Cadeias estocásticas parcimoniosas com aplicações à classificação e filogenia das seqüências de proteínas

Florencia Graciela Leonardi

TESE APRESENTADA
AO
PROGRAMA INTERUNIDADES EM BIOINFORMÁTICA
DA
UNIVERSIDADE DE SÃO PAULO
PARA
OBTENÇÃO DO GRAU DE DOUTOR
EM
CIÊNCIAS

Área de Concentração: **Bioinformática**
Orientador: **Prof. Dr. Antonio Galves**
Co-orientador: **Prof. Dr. Hugo Aguirre Armelin**

Durante a elaboração deste trabalho a autora recebeu apoio financeiro da CAPES.

- São Paulo, março de 2007 -

Cadeias estocásticas parcimoniosas com aplicações à classificação e filogenia das seqüências de proteínas

Este exemplar corresponde à redação final
da tese de doutorado devidamente
corrigida e defendida por
Florencia Graciela Leonardi
e aprovada pela comissão julgadora.

São Paulo, 6 de março de 2007.

Banca examinadora:

- Prof. Dr. Antonio Galves (orientador) - IME-USP.
- Prof. Dr. Roberto Fernández - Université de Rouen, França.
- Profa. Dra. Nancy Lopes Garcia - IMECC-UNICAMP.
- Prof. Dr. Ricardo Fraiman - Universidad de San Andrés, Argentina.
- Prof. Dr. Shaker Chuck Farah - IQ-USP.

Dedico esta tese a Claudio
e aos meus pais, Nélida e Eduardo.

Agradecimentos

Agradeço ao Prof. Antonio Galves pela sua orientação e amizade em todo momento. A sua generosidade e entusiasmo foram um grande incentivo para mim nestes anos.

Agradeço também ao Prof. Hugo Armelin pelo seu apoio e pelas suas idéias, que foram muito importantes para o desenvolvimento deste trabalho.

Sou grata aos Professores Junior Barrera e Hernando del Portillo por ter me ajudado no começo deste doutorado.

Agradeço a todas as pessoas que me ajudaram nestes quatro anos no Brasil, especialmente a Mariela, Silvina e Natalia por ter aberto as portas das suas casas quando precisei.

Finalmente, agradeço aos meus pais, Nélida e Eduardo, pelo seu apoio incondicional e a Claudio por continuar fazendo parte da minha vida.

Resumo

Nesta tese apresentamos alguns resultados teóricos e práticos da modelagem de seqüências simbólicas com cadeias estocásticas parcimoniosas. As cadeias estocásticas parcimoniosas, que incluem as cadeias estocásticas de memória variável, constituem uma generalização das cadeias de Markov de alcance fixo. As seqüências simbólicas às quais foram aplicadas as ferramentas desenvolvidas são as cadeias de aminoácidos. Primeiramente, introduzimos um novo algoritmo, chamado de SPST, para selecionar o modelo de cadeia estocástica parcimoniosa mais ajustado a uma amostra de seqüências. Em seguida, utilizamos esse algoritmo para estudar dois importantes problemas da genômica; a saber, a classificação de proteínas em famílias e o estudo da evolução das seqüências biológicas. Finalmente, estudamos a velocidade de convergência de algoritmos relacionados com a estimação de uma subclasse das cadeias estocásticas parcimoniosas, as cadeias estocásticas de memória variável. Assim, generalizamos um resultado prévio de velocidade exponencial de convergência para o algoritmo PST, no caso de cadeias de memória ilimitada. Além disso, obtemos um resultado de velocidade de convergência para uma versão generalizada do Critério da Informação Bayesiana (BIC), também conhecido como Critério de Schwarz.

Abstract

In this thesis we present some theoretical and practical results, concerning symbolic sequence modeling with parsimonious stochastic chains. Parsimonious stochastic chains, which include variable memory stochastic chains, constitute a generalization of fixed order Markov chains. The symbolic sequences modeled with parsimonious stochastic chains were the sequences of amino acids. First, we introduced a new algorithm, called SPST, to select the model of parsimonious stochastic chain that fits better to a sample of sequences. Then, we use the SPST algorithm to study two important problems of genomics. These problems are the classification of proteins into families and the study of the evolution of biological sequences. Finally, we found upper bounds for the rate of convergence of some algorithms related with the estimation of a subclass of parsimonious stochastic chains; namely, the variable memory stochastic chains. In consequence, we generalize a previous result about the exponential rate of convergence of the PST algorithm, in the case of unbounded variable memory stochastic chains. On the other hand, we proved a result about the rate of convergence of a generalized version of the Bayesian Information Criterion (BIC), also known as Schwarz' Criterion.

Conteúdo

Introdução

As proteínas têm um papel fundamental em quase todas as atividades que se realizam dentro de cada célula viva. As proteínas estão compostas por moléculas menores, chamadas de *aminoácidos*, dos quais existem vinte diferentes tipos e que estão ligados um depois do outro de uma maneira linear, constituindo uma seqüência específica para cada proteína. Esta seqüência está codificada no DNA da célula, em genes específicos, que são transcritos em RNA, o qual é traduzido para formar a seqüência linear de aminoácidos. Dependendo desta seqüência, cada proteína adota uma estrutura tri-dimensional específica, que lhe permite, junto às propriedades físico-químicas dos aminoácidos que a constituem, realizar sua função dentro da célula.

As seqüências de proteínas podem ser determinadas facilmente na atualidade, usando diretamente a molécula de proteína ou através do seqüenciamento dos genes que codificam a seqüência. Contudo, a estrutura de uma proteína é difícil de observar experimentalmente com as tecnologias existentes. De fato, a estrutura de somente uma pequena fração das proteínas conhecidas tem sido resolvida até agora. Dada uma seqüência de proteína isolada, não temos atualmente a capacidade de deduzir a conformação tri-dimensional que ela adotará. E sem um modelo estrutural adequado é muito difícil predizer a função dessa proteína, o que constitui um dos nossos principais objetivos.

Diferentes seqüências de proteínas, tanto dentro de um único organismo como em diferentes organismos, possuem similaridades notáveis. De fato, as proteínas conhecidas podem ser organizadas de uma maneira hierárquica, baseada na aparente conservação da seqüência, estrutura ou função. Com tantos projetos de seqüenciamento de genomas ao redor do mundo, milhares de novos e hipotéticos genes estão sendo descobertos diariamente. Ao redor de um milhão de seqüências de proteínas são conhecidas na atualidade, tornando impossível a análise manual desse conjunto cada vez maior.

O objetivo desta tese é apresentar novas ferramentas matemáticas e computacionais para a análise das seqüências de proteínas. Essa análise está dirigida tanto ao estudo da função quanto ao estudo da evolução das seqüências de proteínas. As ferramentas matemáticas apresentadas estão inseridas dentro do marco de referência da análise estocástica de seqüências simbólicas. Essa abordagem está baseada na inferência de padrões probabilísticos de dependências locais

entre os aminoácidos que constituem as proteínas de um determinado grupo ou *família*.

Mais especificamente, dada uma seqüência simbólica $X_1, X_2, \ldots$ assumindo valores num alfabeto finito (por exemplo, $X_1, X_2, \ldots$ poderia ser uma cadeia de aminoácidos), tentamos predizer cada novo símbolo X_n como função do *passado* $X_1, \ldots, X_{n-1}$. Assumindo que o passado relevante tem um comprimento finito e fixo k, isso poderia ser feito estimando as probabilidades de transição

$$\mathbb{P}(X_n = x_n \mid X_{n-1} = x_{n-1}, \ldots, X_{n-k} = x_{n-k}).$$

Esse modelo, chamado de cadeias de Markov de alcance k, tem a desvantagem de que o número de parâmetros que devem ser estimados cresce exponencialmente com a ordem k. Mas poderia acontecer que o valor de k não fosse fixo, senão que variasse com cada passado e, portanto, com as cadeias de Markov de alcance fixo estaríamos estimando muitas probabilidades que na verdade seriam iguais. Assim surgiram as *cadeias estocásticas de memória variável* (Rissanen, 1983), nas quais apenas uma porção do passado é relevante para a determinação do próximo símbolo. O comprimento dessa porção relevante varia de um passado para outro (isso explica o nome de cadeia de *memória variável*). Rissanen chama a porção relevante do passado de *contexto*. O conjunto de contextos de uma cadeia estocástica de memória variável pode ser representada por uma árvore, chamada de *árvore de contextos*.

O modelo de cadeia estocástica de memória variável será um dos exemplos de cadeias parcimoniosas estudadas nesta tese. Desde a sua introdução por Rissanen, no contexto de teoria da informação, as cadeias estocásticas de memória variável têm sido estudadas e utilizadas na modelagem de vários problemas práticos. Como exemplos, dentro da análise de seqüências biológicas, podemos citar as aplicações à identificação de genes (Bülhmann & Wyner, 1999), à classificação de proteínas (Bejerano & Yona, 2001) e à identificação de domínios em proteínas (Bejerano et al., 2001), entre outras. O bom desempenho desse modelo levou à introdução de generalizações ou modificações das cadeias estocásticas de memória variável. Exemplo disso é a generalização das cadeias estocásticas de memória variável para modelar seqüências esparsas; isto é, seqüências nas quais posições relevantes podem estar separadas por outras irrelevantes. Nesse caso podemos citar o modelo de *transductores markovianos esparsos*, utilizado para classificar seqüências de proteínas em Eskin et al. (2000).

No caso específico das seqüências de proteínas é bem conhecido que alguns aminoácidos possuem propriedades físico-químicas semelhantes. Isso possibilitaria que em determinadas posições alguns símbolos pudessem ser substituídos por outros do mesmo tipo, sem mudar a estrutura da proteína. Nesse caso, os contextos estariam dados por seqüências de subconjuntos do alfabeto. Essa é uma generalização dos transductores markovianos esparsos, onde só são

permitidos o conjunto total ou conjuntos unitários nas diferentes posições dos contextos. Assim surgiram as cadeias estocásticas parcimoniosas, que incluem tanto as cadeias estocásticas de memória variável e, portanto, também incluem as cadeias de Markov de alcance fixo, quanto os transductores markovianos esparsos.

Em geral, a estimação das probabilidades de transição, para qualquer tipo de modelo considerado, faz-se utilizando os estimadores de máxima verossimilhança. Esses estimadores têm uma forma bem conhecida. Portanto, o problema de estimação interessante não é a estimação das probabilidades de transição e sim a estimação da árvore de contextos que define o espaço de parâmetros. No caso da estimação de árvores de contextos para cadeias estocásticas de memória variável existem, basicamente, duas abordagens. Poderíamos chamar a essas duas abordagens de *estimação local* e *estimação global*. Os métodos de estimação local, que incluem o algoritmo Contexto (Rissanen, 1983) e o algoritmo PST (Ron et al., 1996), estão baseados na comparação das distribuições associadas a cada ramo terminal da árvore com a do seu ancestral mais próximo. Essa comparação é feita individualmente para cada ramo e de forma seqüencial. Portanto, a decisão sobre um ramo não afeta futuras decisões sobre os outros ramos da árvore. Por outro lado, os métodos de estimação global estão baseados na comparação de todas as árvores de contextos "possíveis", utilizando um critério que é aplicado sobre a árvore como um todo. Os critérios mais conhecidos utilizados na comparação das árvores são o da maximização da verossimilhança penalizada, como no caso do Critério da Informação Bayesiana (BIC), também conhecido como Critério de Schwarz (Schwarz, 1978) e o da maximização da probabilidade *a posteriori*, como no caso da abordagem bayesiana. Baseados na metodologia bayesiana foram propostos algoritmos para estimar a árvore do modelo de transductores markovianos esparsos (Eskin et al., 2000) e também para o modelo de cadeias estocásticas parcimoniosas em geral (Bourguignon & Robelin, 2004). A grande desvantagem desses algoritmos é o seu alto custo computacional. Por exemplo, o algoritmo apresentado em Bourguignon & Robelin (2004) não pode ser utilizado com alfabetos de mais de cinco símbolos, o que torna impossível a sua utilização na modelagem de seqüências de proteínas.

Para finalizar, apresentamos os principais resultados originais da tese, os quais foram organizados da seguinte forma:

- No Capítulo 1 introduzimos um novo algoritmo, chamado de SPST (do inglês *Sparse Probabilistic Suffix Trees*), para estimar a árvore de contextos de uma cadeia estocástica parcimoniosa. Esse algoritmo está inserido dentro da categoria de algoritmos locais de estimação, como descrito anteriormente.

- No Capítulo 2 apresentamos os resultados da aplicação do algoritmo SPST para classificar

seqüências de proteínas em famílias. Também introduzimos uma modificação da etapa de predição do algoritmo, chamada de F-SPST. Esses resultados são comparados aos obtidos por Bejerano & Yona (2001) com o algoritmo PST e foram publicados, junto com a definição dos algoritmos SPST e F-SPST, primeiramente em Leonardi & Galves (2005), em formato de resumo estendido, e posteriormente em Leonardi (2006) como artigo completo.

- No Capítulo 3 apresentamos uma aplicação do algoritmo SPST para o estudo da filogenia das seqüências de proteínas. Para isso é utilizada uma distância entre as árvores de contextos, introduzida em Simovici & Szymon (2006). O estudo é feito em seqüências de globinas e de fatores de crescimento de fibroblastos (FGF). Nos resultados mostramos que essa abordagem consegue identificar relações entre as seqüências que já tinham sido inferidas também por outros métodos.

- No Capítulo 4 demonstramos a velocidade exponencial de convergência do algoritmo PST, introduzido por Ron et al. (1996), no caso de cadeias estocásticas de memória variável, porém ilimitada. Esse resultado generaliza o já obtido para árvores finitas por Galves et al. (2006).

- Finalmente, no Capítulo 5 demonstramos um resultado de velocidade de convergência para um estimador global da árvore de contextos de uma cadeia estocástica de memória variável, com memória não necessariamente limitada. Esse estimador é uma versão generalizada do Critério da Informação Bayesiana (BIC), ou Critério de Schwarz (Schwarz, 1978), permitindo diferentes termos de penalização.

Capítulo 1

Cadeias estocásticas parcimoniosas

Neste capítulo introduzimos um novo algoritmo, chamado de SPST, para estimar a árvore de contextos de uma cadeia estocástica parcimoniosa. Esse algoritmo integra a categoria de algoritmos locais de estimação, como descrito na Introdução. Para compreender melhor o contexto teórico onde estão inseridos o modelo de cadeia estocástica parcimoniosa e o algoritmo SPST, definimos primeiramente alguns conceitos básicos sobre as cadeias estocásticas de memória variável e a sua estimação. Em particular, apresentamos o algoritmo PST, introduzido em (Ron et al., 1996), para estimar a árvore de contextos de uma cadeia estocástica de memória variável.

1.1 Cadeias estocásticas de memória variável

Seja A um alfabeto (conjunto de símbolos) finito. Por exemplo, A pode representar o conjunto dos vinte aminoácidos para as seqüências de proteínas ou quatro nucleotídeos para as seqüências de DNA.

