modelos construídos a partir de alinhamentos estruturais produziram os melhores resultados para ambos os pacotes. O melhor desempenho de HMMER e SAM foi obtido utilizando os alinhamentos da ferramenta MAMMOTH, como mostram as figuras 4.8 e 4.9. Esses resultados estão de acordo com o trabalho (GOUGH, 2002), que mostrou que na presença de alinhamentos de baixa qualidade, ou seja, alinhamentos com muitos buracos, o pacote HMMER atinge um desempenho melhor em relação ao pacote SAM. Os alinhamentos produzidos por MAMMOTH são os alinhamentos que apresentam mais buracos, como pode ser visto na figura 4.3. Esses resultados pode está relacionado à estratégia de seleção de estados, como discutido na sessão 3.4 página 42. Como a estratégia adotada pelo pacote HMMER define os estados de *match* e *insert* ao mesmo tempo que maximiza as probabilidades de emissão e transição, descritos na sessão 3.4.2 página 44, é possível diferenciar regiões conservadas de regiões de inserções mais precisamente. Por lado, a estratégia padrão adotada pelo pacote SAM associa cada coluna do alinhamento a um estado de *match*. O fato das inserções não serem tratadas prejudica o desempenho do pacote SAM, quando alinhamentos de baixa qualidade são utilizados na construção dos modelos.

Entre os alinhamentos primários, os modelos construídos utilizando a ferramenta CLUSTALW apresentaram os melhores resultados. Como os alinhamentos são mais concisos, o pacote SAM obteve melhores resultados em relação ao pacote HMMER, embora as diferenças não sejam significativas, como mostra a tabela 4.5.

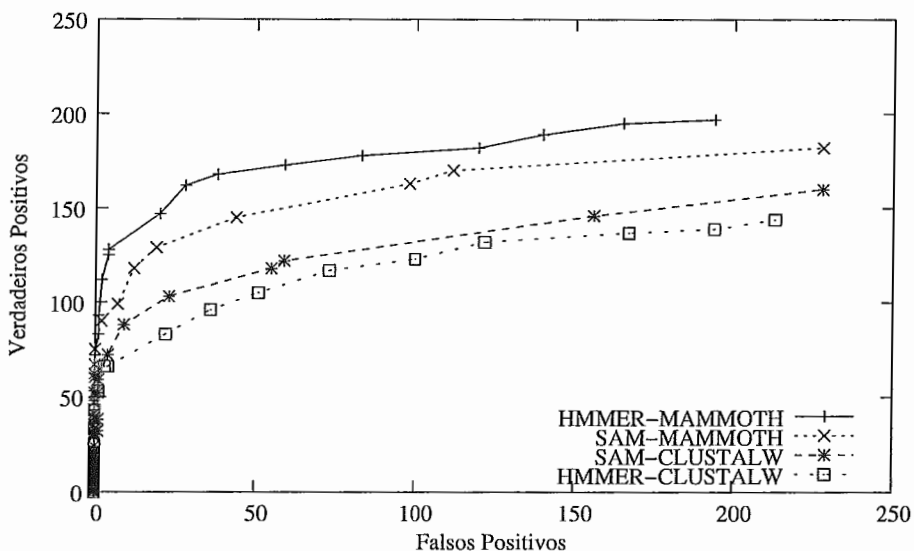


Figura 4.8: Análise de desempenho para o HMMER e SAM, através de curvas ROC, considerando apenas MAMMOTH e CLUSTALW.

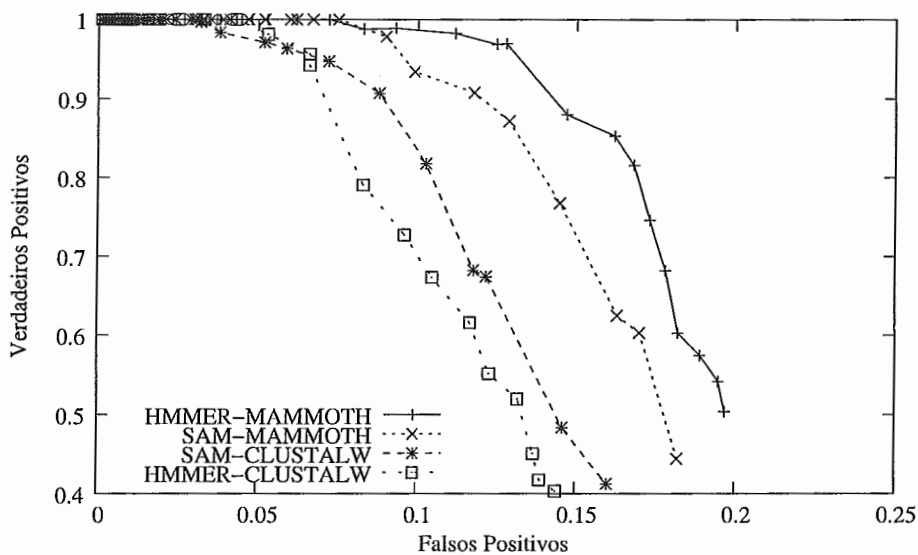


Figura 4.9: Análise de desempenho para o HMMER e SAM, através de curvas *precision* e *recall*, considerando apenas MAMMOTH e CLUSTALW.

	HMMER-MAMM	SAM-MAMM	HMMER-CLUS
SAM-CLUS	0,0013638 (sim)	0,0897833 (não)	0,0277039 (não)
HMMER-CLUS	0,0017298 (sim)	0,0026047 (sim)	
SAM-MAMM	0,0056633 (sim)		

Tabela 4.5: Resultado do *paired t-test* e significância estatística entre os testes realizados para o pacote HMMER e SAM, considerando os alinhamentos produzidos por MAMMOTH e CLUSTALW.

O trabalho (WISTRAND, SONNHAMMER, 2005) desenvolveu duas melhorias nos algoritmos do HMMER. Essas modificações incluem, a implementação de um método baseado em entropia para associar um peso a cada seqüência do alinhamento, como descrito na sessão 3.6 página 53, e a substituição da mistura de distribuições Dirichlet de nove componentes, adotada pelo HMMER, pela mistura de Dirichlet de 20 componentes utilizada pelo SAM, no cálculo das probabilidades de emissão.

Foram realizados testes considerando as duas modificações acima. Os resultados para modelos criados a partir do MAMMOTH são mostrados pelas figuras 4.10 e 4.11. Uma notável diferença foi observada na utilização da mistura de Dirichlet de 20 componentes, demonstrando que as informações a priori incluídas no cálculo das probabilidades aumentam a capacidade de generalização dos modelos. O algoritmo baseado em entropia, para atribuição de pesos as seqüências, não trouxe melhorias devido a baixa identidade entre as seqüências, o que torna difícil a eliminação de redundâncias. A tabela 4.6 mostra os resultados dos *paired t-tests* e as significâncias estatísticas das comparações.

Os resultados para modelos criados a partir do CLUSTALW são mostrados pelas figuras 4.12 e 4.13. A utilização da mistura de Dirichlet do pacote SAM proporcionou um pequeno aumento no desempenho do pacote HMMER, porém sem significância estatística. O algoritmo baseado em entropia apresentou um comportamento similar ao observado no teste com os modelos gerados a partir do MAMMOTH. A tabela 4.7 apresenta os resultados dos *paired t-tests* e as significâncias estatísticas dos testes realizados.

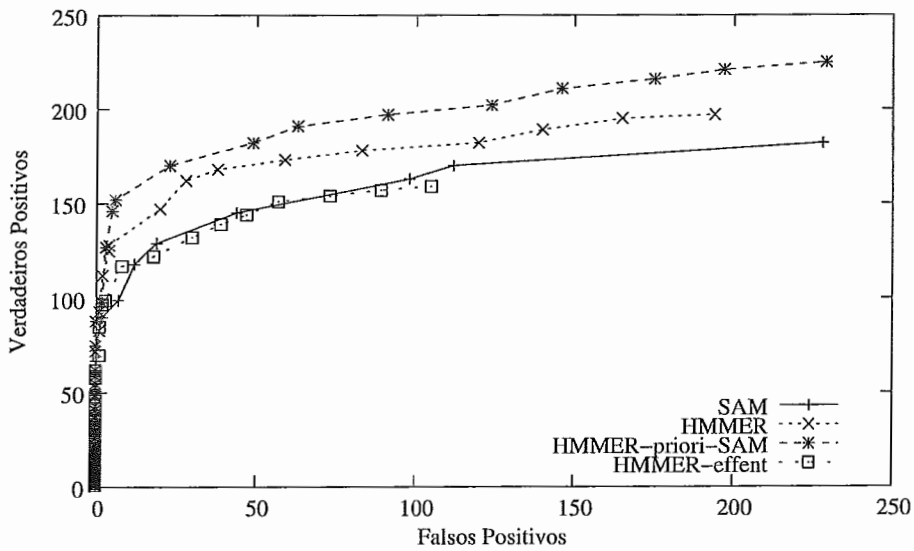


Figura 4.10: Análise de desempenho para o HMMER e SAM, através de curvas ROC, considerando as modificações propostas por (WISTRAND, SONNHAMMER, 2005) e os alinhamentos produzidos por MAMMOTH.

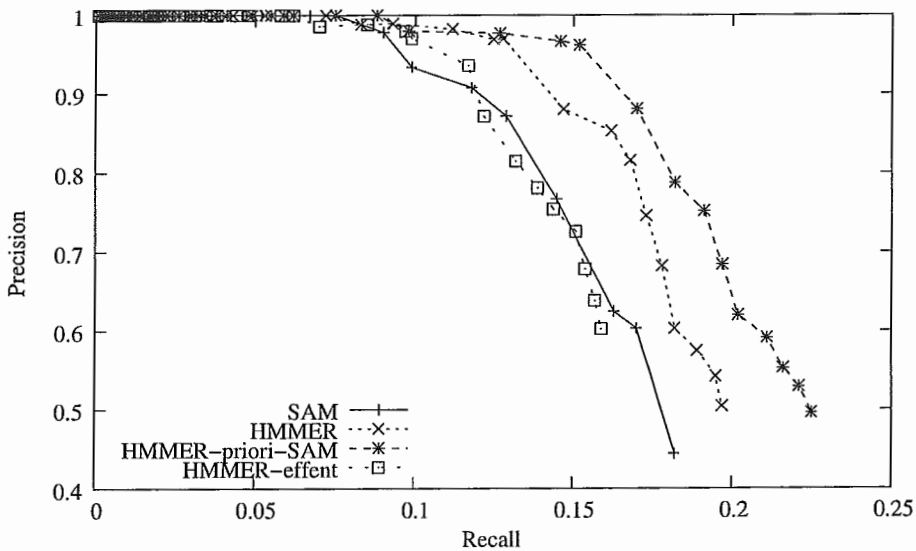


Figura 4.11: Análise de desempenho para o HMMER e SAM, através de curvas *precision* e *recall*, considerando as modificações propostas por (WISTRAND, SONNHAMMER, 2005) e os alinhamentos produzidos por MAMMOTH.

	HMMER	SAM	HMMER-priori-SAM
HMMER-effent	0,0062015 (sim)	0,1865434 (não)	0,0020339 (sim)
HMMER-priori-SAM	0,0131563 (sim)	0,0051761 (sim)	
SAM	0,0056633 (sim)		

Tabela 4.6: Resultado do *paired t-test* e significância estatística entre os testes realizados para o pacote HMMER e SAM, considerando as modificações propostas por (WISTRAND, SONNHAMMER, 2005) e os alinhamentos produzidos por MAMMOTH.