Definição 1.1.1. Seja $(X_t)_{t\in\mathbb{N}}$ um processo estocástico estacionário com valores em A. Diremos que o processo $(X_t)_{t\in\mathbb{N}}$ é uma *cadeia estocástica de memória variável* se para toda seqüência $x_0,\ldots,x_n$ satisfazendo

$$\mathbb{P}[X_0 = x_0,\ldots,X_{n-1} = x_{n-1}] > 0,$$

existe um inteiro k, determinado a partir de $x_{n-k},\ldots,x_{n-1}$ tal que

$$\begin{aligned}\mathbb{P}[X_n = x_n \mid X_{n-1} = x_{n-1},\ldots,X_0 = x_0] = \\ \mathbb{P}[X_n = x_n \mid X_{n-1} = x_{n-1},\ldots,X_{n-k} = x_{n-k}].\end{aligned} \tag{1.1}$$

Assume-se que k é o mínimo inteiro que satisfaz (1.1). Assim, k é o comprimento da *memória* do processo, dado que o *passado* foi $x_0,\ldots,x_{n-1}$. As seqüências finitas dadas pelos

símbolos $(x_{n-k}, \ldots, x_{n-1})$ são chamadas de *contextos*. Note que a ordem em que são escritos os símbolos no contexto é inversa da ordem em que aparecem na expressão (1.1). Como o processo é estacionário, o inteiro k não depende de n. Portanto, simplesmente denotaremos os contextos por $(x_{-k}, \ldots, x_{-1})$.

Numa cadeia estocástica de memória variável, o conjunto de contextos tem a *propriedade do sufixo*. Essa propriedade estabelece que nenhum contexto é um sufixo de um outro contexto. Isso se deduz do fato de ser k o mínimo inteiro que satisfaz (1.1). Mais claramente dito, se $(x_{-k}, \ldots, x_{-1})$ é um contexto, então nenhuma das subseqüências $(x_{-k+j}, \ldots, x_{-1})$, com $1 \leq j \leq k-1$, é um contexto. Essa característica permite representar o conjunto de todos os contextos como uma árvore com raiz. Nessa árvore, cada contexto $(x_{-k}, \ldots, x_{-1})$ é representado por um ramo completo, no qual o primeiro nó a partir da raiz corresponde ao símbolo x_{-1}, e assim sucessivamente até o último nó do ramo (nó terminal) que representa o símbolo x_{-k}.

Exemplo 1.1.2. Consideremos o seguinte exemplo de cadeia estocástica de memória variável sobre o alfabeto binário $A = \{0, 1\}$. Suponhamos que as probabilidades de transição da cadeia $(X_t)_{t\in\mathbb{N}}$ satisfazem a seguinte propriedade

$$\mathbb{P}[X_n = x_n \mid X_0^{n-1} = x_0^{n-1}] = \begin{cases} \mathbb{P}[X_n = x_n \mid X_{n-2}^{n-1} = x_{n-2}^{n-1}], & \text{se } X_{n-1} = 0, \\ \mathbb{P}[X_n = x_n \mid X_{n-1} = x_{n-1}], & \text{se } X_{n-1} = 1, \end{cases}$$

onde a notação x_s^r representa a seqüência $x_s, x_{s+1}, \ldots, x_r$. A expressão acima indica que quando olharmos no passado para predizer o símbolo atual da seqüência, se o símbolo imediato for 1, com essa informação é suficiente para determinar a distribuição de probabilidades de transição. Por outro lado, se o símbolo imediato for 0, essa informação não é suficiente para predizer o próximo símbolo, e precisamos olhar um símbolo a mais no passado. Assim, temos que o conjunto de contextos é $\{(0,0); (1,0); (1)\}$. A árvore de contextos que representa esse processo é a da Figura 1.1(a). Suponhamos também que as probabilidades de transição da cadeia estão dadas por

$$\mathbb{P}[X_n = 0 \mid X_{n-1} = 0, X_{n-2} = 0] = 0.7,$$
$$\mathbb{P}[X_n = 0 \mid X_{n-1} = 0, X_{n-2} = 1] = 0.4$$

e

$$\mathbb{P}[X_n = 0 \mid X_{n-1} = 1] = 0.2.$$

Assim, temos uma distribuição de probabilidades sobre A associada a cada contexto da árvore, que indica a probabilidade do próximo símbolo na seqüência, dado esse contexto. A árvore

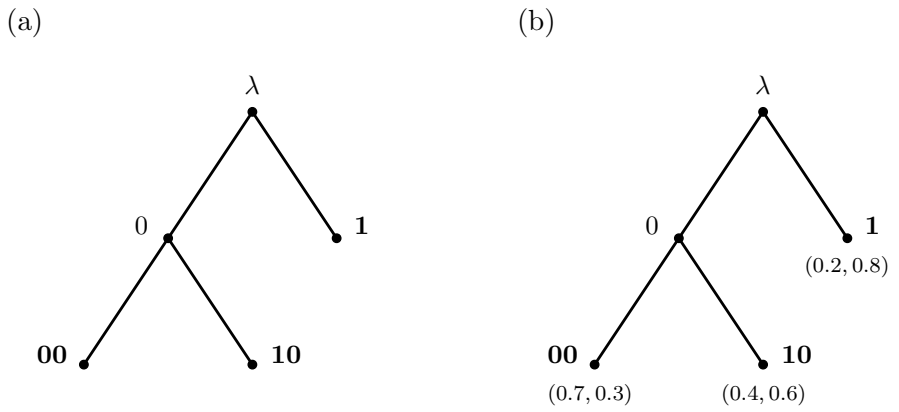

Figura 1.1: Exemplo de árvore de contextos e de árvore probabilística de sufixos de uma cadeia estocástica de memória variável. (a) Representação em forma de árvore do conjunto de contextos $\{(0,0);(1,0);(1)\}$. A raiz da árvore está representada pelo símbolo λ. (b) Árvore de contextos com probabilidades de transição associadas aos ramos.

de contextos junto com as probabilidades de transição definem completamente o modelo de cadeia estocástica de memória variável e é chamada de árvore probabilística de sufixos (ou simplesmente, árvore probabilística). A árvore probabilística para o modelo deste exemplo está representada na Figura 1.1(b). Note que não precisamos estabelecer as probabilidades iniciais porque estamos assumindo estacionaridade e portanto, essas probabilidades podem ser obtidas a partir das probabilidades de transição da cadeia.

1.2 Algoritmos locais de estimação

Nesta seção apresentamos os algoritmos Contexto (Rissanen, 1983) e PST (Ron et al., 1996). Esses algoritmos estão dentro da categoria de algoritmos locais, porque estão baseados num critério que determina o conjunto de contextos individualmente e de forma seqüencial. Essa é uma diferença marcante com relação aos algoritmos globais, que utilizam algum critério sobre a árvore de contextos como um todo. Um exemplo de algoritmo global está dado pelo BIC, apresentado no Capítulo 5, onde estudaremos a sua velocidade de convergência.

Dados dois inteiros $m \leq n$, denotaremos com w_m^n a seqüência $(w_m, \ldots, w_n)$ de símbolos no alfabeto A. O comprimento da seqüência $w = w_m^n$ será denotado por $|w|$ e está definido por $|w| = n - m + 1$. Essa notação também será usada para denotar a quantidade de elementos de conjuntos, e ficará clara pela natureza do argumento. Por exemplo, a quantidade de elementos

de A será denotada com $|A|$. No caso $m > n$, a seqüência w_m^n representará uma seqüência especial, denotada com o símbolo λ, de comprimento $|\lambda| = 0$.

Dadas duas seqüências finitas $w = w_m^n$ e $v = v_i^j$, denotaremos por vw a seqüência com comprimento $|v| + |w|$, obtida pela concatenação das duas seqüências. Em particular, $\lambda w = w\lambda = w$. No caso em que vw denote uma seqüência temporal (como as seqüências condicionais de cadeias estocásticas), assumiremos que v representa o passado mais remoto.

Diremos que a seqüência s é um *sufixo* da seqüência w se existe uma seqüência u, com $|u| \geq 1$, tal que $w = us$. Dada a seqüência finita $w = w_m^n$, denotaremos com suf(w) ao maior sufixo de w; isto é, suf$(w) = w_{m+1}^n$.

Suponhamos que $x_1, \ldots, x_n$ é uma seqüência de símbolos no alfabeto A. Por exemplo, $x_1, \ldots, x_n$ poderia ser uma seqüência de proteína, sobre o alfabeto A dos vinte aminoácidos. Em qualquer caso, o que queremos é estimar uma árvore probabilística de sufixos que melhor descreva a seqüência $x_1, \ldots, x_n$. Para isso, assumiremos que $x_1, \ldots, x_n$ é uma amostra de uma cadeia estocástica $(X_t)_{t\in\mathbb{N}}$.

Para qualquer seqüência w com $1 \leq |w| \leq n$, denotamos com $N_n(w)$ o número de ocorrências da seqüência w na amostra $x_1, x_2, \ldots, x_n$; isto é

$$N_n(w) = \sum_{t=0}^{n-|w|} \mathbf{1}\{x_{t+1}^{t+|w|} = w\}. \tag{1.2}$$

Por outro lado, denotaremos com $N_n(w\cdot)$ a soma $\sum_{b\in A} N_n(wb)$. No caso $w = \lambda$ definimos $N_n(\lambda) = N_n(\lambda\cdot) = n$.

Para qualquer símbolo $a \in A$ e qualquer seqüência w tal que $N_n(w\cdot) \geq 1$, as probabilidades de transição empíricas $\hat{p}_n(a|w)$ estão definidas por

$$\hat{p}_n(a|w) = \frac{N_n(wa)}{N_n(w\cdot)}. \tag{1.3}$$

Se $N_n(w\cdot) = 0$ definimos $\hat{p}_n(a|w) = \frac{1}{|A|}$. Estas definições estão baseadas na estimação por máxima verossimilhança das probabilidades de transição (Billingsley, 1961; Guttorp, 1995). Isso significa que a fórmula de $\hat{p}_n(a|w)$ dada por (1.3) é a que maximiza a expressão

$$\prod_{a\in A} \hat{p}_n(a|w)^{N_n(wa)}.$$

Como foi discutido na Introdução, o problema de interesse na estimação das cadeias estocásticas de memória variável é o da estimação da árvore de contextos. Um algoritmo utilizado para esse fim é o algoritmo Contexto, introduzido por Rissanen (1983) no contexto de teoria da informação. A consistência de uma versão sensivelmente diferente do algoritmo Contexto

foi provada em Bülhmann & Wyner (1999) e generalizada para o caso de árvores de memória ilimitada em Ferrari & Wyner (2003) e em Duarte et al. (2006).

O algoritmo Contexto está baseado na construção de uma árvore máxima que é possível estimar com a amostra disponível (e cujos ramos não superam em comprimento um valor pre-estabelecido). Em seguida, os ramos dessa árvore são cortados utilizando um critério baseado na entropia cruzada. Mais especificamente, a árvore estimada com o algoritmo Contexto (Rissanen, 1983) é a maior árvore de contextos $\hat{\tau}_n^k$, de profundidade menor ou igual a k, tal que

$$\Delta\hat{H}_n(w) = N_n(w\cdot)\hat{H}_n(w) - \sum_{b\in A} N_n(bw\cdot)\hat{H}_n(bw) \geq K_n,$$

para toda seqüência w tal que $bw \in \hat{\tau}_n^k$ para algum $b \in A$. Nesse caso, K_n é um ponto de corte escolhido pelo usuário, satisfazendo $K_n \sim c\log(n)$, onde c é uma constante, e $\hat{H}_n(s)$ corresponde à entropia da distribuição $\hat{p}_n(\cdot|s)$, definida por

$$\hat{H}_n(s) = -\sum_{a\in A} \hat{p}_n(a|s)\log\hat{p}_n(a|s).$$

É fácil ver que o operador $\Delta\hat{H}_n(w)$ corresponde aproximadamente ao logaritmo da razão de verossimilhança entre dois modelos: o modelo no qual as seqüências do tipo bw, com $b \in A$, são contextos e o modelo no qual w é um contexto. Em geral, $\Delta\hat{H}_n(w)$ não é exatamente igual ao logaritmo da razão de verossimilhança pela nossa definição dos contadores $N_n(w)$, mas podemos vê-lo como uma aproximação, dado que essa diferença depende, como muito, dos primeiros k símbolos e está dada pela possível diferença de ordem entre os dois modelos.

Por outro lado, o operador $\Delta\hat{H}_n(w)$ pode ser reescrito como

$$\Delta\hat{H}_n(w) = \sum_{b\in A} N_n(bw\cdot)D(\hat{p}_n(\cdot|bw)\|\hat{p}_n(\cdot|w)),$$

onde D está definido por

$$D(p\,\|q) = \sum_{a\in A} p(a)\log\frac{p(a)}{q(a)},$$

para duas distribuições de probabilidade p e q sobre A. Aqui, $p(a)\log\frac{p(a)}{q(a)}$ é definido com 0 se $p(a) = 0$ e $+\infty$ se $p(a) > q(a) = 0$. Esse operador é chamado de *divergência* ou *entropia relativa* entre as distribuições p e q. Ele também é chamado algumas vezes de *distância de Kullback-Leibler*, mesmo não sendo simétrico, condição necessária para ser uma distância. Uma propriedade relevante desse operador é que ele é sempre não negativo e é zero se e somente se $p = q$. Essa é uma instância particular do seguinte

Lema 1.2.1. *Para números não negativos* $p_1, \ldots, p_n$ *e* $q_1, \ldots, q_n$ *vale que*

$$\Big(\sum_{i=1}^{n} p_i\Big) \log \frac{\sum_{i=1}^{n} p_i}{\sum_{i=1}^{n} q_i} \;\leq\; \sum_{i=1}^{n} p_i \log \frac{p_i}{q_i},$$

com igualdade se e somente se $p_i = c\, q_i$ *para todo* $i = 1, \ldots, n$.

Esse resultado se deriva diretamente da aplicação da desigualdade de Jensen e é conhecido como *desigualdade log-suma*.

Mesmo sendo muito similar, o algoritmo PST difere do algoritmo Contexto em dois fatos. O primeiro é que o algoritmo PST vai construindo a árvore e vai realizando os testes entre as distribuições de maneira simultânea. Essa diferença não é muito relevante, já que é fácil provar que ambas abordagens, utilizando um mesmo critério de corte, devolvem a mesma árvore. Uma outra diferença é que o critério utilizado na comparação das distribuições está baseado numa distância propriamente dita entre as distribuições e não na entropia cruzada. Assim, o critério de corte do algoritmo PST está definido por

$$\Delta_n^{\mathrm{PST}}(w) = \max_{a \in A} |\hat{p}_n(a|w) - \hat{p}_n(a|\mathrm{suf}(w))|.$$

Desta forma, dado $\delta > 0$, se definimos $A_1^j = \{w \colon 1 \leq |w| \leq j\}$ temos que a árvore estimada com o algoritmo PST está dada por

$$\hat{\tau}_n^k = \{w \in A_1^k \colon \Delta_n^{\mathrm{PST}}(w) > \delta \text{ e } \Delta_n(uw) \leq \delta \; \forall u \in A_1^{k-|w|}\},$$

onde no caso $|w| = k$ temos que $A_1^{k-|w|} = \emptyset$.

Um esquema com as etapas do algoritmo PST é mostrado na Figura 1.2. Maiores detalhes sobre esse algoritmo e sobre a sua otimização linear podem ser encontrados em Bejerano (2003).

1.3 Cadeias estocásticas parcimoniosas

Nesta seção introduzimos uma definição de cadeia estocástica parcimoniosa, que inclui a definição de cadeia estocástica de memória variável. Esta definição está inspirada na idéia de economia na quantidade de parâmetros que devem ser estimados, que está presente na motivação das cadeias de memória variável, em relação com as cadeias de Markov de ordem fixa.