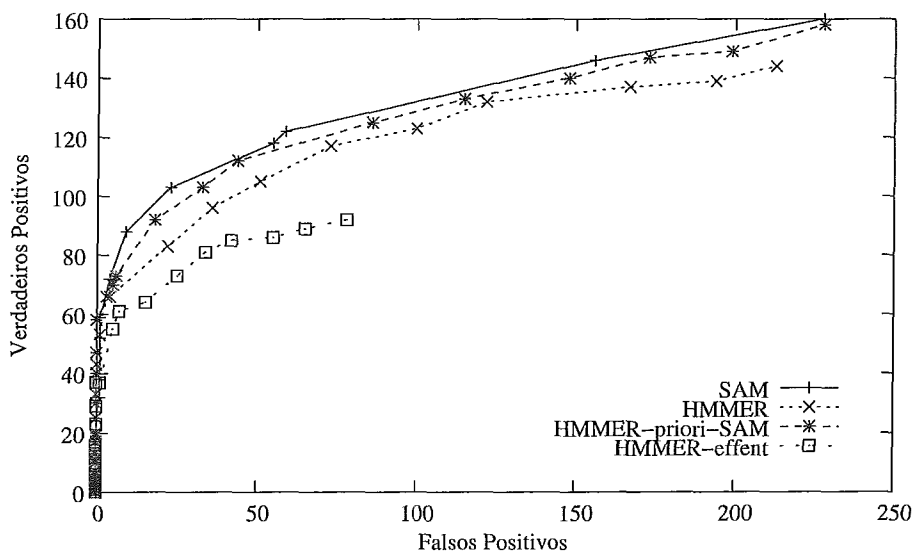


Figura 4.12: Análise de desempenho para o HMMER e SAM, através de curvas ROC, considerando as modificações propostas por (WISTRAND, SONNHAMMER, 2005) e os alinhamentos produzidos por CLUSTALW.

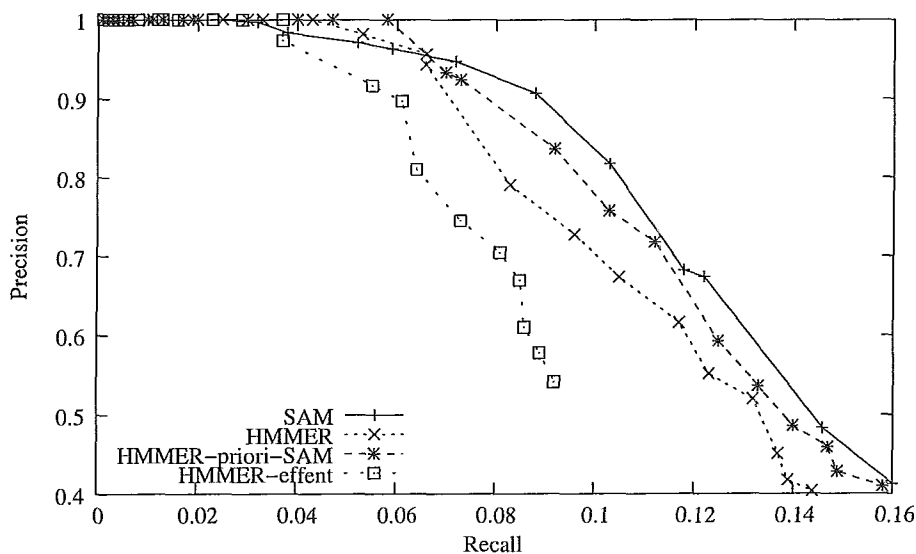


Figura 4.13: Análise de desempenho para o HMMER e SAM, através de curvas *precision* e *recall*, considerando as modificações propostas por (WISTRAND, SONNHAMMER, 2005) e os alinhamentos produzidos por CLUSTALW.

	HMMER	SAM	HMMER-priori-SAM
HMMER-effent	0,0041053 (sim)	0,1865434 (não)	0,0020339 (sim)
HMMER-priori-SAM	0,0232574 (não)	0,0314675 (não)	
SAM	0,0277039 (não)		

Tabela 4.7: Resultado do *paired t-test* e significância estatística entre os testes realizados para o pacote HMMER e SAM, considerando as modificações propostas por (WISTRAND, SONNHAMMER, 2005) e os alinhamentos produzidos por CLUSTALW.

Embora os pacotes HMMER e SAM sejam mais eficientes, na detecção de homólogos distantes, quando comparados a métodos baseados em alinhamentos par a par, (GOUGH, KARPLUS, et al., 2001; KARPLUS, BARRET, et al., 1998; PARK, KARPLUS, et al., 1998), a detecção de homologias não atinge 30% dos verdadeiros positivos existentes. Esse fato pode ser explicado devido a alta divergência entre as seqüências do conjunto de treinamento. Os modelos construídos a partir dessas seqüências apresentam dificuldades para distinguir entre homólogos distantes e verdadeiros negativos. Os testes realizados demonstraram que o desempenho dos pHMMs está diretamente ligado ao alinhamento de seqüências fornecido como entrada e a qualidade das informações a priori utilizadas. Os melhores resultados foram obtidos pelos modelos produzidos através de alinhamentos estruturais, pois esses modelos são capazes de representar padrões relevantes presentes nos alinhamentos, e isso auxilia a detecção de homólogos distantes. As informações a priori são indispensáveis à criação de modelos mais genéricos, principalmente na presença de alinhamentos estruturais, onde a ausência de dados observados em algumas colunas do alinhamento é mais evidente.

CAPÍTULO 5

Adicionando Informações Estruturais à *Profiles* HMM

Padrões estruturais são encontrados em proteínas com comprovada origem evolutiva. Proteínas homólogas possuem um conjunto de conformações que governam suas funções. O uso dessas informações para auxiliar o processo de anotação e inferência de função em proteínas, torna-se cada vez mais freqüente, como mostram os trabalhos (WANG, SAMUDRALA, 2005; ALEXANDROV, GERSTEIN, 2004; GOYON, TUFFÉRY, 2004; BYSTROFF, BAKER, 2000). Grande parte desses trabalhos buscam as similaridades estruturais considerando apenas as estruturas terciárias, sessão 2.3.3 página 28. No entanto, o trabalho (CHAKRABARTI, SOWDHAMIMI, 2004), mostrou que a conservação de padrões estruturais não está diretamente relacionada a rigidez espacial dos átomos, que formam as moléculas das proteína. O objetivo deste capítulo é introduzir uma nova metodologia, baseada em pHMMs e diferentes propriedades estruturais, aplicada ao problema de detecção de homólogos distantes. Basicamente, nós criamos um novo algoritmo de atribuição de pesos às seqüências, descrito na sessão 5.1. A partir desse algoritmo foram construídos cinco pHMMs diferentes, para cada conjunto de seqüências alinhadas. Desses cinco, três são baseados no trabalho (CHAKRABARTI, SOWDHAMIMI, 2004), que considera relevante as informações de estruturas secundárias, acessibilidade e empacotamento (*packing*) de aminoácidos, abordados pelas sessões 5.1.2, 5.1.3 e 5.1.4, respectivamente. Um pHMM foi baseado em informações de estrutura terciária, sessão 5.1.5 e um na estru-

tura primária baseado no modelo do HMMER sem modificações. A classificação de novas seqüências combina a classificação dos diferentes pHMMs, abordada na sessão 5.2. Os resultados são discutidos na sessão 5.3. A figura 5.1 mostra o esquema descrito. Um conjunto de estruturas de proteínas homólogas são alinhadas através da ferramenta MAMMOTH (ATTWOOD, BRADLEY, et al., 2005). Essa ferramenta produz como saída o alinhamento estrutural convertido em alinhamento primário e as coordenadas do alinhamento estrutural. A partir dessas saídas foram construídos os cinco pHMMs. O pHMM 1D foi construído somente a partir do alinhamento primário. Os modelos pHMM 2D, pHMM Acc e pHMM Oi foram construídos também utilizando o pacote JOY (MIZUGUCHI, DEANE, et al., 1998) para extrair das estruturas tridimensionais as informações de estrutura secundária, acessibilidade e empacotamento de aminoácidos. Essas informações foram empregadas na construção da matriz de pesos estruturais M'_s descrita na sessão 5.1, que foi utilizada durante o aprendizado dos pHMMs. A matriz M'_s para o modelo pHMM 3D foi construída determinando o (*homologous core structure*) HSP (MATSUO, BRYANT, 1999). A classificação de novas seqüências combina a classificação dos cinco pHMMs.

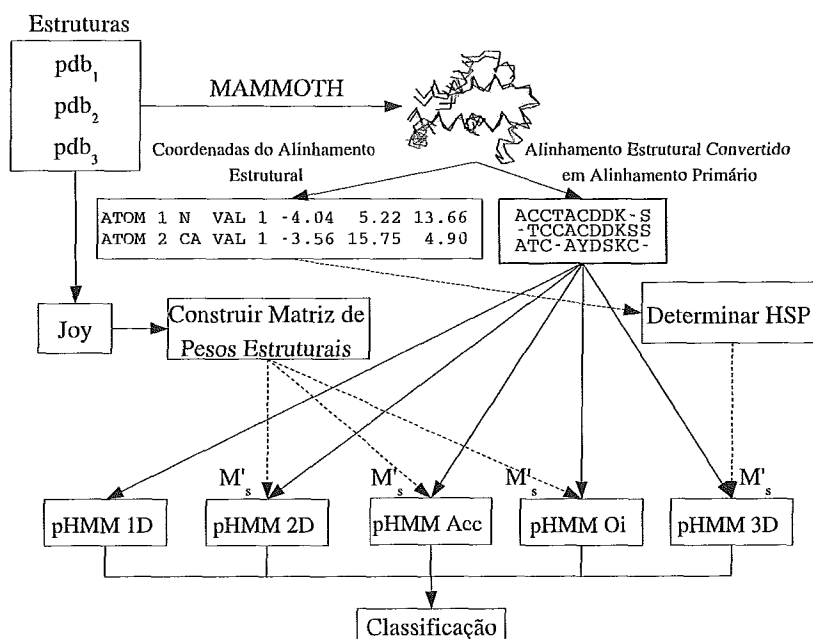


Figura 5.1: Esquema HMMER-STRUCT.

5.1 Modificação do Algoritmo de Atribuição de Pesos às Seqüências

5.1.1 Metodologia para Atribuição de Pesos Estruturais

O capítulo 4, mostrou que a utilização de alinhamentos estruturais melhora o desempenho de pHMMs. Entretanto, muita informação relevante é perdida quando alinhamentos estruturais são usados apenas como alinhamentos primários. Quando proteínas que apresentam baixa similaridade seqüencial são alinhadas estruturalmente, ocorre a sobreposição entre algumas regiões espaciais. Os aminoácidos envolvidos nessas regiões formam a estrutura mínima necessária para identificar novos homólogos. Sendo assim, *pode-se argumentar que esses aminoácidos devem receber pesos maiores quando comparados a outros aminoácidos sem representatividade estrutural*. O mesmo argumento se aplica a outras características estruturais que são conservadas em proteínas da mesma família. O programa *hmmbuild* responsável pela construção dos modelos, no pacote HMMER, foi modificado para incorporar todas as alterações descritas nesta sessão.

A proposta deste trabalho é construir diversos modelos adicionando informações estruturais sob diferentes aspectos. Para isso, foi mantido o modelo construído pelo pacote HMMER, utilizando alinhamentos estruturais e foram adicionados quatro modelos construídos de acordo com as sessões 5.1.2, 5.1.5, 5.1.3 e 5.1.4.

Todos os algoritmos que associam pesos as seqüências, descrito na sessão 3.6 página 53, atribuem o mesmo peso a todos os aminoácidos de determinada proteína. Sendo assim, o alinhamento múltiplo pode ser representado pela matriz M_c de dimensão $N \times L$, representada em 5.1, onde N é o número de proteínas e L o tamanho do alinhamento. Na matriz M_c , cada elemento w_i corresponde ao peso de um aminoácido e $\sum_i^N w_i = N$.

$$M_c = \begin{pmatrix} w_1 & \dots & w_1 \\ \vdots & \vdots & \vdots \\ w_N & \dots & w_N \end{pmatrix} \quad (5.1)$$

Para representar os pesos estruturais de cada aminoácido foi criada uma matriz M_s de dimensão $N \times L$, representada em 5.2, onde cada elemento m_{ij} representa o peso estrutural atribuído ao j -ésimo aminoácido da i -ésima proteína no alinhamento. As próximas sessões mostram como os pesos estruturais são encontrados.

$$M_s = \begin{pmatrix} m_{11} & \dots & m_{1L} \\ \vdots & \vdots & \vdots \\ m_{N1} & \dots & m_{NL} \end{pmatrix} \quad (5.2)$$

Os pesos da matriz M_s foram combinados com os pesos fornecidos pela matriz M_c , para que cada aminoácido receba um peso, de acordo com sua importância estrutural e o peso atribuído a proteína a qual pertence. Dessa forma, a matriz M_c foi multiplicada pela matriz transposta M_s^T , resultando na matriz representada por 5.3.

$$M'_s = M_c M_s^T = \begin{pmatrix} w_1 m_{11} & \dots & w_1 m_{1L} \\ \vdots & \vdots & \vdots \\ w_N m_{N1} & \dots & w_N m_{NL} \end{pmatrix} \quad (5.3)$$

Antes de utilizar a matriz M'_s , no cálculo das probabilidades do modelo, é necessário normalizá-la. Para isso, cada elemento m_{ij} da matriz foi multiplicado por um fator α_j , tal que, $\sum_i^N \alpha_j m_{ij} = N - g$, onde g é o número de buracos da coluna j . Após a normalização as probabilidades de emissão do modelo são calculadas através da fórmula 5.4, onde $c_i(\sigma)$ é o contador real do aminoácido σ no estado i e $a(\sigma)$ é a informação a priori do aminoácido σ , como discutido na sessão 3.5 página 46.

$$E_i(\sigma) = \frac{c_i(\sigma) + a(\sigma)}{\sum_j c_i(\sigma_j) + a(\sigma_j)} \quad (5.4)$$

O contador $c_i(\sigma)$ é obtido pela fórmula 5.5, onde m_{ij} é elemento da matriz M'_s .

$$c_i(\sigma) = \sum_i^N f(\sigma) \quad \dots \quad f(\sigma) = \begin{cases} m_{ij}, & \text{se } \sigma \text{ é o aminoácido da posição } ij \\ 0, & \text{caso contrário} \end{cases} \quad (5.5)$$

As probabilidades de transição são calculadas através da fórmula 5.6, onde c_{kl} é o contador de transições do estado k para o estado l , a_{kl} é a probabilidade a priori, e $k, l \in \{M, I, D\}$, (*match*, *insert* e *delete*).

$$t_{kl} = \frac{c_{kl} + a_{kl}}{\sum_l c_{kl} + a_{kl}} \quad (5.6)$$

O contador de transições c_{kl} é obtido através da fórmula 5.7.

$$c_{kl} = \sum_i^N f_{kl} \quad \dots \quad f_{kl} = \begin{cases} \frac{m_{ik} + m_{il}}{2}, & \text{se } k, l \in \{M, I\} \\ m_{ik}, & \text{se } l \in \{D\} \text{ e } k \notin \{D\} \\ m_{il}, & \text{se } k \in \{D\} \text{ e } l \notin \{D\} \\ 1, & \text{se } k, l \in \{D\} \end{cases} \quad (5.7)$$

5.1.2 Estruturas Secundárias

Grande parte da conservação entre proteínas homólogas deve-se a presença de elementos de estrutura secundária. Esses elementos são divididos em três classes principais: alpha-hélices, folhas-beta e *loops*, discutidos na sessão 2.3.2 página 25. Os motivos, regiões conservadas dentro de proteínas homólogas, são formados, em grande parte, pela combinação de elementos de estruturas secundárias.

O programa SSTRUC, que faz parte do pacote JOY (MIZUGUCHI, DEANE, et al., 1998), extrai os elementos de estruturas secundárias de estruturas depositadas no banco de dados PDB (HELEN, WESTBROOK, et al., 2000). A partir dessas informações foi construída a matriz M'_s , sessão 5.2. O objetivo é representar o alinhamento estrutural sob o enfoque dos elementos de estrutura secundária.

Cada elemento m_{ij} , da matriz M'_s , representa o peso associado ao aminoácido de acordo com o elemento de estrutura secundária ao qual está associado. Embora o sítio ativo de uma proteína possa ser encontrado nas região de *loops*, as inserções de novos resíduos também ocorrem nessas regiões. Por esse motivo, pesos menores foram atribuídos aos *loops*, por se tratar de regiões menos conservadas. A figura 5.2

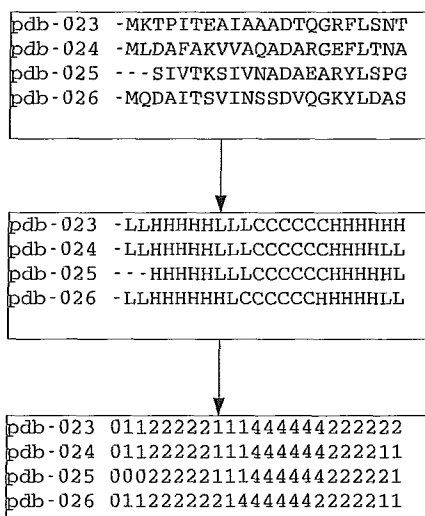


Figura 5.2: Matriz de pesos a partir de informações de estrutura secundária.

mostra a matriz construída a partir das informações obtidas pelo JOY. Foi atribuído o peso um para as regiões de *loop*, peso dois para alphas-hélices e peso quatro para as folhas-beta. Esses valores foram baseados no trabalho (DEANE, PERDERSEN, et al., 2003), que realizou um estudo sobre a conservação de elementos de estruturas secundárias em diversas família de proteínas.

5.1.3 Acessibilidade dos Aminoácidos

Os aminoácidos, de maneira geral, podem ser divididos em dois grupos: os que possuem molécula polar, e por isso são solúveis em meios aquosos e os que apresentam molécula apolar e conseqüentemente são inacessíveis, ou seja, não são solúveis em água. O efeito hidrofóbico, discutido na sessão 2.3.3 página 28, é um fator determinante para a conformação da proteína. Aminoácidos com alta inacessibilidade formam o interior da proteína, pois tendem a se distanciar da superfície, onde o contato com soluções aquosa é mais intenso. Esses aminoácidos são mais conservados, pois as inserções e exclusões de resíduos ocorrem, em grande parte, na superfície da proteína, (DEANE, PERDERSEN, et al., 2003).

As informações sobre a acessibilidade dos resíduos foram obtida através do programa PSA (LEE, RICHARDS, 1971), que integra o pacote JOY. A partir dessas informações foi construída a matriz M'_s com base no alinhamento estrutural fornecido

pelo MAMMOTH. Cada elemento m_{ij} representa a acessibilidade do aminoácido correspondente. O valor um foi atribuído aos aminoácidos considerados acessíveis e o valor três aos aminoácidos inacessíveis. Esses valores foram baseados no trabalho (CHAKRABARTI, SOWDHAMIMI, 2004), que através de um estudo concluiu que aminoácidos inacessíveis são três vezes mais conservados em relação aos demais. O objetivo é privilegiar as regiões do alinhamento, cujos os aminoácido possuem alto grau de inacessibilidade.

Após a construção da matriz M'_s as probabilidades do modelo foram calculadas de acordo com a sessão 5.2.

5.1.4 Empacotamento dos Aminoácidos

A estrutura terciária das proteínas é formada por diversas ligações e interações químicas, sessão 2.3.3 página 28. Nas regiões onde essas ligações são mais intensas ocorre a formação de pacotes (*packing*) ou empacotamento. Os aminoácidos envolvidos nessas regiões tendem a ser mais conservados devido ao número de interações com outros aminoácidos. Para medir o grau de empacotamento dos aminoácidos T.J Ooi criou uma medida denominada *Ooi number* (NISHIKAWA, OOI, 1986). Para um aminoácido i o valor *Ooi number* é calculado, contabilizando o número de vizinhos de i em uma esfera de raio 14 Å, essa contagem é realizada considerando as distâncias entre o $C\alpha$ de i e o $C\alpha$ de cada um de seus vizinhos. Sendo considerado vizinho de i apenas os aminoácidos que possuem interação direta com i . A figura 5.3 mostra um trecho da saída do programa JOY, destacando os valores de *Ooi number* para a proteína Dehaloperoxidase da família da globinas.

```
1ew6
Ooi number
233244334322333444455544432222332332
232222322332222432343344335444433322
334333443343221232233334543553454334
2221223333533454454334323232*
```

Figura 5.3: Valores de *Ooi number* para a proteína Dehaloperoxidase.

A matriz de pesos estruturais M'_s foi construída atribuindo a cada elemento m_{ij} a medida *Ooi number* diretamente. Essa matriz foi empregada no cálculo das probabilidades de emissão e transição, como descrito na sessão 5.2.

5.1.5 Estruturas Terciárias

Os algoritmos para alinhamentos estruturais transladam e rotacionam as estruturas das proteínas, com o objetivo de encontrar a máxima sobreposição entre os átomos (HIGGINS, TAYLOR, 2000). A partir de um alinhamento estrutural de proteínas homólogas, pode-se observar um conjunto de átomos cujas coordenadas tridimensionais sofrem variações mínimas, chamado de HSP (*homologous core structure*), (MATSUO, BRYANT, 1999). De acordo com os trabalhos (GERSTEIN, ALTMAN, 1995) e (ALTMAN, GERSTEIN, 1995), HSPs podem ser utilizados para caracterizar proteínas da mesma origem evolutiva.

Para detectar os HSPs das proteínas foi desenvolvido um módulo para extrair essas informações de alinhamentos estruturais. Para isso, foi necessário determinar o grau de sobreposição entre os átomos alinhados. O programa MAMMOTH foi escolhido, por apresentar melhores resultados em relação ao programa 3DCOFFEE, como foi mostrado na sessão 4.5 página 72. MAMMOTH trabalha alinhando apenas os carbonos alphas da cadeia principal da proteína, dessa forma, a sobreposição entre os átomos corresponde a sobreposição entre os próprios aminoácidos.