Denotaremos por $\mathcal{P}_A$ o conjunto de partes do alfabeto A. Isto é,

$$\mathcal{P}_A = \{w_i \colon w_i \subset A\}.$$

PST $(N,\ \delta,\ k)$

1. *Inicialização*: seja τ uma árvore consistindo unicamente do nó raiz (identificado com a seqüência λ), e seja

$$S = \{a\colon a \in A \text{ e } N_n(a\cdot) \geq N\}\,.$$

2. *Construindo a árvore*: enquanto $S \neq \emptyset$, pegue qualquer $w \in S$ e faça:
   (a) Remova w de S.
   (b) Se $\Delta_n^{\text{PST}}(w) > \delta$ então adicione a τ os nós faltantes correspondente à seqüência w.
   (c) Se $|w| < k$ adicione a S as seqüências $\{aw\colon a \in A \text{ e } N_n(aw\cdot) \geq N\}$ (se existir alguma).
3. *Estimação das probabilidades de transição*: associe a cada seqüência $w \in \tau$ a distribuição de probabilidades sobre A dada por

$$\{\hat{p}_n(a|w)\colon a \in A\}.$$

Figura 1.2: Algoritmo PST. Os parâmetros que devem ser escolhidos pelo usuário são: N, o número mínimo de vezes que um contexto tem que ser visto na amostra; δ, o ponto de corte e k, a profundidade máxima da árvore, dada pelo comprimento do seu maior contexto.

Os elementos de $\mathcal{P}_A^j$ serão denotados por $w = (w_{-j}, \ldots, w_{-1})$, como no caso dos contextos das cadeias estocásticas de memória variável. A única diferença entre eles é que no caso anterior, cada elemento w_{-i} representava um símbolo no alfabeto A, e agora w_{-i} representa um subconjunto de A.

Assim, denotaremos com $\mathcal{P}_A^*$ o conjunto de todas as seqüências finitas de elementos em $\mathcal{P}_A$; isto é,

$$\mathcal{P}_A^* = \bigcup_{j=1}^{\infty} \mathcal{P}_A^j.$$

Definição 1.3.1. Seja $(X_t)_{t\in\mathbb{N}}$ um processo estocástico estacionário com valores em A. Diremos que o processo $(X_t)_{t\in\mathbb{N}}$ é uma *cadeia estocástica parcimoniosa* se existe um conjunto $\tau \subset \mathcal{P}_A^*$ tal que:

1. Para toda seqüência $x_0, \ldots, x_n$ satisfazendo

$$\mathbb{P}[X_0 = x_0, \ldots, X_{n-1} = x_{n-1}] > 0,$$

existe um elemento $(w_{-k}, \ldots, w_{-1}) \in \tau$ tal que

$$\mathbb{P}[X_n = x_n \mid X_{n-1} = x_{n-1}, \ldots, X_0 = x_0] = \mathbb{P}[X_n = x_n \mid X_{n-k} \in w_{-k}, \ldots, X_{n-1} \in w_{-1}]. \quad (1.4)$$

2. Se $(w_{-k}, \ldots, w_{-1})$ e $(\bar{w}_{-\bar{k}}, \ldots, \bar{w}_{-1})$ pertencem a τ e existe um j tal que $w_{-i} \cap \bar{w}_{-i} \neq \emptyset$, para $i = 1, \ldots, j$, então $w_{-i} = \bar{w}_{-i}$ para $i = 1, \ldots, j$.

3. O conjunto τ é o "mínimo" que satisfaz 1. e 2. Isto é; se $\bar{\tau}$ satisfaz 1. e 2. então, para todo $(\bar{w}_{-\bar{k}}, \ldots, \bar{w}_{-1}) \in \bar{\tau}$ existe $(w_{-k}, \ldots, w_{-1}) \in \tau$ tal que $\bar{k} \geq k$ e $\bar{w}_j \subset w_j$ para todo $j = 1, \ldots, k$.

Como pode-se observar, agora os contextos do modelo estão dados pelas seqüências de subconjuntos $(w_{-k}, \ldots, w_{-1}) \in \tau$. A representação em árvore do conjunto de contextos é análoga à representação para o caso de cadeias estocásticas de memória variável.

Exemplo 1.3.2. Suponhamos que as probabilidades de transição da cadeia $(X_t)_{t\in\mathbb{N}}$ sobre o alfabeto $A = \{0, 1\}$ satisfazem a seguinte propriedade

$$\mathbb{P}[X_n = x_n \mid X_0^{n-1} = x_0^{n-1}] = \mathbb{P}[X_n = x_n \mid X_{n-2} = x_{n-2}].$$

A expressão acima indica que o símbolo na posição $n-1$ não é relevante para predizer o símbolo atual da seqüência e o único símbolo relevante é o da posição $n-2$. Assim, o conjunto de contextos é $\{(\{0\}, \{0, 1\}); (\{1\}, \{0, 1\})\}$. Se tentássemos modelar esse processo como uma cadeia estocástica de memória variável, teríamos que o conjunto de contextos nesse caso seria $\{(0, 0); (0, 1); (1, 0); (1, 1)\}$ e portanto, a árvore de contextos seria uma árvore completa de profundidade 2. Assim, muitos parâmetros seriam iguais, como por exemplo as distribuições associadas aos contextos $(0, 1)$ e $(1, 1)$. No caso da modelagem com cadeias estocásticas parcimoniosas, o conjunto de parâmetros que devem ser estimados se reduz à metade e, portanto, a estimação das probabilidades de transição se torna mais precisa. A árvore de contextos que representa o processo deste exemplo é a da Figura 1.3(a). Se supomos que as probabilidades de transição da cadeia estão dadas por

$$\mathbb{P}[X_n = 0 \mid X_{n-2} = 0] = 0.7$$
$$\mathbb{P}[X_n = 0 \mid X_{n-2} = 1] = 0.4,$$

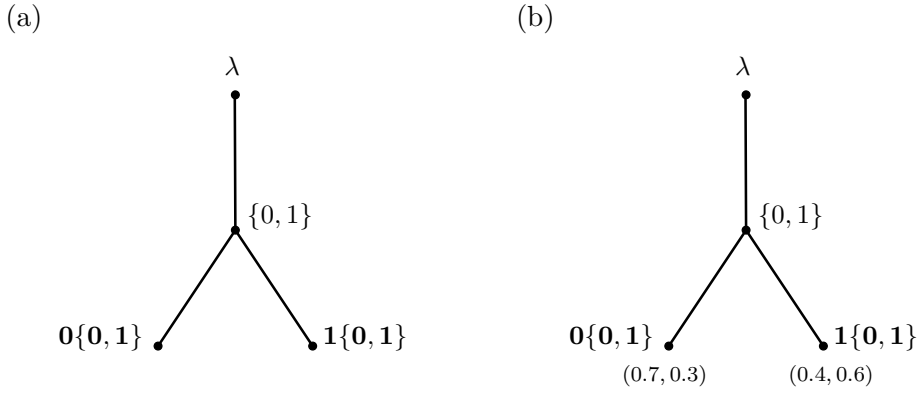

Figura 1.3: Exemplo de árvore de contextos parcimoniosos sobre o alfabeto $A = \{0, 1\}$. (a) Representação em forma de árvore do conjunto de contextos $\{(\{0\}, \{0, 1\}); (\{1\}, \{0, 1\})\}$. A raiz da árvore está representada pelo símbolo λ. (b) Árvore de contextos parcimoniosos com probabilidades de transição associadas aos ramos.

temos uma distribuição de probabilidades sobre A associada a cada contexto da árvore. A árvore probabilística correspondente é a da Figura 1.3(b).

1.4 Algoritmo SPST

Nesta seção introduzimos uma variação do algoritmo PST para identificar os contextos $w = w_{-k}^{-1}$ onde, para cada $i = 1, \ldots, k$, temos que w_{-i} é um subconjunto de A (e não um símbolo como era no caso das cadeias estocásticas de memória variável). As definições de comprimento, concatenação, etc. continuam como anteriormente e serão diferenciadas pelo contexto onde se inserem. Vale notar que dado um contexto parcimonioso w, denotamos seu comprimento com $|w|$ (ou seja, a quantidade de subconjunto de A que compõem w) e para cada $i = 1, \ldots, |w|$ temos que $|w_i|$ denota a quantidade de elementos no conjunto w_i. Para simplificar a notação identificaremos o subconjunto unitário $\{a\}$ com o símbolo a e usaremos a notação a em ambos os casos.

Por outro lado, os contadores definidos anteriormente são extendidos para o caso em que o argumento é um contexto parcimonioso. Assim, temos que

$$N_n(w) = \sum_{t=0}^{n-|w|} \mathbf{1}\{x_{t+1}^{t+|w|} \in w\}, \tag{1.5}$$

onde $x_{t+1}^{t+|w|} \in w$ significa que $x_{t+1} \in w_1, \ldots, x_{t+|w|} \in w_{|w|}$. Como antes, $N_n(w\cdot)$ denotará a soma $\sum_{b\in A} N_n(wb)$.

Dado um contexto parcimonioso w e dois subconjuntos de A, u_1 e u_2, definimos o operador $\Delta_n^{\text{SPST}}(w, u_1, u_2)$ como uma aproximação do logaritmo da razão de verossimilhança entre o modelo que tem como contextos a u_1w e u_2w e o modelo que tem só a $(u_1 \cup u_2)w$ como contexto. Essa idéia é a mesma utilizada na definição de $\Delta\hat{H}_n(w)$ no caso do algoritmo Contexto. Assim, temos que

$$\begin{aligned}\Delta_n^{\text{SPST}}(w, u_1, u_2) = &\sum_{a\in A} N_n(u_1wa) \log \hat{p}_n(a|u_1w) + \sum_{a\in A} N_n(u_2wa) \log \hat{p}_n(a|u_2w) \\ &- \sum_{a\in A} N_n((u_1 \cup u_2)wa) \log \hat{p}_n(a|(u_1 \cup u_2)w)\,.\end{aligned}$$

Note que o contexto $(u_1 \cup u_2)w$ representa a concatenação do subconjunto de A dado por $u_1 \cup u_2$ com o contexto w. Assim, temos que

$$N_n((u_1 \cup u_2)w) = N_n(u_1w) + N_n(u_2w).$$

Usando essas definições podemos especificar como funciona o algoritmo SPST. Os parâmetros que devem ser especificados pelo usuário são

1. k, a profundidade máxima da árvore;

2. N, o número mínimo de vezes que um contexto parcimonioso deve aparecer na amostra para ser considerado e

3. r, o ponto de corte que estabelece que os subconjuntos u_1 e u_2 devem ser unidos.

O algoritmo SPST funciona da seguinte forma. Ele começa com uma árvore τ que consiste unicamente do nó raiz. Em cada passo, para cada contexto parcimonioso w da árvore e para cada símbolo $a \in A$, o descendente a é adicionado ao nó correspondente a w se $N_n(aw, \cdot) \geq N$. Em seguida, para cada par de descendentes de w, u_1 e u_2, calculamos o operador $\Delta_n^{\text{SPST}}(w, u_1, u_2)$ e escolhemos o par de descendentes (u_1, u_2) que realiza o mínimo de $\Delta_n^{\text{SPST}}(w, u_1, u_2)$. Se esse mínimo é menor que r, os conjuntos u_1 e u_2 são unidos para formar o nó $u_1 \cup u_2$. Esse procedimento é iterado com o novo conjunto de descendentes de w, até que não possam ser unidos mais descendente de w. Tomar o mínimo de $\Delta_n^{\text{SPST}}(w, u_1, u_2)$ entre todos os possíveis pares (u_1, u_2) implica a independência da ordem em que os testes são realizados. Para concluir a construção da árvore associamos a cada nó uma distribuição de probabilidades de transição. Essa distribuição contem a probabilidade de cada símbolo em A

SPST (N, r, k)

1. *Inicialização*: seja τ uma árvore consistindo unicamente do nó raiz (identificado com a seqüência λ), e seja

$$S = \{a\colon a \in A \text{ e } N_n(a\cdot) \geq N\}\,.$$

2. *Construindo a árvore*: enquanto $S \neq \emptyset$ faça:

   (a) Remova w de S e adicione-o a τ. Em seguida remova todos os elementos $w' \in S$ tais que $\text{suf}(w') = \text{suf}(w)$ e adicione-os a τ.

   (b) Seja

$$r' = \min\{\Delta_n^{\text{SPST}}(\text{suf}(w), u_i, u_j)\colon u_i\text{suf}(w), u_j\text{suf}(w) \in \tau\}$$

   e

$$(u_{i^*}, u_{j^*}) = \text{argmin}\{\Delta_n^{\text{SPST}}(\text{suf}(w), u_i, u_j)\colon u_i\text{suf}(w), u_j\text{suf}(w) \in \tau\}.$$

   Se $r' < r$ junte u_{i^*} e u_{j^*} num único nó (faça a união desses conjuntos).

   (c) Repita o passo (b) até que não possam ser realizadas mais modificações na árvore τ.

   (d) Se $|w| < k$, para cada contexto da forma $u_i\text{suf}(w) \in \tau$ adicione o conjunto $\{au_i\text{suf}(w)\colon a \in A \text{ e } N_n(au_i\text{suf}(w)\cdot) \geq N\}$ a S.

3. *Estimação das probabilidades de transição*: associe a cada contexto $w \in \tau$ a distribuição de probabilidades sobre A dada por

$$\{\hat{p}_n(a|w)\colon a \in A\}.$$

Figura 1.4: Algoritmo SPST. Os parâmetros que devem ser escolhidos pelo usuário são: N, o número mínimo de vezes que um contexto parcimonioso tem que ser visto na amostra; r, o ponto de corte que estabelece que dois subconjuntos de A devem ser unidos e k, a profundidade máxima da árvore, dada pelo comprimento do seu maior contexto.

dado o contexto parcimonioso representado pelo caminho entre o nó e a raiz da árvore. As probabilidades de transição são estimadas usando os estimadores de máxima verossimilhança,

definidos pela expressão (1.3) e os contadores (1.5). Um esquema com as etapas do algoritmo SPST é mostrado na Figura 1.4.

Capítulo 2

Classificação de proteínas

Neste capítulo apresentamos os resultados obtidos a partir da aplicação do algoritmo SPST para classificar proteínas em famílias. Também introduzimos uma variação na etapa de predição das seqüências que melhora significativamente o desempenho do algoritmo. Essa variação é chamada de F-SPST. Esses resultados são comparados com os obtidos previamente por Bejerano & Yona (2001), utilizando o algoritmo PST. Os resultados deste capítulo foram publicados primeiramente em Leonardi & Galves (2005), em forma de resumo estendido, e recentemente em Leonardi (2006) como artigo completo.

2.1 O problema da classificação de proteínas

Como foi discutido na Introdução, as moléculas de proteínas estão envolvidas em quase todas as atividades que ocorrem dentro de cada célula viva. As proteínas encarregam-se, por exemplo, do transporte e armazenamento de moléculas menores, da constituição de membranas, da catalização de reações químicas, da transmissão de sinais, etc.

Cada proteína é uma macromolécula complexa. Ainda assim, as proteínas estão compostas por pequenos blocos mais simples, escolhidos de um conjunto limitado, que estão ligados um depois do outro de uma maneira linear. Nesse conjunto existem vinte diferentes blocos, que são chamados de *aminoácidos*. Todos os aminoácidos possuem uma estrutura molecular similar, mas com algumas diferenças. Essas diferenças conferem aos aminoácidos diversas propriedades bioquímicas. Segundo estas ou outras propriedades, os aminoácidos podem ser agrupados em diferentes classes. Um exemplo do agrupamento dos aminoácidos segundo algumas propriedades físico-químicas pode ser visto na Figura 2.1.