A distância entre as posições espaciais dos aminoácidos é a *distância euclidiana*, fórmula 5.8, que representa a menor distância entre dois pontos em um espaço n -dimensional.

$$d_{a,b} = \sqrt{(X_a - X_b)^2 + (Y_a - Y_b)^2 + (Z_a - Z_b)^2} \quad (5.8)$$

Para associar um peso a cada aminoácido, de acordo com a posição espacial ocupada por ele, foi necessário determinar a *distância relativa* d_{ij} , entre cada aminoácido da proteína i presente na coluna j do alinhamento e os demais aminoácidos da

mesma coluna. Essa distância foi obtida através da média das distâncias euclidianas, de acordo com a fórmula 5.9.

$$d_{ij} = \frac{\sum_{b=1}^{n-1} d_{i,j,i+1j}}{n-1} \quad (5.9)$$

O inverso das distâncias relativas d_{ij} foi utilizado para calcular o grau de sobreposição entre os aminoácidos de cada coluna j do alinhamento múltiplo. Essas distâncias foram normalizadas, através da fórmula 5.10, onde d_{min} corresponde a menor distância relativa, e d_{max} a distância máxima. Para representar a importância posicional de cada aminoácido, adotou-se o valor zero para nenhuma importância estrutural e um para máxima.

$$m_{ij} = \frac{d_{ij} - d_{min}}{d_{max} - d_{min}} \quad (5.10)$$

Após a normalização, os valores m_{ij} foram usados para compor a matriz M'_s e as probabilidades do modelo foram calculadas de acordo com a sessão 5.2.

5.2 Biblioteca de Modelos

A partir de cada alinhamento estrutural foram construídas quatro matrizes M'_s , de acordo com a sessão 5.1, sendo que cada matriz destaca uma característica estrutural relevante, como descrito nas sessões anteriores. A união dos modelos, que representam diferentes propriedades estruturais, forma uma *biblioteca* ou banco de dados de modelos, capaz de representar padrões sob diferentes aspectos, em proteínas homólogas. O programa *hmmpfam*, que faz parte do pacote HMMER, foi utilizado para realizar busca na base de dados de modelos. Bibliotecas de modelos tem sido utilizadas por diversos trabalhos, (BATEMAN, COIN, et al., 2004; HAFT, SELENGUT, et al., 2003; GOUGH, 2002), e apresentam melhores resultados quando comparadas a modelos individuais. O cálculo do *e-value*, discutido na sessão 3.7.4 página 59, tende a ser mais apurado, por considerar a confiança em diferentes modelos. Em geral, se uma proteína é classificada por vários modelos relacionados, as chances da classificação estar correta são maiores quando comparada a classificação de um

único modelo.

5.3 Resultados

Os procedimentos dos testes realizados são idênticos aos apresentados na sessão 4.4 página 70. As figuras 5.4 e 5.5 mostram os resultados, através de curvas ROC e *precision* e *recall*, comparando apenas os modelos que compõem a ferramenta HMMER-STRUCT. A tabela 5.1 mostra os resultados dos *paired t-tests* e as significâncias estatísticas das comparações entre os modelos.

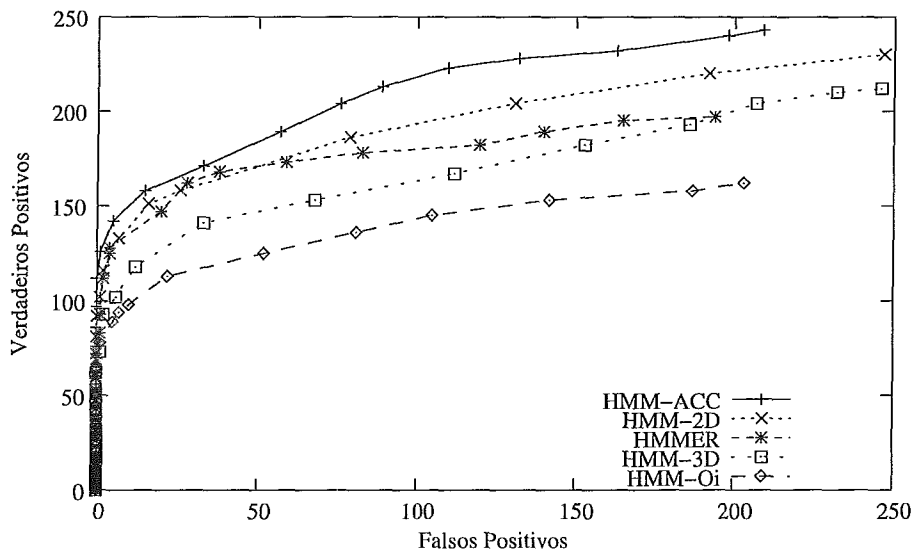


Figura 5.4: Comparação entre os modelos do HMMER-STRUCT, através de curvas ROC, considerando os alinhamentos produzidos por MAMMOTH.

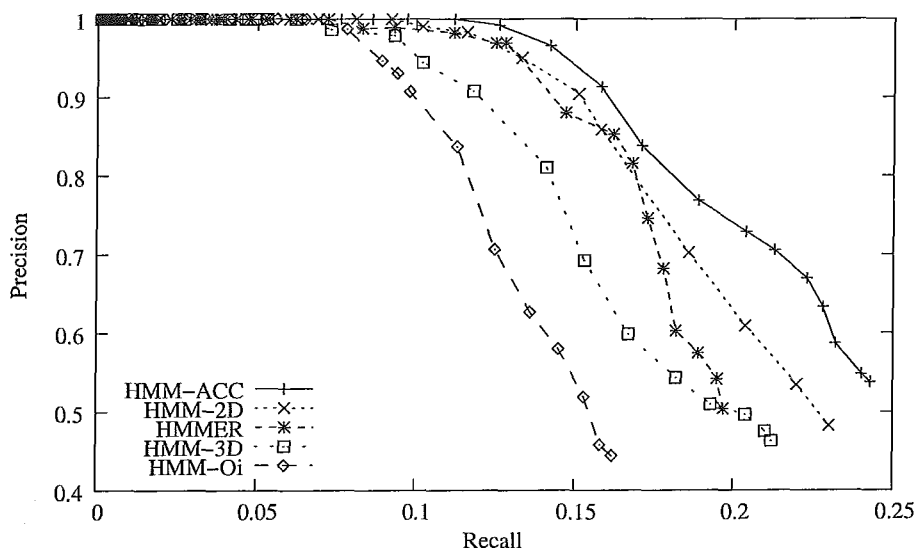


Figura 5.5: Comparação entre os modelos do HMMER-STRUCT, através de curvas *precision* e *recall*, considerando os alinhamentos produzidos por MAMMOTH.

	HMMER	HMM-2D	HMM-3D	HMM-ACC
HMM-Oi	0,0546380 (não)	0,0000001 (sim)	0,0000007 (sim)	0,0000066 (sim)
HMM-ACC	0,0000004 (sim)	0,0113760 (sim)	0,0111504 (sim)	
HMM-3D	0,0000005 (sim)	0,0059098 (sim)		
HMM-2D	0,0000002 (sim)			

Tabela 5.1: Resultado do *paired t-test* e significância estatística dos testes realizados entre os modelos do HMMER-STRUCT, considerando os alinhamentos produzidos por MAMMOTH.

Observando as figuras 5.4 e 5.5, e a tabela 5.1 é possível concluir que o modelo que apresentou o melhor desempenho foi o modelo baseado em informações de acessibilidade (HMM-ACC). De fato, os aminoácidos não solúveis em água são mais conservados, formando o interior da proteína, que é menos predisposto a eventos de mutação. Os modelos baseados em informações de estruturas secundárias (HMM-2D), também apresentaram resultados estatisticamente significante em relação aos demais. Como discutido na sessão 2.3.2 página 25, os elementos de estruturas secundária são responsáveis em média por 60% dos aminoácidos conservados das proteínas. Por outro lado, os modelos (HMM-3D) e (HMM-Oi) apresentaram desempenho inferior ao pacote HMMER. O comportamento dos modelos, baseados em informações tridimensionais, pode ser explicado pelo fato dessas informações serem usadas apenas para dar mais pesos aos aminoácidos do alinhamento múltiplo, que consi-

dera apenas seqüências primárias. O modelo (HMM-3D) é baseado em informações tridimensionais, porém não representa estruturas terciárias propriamente, para isso seria necessário profundas alterações na arquitetura de pHMMs. Os modelos baseados na medida O_i podem estar sofrendo de *overfitting*, devido a escolhas dos pesos. Um estudo detalhado desses pesos será realizado em trabalhos futuros, como será discutido na sessão 6.2 página 98.

Os resultados dos diferentes modelos foram combinados variando o número de classificadores, as curvas 5.6 e 5.7 mostram as curvas ROC e *precision* e *recall*, respectivamente. Na legenda das figuras o termo HMMER-STRUCT- x deve ser lido como a união dos resultados de pelo menos x classificadores.

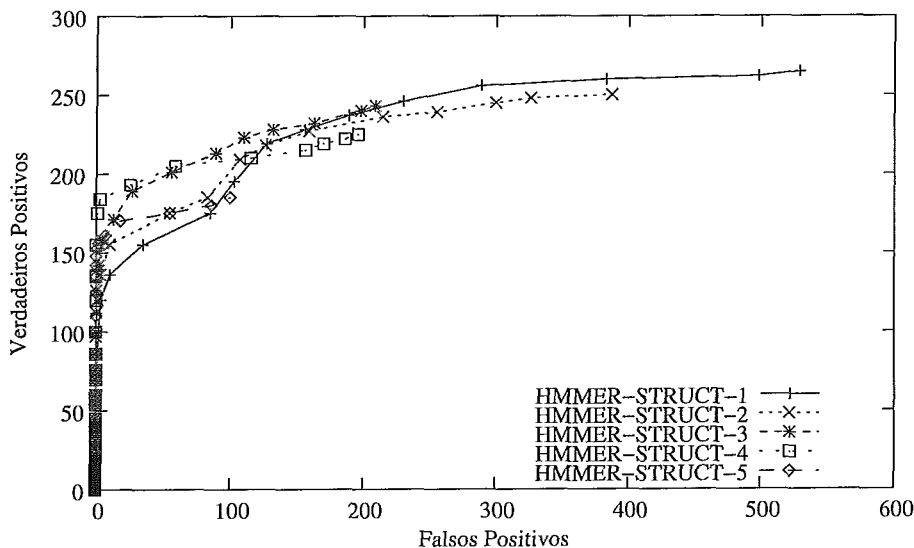


Figura 5.6: Avaliação do desempenho do HMMER-STRUCT, através de curvas ROC, variando o número de classificadores e considerando os alinhamentos produzidos por MAMMOTH.

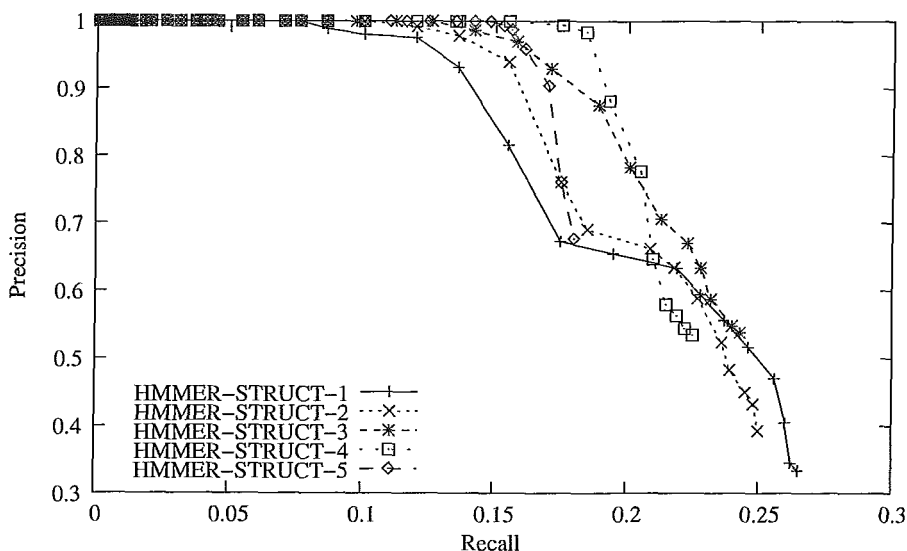


Figura 5.7: Avaliação do desempenho do HMMER-STRUCT, através de curvas *precision* e *recall*, variando o número de classificadores e considerando os alinhamentos produzidos por MAMMOTH.