Um problema central em genômica é a determinação da função de uma proteína utilizando a informação contida na sua cadeia de aminoácidos (Rust et al., 2002; Karp, 2002). Atualmente, os métodos mais utilizados para a obtenção de uma hipótese sobre a função de uma proteína estão baseados na busca de similaridade por alinhamento (Smith & Waterman, 1981). Uma grande desvantagem dessa metodologia é a sua complexidade computacional quadrática, o que

G	Gly	Glicina	S	Ser	Serina
A	Ala	Alanina	T	Thr	Treonina
V	Val	Valina	C	Cys	Cisteína
L	Leu	Leucina	N	Asn	Asparagina
I	Ile	Isoleucina	Q	Gln	Glutamina
P	Pro	Prolina	R	Arg	Arginina
M	Met	Metionina	K	Lys	Lisina
F	Phe	Fenilalanina	H	His	Histidina
Y	Tyr	Tirosina	D	Asp	A. Aspártico
W	Trp	Triptofano	E	Glu	A. Glutâmico

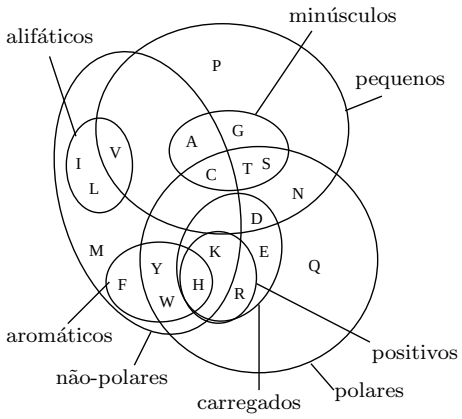

Figura 2.1: Os vinte aminoácidos. Códigos de uma e três letras dos aminoácidos (esquerda). Diagrama de Venn, adaptado de Taylor (1986) e extraído de Bejerano (2003), agrupando os aminoácidos segundo algumas propriedades físico-químicas (esquerda).

levou à introdução de algoritmos heurísticos para a comparação de seqüências em grandes conjuntos de dados, como por exemplo BLAST (Altschul et al., 1997) ou FASTA (Pearson, 2000). Uma outra abordagem, ainda mais relacionada com a que apresentaremos aqui, utiliza uma classe de modelos chamada de *Cadeias de Markov Ocultas* (HMM, do inglês *Hidden Markov Models*) (Rabiner, 1986). Esse método, embora muito utilizado na modelagem de seqüências biológicas, apresenta a desvantagem de ter uma quantidade muito grande de parâmetros que devem ser estimados e, por outro lado, os algoritmos utilizados para a estimação não garantem a escolha do modelo ótimo.

Recentemente, Bejerano & Yona (2001) propôs a utilização das cadeias estocásticas de memória variável no problema de classificação de proteínas em famílias. Nesse caso foi utilizado o algoritmo PST, apresentado no Capítulo 1, para estimar a árvore de contextos e as probabilidades de transição do modelo. No artigo citado foi mostrado que esse método consegue detectar mais seqüências relacionadas do que Gapped-BLAST (uma variação de BLAST) e que é quase tão sensível quanto HMM. Uma grande vantagem do modelo de cadeias estocásticas de memória variável é que ele não depende de alinhamentos múltiplos das seqüências e que ele pode ser estimado em tempo e espaço linear (Apostolico & Bejerano, 2000).

2.2 Aplicação do algoritmo SPST para classificar proteínas em famílias

Uma família de proteínas, como foi originariamente definido por Dayhoff et al. (1978), é um grupo de proteínas com uma função similar que têm uma identidade de aminoácidos maior que

50% quando alinhadas. Uma família de proteínas compreende proteínas com a mesma função em diferentes organismos (ortólogos) ou proteínas no mesmo organismo (parólogos), derivados de duplicações gênicas e re-arranjamentos.

Dada uma família de proteínas $\mathcal{F}$ e uma nova seqüência de aminoácidos s, o objetivo final é saber se s pertence a $\mathcal{F}$ ou não. Para responder essa pergunta primeiramente estimamos um modelo para a família $\mathcal{F}$, utilizando as seqüências já classificadas dentro da família. Em seguida, calculamos uma pontuação utilizando o modelo estimado e, dependendo desse valor, classificamos a seqüência s como pertencente à família $\mathcal{F}$ ou não.

O modelo construído para a família $\mathcal{F}$ será uma cadeia estocástica parcimoniosa, obtida a partir da estimação da árvore probabilística correspondente. Para estimar essa árvore probabilística utilizaremos o algoritmo SPST, introduzido no capítulo anterior. Denotemos com $(\hat{\tau}_n, \hat{p}_n)$ a árvore probabilística estimada, onde $\hat{\tau}_n$ representa a árvore de contextos parcimoniosos e $\hat{p}_n$ é o conjunto composto pelas distribuições associadas a cada contexto. Nesse caso, n representa o tamanho total da amostra; isto é, a soma dos comprimentos das seqüências pertencentes à família $\mathcal{F}$.

Dada a seqüência $s = s_1, s_2, \ldots, s_m$, sua pontuação no modelo estimado para a família $\mathcal{F}$ está dada por

$$S(s) = \frac{1}{|s|} \log \left[\bar{p}_n(s)\right],$$

onde $\bar{p}_n(s)$ é a probabilidade *suavizada* da seqüência s no modelo dado por $(\hat{\tau}_n, \hat{p}_n)$. Isto é,

$$\bar{p}_n(s) = \prod_{i=1}^{m} \left[(1 - |A|\gamma)\, \hat{p}_n(s_i | w(s_1, \ldots, s_{i-1})) + \gamma \right],$$

onde $w = w(s_1, \ldots, s_{i-1})$ é o contexto parcimonioso na árvore $\hat{\tau}_n$ tal que $s_1^{i-1} \in w$ e γ é um valor positivo utilizado para evitar probabilidades iguais a zero e, portanto, uma pontuação igual a $-\infty$. O parâmetro γ deve satisfazer a condição $0 < \gamma < \frac{1}{|A|}$.

Assim, dada uma pontuação mínima que chamaremos de $S_{\min}$, diremos que a seqüência s pertence à família $\mathcal{F}$ se $S(s) \geq S_{\min}$.

2.3 Algoritmo F-SPST

Algumas vezes, a região de alta similaridade entre as seqüências de uma mesma família de proteínas é consideravelmente menor que o comprimento das seqüências. Isso se deve a que as proteínas podem estar compostas por diferentes domínios, que realizam diferentes funções e, portanto, essas proteínas pertencem a mais de uma família. Por esse motivo, o cálculo da pontuação S sobre a seqüência inteira pode não ser apropriado em alguns casos. Baseados nesse

fato, propomos uma mudança no cálculo da pontuação S. Essa nova pontuação é chamada de S'. Nesse caso, fixamos um inteiro M e, para as seqüências com comprimento $m > M$, calculamos o valor $S'(s)$ por

$$S'(s) = \max_{j=0,\ldots,m-M} \{ \frac{1}{M} \log [\, \bar{p}_n(s_{j+1}^{j+M}) \,] \},$$

onde $\bar{p}_n(s_{j+1}^{j+M})$ está dado por

$$\bar{p}_n(s_{j+1}^{j+M}) = \prod_{i=j+1}^{j+M} [\, (1 - |A|\gamma)\, \hat{p}_n(s_i | w(s_1, \ldots, s_{i-1})) + \gamma \,].$$

No caso $m \leq M$, a pontuação S' coincide com a pontuação S. O algoritmo que implementa a pontuação S' é chamado de F-SPST.

2.4 Implementação e resultados

Com o propósito de testar os algoritmos SPST e F-SPST, e compará-los com os resultados obtidos por (Bejerano & Yona, 2001) com o algoritmo PST, usamos famílias de proteínas da base de dados Pfam (Bateman et al., 2004), na versão 1.0. Essa base de dados contem 175 famílias de proteínas, derivadas da base de dados SWISSPROT 33 (Boeckmann et al., 2004). Em primeiro lugar, selecionamos as primeiras 50 famílias, na ordem alfabética, que continham mais de 10 seqüências. Para cada família nesse conjunto, estimamos um modelo de cadeia estocástica parcimoniosa, utilizando o algoritmo SPST. Usamos 4/5 das seqüências em cada família para estimar o modelo. Em seguida, calculamos a pontuação S, no caso do algoritmo SPST e S', no caso do algoritmo F-SPST, para cada uma das seqüências da base de dados SWISSPROT 33. Essa base de dados contem 52.205 seqüências. Dentre essas seqüências encontram-se todas as seqüências de todas as famílias, tanto as usadas para estimar os modelos quanto as que ficaram por fora da estimação. Finalmente, as seqüências foram ordenadas segundo o valor da pontuação, no sentido decrescente.

Para estabelecer o ponto de corte $S_{\min}$, com o qual decidimos se uma seqüência pertence à família ou não, utilizamos o *critério do número de equivalência* (Pearson, 1995). Esse critério estabelece como ponto de corte o valor da pontuação no qual o número de falsos positivos (i.e. proteínas não pertencentes à família com uma pontuação acima do ponto de corte) é igual ao número de falsos negativos (i.e. proteínas pertencentes á família com uma pontuação por baixo do ponto de corte). Esse é um ponto de balanço entre a seletividade e a sensibilidade do modelo. Uma proteína pertencente à família cuja pontuação for maior que o ponto de corte (verdadeiro positivo) é considerada detectada com sucesso. A qualidade do modelo é medida

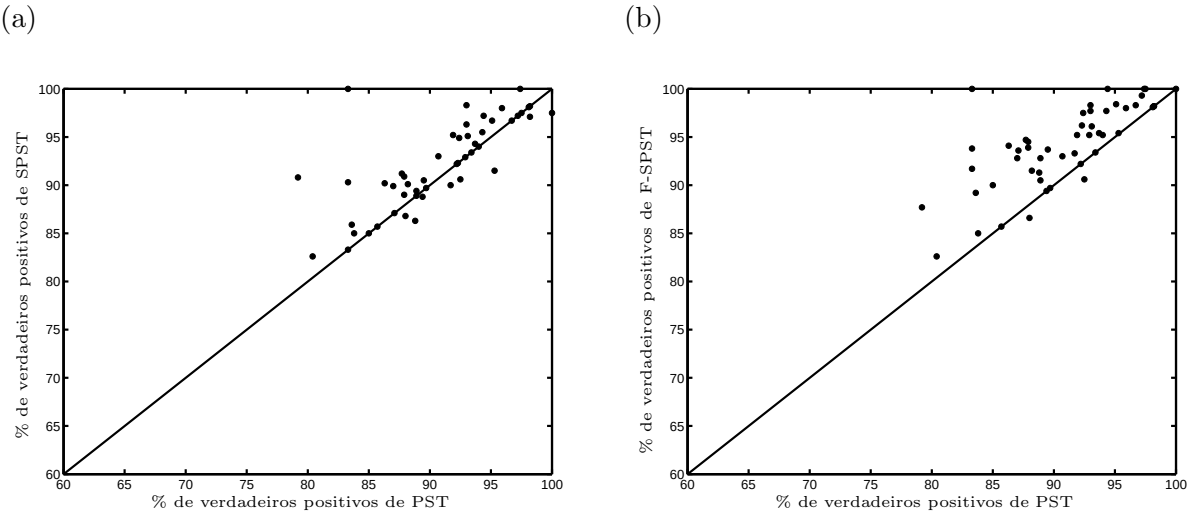

Figura 2.2: Comparação gráfica dos desempenhos dos algoritmos SPST e F-SPST, em relação com o algoritmo PST. (a) SPST vs. PST. (b) F-SPST vs. PST

através da porcentagem de verdadeiros positivos detectados, com referência ao número total de seqüências na família.

A implementação do algoritmo SPST, e sua variação no algoritmo F-SPST, foi feita em ANSI C e compilada usando `gcc`. O código desse programa é livre e pode ser obtido a partir do site `http://www.ime.usp.br/~leonardi/spst/`.

O tempo de estimação de uma árvore probabilística num processador AMD (Athlon) 851 Mhz PC, para uma família de proteínas na base de dados Pfam 1.0, variou entre 2 s e 49 min. O cálculo das pontuações para as 52.205 seqüências na base de dados SWISSPROT 33 levou, em média, 1 min 40 s para o algoritmo SPST e 1 min 5 s para o algoritmo F-SPST.

A Tabela 2.1 mostra as porcentagens de classificação obtidas com os algoritmos SPST e F-SPST, junto com os resultados publicados do algoritmo PST (Bejerano & Yona, 2001). Também é mostrado o número de falsos positivos de cada algoritmo. Como conseqüência da forma em que é calculado o ponto de corte, as porcentagens de falsos positivos é igual a 100% menos a porcentagem de verdadeiros positivos. Por exemplo, no caso da família *7tm_1*, a porcentagem de verdadeiros positivos detectados com o algoritmo F-SPST é 97.7%, portanto, a porcentagem de falsos positivos é 2.3%. Isso dá 12 seqüências classificadas erroneamente como sendo membros da família *7tm_1*. A Figura 2.2 mostra uma comparação gráfica entre os resultados dos algoritmos SPST e F-SPST, em relação aos resultados do algoritmo PST.

Em Bejerano & Yona (2001) foi mostrado que o desempenho do algoritmo PST pode ser

Tabela 2.1: Comparação de PST, SPST e F-SPST. As famílias estão ordenadas alfabeticamente e correspondem às primeiras 50 famílias com mais de 10 seqüências na base de dados Pfam 1.0. O número de seqüências em cada família é mostrado na segunda coluna. As outras seis colunas, dois para cada algoritmo, indicam a porcentagem de verdadeiros positivos detectados, tendo como referência o tamanho de cada família, e o número de falsos positivos, respectivamente. Os resultados do algoritmo PST foram extraídos de (Bejerano & Yona, 2001). O conjunto de parâmetros para estimar os modelos de cadeia estocástica parcimoniosa, tanto no algoritmo SPST quanto no F-SPST, foram: $N = 3$, $r = 3.8$ e $k = 20$. O parâmetro de suavização das probabilidades de transição utilizado foi $\gamma = 0.001$. O valor de M usado no algoritmo F-SPST foi $M = 80$ para todas as famílias.