Os melhores resultados foram obtidos combinando pelo menos três classificadores. As figuras 5.8 e 5.9 mostram os resultados, através de curvas ROC e *precision* e *recall*, comparando HMMER-STRUCT-3, HMMER e SAM, considerando apenas os alinhamentos produzidos por MAMMOTH. A tabela 5.2 apresenta os resultados dos *paired t-tests* e as significâncias estatísticas das comparações.

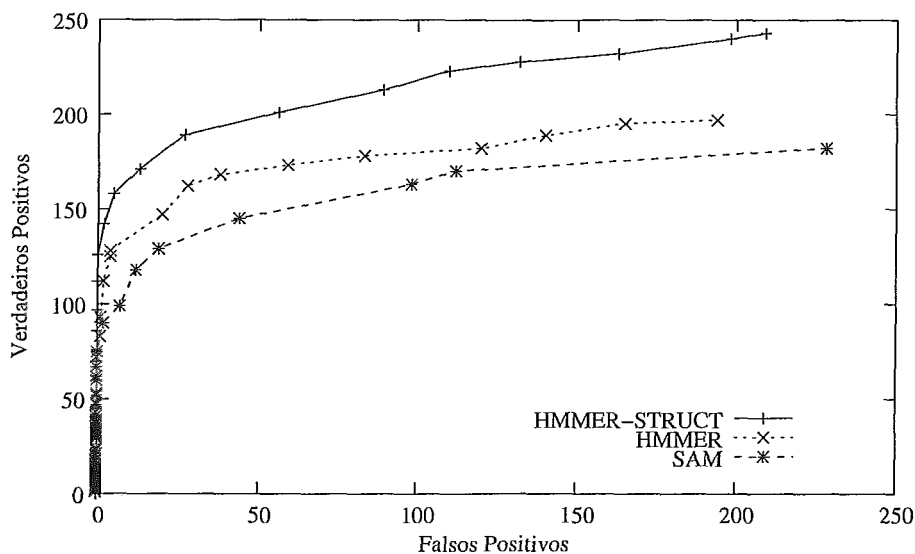


Figura 5.8: Comparação entre HMMER, SAM e HMMER-STRUCT, através de curvas ROC, considerando os alinhamentos produzidos por MAMMOTH.

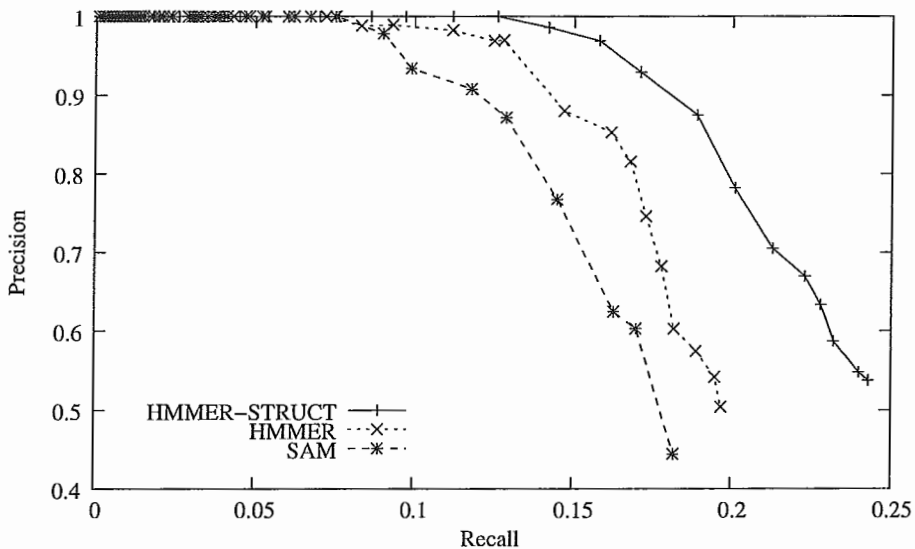


Figura 5.9: Comparação entre HMMER, SAM e HMMER-STRUCT, através de curvas *precision* e *recall*, considerando os alinhamentos produzidos por MAMMOTH.

	HMMER-STRUCT
SAM	0,0018652 (sim)
HMMER	0,0017503 (sim)

Tabela 5.2: Resultado do *paired t-test* e significância estatística dos testes realizados entre HMMER, SAM e HMMER-STRUCT-3, considerando os alinhamentos produzidos por MAMMOTH.

CAPÍTULO 6

Conclusão

6.1 Contribuições

A precisão dos métodos de detecção de homologias é de vital importância para os processos de anotação de novas seqüências. O principal desafio dessa área é direcionado à detecção de homologias distantes. O objetivo é diminuir o número de novas seqüências sem correspondência em bancos de dados públicos empregando métodos mais eficientes.

O crescimento das bases de dados que armazenam informações estruturais de proteínas, pode ser visto como uma motivação ao aprimoramento e surgimento de métodos baseados nessas informações. Diversos trabalhos estão focados na utilização de informações estruturais, como forma de melhorar alinhamentos múltiplos de seqüências (CHAKRABARTI, BHARDWAJ, et al., 2004); criar assinaturas com objetivo de personalizar proteínas homólogas (WANG, SAMUDRALA, 2005); e identificar motivos estruturais que caracterizem famílias de proteínas e auxiliem a detecção de homólogos distantes (CHAKRABARTI, SOWDHAMINI, 2004).

Os alinhamentos estruturais captam as similaridades espaciais entre as coordenadas dos átomos de um conjunto de proteínas. Esses alinhamentos provêm informações significantes sobre as relações das amostras do conjunto, principalmente quando as seqüências analisadas são muito divergentes. O capítulo 4 mostrou que

alinhamentos estruturais, utilizados na construção de pHMMs, produzem melhores resultados em relação a alinhamentos primários.

Todavia quando alinhamentos estruturais são utilizados apenas provendo alinhamentos primários, outras características estruturais relevantes são ignoradas. A proposta deste trabalho foi construir um conjunto de pHMMs, cada um deles incorporando informação de diferentes propriedades estruturais. Cada modelo construído representa um tipo de característica estrutural. Essas propriedades estruturais foram aplicadas no treinamento dos pHMMs alterando as distribuições dos aminoácidos, ou seja, foi desenvolvido um novo método de atribuição de pesos às seqüências baseado em informações estruturais. Nossa abordagem não utiliza nenhum tipo de alfabeto especial para representar elementos estruturais, tais como os trabalhos (ALEXANDROV, GERSTEIN, 2004; GOYON, TUFFÉRY, 2004; BYSTROFF, BAKER, 2000); sendo assim, a inferência dos pHMMs aplica-se a seqüências primárias, não sendo necessário fazer uso de informações através de métodos de previsão de estruturas.

O método proposto neste trabalho constrói cinco modelos a partir de um único conjunto de proteínas homólogas. A junção desses modelos possibilitou a criação de uma biblioteca ou base de dados de modelos. O grau de confiança numa classificação aumenta a medida que vários modelos classificam a mesma proteína como positivo. Uma análise variando o número de classificadores mostrou que os melhores resultados foram obtidos quando pelo menos três modelos concordaram com a classificação.

Uma vantagem deste método é a sua flexibilidade, dado que pode ser aplicado mesmo quando existe pouca informação estrutural. No pior caso, dadas N seqüências e apenas duas informações estruturais o biólogo pode construir um classificador. Esta flexibilidade é importante, considerando que o número de proteínas com informações estruturais disponíveis é muito inferior ao número de proteínas anotadas.

6.2 Trabalhos Futuros

As características estruturais das proteínas, quando devidamente exploradas, aumentam a sensibilidade de pHMMs. Esse trabalho ateve-se ao uso de informações de estruturas secundária, terciária, empacotamento e acessibilidade de aminoácidos. Porém, existem outras características que não foram consideradas, tais como pontes de hidrogênio. Essas informações são relevantes porém complexas, pois essas ligações podem envolver átomos da própria cadeia principal (responsáveis pela formação de estruturas secundárias), átomos da cadeia lateral e átomos do grupo amino ou átomos do grupo carboxílico da cadeia principal, ou ainda entre átomos das cadeias laterais. Sendo assim, um estudo sobre a influência de cada uma dessas ligação deve ser realizado, antes de inclui-las na construção de pHMMs.

Profiles HMMs podem ser usados de duas formas: generativa ou discriminativa. O corrente trabalho adotou a forma generativa, onde os modelos são criados e as inferências são realizadas independentemente. Na forma discriminativa, um banco de dados de modelos é criado, e cada modelo representa um conjunto de homólogos. As inferências são realizadas comparando cada amostra com o todo o banco de dados. Esse método possui uma comprovada eficiência, pois as amostras podem ser classificadas pelos modelos mais próximos. O banco de dados *Super-Family* (MADERA, VOGEL, et al., 2004) é um exemplo de pHMMs discriminativos. Esse banco é formado por pHMMs, construídos a partir do pacote SAM, e representam todas as super famílias de proteínas com estrutura definida. No entanto, *Super-Family* não faz uso de alinhamentos ou características estruturais. A proposta é aprimorar a base de dados *Super-Family* substituindo o pacote SAM pelo pacote HMMER-STRUCT.

A importância de cada propriedade estrutural, dos modelo que integram HMMER-STRUCT, foi baseada em trabalhos prévios (CHAKRABARTI, SOWDHAMIMI, 2004; DEANE, PERDERSEN, et al., 2003; NISHIKAWA, OOI, 1986). No entanto, um estudo desses pesos torna-se necessário, para verificar se os modelos construídos são ótimos, ou seja, se a alteração desses pesos não acarretaria em melhor desempe-

nho. Além disso, o modelo HMM-3D, discutido na sessão 5.1.5 página 89, pode ser aprimorado, no sentido de representar estruturas tridimensionais e realizar comparações a nível espacial.

O pacote HMMER-STRUCT possui cinco modelos que são utilizados de forma independente, o objetivo é aproveitar o melhor de cada modelo e trabalhar com as informações disponíveis. Outros modelos produzidos por diferentes ferramentas podem ser acoplados ao pacote HMMER-STRUCT, inclusive ferramentas que implementam outras abordagens para a detecção de homologias distantes, tais como THMM (QIAN, GOLDSTEIN, 2004), PSI-BLAST (ALTSCHUL, MADDEN, et al., 2000), entre outras.

A sessão 5.3 mostrou que os melhores resultados foram obtidos combinando no mínimo três modelos. Uma análise adicional deve ser empregada no sentido de extrair regras lógicas que melhor combine os resultados dos pHMMs. Os pHMMs podem ter pesos diferente e os resultados podem ser unidos através de um *sistema de votação*. O sistema de votação deve ser individual para cada conjunto de homólogos, representados pelos pHMMs. A proposta é descobrir o melhor sistema de votação empregando técnicas de validação cruzada

Referências Bibliográficas

- ALBERTS, B., BRAY, D., LEWIS, J., RAFF, M., ROBERTS, K., , WATSON, J., 2002, *Molecular Biology of the Cell*, Garland Science.
- ALEXANDROV, V., GERSTEIN, M., 2004, “Using 3D Hidden Markov Models that explicitly represent spatial coordinates to model and compare protein structures”, *BMC Bioinformatics*, v. 5, n. 2, pp. 1–10.
- ALTMAN, R., GERSTEIN, M., 1995, “Finding an average core structures: application to the globins”, *Journal of Molecular Biology*, v. 251, n. 1, pp. 165–175.
- ALTSCHUL, F., GISH, W., MILLER, W., MYERS, E., , LIPMAN, D., 1990, “A basic local alignment search tool”, *Journal of Molecular Biology*, v. 215, n. 3, pp. 403–410.
- ALTSCHUL, S., MADDEN, T., SCHAFFER, A., ZHANG, J., ZHANG, Z., MILLER, W., , LIPMAN, D., 2000, “PSI-BLAST searches using hidden markov models of structural repeats: prediction of an unusual sliding DNA clamp and of beta-propellers in UV-damaged DNA-binding protein”, *Nucleic Acids Research*, v. 28, n. 18, pp. 3570–3580.
- ANDREEVA, A., HOWORTH, D., BRENNER, S., HUBBARD, T., CHOTHIA, C., , MURZIN, A., 2004, “SCOP database in 2004: refinements integrate structure and sequence family data”, *Nucleic Acids Research*, v. 32, n. 1, pp. 226–229.