Família	No. de seqüências	PST		SPST		F-SPST	
		% v.p.	No. f.p.	% v.p.	No. f.p.	% v.p.	No. f.p.
7tm_1	515	93.0	36	96.3	19	97.7	12
7tm_2	36	94.4	2	97.2	1	100.0	0
7tm_3	12	83.3	2	100.0	0	100.0	0
AAA	66	87.9	8	90.9	6	93.9	4
ABC_tran	269	83.6	44	85.9	38	89.2	29
actin	142	97.2	4	97.2	4	99.3	1
adh_short	180	88.9	20	89.4	19	92.8	13
adh_zinc	129	95.3	6	91.5	11	95.4	6
aldedh	69	87.0	9	89.9	7	92.8	5
alpha-amylase	114	87.7	14	91.2	10	94.7	6
aminotran	63	88.9	7	88.9	7	90.5	6
ank	83	88.0	10	86.8	11	86.6	11
arf	43	90.7	4	93.0	3	93.0	3
asp	72	83.3	12	90.3	7	91.7	6
ATP-synt_A	79	92.4	6	94.9	4	97.5	2
ATP-synt_ab	180	96.7	6	96.7	6	98.3	3
ATP-synt_C	62	91.9	5	95.2	3	95.2	3
beta-lactamase	51	86.3	7	90.2	5	94.1	3
bZIP	95	89.5	10	90.5	9	93.7	6
C2	78	92.3	6	92.3	6	96.2	3
cadherin	31	87.1	4	87.1	4	93.6	2
cellulase	40	85.0	6	85.0	6	90.0	4
cNMP_binding	42	92.9	3	92.9	3	95.2	2
COesterase	61	91.7	5	90.0	6	93.3	4

continua na página seguinte

continuação da página anterior

Família	No. de seqüências	PST		SPST		F-SPST	
		% v.p.	No. f.p.	% v.p.	No. f.p.	% v.p.	No. f.p.
connexin	40	97.5	1	97.5	1	100.0	0
copper-bind	61	95.1	3	96.7	2	98.4	1
COX1	80	83.8	13	85.0	12	85.0	12
COX2	109	98.2	2	98.2	2	98.2	2
cpn10	57	93.0	4	98.3	1	98.3	1
cpn60	84	94.0	5	94.0	5	95.2	4
crystall	53	98.1	1	98.1	1	98.1	1
cyclin	80	88.8	9	86.3	11	91.3	7
Cys-protease	91	87.9	11	89.0	10	94.5	5
cystatin	53	92.5	4	90.6	5	90.6	5
Cys_knot	61	93.4	4	93.4	4	93.4	4
cytochrome_b_C	130	79.2	27	90.8	12	87.7	16
cytochrome_b_N	170	98.2	3	97.1	5	98.2	3
cytochrome_c	175	93.7	11	94.3	10	95.4	8
DAG_PE-bind	68	89.7	7	89.7	7	89.7	7
DNA_methylase	48	83.3	8	83.3	8	93.8	3
DNA_pol	46	80.4	9	82.6	8	82.6	8
dsrm	14	85.7	2	85.7	2	85.7	2
E1-E2_ATPase	102	93.1	7	95.1	5	96.1	4
efhand	320	92.2	25	92.2	25	92.2	25
EGF	169	89.4	18	88.8	19	89.4	18
enolase	40	100.0	0	97.5	1	100.0	0
fer2	88	94.3	5	95.5	4	97.7	2
fer4	152	88.2	18	90.1	15	91.5	13
fer4_NifH	49	95.9	2	98.0	1	98.0	1
FGF	39	97.4	1	100.0	0	100.0	0

incrementado aumentando o número de nós da árvore. Para os algoritmos SPST e F-SPST, esse fato depende dos valores dos parâmetros k e N. No caso dos outros parâmetros, não conhecemos ainda quais seriam as melhores escolhas desses valores. Portanto, estudamos o desempenho do algoritmo F-SPST como função dos parâmetros r, γ e M. Cinco famílias da Tabela 2.1 foram escolhidas aleatoriamente e foram estimadas as respectivas árvores probabilísticas, variando os valores do parâmetro correspondente. O desempenho do algoritmo F-SPST não foi afetado significativamente no caso dos parâmetros γ e M (Figura 2.3a-b), mas foi decrescendo com valores maiores de r (Figura 2.3c). Esse fato é devido a que quando r aumenta, o número de

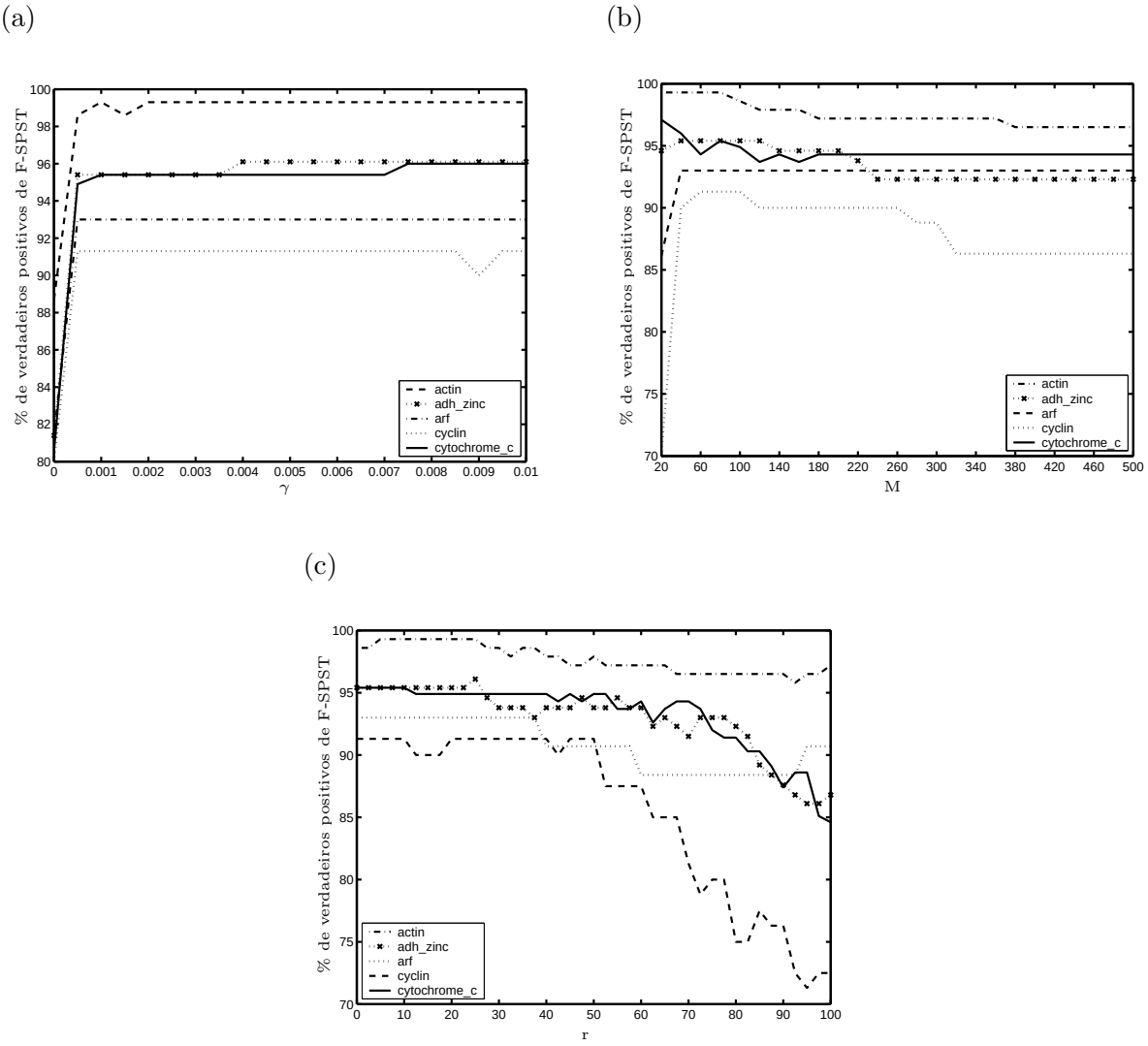

Figura 2.3: Avaliação do desempenho do algoritmo F-SPST como função dos parâmetros γ (a), M (b) e r (c). O eixo horizontal mostra os valores do parâmetro usados para estimar a árvore probabilística correspondente a cada família. Os valores dos demais parâmetros foi mantido igual ao usado nos resultados da Tabela 2.1. Esta avaliação foi feita para 5 famílias escolhidas aleatoriamente.

nós da árvore diminui (ainda mais nós são juntados) e nesse caso temos um modelo sobestimado. Mesmo assim, é interessante notar que para todos os valores de r inferiores a 50, o desempenho do algoritmo F-SPST para as cinco famílias foi mantido acima de 90%.

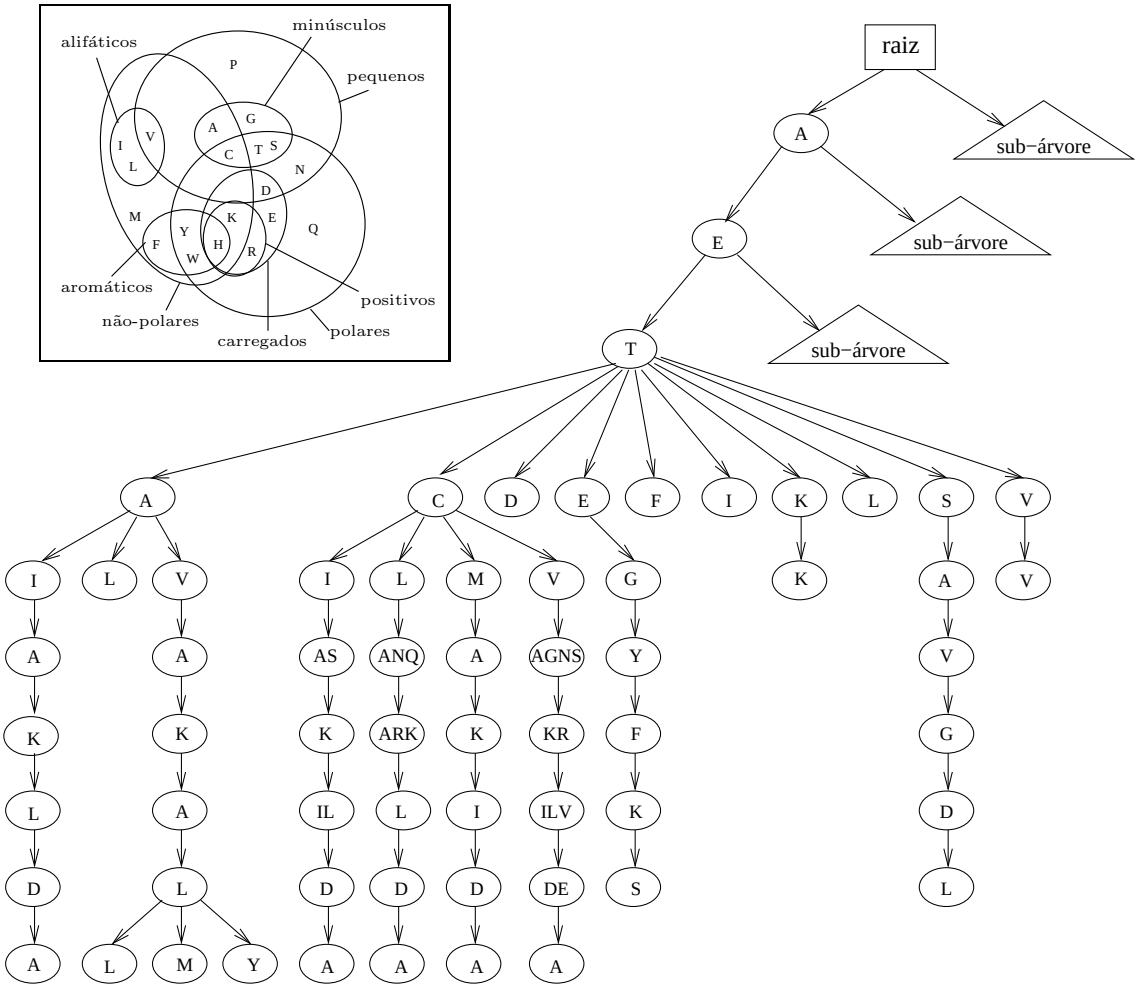

Figura 2.4: Uma árvore de contextos parcimoniosos estimada com o algoritmo SPST. Essa árvore corresponde às seqüências da família AAA (família de *ATPases* associadas com várias atividades celulares). Em cada nó da árvore vemos um subconjunto de aminoácidos correspondente a uma posição em um contexto parcimonioso (as chaves que denotam os subconjuntos foram omitidas). Alguns nós da árvore se correspondem com os grupos de aminoácidos com propriedades físico-químicas semelhantes (mostrados no quadro acima).

Uma propriedade muito interessante do algoritmo SPST aparece quando comparamos os contextos parcimoniosos na árvore estimada com as classes obtidas ao agrupar os aminoácidos segundo suas propriedades físico-químicas, como foi mostrado na Figura 2.1. Por exemplo, a

árvore estimada para a família AAA (família de *ATPases* associadas com várias atividades celulares) tem como nó na árvore o subconjunto de aminoácidos $\{I, V, L\}$. Esse subconjunto corresponde exatamente ao grupo dos aminoácidos alifáticos. A Figura 2.4 mostra uma porção dessa árvore junto ao agrupamento dos aminoácidos.

Os resultados apresentados neste capítulo sugerem que os algoritmos SPST e F-SPST podem melhorar a classificação obtida com o algoritmo PST. No caso do algoritmo SPST, isso é devido provavelmente ao fato que o modelo de cadeia estocástica parcimoniosa descreve melhor a natureza *esparsa* das seqüências de proteínas, à vez que se beneficia com a redução no número de parâmetros que devem ser estimados. No caso do algoritmo F-SPST fica claro que uma comparação local das seqüências de proteínas é mais apropriada que a comparação global, dada a composição em domínios das seqüências de proteínas.

Capítulo 3

Análise filogenética de proteínas

Neste capítulo apresentamos os resultados das aplicações do algoritmo SPST para o estudo da similaridade entre as seqüências de proteínas. Para isso, utilizamos uma distância entre as árvores de contextos associadas às seqüências. Essa distância leva em consideração a estrutura das árvores e indiretamente as probabilidades de transição. Aplicamos esse método para estudar a filogenia de seqüências de proteínas pertencentes a família dos *fatores de crescimento de fibroblastos* (FGF) e à família das *globinas*. Esses resultados constituem um estudo preliminar das aplicações da modelagem com cadeias estocásticas parcimoniosas à inferência da evolução de seqüências simbólicas, que deve ser aprofundado. Uma versão preliminar desta abordagem foi apresentada em forma de cartaz no Congresso da Associação Brasileira de Bioinformática e Biologia Computacional, X-meeting 2005 (Leonardi, 2005). Uma versão em formato de artigo completo, contendo os resultados deste capítulo, está sendo redigida atualmente (Leonardi et al., 2006).

3.1 Um espaço métrico de árvores

As definições apresentadas nesta seção estão baseadas na noção de β-entropia para partições, introduzida em Simovici & Szymon (2002). É fácil provar que cada árvore de contextos τ define uma partição do conjunto A^j (conjunto de todas as seqüências finitas sobre A de tamanho j), para cada $j \geq d(\tau)$, onde $d(\tau)$ denota a profundidade da árvore τ; isto é

$$d(\tau) = \sup\{|w| : w \in \tau\}.$$

Portanto, a tradução da noção de entropia para árvores de contextos é direta. Com a noção de entropia e a definição de partição máxima entre duas partições, deriva-se a definição de distância introduzida em (Simovici & Szymon, 2006). Essa será a distância que utilizaremos para estudar a similaridade entre as seqüências de proteínas.

Dado um contexto parcimonioso $w = (w_{-j}, \ldots, w_{-1})$, denotaremos com $[w]$ ao produto dos cardinais dos conjuntos w_i; isto é

$$[w] = \prod_{i=1}^{j} |w_{-i}|\,,$$

onde, como antes, $|w_i|$ representa o número de elementos no conjunto w_i. O valor de $[w]$ equivale à quantidade de seqüências $s = (s_{-j}, \ldots, s_{-1}) \in A^j$ tais que $s_{-i} \in w_{-i}$, para todo $i = 1, \ldots, j$.

Definição 3.1.1. Dada uma árvore de contextos τ e um número real $\beta > 0$, definimos a β-entropia da árvore τ por

$$H_\beta(\tau) = \frac{1}{2^{1-\beta} - 1} \Big(\sum_{w \in \tau} \big[[w]\,|A|^{-|w|}\big]^\beta - 1 \Big), \quad \text{se } \beta \neq 1,$$

e

$$H_\beta(\tau) = -\sum_{w \in \tau} [w]\,|A|^{-|w|} \cdot \log_2\big[[w]\,|A|^{-|w|}\big], \quad \text{se } \beta = 1.$$

Exemplo 3.1.2. Consideremos a árvore τ da Figura 3.1(a), sobre o alfabeto $A = \{a, b, c, d\}$. Calculemos $H_\beta(\tau)$, com $\beta = 1$. Neste caso temos três contextos, $w^1 = (\{a, b, c\}, \{ac\})$, $w^2 = (d, \{ac\})$ e $w^3 = (\{bd\})$. Assim, temos que $[w^1] = 6$, $[w^2] = 2$ e $[w^3] = 2$. Portanto,

$$\begin{aligned} H_\beta(\tau) &= -\big(\,6\,|A|^{-2} \cdot \log_2\big[6\,|A|^{-2}\big] + 2\,|A|^{-2} \cdot \log_2\big[2\,|A|^{-2}\big] + 2\,|A|^{-1} \cdot \log_2\big[2\,|A|^{-1}\big]\,\big) \\ &= 0.375 \cdot 1.415037 + 0.125 \cdot 3 + 0.5 \cdot 1 \\ &= 1.405639. \end{aligned}$$

Dados dois contextos parcimoniosos $w = (w_{-j} \ldots, w_{-1})$ e $\bar{w} = (\bar{w}_{-\bar{\jmath}}, \ldots, \bar{w}_{-1})$, definimos a intersecção entre w e $\bar{w}$ (assumindo sem perda de generalidade que $j \geq \bar{\jmath}$) por $w \cap \bar{w} = (w_{-j}, \ldots, w_{-(\bar{\jmath}+1)}, w_{-\bar{\jmath}} \cap \bar{w}_{-\bar{\jmath}}, \ldots, w_{-1} \cap \bar{w}_{-1})$, se $w_i \cap \bar{w}_i \neq \emptyset$ para todo $i = 1, \ldots, \bar{\jmath}$. No caso $w_i \cap \bar{w}_i = \emptyset$ para algum $i = 1, \ldots, \bar{\jmath}$ definimos $w \cap \bar{w} = \emptyset$.

Dadas duas árvores de contextos parcimoniosos, $\tau = \{w^1, \ldots, w^n\}$ e $\bar{\tau} = \{\bar{w}^1, \ldots, \bar{w}^m\}$, definimos a árvore máxima entre τ e $\bar{\tau}$ por

$$\tau \vee \bar{\tau} = \{w^i \cap \bar{w}^j : w^i \cap \bar{w}^j \neq \emptyset \text{ para } i = 1, \ldots, n;\ j = 1, \ldots, m\}\,. \tag{3.1}$$

Exemplo 3.1.3. Consideremos as árvores de contextos parcimoniosos da Figura 3.1(a)-(b), sobre o alfabeto $A = \{a, b, c, d\}$. Chamemos essas árvores de τ e $\bar{\tau}$, respectivamente. Fazendo a intersecção de cada contexto da árvore τ com cada contexto da árvore $\bar{\tau}$ obtemos os contextos

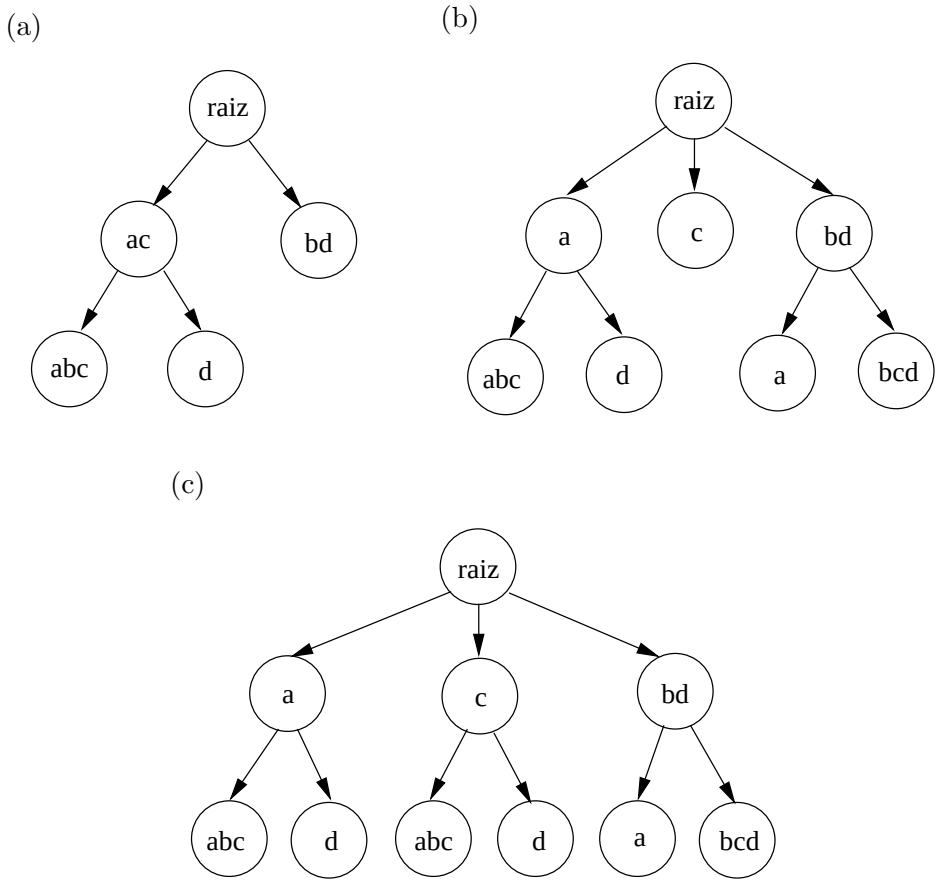

Figura 3.1: Árvore máxima de duas árvores de contextos parcimoniosos sobre o alfabeto $A = \{a, b, c, d\}$. (a) Árvore τ. (b) Árvore $\bar{\tau}$. (c) Árvore $\tau \vee \bar{\tau}$.

da árvore $\tau \vee \bar{\tau}$, excluindo aqueles cuja intersecção é vazia. Por exemplo, a intersecção dos contextos $(\{a, b, c\}, \{ac\})$ e $(\{a, b, c\}, a)$ resulta no contexto $(\{a, b, c\}, a)$ e a intersecção dos contextos $(\{a, b, c\}, \{a, c\})$ e (c) resulta no contexto $(\{a, b, c\}, c)$, ambos pertencentes a árvore $\tau \vee \bar{\tau}$. A árvore $\tau \vee \bar{\tau}$ é mostrada na Figura 3.1(c).

Seguindo Simovici & Szymon (2006) obtemos a seguinte definição de distância entre duas árvores de contextos parcimoniosos.

Definição 3.1.4. Dadas duas árvores de contextos parcimoniosos, τ e $\bar{\tau}$, definimos a β-distância entre τ e $\bar{\tau}$ por

$$d_\beta(\tau, \bar{\tau}) = 2\, H_\beta(\tau \vee \bar{\tau}) - H_\beta(\tau) - H_\beta(\bar{\tau})\,. \tag{3.2}$$

3.2 Implementação e resultados

A β-distância definida por (3.2) foi implementada para estudar a similaridade entre as seqüências de proteínas de algumas famílias. O objetivo é estudar a filogenia dessas seqüências, a partir da sua representação como árvores de contextos parcimoniosos. O programa que implementa a β- distância utiliza o código do algoritmo SPST para estimar as árvores de contextos parcimoniosos correspondente a cada seqüência e é chamado de Phyl-SPST. Ele está disponível na página `http://www.ime.usp.br/~leonardi/phyl-spst`.

Para testar o desempenho da β-distância primeiramente foi aplicado o algoritmo Phyl-SPST às 22 seqüências de proteínas da família FGF encontradas em humanos. Essas seqüências têm sido bastante estudadas na literatura e análises filogenéticas sugerem que essas 22 seqüências podem ser agrupadas em 7 sub-famílias (Itoh & Ornitz, 2004). As seqüências utilizadas na análise foram obtidas da base de dados Pfam (Bateman et al., 2004). Para cada domínio dos 22 desta família foi estimada uma árvore de contextos parcimoniosa, utilizando o algoritmo SPST. Em seguida, foi calculada a β-distância definida por (3.2), para cada par de árvores, e foram organizados os resultados numa matriz de distâncias. Usando como entrada essa matriz de distâncias foi aplicado primeiramente o algoritmo KITSCH e, em seguida, o algoritmo DRAWTREE, ambos do pacote Phylip3.65 (Felsenstein, 2004), para construir uma árvore filogenética. A Figura 3.2 mostra a árvore filogenética obtida com esse procedimento. Os grupos de seqüências obtidos por esse método coincidem exatamente com as 7 sub-famílias encontradas na literatura e a árvore se corresponde com a análoga apresentada em Itoh & Ornitz (2004, Figure 1).

Uma segunda aplicação do programa Phyl-SPST teve o objetivo de estudar a similaridade entre as seqüências da família de proteínas chamadas de *globinas*. As seqüências utilizadas correspondem a todas as seqüências de vertebrados pertencentes ao grupo de globinas analisadas na base de dados PALI (Sujatha et al., 2001). Nesse grupo encontramos 41 seqüências, que foram obtidas posteriormente da base de dados SCOP (Murzin et al., 1995). Como no caso da família FGF, foi aplicado o algoritmo Phyl-SPST a essas 41 seqüências. Em seguida, foi utilizado o algoritmo NEIGHBOR e DRAWGRAM do pacote Phylip3.65 para construir a árvore filogenética. Essa árvore pode ser observada na Figura 3.3.

Com o objetivo de comparar o desempenho do algoritmo Phyl-SPST com outras metodologias mais utilizadas na prática foi obtida uma outra árvore filogenética para as mesmas 41 seqüências de globinas. Neste caso foi seguido o seguinte procedimento. Primeiro as 41 seqüências foram alinhadas utilizando o pacote ClustalW (Thompson et al., 1994). A partir desse alinhamento foi calculada a distância PAM (Dayhoff et al., 1978), utilizando o programa

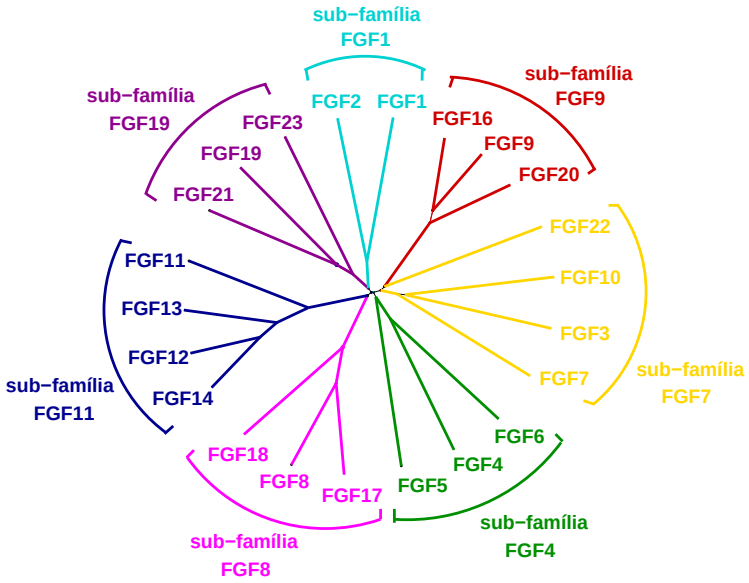

Figura 3.2: Sub-famílias obtidas com a análise filogenética dos 22 domínios de FGF em humanos, utilizando a β-distância entre as árvores de contextos parcimoniosos correspondentes a cada seqüência. A árvore filogenética foi obtida utilizando os algoritmos KITSH e DRAWTREE, do pacote Phylip3.65.

PROTDIST do pacote Phylip3.65. Por último, foi obtida a árvore filogenética através da utilização dos programas NEIGHBOR e DRAGRAM, como no caso da análise anterior. Esta árvore filogenética, baseada na distância PAM, é apresentada na Figura 3.4. A partir desses resultados podemos observar que as duas abordagens utilizadas identificam basicamente as mesmas relações filogenéticas entre as seqüências. Contudo, algumas diferenças podem ser identificadas nos grupos de seqüências próximas. Uma vantagem da abordagem baseada em árvores de contextos é que não é necessário obter preliminarmente um alinhamento múltiplo das seqüências.

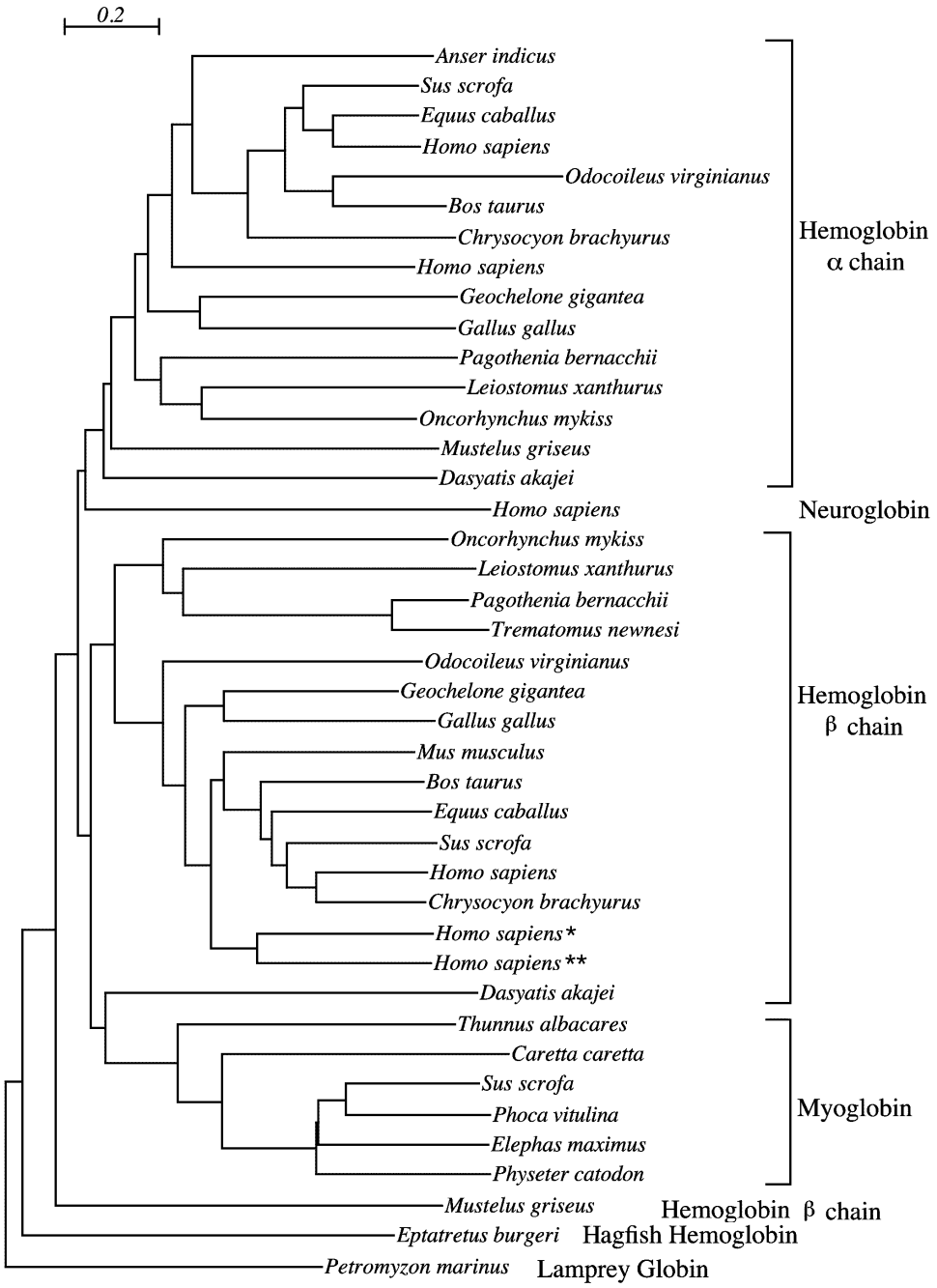

Figura 3.3: Árvore filogenética baseada na β-distância, para 41 seqüências de globinas em vertebrados.