- ATTWOOD, T., BRADLEY, P., FLOWER, D., GAULTON, A., MAUDLING, N.,
, MITCHELL, A., 2005, "A new progressive-iterative algorithm for multiple
structure alignment", *Bioinformatics*, v. 21, n. 15, pp. 3255–3263.
- BAE, K., MALLICK, B., ELSIK, C., 2005, "Prediction of protein interdomain linker
regions by a hidden Markov model", *Bioinformatics*, v. 21, n. 10, pp. 2264–2270.
- BAGOS, P., LIAKOPOULOS, T., HAMODRAKAS, S., 2002, "Faster gradient des-
cent training of hidden Markov models, using individual learning rate adapta-
tion", Technical report, Department of Cell Biology and Biophysics Faculty of
Biology, University of Athens.
- BAIROCH, A., BOECKMANN, B., FERRO, S., , GASTEIGER, E., 2005, "Swiss-
Prot: juggling between evolution and stability", *Briefings in Bioinformatics*, v.
5, n. 1, pp. 39–55.
- BALBI, P., CHAUVIN, Y., 1994, "Smooth on-line learning algorithms fo hidden
Markov models", *Neural Computation*, v. 6, n. 2, pp. 305–316.
- BALDI, P., BRUNAK, S., 2001, *Bioinformatics: The Machine Learning Approach*,
The Mit Press, Massachusetts USA.
- BARRETT, C., HUGHEY, R., KARPLUS, K., 1997, "Scoring hidden Markov Mo-
dels", *Computer Applications in the Biosciences*, v. 13, n. 2, pp. 191–199.
- BATEMAN, A., COIN, L., DURBIN, R., FINN, R., HOLLICH, V., GRIFFITHS-
JONES, S., KHANNA, A., MARSHALL, M., MOXON, S., SONNHAMMER,
E., STUDHOLME, D., YEATS, C., , EDDY, S., 2004, "The Pfam Protein
Families Database", *Nucleic Acids Research*, v. 32, n. 1, pp. 138–141.
- BAUM, L., 1972, "An equality and associated maximization technique in statistical
estimation for probabilistic functions of markov process", *Inequalities*, v. 12, n.
3, pp. 1–8.
- BECK, J., SHULTZ, E., 1986, "The use of relative operating characteristic (ROC)
curves in test performance evaluation", *Archives Pathology and Laboratory Me-
dicine*, v. 110, n. 1, pp. 13–20.

- BERGER, J., 1985, *Statistical Decision Theory and Bayesian Analysis*, Springer-Verlag, New York.
- BERTSEKAS, D., 1995, *Dynamic Programming and Optimal Control*, Athena Scientific.
- BILENKO, M., MOONEY, R., 2003, "On evaluation and training-set construction for duplicate detection".
- BIRD, A., 1982, "CpG islands as gene markers in the vertebrate nucleus", *Trends in Genetics*, v. 3, pp. 342–347.
- BOECKMANN, B., BAIROCH, A., APWEILER, R., BLATTER, M., ESTREICHER, A., GASTEIGER, E., MARTIN, M., MICHOUD, K., O'DONOVAN, C., PHAN, I., 2003, "The Swiss-Prot protein knowledgebase and its supplement TrEMBL in 2003", *Nucleic Acids Research*, v. 31, n. 1, pp. 365–370.
- BONANNO, J., ALMO, S., BRESNICK, A., CHANCE, M., FISER, A., SWAMINATHAN, S., JIANG, J., STUDIER, F., SHAPIRO, L., LIMA, C., GAASTERLAND, T., SALI, A., BAIN, K., FEIL, I., GAO, X., LORIMER, D., RAMOS, A., SAUDER, J., WASSERMAN, S., EMTAGE, S., DAMICO, K., BURLEY, S., 2005, "New York-Structural GenomiX Research Consortium (NYSGXRC): A Large Scale Center for the Protein Structure Initiative", *Journal of Structural and Functional Genomics*, v. 6, n. 3, pp. 225–232.
- BOURNE, P., WEISSIG, H., 2003, *Structural Bioinformatics*, chapter Fundamentals of protein structure, pp. 15–36 Sinauer Associates.
- BRANDEN, C., TOOZE, J., 1991a, *Introduction to protein Structure*, chapter The building blocks, pp. 3–9 Garland Publishing.
- BRANDEN, C., TOOZE, J., 1991b, *Introduction to protein Structure*, chapter Motifs of protein structure, pp. 11–29 Garland Publishing.
- BREJOVA, B., BROWN, D., LI, M., VINAR, T., 2005, "ExonHunter: a comprehensive approach to gene finding", *Bioinformatics*, v. 21, n. 1, pp. 57–65.

- BROWN, M., HUGHEY, R., KROGH, A., MIAN, I., SJÖLANDER, K., , HAUS-
SLER, D., 1993, “Using Dirichlet Mixture Priors to Derive Hidden Markov
Models for Protein Families”, In: Hunter, L., Searls, D., Shavlik, J., , editors,
Proc. of First Int. Conf. on Intelligent Systems for Molecular Biology, pp. 47–
55, Menlo Park, CA AAAI/MIT Press.
- BRYANT, D., MOULTON, V., 2004, “Neighbor-Net: An Agglomerative Method for
the Construction of Phylogenetic Networks”, *Molecular Biology and Evolution*,
v. 21, n. 2, pp. 255–265.
- BUCKLAND, M., GEY, F., 1994, “The relationship between Recall and Precision”,
Journal of the American Society for Information Science, v. 45, n. 1, pp. 12–19.
- BYSTROFF, C., BAKER, D., 2000, “HMMSTR: A hidden Markov model for local
sequence-structure correlation in proteins”, *Journal of Molecular Biology*, v. 301,
n. 18, pp. 173–190.
- CAMPROUX, A., TUFFERY, P., 2005, “Hidden Markov model-derived structural
alphabet for proteins: the learning of protein local shapes captures sequence
specificity”, *Biochimica et Biophysica Acta (BBA)*, v. 1724, n. 3, pp. 394–403.
- CATHY, H., WU, H., NIKOLSKAYA, A., HONGZHAN, H., LAI-SU, L., DAR-
REN, A., VINAYAKA, C., ZHANG-ZHI, H., MAZUMDER, R., SANDEEP,
K., KOURTESIS, P., LEDLEY, R., SUZEK, B., ARMINSKI, L., CHEN, Y.,
ZHANG, J., CARDENAS, J., CHUNG, S., CASTRO, J., DINKOV, G., , BAR-
KER, W., 2004, “PIRSF: family classification system at the Protein Information
Resource”, *Nucleic Acids Research*, v. 32, n. 1, pp. 112–114.
- CHAKRABARTI, S., BHARDWAJ, N., ANAND, P., , SOWDHAMINI, R., 2004,
“Improvement of alignment accuracy utilizing sequentially conserved motifs”,
BMC Bioinformatics, v. 5, n. 1, pp. 167–179.
- CHAKRABARTI, S., SOWDHAMIMI, R., 2004, “Regions of minimal structural va-
riation among members of protein domain superfamilies: application to remote

- homology detection and modelling using distant relationships”, *Bioinformatics*, v. 569, n. 1, pp. 31–36.
- CHAKRABARTI, S., SOWDHAMINI, R., 2004, “Regions of minimal structural variation among members of protein domain superfamilies: application to remote homology detection and modelling using distant relationships”, *FEBS*, v. 569, n. 1, pp. 31–36.
- CHOR, B., HENDY, M., HOLLAND, B., , PENNY, D., 2000, “Multiple Maxima of Likelihood in Phylogenetic Trees: An Analytic Approach”, *Molecular Biology and Evolution*, v. 17, n. 10, pp. 1529–1541.
- CHURCHILL, G., 1989, “Stochastic models for heterogeneous DNA sequences”, *Bulletin of Mathematical Biology*, v. 51, n. 1, pp. 79–94.
- COLLINS, M., 2002, “Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms”, Technical report, AT&T Lab Research New Jersey.
- CONANT, G., LEWIS, P., 2001, “Effects of Nucleotide Composition Bias on the Success of the Parsimony Criterion in Phylogenetic Inference”, *Molecular Biology and Evolution*, v. 18, n. 6, pp. 1024–1033.
- COOPER, G., 2000, *The Cell A Molecular Approach*, Sinauer Associates.
- COTTON, J., 2005, “Rates and patterns of gene duplication and loss in the human genome”, *Proceedings of the Royal Society of London*, v. 272, n. 1560, pp. 277–283.
- COYNE, J., ORR, H., 2004, *Speciation*, Sinauer Associates.
- CRAVEN, M., SLATTERY, F., 2001, “Relational learning with statistical prediction invention: better models for hypertext”, *Machine Learning*, v. 43, n. 1, pp. 97–119.
- DAVIDS, W., FUXELIUS, H., ANDERSSON, S., 2003, “Comparative and Functional Genomics”, *John Wiley & Sons, Ltd*, v. 4, n. 5, pp. 537–541.

- DAYHOFF, M., SCHWARTZ, R., ORCUTT, B., 1978, "A model of evolutionary change in proteins", *Atlas of Protein Sequence and Structure*, v. 5, n. 1, pp. 345–352.
- DEANE, C., PERDERSEN, J., LUNTER, G., 2003, "Insertions and deletions in protein alignment".
- DEGROOT, M., 1987, *Probability and Statistics*, Addison-Wesley.
- DENNIS, A., LIPMAN, D., OSTELL, J., WHEELER, D., 2004, "GenBank", *Nucleic Acids Research*, v. 33, n. 1, pp. 34–38.
- DESPER, R., GASCUEL, O., 2004, "Theoretical Foundation of the Balanced Minimum Evolution Method of Phylogenetic Inference and Its Relationship to Weighted Least-Squares Tree Fitting", *Molecular Biology and Evolution*, v. 21, n. 3, pp. 587–598.
- DURBIN, R., EDDY, S., KROGH, A., MITCHISON, G., 1998, *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*, Cambridge University Press, Cambridge UK.
- EDDY, S., 1996, "Hidden markov models", *Current Opinion in Structural Biology*, v. 6, n. 3, pp. 361–365.
- EDDY, S., 1997, "Maximum likelihood fitting of extreme value distribution", Technical report, Washington University School of medicine.
- EDDY, S., 1998, "Profile hidden Markov models", *Bioinformatics*, v. 14, n. 9, pp. 755–763.
- EDDY, S., 2003, *HMMER User Guide: Biological sequence analysis using profiles hidden Markov models*.
- EDGAR, C., 2004, "MUSCLE: multiple sequence alignment with high accuracy and high throughput", *Nucleic Acids Research*, v. 32, n. 5, pp. 1792–1797.

- EDGAR, R., SJOLANDER, K., 2004, "COACH: profile-profile alignment of protein families using hidden Markov models", *Bioinformatics*, v. 20, n. 8, pp. 1309–1318.
- ESPADALER, J., 2005, "Detecting remote related proteins by their interactions and sequence similarity", *PNAS*, v. 102, n. 20, pp. 7151–7156.
- ESPADALER, J., ARAGUES, R., ESWAR, N., MARTI-RENOM, M., QUEROL, E., AVILES, F., SALI, A., , OLIVA, B., 2005, "Detecting remotely related proteins by their interactions and sequence similarity", *National Academy of Sciences*, v. 102, n. 20, pp. 7151–7156.
- FAWCETT, T., 2004, "ROC Graphs: Notes and Practical Considerations for Researchers".
- GERSTEIN, M., ALTMAN, R., 1995, "Average core structures and variability measures for protein families: application to the immunoglobulins", *Journal of Molecular Biology*, v. 251, n. 1, pp. 165–175.
- GERSTEIN, M., LEVITT, M., 1996, "Using iterative dynamic programming to obtain accurate pairwise and multiple alignments of protein structures", *International Conference on Intelligent System for Molecular Biology*, v. 4, n. 6, pp. 59–67.
- GERSTEIN, M., SONNHAMMER, E., CHOTHIA, C., 1994, "Volume changes in protein evolution", *Journal of Molecular Biology*, v. 236, n. 4, pp. 1067–1078.
- GOLDSTEIN, L., WATERMAN, M., 1997, "Approximation to profile score distribution", *Journal of computational biology*, v. 1, n. 2, pp. 93–104.
- GOUGH, J., 2002, *Hidden Markov Models and their application to genome analysis in the context of protein structure*, Ph.D. thesis, Sidney Sussex College.
- GOUGH, J., KARPLUS, K., HUGHEY, R., , CHOTHIA, C., 2001, "Assignment of homology to genome sequences using a library of hidden Markov models that represent all proteins of known structure", *Journal of Molecular Biology*, v. 313, n. 4, pp. 903–919.

- GOYON, F., TUFFÉRY, P., 2004, “SA-Search: A web tool for protein structure mining based on structural alphabet”, *Nucleic Acids Research*, v. 32, pp. 545–548.
- GRIBSKOV, M., MCLACHLAN, A., EISENBERG, D., 1987, “Profile analysis: detection of distantly related proteins”, *National Academy of Sciences*, v. 84, n. 1, pp. 4355–4358.
- GRUNDY, W., BAKER, M., 1997, “Meta-MEME: Motif-based Hidden Markov Models of Protein Families”, *Computer Applications in the Biosciences*, v. 13, n. 4, pp. 397–406.
- GUDA, C., LU, S., SCHEEFF, E., BOURNE, E., , SHINDYALOV, I., 2004, “CE-MC: a multiple protein structure alignment server”, *Nucleic Acids Research*, v. 32, n. 2, pp. 100–103.
- HAFT, D., SELENGUT, J., WHITE, O., 2003, “The TIGRFAMs database of protein families”, *Nucleic Acids Research*, v. 31, n. 1, pp. 371–373.
- HAYKIN, S., 2001, *Redes Neurais: Princípios e prática*, Bookman.
- HELEN, M., WESTBROOK, J., FENG, Z., GILLILAND, G., BHAT, T., WEISSIG, H., SHINDYALOV, I., , BOURNE, P., 2000, “The Protein Data Bank”, *Nucleic Acids Research*, v. 28, n. 1, pp. 235–242.
- HENIKOFF, S., HENIKOFF, J., 1991, “Automated assembly of protein blocks for database searching”, *Nucleic Acids Research*, v. 19, n. 23, pp. 6565–6572.
- HENIKOFF, S., HENIKOFF, J., 1992, “Amino Acid Substitution Matrices from Protein Blocks”, *Proceedings of the National Academy of Sciences*, v. 89, n. 1, pp. 10915–10919.
- HENIKOFF, S., HENIKOFF, J., PIETROKOVSKI, S., 1999, “Blocks+: A non-redundant database of protein alignment blocks derived from multiple compilations”, *Bioinformatics*, v. 15, n. 6, pp. 471–479.

- HIGGINS, D., TAYLOR, W., 2000, *Bioinformatics: Sequence, structure and data-banks*, chapter Comparison of protein three-dimensional structures, pp. 15–46 Oxford University Press.
- HONGZHAN, H., WINONA, C., CHEN, C. Y., , WU, C. H., 2003, “iProClass: an integrated database of protein family, function and structure information”, *Nucleic Acids Research*, v. 31, n. 1, pp. 390–392.
- HOU, Y., HSU, W., LEE, M., , BYSTROFF, C., 2004a, “Remote homolog detection using local sequence-structure correlations”, *Journal of Molecular Biology*, v. 340, n. 2, pp. 385–395.
- HOU, Y., HSU, W., LEE, M., , BYSTROFF, C., 2004b, “Remote homology detection using local sequence-structure correlations”, *PROTEINS: Structure, Function and Bioinformatics*, v. 57, n. 3, pp. 518–530.
- HUANG, X., MILLER, W., 1991, “A Time-Efficient, Linear-Space Local Similarity Algorithm”, *Advances in Applied Mathematics*, v. 12, n. 1, pp. 337–357.
- HUANG, X., ZHANG, J., 1996, “Methods for comparing a DNA sequence with a protein sequence”, *Computer Applications in the Biosciences*, v. 12, n. 6, pp. 497–506.
- HUGHEY, R., KARPLUS, K., KROGH, A., 2003, *SAM: Sequence Alignment and Modeling*.
- HUGHEY, R., KROGH, A., 1996a, “Hidden Markov models for sequence analysis: extension and analysis of the basic method”, *Computer Applications in the Biosciences*, v. 12, n. 2, pp. 95–107.
- HUGHEY, R., KROGH, A., 1996b, “Hidden markov models for sequence analysis: extension and analysis of the basic method”, *Computer Applications in the Biosciences*, v. 12, n. 2, pp. 95–107.
- HULO, N., SIGRIST, C., SAUX, V., BORDOLI, L., GATTIKER, A., CASTRO, E., BUCHER, P., , BAIROCH, A., 2004, “Recent improvements to the PROSITE database”, *Nucleic Acids Research*, v. 32, n. 1, pp. 134–137.

- ISLAS, S., BECERRA, A., LUISI, P., , LAZCANO, A., 2004, “Comparative genomics and the gene complement of a minimal cell”, *Origin of Life Evolution*, v. 34, n. 1, pp. 243–256.
- J. HUELSENBECK, F. R., 2001, “MRBAYES: Bayesian inference of phylogenetic tree”, *Bioinformatics*, v. 17, n. 8, pp. 754–755.
- JONES, S., BATEMAN, A., 2002, “The use of structure information to increase alignment accuracy does not aid homologue detection with profiles HMMs”, *Bioinformatics*, v. 18, n. 9, pp. 1243–1249.
- WANG, K., SAMUDRALA, R., 2005, “FSSA: A novel method for identifying functional signatures from structural alignments”, *Bioinformatics*, v. 21, n. 13, pp. 2969–2977.
- KARLIN, S., ALTSCHUL, S., 1990, “Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes”, *Proceedings of the National Academy of Sciences USA*, v. 87, n. 6, pp. 2264–2268.
- KARPLUS, K., 1995, “Regularizers for estimation distributions of aminoacids from small samples”, Technical report, University of California.
- KARPLUS, K., BARRET, C., HUGHEY, R., 1998, “Hidden Markov models for detecting remote protein homologies”, *Bioinformatics*, v. 14, n. 10, pp. 846–856.
- KARPLUS, K., KARCHIN, R., SHACKELFORD, G., , HUGHEY, R., 2005, “Calibrating E-values for hidden Markov models with reverse-sequence null models”, *Bioinformatics*, v. 6, n. 2, pp. 305–316.
- KAZUTAKA, K., KAZUHARU, M., KEI-ICHI, K., , TAKASHI, M., 2002, “MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform”, *Nucleic Acids Research*, v. 30, n. 14, pp. 3059–3066.
- KIRKPATRICK, M., BARTON, N., 2004, “Chromosome inversions, local adaptation, and speciation”, *Molecular Biology and Evolution*, v. 21, n. 10, pp. 1820–1830.

- KLOTZ, I., LANGERMAN, N., DARNALL, D., 1970, "Quarternary structure of proteins", *Annual Review of Biochemistry*, v. 39, n. 2, pp. 25–62.
- KNUDSEN, B., MIYAMOTO, M., 2003, "Sequence alignments and pair hidden Markov models using evolutionary history", *Journal of Molecular Biology*, v. 333, n. 2, pp. 453–460.
- KRETSINGER, R., 1980, "Structure and evolution of calcium modulated proteins", *CRC Critical Reviews in Biochemistry*, v. 8, n. 2, pp. 119–174.
- KROGH, A., 1994, "Hidden Markov models for labeled sequences", In: *Proceedings of the 12th IAPR International Conference on Pattern Recognition*, pp. 140–144 IEEE Computer Society Press.
- KROGH, A., BROWN, M., MIAN, I., SJOLANDER, K., , HAUSSLER, D., 1994, "Hidden markov models in computational biology applications to protein modeling", *Journal of Molecular Biology*, v. 235, n. 5, pp. 1501–1531.
- KROGH, A., MITCHISON, G., 1995, "Maximum entropy weighting of aligned sequences of protein or DNA", *Proceedings of the third international conference on intelligent system for molecular biology*, v. 3, pp. 215–221.
- KUMAR, S., TAMURA, K., NEI, M., 2004, "MEGA3: Integrated Software for Molecular Evolutionary Genetics Analysis and Sequence Alignment", *Briefings in Bioinformatics*, v. 5, n. 1, pp. 150–163.
- LANDER, E., LINTON, L., BIRREN, B., NUSBAUM, C., ZODY, M., BALDWIN, J., DEVON, K., DEWAR, K., DOYLE, M., FITZHUGH, W., , FUNKE, R., 2001, "Initial sequencing and analysis of the human genome", *Nature*, v. 409, n. 6822, pp. 860–921.
- LARGET, B., SIMON, D., 2005, "Markov Chasin Monte Carlo Algorithms for the Bayesian Analysis of Phylogenetic Trees", *Molecular Biology and Evolution*, v. 16, n. 6, pp. 750–759.
- LAZIKANI, B., JUNG, J., XIANG, Z., , HONIG, B., 2001, "Protein structure prediction", *Current Opinion in Chemical Biology*, v. 5, n. 1, pp. 51–56.