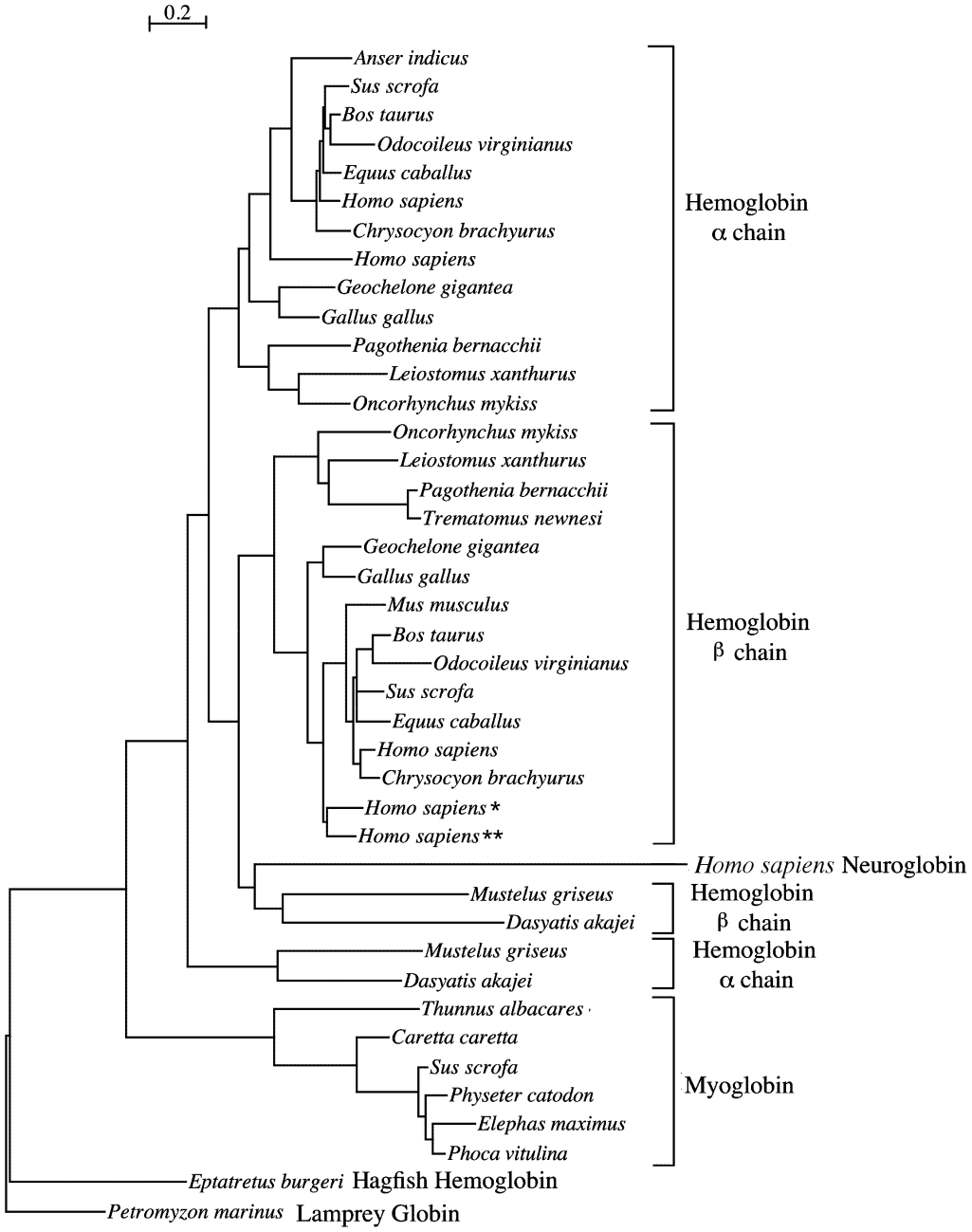

Figura 3.4: Árvore filogenética baseada na distância PAM, para 41 seqüências de globinas em vertebrados.

Capítulo 4

Velocidade de convergência do algoritmo PST

Neste capítulo demonstramos a velocidade exponencial de convergência do algoritmo PST, para o caso de cadeias estocásticas de memória variável, com memória não necessariamente limitada. Esse resultado generaliza o obtido por Galves et al. (2006) para as cadeias de memória limitada.

4.1 Árvores probabilísticas de sufixos

Como nos capítulos anteriores, A representará um alfabeto finito, de tamanho $|A|$. Esse será o espaço de estados de todas as cadeias estocásticas consideradas. Lembraremos aqui algumas notações introduzidas no Capítulo 1. Para $m \leq n$ denotamos com w_m^n a seqüência $(w_m, \ldots, w_n)$ de símbolos no alfabeto A, de comprimento $|w| = n - m + 1$. Uma seqüência w_m^n, com $m > n$ é denotada com o símbolo especial λ e tem comprimento $|\lambda| = 0$. Dadas duas seqüências $w = w_m^n$ e $v = v_j^k$, denotamos por vw a seqüência com comprimento $|v| + |w|$, obtida pela concatenação das duas seqüências. Em particular, $\lambda w = w\lambda = w$. No caso em que vw denote uma seqüência temporal (como as seqüências condicionais de cadeias estocásticas), assumiremos que v se refere ao passado mais remoto. A definição de concatenação estende-se também ao caso em que v denota uma seqüência semi-infinita, do tipo $v = v_{-\infty}^{-1}$.

Diremos que seqüência s é um *sufixo* da seqüência w se existe uma seqüência u, com $|u| \geq 1$, tal que $w = us$. Dada a seqüência finita $w = w_m^n$, denotamos com $\mathrm{suf}(w)$ ao maior sufixo de w; isto é, $\mathrm{suf}(w) = w_{m+1}^n$.

O conjunto A^j denota todas as seqüências de símbolos em A tais que $|w| = j$. Definimos o conjunto A^* como o conjunto de todas as seqüências finitas sobre A; isto é

$$A^* = \bigcup_{j=1}^{\infty} A^j.$$

Para qualquer inteiro $h \geq 1$ denotamos com A_1^h o conjunto de todas as seqüências finitas de

comprimento como máximo h; isto é,

$$A_1^h = \bigcup_{j=1}^{h} A^j.$$

Definição 4.1.1. Um subconjunto enumerável τ de A^* é uma *árvore* se nenhuma seqüência $s_{-j}^{-1} \in \tau$ é um sufixo de uma outra seqüência $w_{-i}^{-1} \in \tau$. Essa propriedade é chamada de *propriedade do sufixo.*

Definimos a *profundidade* de uma árvore τ como

$$d(\tau) = \sup\{|w| : w \in \tau\}.$$

No caso em que $d(\tau)$ é finito, τ contem um número finito de seqüências. Nesse caso usaremos a notação $|\tau|$ para referirmos à quantidade de seqüências da árvore τ.

Definição 4.1.2. Diremos que a árvore τ é *ilimitada* se $d(\tau) = +\infty$.

Uma árvore τ é chamada de *irredutível* se nenhuma seqüência $w \in \tau$ pode ser substituída por um sufixo dela sem violar a propriedade do sufixo. A noção de árvore irredutível generaliza a noção de árvore *completa*, usualmente utilizada no contexto das cadeias estocásticas de memória variável. A partir daqui a palavra árvore sempre denotará uma árvore irredutível.

É fácil ver que o conjunto τ pode ser identificado com o conjunto de folhas (nós terminais) de uma árvore com raíz com um número enumerável de ramos finitos.

Definição 4.1.3. Uma *árvore probabilística de sufixos sobre* A é um par ordenado (τ, p) tal que

1. τ é uma árvore irredutível;
2. $p = \{p(\cdot|w); w \in \tau\}$ é uma família de probabilidades de transição sobre A.

Consideremos uma cadeia estocástica estacionária $\{X_t : t \in \mathbb{Z}\}$ sobre o alfabeto A. Dada a seqüência $w \in A^j$ denotamos por

$$p(w) = \mathbb{P}(X_1^j = w)$$

a probabilidade estacionária do cilindro definido pela seqüência w. Se $p(w) > 0$, escrevemos

$$p(a|w) = \mathbb{P}(X_0 = a \mid X_{-j}^{-1} = w).$$

No que segue usaremos a notação X_t para designar o processo $\{X_t : t \in \mathbb{Z}\}$.

Definição 4.1.4. A seqüência $w \in A^j$ é um *contexto* para o processo X_t se $p(w) > 0$ e para toda seqüência $x_{-\infty}^{-1}$ tal que w é um sufixo de $x_{-\infty}^{-1}$ temos que

$$\mathbb{P}(X_0 = a \mid X_{-\infty}^{-1} = x_{-\infty}^{-1}) = p(a|w), \quad \text{para todo } a \in A, \tag{4.1}$$

e nenhum sufixo de w tem essa propriedade.

Com essa definição podemos ver que o conjunto de todos os contextos de um processo X_t constitui uma árvore irredutível. Essa árvore será chamada de *árvore de contextos* associada ao processo X_t. Note também que o par composto pela árvore de contextos e pelas probabilidades de transição, dadas por (4.1), constitui a árvore probabilística de sufixos (τ, p) associada ao processo X_t.

Definição 4.1.5. Uma árvore probabilística de sufixos (τ, p) é do *tipo-A* se suas probabilidades de transição p satisfazem

1. $\sum_{a \in A} \inf_{w \in \tau} p(a|w) > 0$;
2. $\lim_{k \to +\infty} \beta_k = 0$, onde a seqüência $\{\beta_k\}_{k \in \mathbb{N}}$ está definida por

$$\beta_0 := \sup\{|p(a|w) - p(a|v)|, a \in A, v, w \in \tau \text{ com } w_{-1} \neq v_{-1}\}$$

e para $k \geq 1$,

$$\beta_k := \sup\{|p(a|w) - p(a|v)|, a \in A, v, w \in \tau \text{ com } w_{-k}^{-1} = v_{-k}^{-1}\}. \tag{4.2}$$

A seqüência $\{\beta_k\}_{k \in \mathbb{N}}$ é chamada de *taxa de continuidade*.

A partir deste momento assumiremos que a árvore probabilística de sufixos (τ, p) é do tipo-A, com taxa de continuidade somável; isto é

$$\beta := \sum_{k \in \mathbb{N}} \beta_k < +\infty.$$

Para evitar alguns problemas técnicos da demonstração do teorema de velocidade de convergência também assumiremos que $\beta_0 < 1$.

No caso de uma árvore probabilística do tipo-A e com taxa de continuidade somável, o argumento de acoplamento maximal usado em Fernández & Galves (2002) implica a unicidade da lei da cadeia associada a ela.

Dada uma árvore probabilística de sufixos (τ, p), associada ao processo X_t e um inteiro $k \geq 1$, denotaremos por (τ^k, p^k) a árvore probabilística de sufixos dada por

$$\tau^k = \{w \in \tau; |w| \leq k\} \cup \{w_{-k}^{-1}; w \in \tau, |w| > k\} \tag{4.3}$$

e para todo $w \in \tau^k$,

$$p^k(a|w) = \mathbb{P}(X_0 = a \mid X_{-1} = w_{-1}, \ldots, X_{-|w|} = w_{-w}) \,. \tag{4.4}$$

A árvore τ^k corresponde à árvore τ truncada ao nível k. Como exemplo podem ser observadas as árvores probabilísticas da Figura 4.1.

Seja, para $k \geq 1$,

$$D_k = \min_{w \in \tau^k} \max_{a \in A} \{ \, |p(a|w) - p(a|\text{suf}(w))| \, \},$$

onde, no caso $\text{suf}(w) = \lambda$ definimos $p(a|\lambda) = p(a)$. Por outro lado, seja

$$\epsilon_k = \min_{w \colon |w| \leq k} \{ \, p(w) \colon p(w) > 0 \, \}.$$

Condição 4.1.6. A partir de agora assumiremos que o processo X_t, com árvore probabilística de sufixos (τ, p), satisfaz as seguintes condições:

1. (τ, p) é do tipo-A, com taxa de continuidade somável e $\beta_0 < 1$.
2. $D_k > 0$ para todo $k \geq 1$.

Exemplo 4.1.7. Seja (τ, p) a árvore probabilística de sufixos da Figura 4.1(a). O conjunto de símbolos desse processo é o alfabeto $A = \{0, 1\}$. Consideremos as probabilidades de transição dadas por

$$q_l = p(0|1\underbrace{00\ldots0}_{l\ 0's}) = 1 - p(1|1\underbrace{00\ldots0}_{l\ 0's}), \tag{4.5}$$

onde $(q_l)_{l \geq 0}$ é uma seqüência de números positivos, somável e estritamente decrescente. Nesse e em todos os casos seguintes, a seqüência condicional na expressão (4.5) vai desde o passado mais remoto ao mais próximo. Isto é,

$$p(0|1\underbrace{00\ldots0}_{l\ 0's}) = \mathbb{P}(X_0 = 0 \mid X_{-1} = 0, \ldots, X_{-l} = 0,\, X_{-(l+1)} = 1) \,.$$

Assim, no caso das probabilidades de transição dadas por (4.5) temos que

$$\beta_k = \sup_{i,j \geq k} \{|q_i - q_j|\} = q_k.$$

Portanto, essa família de cadeias estocásticas tem taxa de continuidade somável. Por outro lado temos que

$$1 - p(1) = p(0) = p(00) + p(10) = p(00) + q_0 p(1).$$

Assim,

$$p(00) = 1 - p(1)[1 + q_0].$$

Em geral, para $l \geq 2$ é fácil ver que

$$p(\underbrace{00\ldots0}_{l\ 0's}) = 1 - p(1)[1 + q_0 + \cdots + \prod_{j=0}^{l-2} q_j].$$

Desta forma temos que

$$\begin{aligned} p(0|\underbrace{00\ldots0}_{l\ 0's}) &= \frac{1 - p(1)[1 + q_0 + \cdots + \prod_{j=0}^{l-1} q_j]}{1 - p(1)[1 + q_0 + \cdots + \prod_{j=0}^{l-2} q_j]} \\ &= 1 - \frac{p(1)\prod_{j=0}^{l-1} q_j}{1 - p(1)[1 + q_0 + \cdots + \prod_{j=0}^{l-2} q_j]}. \end{aligned}$$

Portanto, vemos que $D_k > 0$ para todo $k \geq 1$. Concluímos que a família de cadeias estocásticas com árvore probabilística de sufixos como a da Figura 4.1(a) e probabilidades como em (4.5) satisfazem a Condição 4.1.6. Um exemplo particular disso é dado por

$$q_l = (\frac{1}{l+2})^2, \qquad \text{para } l \geq 0.$$

4.2 Estimação com o algoritmo PST

No que resta deste capítulo assumiremos que $x_1, x_2, \ldots$ é uma amostra de uma cadeia estocástica associada à árvore probabilística de sufixos (τ, p), que satisfaz a Condição 4.1.6. Nesse caso diremos que $x_1, x_2, \ldots$ é uma *realização* de (τ, p).