- LEE, B., RICHARDS, F., 1971, "The interpretation of protein structure: estimation of static accessibility", *Journal of Molecular Biology*, v. 14, n. 3, pp. 379–400.
- LETUNIC, I., COPLEY, R., SCHMIDT, S., CICCARELLI, F., DOERKS, T., SCHULTZ, J., PONTING, C., BORK, P., 2004, "SMART 4.0: towards genomic data integration", *Nucleic Acids Research*, v. 32, n. 1, pp. 142–144.
- LIN, K., SIMOSSIS, V., TAYLOR, W., HERINGA, J., 2005, "A simple and fast secondary structure prediction method using hidden neural networks", *Bioinformatics*, v. 21, n. 2, pp. 152–159.
- LUPYAN, D., LEO-MACIAS, A., 2005, "A new progressive-iterative algorithm for multiple structure alignment", *Bioinformatics*, v. 21, n. 15, pp. 3255–3263.
- MADERA, M., GOUGH, J., 2002, "A comparison of profile hidden Markov model procedure for remote homology detection", *Nucleic Acids Research*, v. 30, n. 19, pp. 4321–4328.
- MADERA, M., VOGEL, C., KUMMERFELD, S., CHOTHIA, C., GOUGH, J., 2004, "The SUPERFAMILY database in 2004: additions and improvements", *Nucleic Acids Research*, v. 32, n. 1, pp. 235–239.
- MAJOROS, W., PERTEA, M., SALZBERG, S., 2005, "Efficient implementation of a generalized pair hidden Markov model for comparative gene finding", *Bioinformatics*, v. 21, n. 9, pp. 1782–1788.
- MAMITSUKA, H., 2005, "Finding the biologically optimal alignment of multiple sequences", *Artificial Intelligence in Medicine*, v. 35, n. 2, pp. 9–18.
- MATSUO, Y., BRYANT, S., 1999, "Identification of homologous core structures", *Proteins*, v. 35, n. 1, pp. 70–79.
- MENDEL, M., 1992, "A commercial large-vocabulary discrete speech recognition system: Dragon Dictate", *Language Speech*, v. 35, n. 1, pp. 237–246.
- METZ, C., 1978, "Basic principles of ROC analysis", *Seminar in Nuclear Medicine*, v. 8, n. 4, pp. 283–298.

- MITCHELL, T., 1997, *Machine Learning*, McGraw-Hill.
- MIZUGUCHI, K., DEANE, C., BLUNDELL, T., JOHNSON, M., , OVERINGTON, J., 1998, "JOY: protein sequence-structure representation and analysis", *Bioinformatics*, v. 14, n. 7, pp. 617–623.
- MOULT, J., FIDELIS, K., ZEMLA, A., , HUBBARD, T., 2003, "Critical assessment of methods of protein structure prediction (CASP)", *Fifth Meeting on the Critical Assessment of Techniques for Protein Structure Prediction*, v. 53, n. 6, pp. 334–339.
- MOULTON, G., NORDLE, A., PAINE, K., TAYLOR, P., UDDIN, A., , ZYGOURI, C., 2003, "PRINTS and its automatic supplement, prePRINTS", *Nucleic Acids Research*, v. 31, n. 1, pp. 400–402.
- NEEDLEMAN, S., WUNSCH, C., 1970, "A general method applicable to the search for similarities in the amino acid sequence of two proteins", *Journal of Molecular Biology*, v. 48, n. 1, pp. 443–453.
- NEI, M., KUMAR, S., 2000, *Molecular Evolution and Phylogenetics*, Oxford University Press.
- NISHIKAWA, K., OOI, T., 1986, "Radial locations of amino acid residues in a globular protein: correlation with the sequence", *Journal of Biochemistry*, v. 100, n. 4, pp. 1043–1047.
- NOTREDAME, C., HIGGINS, D., HERINGA, J., 2000, "T-coffee: a novel method for fast and accurate multiple sequence alignment", *Computer Applications in the Biosciences*, v. 302, n. 1, pp. 205–217.
- PARK, J., KARPLUS, K., BARRETT, C., HUGHEY, R., HAUSSLER, D., HUBBARD, T., , CHOTHIA, C., 1998, "Sequence comparisons using multiples sequence detect three times as many remote homologues as pairwise methods", *Journal of Molecular Biology*, v. 284, n. 4, pp. 1201–1210.

- PAULING, L., COREY, R., BRANSON, H., 1951, "The structure of protein: two hydrogen-bonded helical configurations of the polypeptide chain", *National Academy of Sciences*, v. 37, n. 1, pp. 205–211.
- PEARSON, W., 1985, "Rapid and sensitive sequence comparisons with FASTP and FASTA", *Methods Enzymol*, v. 183, n. 1, pp. 63–98.
- PETREY, D., XIANG, Z., TANG, C., XIE, L., GIMPELEV, M., MITROS, T., SOTO, C., GOLDSMITH-FISCHMAN, S., KERNYTSKY, A., SCHLESSINGER, A., KOH, I., ALEXOV, E., , HONIG, B., 2003, "Using multiple structure alignments, fast model building, and energetic analysis in fold recognition and homology modeling", *Proteins*, v. 53, n. 6, pp. 430–435.
- PORTO, A., BARBOSA, V., 2005, "A methodology for determining amino-acid substitution matrices from set covers".
- PRESS, W., TEUKOLSKY, S., VETTERLING, W., , FLANNERY, B., 1992, *Numerical Recipes in C*, Cambridge University Press, Cambridge UK.
- QIAN, B., GOLDSTEIN, R., 2004, "Performance of an iterated T-HMM for homology detection", *Bioinformatics*, v. 20, n. 14, pp. 2175–2180.
- RABINER, L., 1989, "A tutorial on hidden Markov models and selected applications in speech recognition", pp. 267–296.
- RUSSEL, S., NORVIG, P., 2002, *Artificial Intelligence: A Modern Approach*, Prentice-Hall.
- S. HENIKOFF, J. H., 1994, "Position-based sequence weights", *Molecular Biology*, v. 243, n. 4, pp. 574–578.
- SANTNER, T., DUFFY, D., 1989, *The Statistical Analysis of Discret Data*, Springer-Verlag, New York.
- SCOTT, M., MATSUDAIRA, P., LODISH, H., DARNELL, J., ZIPURSKY, L., KAISER, C., BERK, A., , KRIEGER, M., 2000, *Molecular Cell Biology*, Von Hoffman Press.

- SETUBAL, J., MEIDANIS, J., 1997, *Introduction to Computational Molecular Biology*, PWS.
- SIBBALD, P., ARGOS, P., 1990, "Weighting aligned protein or nucleic acid sequences to correct for unequal representation", *Journal of Molecular Biology*, v. 216, n. 4, pp. 813–818.
- SIEPEL, A., HAUSSLER, D., 2004, "Phylogenetic Estimation of Context-Dependent Substitution Rates by Maximum Likelihood", *Molecular Biology and Evolution*, v. 21, n. 3, pp. 468–488.
- SJÖLANDER, K., 1997, *A bayesian-information theoretic method for evolutionary inference in proteins*, Ph.D. thesis, University of California.
- SJOLANDER, K., KARPLUS, K., BROWN, M., HUGHEY, R., KROGH, A., MIAN, I., , HAUSSLER, D., 1996, "Dirichlet mixtures: a method for improving detection of weak but significant protein sequence homology", *Computer Applications in the Biosciences*, v. 12, n. 4, pp. 327–345.
- SMITH, N., WEBSTER, M., ELLEGREN, H., 2002, "Deterministic mutation rate variation in the human genome", *Genome Research*, v. 12, n. 9, pp. 1350–1356.
- SMITH, T., WATERMAN, M., 1981, "Identification of common molecular subsequences", *Journal of Molecular Biology*, v. 147, n. 1, pp. 195–197.
- SÖDING, J., 2005, "Protein Homology detection by HMM-HMM comparison", *Bioinformatics*, v. 21, n. 7, pp. 951–960.
- SPENCER, M., SUSKO, E., ROGER, A., 2005, "Likelihood, Parsimony, and Heterogeneous Evolution", *Molecular Biology and Evolution*, v. 22, n. 5, pp. 1161–1164.
- SULLIVAN, O., SUHRE, K., ABERGEL, C., HIGGINS, D., , NOTREDAME, C., 2004, "3DCoffee: combining protein sequences and structures within multiple sequence alignments", *Journal of Molecular Biology*, v. 340, n. 2, pp. 385–395.

- TAKEDA-SHITAKA, M., TAKAYA, D., CHIBA, C., TANAKA, H., , UMEYAMA, H., 2004, "Protein structure prediction in structure based drug design", *Current Medicinal Chemical*, v. 11, n. 5, pp. 551–558.
- TANG, C., XIE, L., KOH, I., POSY, S., ALEXOV, E., , B, B. H., 2003, "On the role of structural information in remote homology detection and sequence alignment: new methods using hybrid sequence profiles", *Jornal of Molecular Biology*, v. 334, n. 5, pp. 1043–1062.
- TATUSOV, R., 1994, "Detection of conserved segments in proteins: iterative scanning of sequence databases with alignment blocks", *National Academy of Sciences*, v. 91, n. 1, pp. 12091–12095.
- THOMPSON, J., GIBSON, T., 1994a, "CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice", *Computer Applications in the Biosciences*, v. 22, n. 22, pp. 4673–4680.
- THOMPSON, J., GIBSON, T., 1994b, "Improved sensitivity of profile searches through the use of sequence weights and gap excision", *Computer Applications in the Biosciences*, v. 10, n. 1, pp. 19–29.
- VINGRON, M., ARGOS, P., 1990, "A fast and sensitive multiple sequence alignment algorithm", *Computer Applications in the Biosciences*, v. 5, n. 2, pp. 115–121.
- VLADIMIR, B., TAN, S., SUZUKI, Y., , SUGANO, S., 2004, "Promoter prediction analysis on the whole human genome", *Nature Biotechnology*, v. 22, n. 1, pp. 1467–1473.
- WALLE, I., I., WYNS, L., 2004, "Align-m: a new algorithm for multiple alignment of highly divergent sequences Source", *Bioinformatics*, v. 20, n. 9, pp. 1428–1435.
- WANG, B., ADAMS, M., DAILEY, H., DELUCAS, L., LUO, M., ROSE, J., BUNZEL, R., DAILEY, T., HABEL, J., HORANYI, P., JENNEY, F., KATAEVA, I., LEE, H., LI, S., LI, T., LIN, D., LIU, Z., LUAN, C., MAYER, M., NAGY, L., NEWTON, M., NG, J., POOLE, F., SHAH, A., SHAH, C., SUGAR, F., , XU,

- H., 2005, "Protein Production and Crystallization at SECSG - An Overview", *Journal of Structural and Functional Genomics*, v. 6, n. 3, pp. 233–243.
- WANG, J., HANNENHALLI, S., 2005, "Generalizations of Markov model to characterize biological sequences", *BMC Bioinformatics*, v. 6, n. 219, pp. 4–8.
- WEBSTER, M., SMITH, N., LERCHER, M., , ELLEGREN, H., 2004, "Gene expression, synteny, and local similarity in human noncoding mutation rates", *Molecular Biology and Evolution*, v. 21, n. 10, pp. 1820–1830.
- WISTRAND, M., SONNHAMMER, E., 2005, "Improved profile HMM performance by assessment of critical algorithmic in SAM and HMMER", *BMC Bioinformatics*, v. 6, n. 1, pp. 99.
- XU, W., MIRANKER, D., 2004, "A metric model of amino acid substitution", *Bioinformatics*, v. 20, n. 8, pp. 1214–1221.
- YANG, A., WANG, L., 2003, "Local structure prediction with local structure-based sequence profiles", *Proteins*, v. 19, n. 10, pp. 1267–1274.
- YANG, Z., RANNALA, B., 1997, "Bayesian phylogenetic inference using DNA sequences: a Markov Chain Monte Carlo Method", *Molecular Biology and Evolution*, v. 14, n. 7, pp. 717–724.
- YIN, P., HARTEMINK, A., 2005, "Theoretical and practical advances in genome halving", *Bioinformatics*, v. 21, n. 7, pp. 277–283.
- YU, Y., ALTSCHUL, S., 2005, "The construction of amino acid substitution matrices for the comparison of proteins with non-standard compositions", *Bioinformatics*, v. 21, n. 7, pp. 902–911.
- ZAHA, A., 2003, *Biologia molecular básica*, chapter A célula e seus componentes moleculares, pp. 13–34 Mercado Aberto.