Como antes, para qualquer seqüência w com $1 \leq |w| \leq n$, denotamos com $N_n(w)$ o número de ocorrências da seqüência na amostra $x_1, x_2, \ldots, x_n$; isto é

$$N_n(w) = \sum_{t=0}^{n-|w|} \mathbf{1}\{x_{t+1}^{t+|w|} = w\} \tag{4.6}$$

e $N_n(w\cdot)$ denotará a soma $\sum_{b\in A} N_n(wb)$. Por outro lado, definimos $N_n(\lambda) = N_n(\lambda\cdot) = n$.

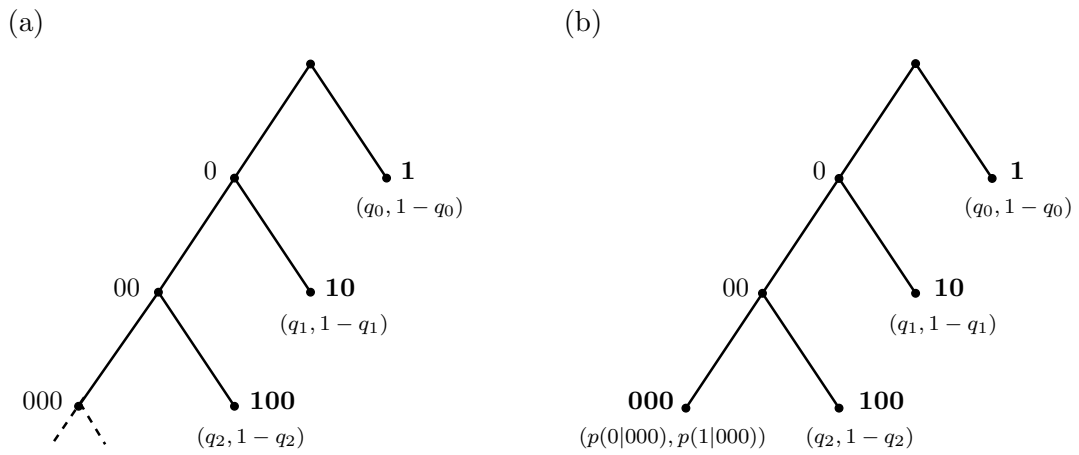

Figura 4.1: Exemplo de árvore probabilística de sufixos sobre o alfabeto $A = \{0, 1\}$. As seqüências correspondentes aos nós terminais constituem os contextos do processo. (a) Árvore de contextos de memória ilimitada (ver Exemplo 4.1.7). (b) Árvore em (a) truncada ao nível 3.

Para qualquer símbolo $a \in A$ e qualquer seqüência w tal que $N_n(w\cdot) \geq 1$, as probabilidades de transição empíricas $\hat{p}_n(a|w)$ estão definidas por

$$\hat{p}_n(a|w) = \frac{N_n(wa)}{N_n(w\cdot)}. \tag{4.7}$$

No caso $N_n(w\cdot) = 0$ definimos $\hat{p}_n(a|w) = \frac{1}{|A|}$.

O algoritmo PST, apresentado no Capítulo 1, está definido da seguinte forma. Em primeiro lugar, definimos para qualquer seqüência finita $w \in A^*$:

$$\Delta_n^{\mathrm{PST}}(w) = \max_{a \in A} |\hat{p}_n(a|w) - \hat{p}_n(a|\mathrm{suf}(w))|.$$

O operador $\Delta_n^{\mathrm{PST}}(w)$ calcula uma distância entre a distribuição de probabilidades de transição associada à seqüência w e a associada à seqüência $\mathrm{suf}(w)$, ambas estimadas a partir da expressão (4.7). No que resta deste capítulo usaremos a notação $\Delta_n(w)$ para referirmos ao operador $\Delta_n^{\mathrm{PST}}(w)$.

Definição 4.2.1. Dado $\delta > 0$ e $k < n$, a árvore estimada pelo algoritmo PST está dada por

$$\hat{\tau}_n^k = \{w \in A_1^k : \Delta_n(w) > \delta \ \wedge \ \Delta_n(uw) \leq \delta \ \ \forall u \in A_1^{k-|w|}\},$$

onde no caso $|w| = k$ temos que $A_1^{k-|w|} = \emptyset$.

É fácil ver que $\hat{\tau}_n^k$ é uma árvore. Além disso, a forma como definimos $\hat{p}_n(\cdot|\cdot)$ em (4.7) associa uma medida de probabilidade para cada seqüência em $\hat{\tau}_n^k$.

4.3 Desigualdades exponenciais para as probabilidades empíricas

O resultado principal deste capítulo é a demonstração da velocidade exponencial de convergência do algoritmo PST no caso de árvores com memória finita, porém ilimitada. Como já foi mencionado, esse resultado generaliza o obtido por Galves et al. (2006) para o caso de árvores finitas.

O principal ingrediente da demonstração da velocidade de convergência do algoritmo PST são algumas desigualdades exponenciais, que serão demonstradas nesta seção. Essas desigualdades foram obtidas em um contexto geral em Dedecker & Prieur (2005, Proposition 5), a partir de um resultado em Dedecker & Doukhan (2003, Proposition 4). Nesta seção mostraremos que esses resultados podem ser aplicados no caso de cadeias estocásticas de memória variável não necessariamente limitada.

O resultado principal desta seção é o seguinte

Teorema 4.3.1. *Para qualquer seqüência finita w e qualquer $t > 0$ vale que*

$$\mathbb{P}(\,|N_n(w) - (n - |w| + 1)p(w)| \,>\, t\,) \,\leq\, e^{\frac{1}{e}} \exp\Big[\frac{-t^2 C}{(n - |w| + 1)(|w| + 1)}\Big], \tag{4.8}$$

onde

$$C = \frac{1 - \beta_0}{8e(1 + \beta)}.$$

A demonstração segue o caminho desenvolvido em Galves et al. (2006). Para isso, precisamos de um resultado de mistura para um processo estocástico consistente com uma árvore probabilística de sufixos (τ, p), de memória não necessariamente limitada. Esse resultado é o seguinte

Lema 4.3.2. *Existe uma seqüência $\{\beta_k^*\}_{k\geq 0}$, com $\sum_{k\geq 0} \beta_k^* < +\infty$, tal que para qualquer $i \geq 1$, qualquer $k \geq i$, qualquer $j \geq 1$ e qualquer seqüência finita w_1^j, vale a seguinte desigualdade*

$$\sup_{x_1^i \in A^i} |\mathbb{P}(X_k^{k+j-1} = w_1^j \mid X_1^i = x_1^i) - p(w_1^j)| \leq (j + 1)\, \beta_{k-i}^* \,. \tag{4.9}$$

Além disso,

$$\sum_{k\geq 0} \beta_k^* \;\leq\; \frac{2\,(1 + \beta)}{1 - \beta_0}.$$

Demonstração. Definimos $\beta_0^* = 1$, logo (4.9) vale trivialmente para $k = i$. Por outro lado é fácil ver que para todo $i \geq 1$,

$$\inf_{u \in A^\infty} p(a|x_{-i}^{-1}u_{-\infty}^{-i-1}) \;\leq\; p(a|x_{-i}^{-1}) \;\leq\; \sup_{u \in A^\infty} p(a|x_{-i}^{-1}u_{-\infty}^{-i-1}), \tag{4.10}$$

onde A^∞ denota o conjunto de todas as seqüências semi-infinitas $u_{-\infty}^{-1}$. Para uma demonstração das desigualdades acima ver, por exemplo, Fernández & Galves (2002, Proposition 3). Usando esse fato e a condição de estacionaridade é suficiente provar que para todo $k \geq 0$,

$$\sup_{x \in A^\infty} |\mathbb{P}(X_k^{k+j-1} = w_1^j \mid X_{-\infty}^{-1} = x_{-\infty}^{-1}) - p(w_1^j)| \leq (j+1)\,\beta_k^*. \tag{4.11}$$

Note que para todos os passados $x_{-\infty}^{-1}$ temos que

$$\left|\mathbb{P}(X_k^{k+j-1} = w_1^j \mid X_{-\infty}^{-1} = x_{-\infty}^{-1}) - p(w_1^j)\right|$$

é igual a

$$\Big|\int_{u \in A^\infty} \big[\mathbb{P}(X_k^{k+j-1} = w_1^j | X_{-\infty}^{-1} = x_{-\infty}^{-1}) - \mathbb{P}(X_k^{k+j-1} = w_1^j | X_{-\infty}^{-1} = u_{-\infty}^{-1})\big]\, dp(u)\Big|.$$

A última expressão pode ser limitada superiormente por

$$\int_{u \in A^\infty} \left|\mathbb{P}(X_k^{k+j-1} = w_1^j | X_{-\infty}^{-1} = x_{-\infty}^{-1}) - \mathbb{P}(X_k^{k+j-1} = w_1^j | X_{-\infty}^{-1} = u_{-\infty}^{-1})\right| dp(u).$$

Agora, usando os resultados em Bressaud et al. (1999, Corollary 1) temos que

$$\left|\mathbb{P}(X_k^{k+j-1} = w_1^j \mid X_{-\infty}^{-1} = x_{-\infty}^{-1}) - \mathbb{P}(X_k^{k+j-1} = w_1^j \mid X_{-\infty}^{-1} = u_{-\infty}^{-1})\right| \leq\; (j+1)\mathbb{P}(S_k = 0),$$

onde $(S_n)_{n \in \mathbb{N}}$ é uma cadeia de Markov sobre $\mathbb{N}$ com probabilidades de transição dadas por

$$\mathbb{P}(S_{n+1} = x+1 \mid S_n = x) = 1 - \beta_x,$$
$$\mathbb{P}(S_{n+1} = 0 \mid S_n = x) = \beta_x$$

e probabilidade inicial $P(S_0 = 0) = 1$. Para $k \geq 1$ definimos $\beta_k^* = \mathbb{P}(S_{k-1} = 0)$. Assim temos que $\beta_2^* = \beta_0 < 1$. Pode ser visto que pelo fato de ser $\{\beta_k\}_{k \geq 0}$ somável, a seqüência $\{\beta_k^*\}_{k \geq 0}$

também é somável (Bressaud et al., 1999, Proposition 2). Por outro lado, temos que

$$\begin{aligned}
\prod_{k\geq 2}(1-\beta_k^*) &= \prod_{k\geq 1}(1-\mathbb{P}(S_k=0)) \\
&= \prod_{k\geq 1}\sum_{x=1}^{k}\mathbb{P}(S_k=x) \\
&= \prod_{k\geq 1}\sum_{x=1}^{k}\mathbb{P}(S_k=x \mid S_{k-1}=x-1)\mathbb{P}(S_{k-1}=x-1) \\
&= \prod_{k\geq 1}\sum_{x=1}^{k}(1-\beta_{x-1})\mathbb{P}(S_{k-1}=x-1) \\
&\geq \prod_{k\geq 1}(1-\beta_{k-1})\mathbb{P}(S_{k-1}=k-1) \\
&= \prod_{k\geq 1}(1-\beta_{k-1})\prod_{i=0}^{k-2}(1-\beta_i) \\
&\geq \big[\prod_{k\geq 0}(1-\beta_k)\big]^2 > 0\,. \qquad (4.12)
\end{aligned}$$

Assim,

$$-\sum_{k\geq 2}\ln(1-\beta_k^*) \leq -2\sum_{k\geq 0}\ln(1-\beta_k).$$

Agora, usando a expansão em série de Taylor da função logaritmo é fácil ver que

$$x \leq -\ln(1-x) \leq \frac{x}{1-c}, \qquad (4.13)$$

para todo $x \in (-1, c]$. Desta forma, vemos que

$$\sum_{k\geq 0}\beta_k^* \leq \frac{2\,(1+\sum_{k\geq 0}\beta_k)}{1-\beta_0}.$$

Esse fato conclui a demonstração do Lema 4.3.2. □

A seguir apresentamos a demonstração do Teorema 4.3.1. Essa demonstração usa fortemente a propriedade de mistura para cadeias estocásticas com taxa de continuidade somável, demonstrada no Lema 4.3.2.

Demonstração do Teorema 4.3.1. Seja w uma seqüência finita. Seguindo o mesmo caminho que em Galves et al. (2006, Theorem 2.5) obtemos, a partir do resultado em Dedecker & Doukhan

(2003, Proposition 4), que para todo $p \geq 2$ vale

$$\begin{aligned}\|N_n(w) - \mathbb{E}(N_n(w))\|_p &\leq \Big(2p \sum_{i=1}^{n-|w|+1} \sum_{k=i}^{n-|w|+1} \sup_{x_1^i \in A^i} |\mathbb{P}(X_k^{k+|w|-1} = w \,|\, X_1^i = x_1^i) - p(w)|\Big)^{\frac{1}{2}} \\ &\leq (2p(n-|w|+1)(|w|+1)\beta^*)^{\frac{1}{2}},\end{aligned}$$

onde $\beta^* = \sum_{k\geq 0} \beta_k^*$. Logo, também obtemos, a partir de Dedecker & Prieur (2005, Proposition 5), que para todo $t > 0$,

$$\mathbb{P}(|N_n(w) - (n-|w|+1)p(w)| > t) \leq e^{\frac{1}{e}} \exp\big[\frac{-t^2 C}{(|w|+1)(n-|w|+1)}\big],$$

onde

$$C = \frac{1-\beta_0}{8e(1+\beta)}.$$

□

Como conseqüência direta do Teorema 4.3.1 obtemos os seguintes corolários.

Corolário 4.3.3. *Para qualquer seqüência finita w tal que $p(w) > 0$ e para qualquer r satisfazendo $r < (n-|w|+1)p(w)$ vale que*

$$\mathbb{P}(N_n(w) \leq r) \leq e^{\frac{1}{e}} \exp\big[-(n-|w|+1)\,\frac{[p(w) - \frac{r}{n-|w|+1}]^2 C}{|w|+1}\big]. \tag{4.14}$$

Demonstração. Como $r < (n-|w|+1)p(w)$ temos que

$$\begin{aligned}\mathbb{P}(N_n(w) \leq r) &= \mathbb{P}(N_n(w) - (n-|w|+1)p(w) \leq r - (n-|w|+1)p(w)) \\ &\leq \mathbb{P}(|N_n(w) - (n-|w|+1)p(w)| \geq (n-|w|+1)p(w) - r)\end{aligned}$$

Usando o Teorema 4.3.1 podemos limitar superiormente o termo direito da última desigualdade por

$$e^{\frac{1}{e}} \exp\big[-(n-|w|+1)\,\frac{[p(w) - \frac{r}{n-|w|+1}]^2 C}{|w|+1}\big].$$

□

Corolário 4.3.4. *Para qualquer seqüência finita w, qualquer símbolo $a \in A$ tal que $p(wa) > 0$ e qualquer $n > |w|$ vale que*

$$\mathbb{P}\big(|\hat{p}_n(a|w) - p(a|w)| > t\big) \leq 3\, e^{\frac{1}{e}} \exp\big[-(n-|w|)\,\frac{\min(t^2,1)p(wa)^2 C}{4(|w|+2)}\big]. \tag{4.15}$$

Demonstração. A desigualdade (4.15) segue de (4.8) e (4.14), como detalhado a seguir. Temos que

$$\mathbb{P}\big(|\hat{p}_n(a|w) - p(a|w)| > t\big) \leq \mathbb{P}\big(|\hat{p}_n(a|w) - p(a|w)| > t, N_n(w\cdot) \geq 1\big) + \mathbb{P}\big(N_n(w\cdot) = 0\big).$$

Assim, como $N_n(w\cdot) = 0$ implica $N_n(wb) = 0$ para todo $b \in A$, podemos usar o Corolário 4.3.3, obtendo

$$\mathbb{P}\big(N_n(w\cdot) = 0\big) \leq e^{\frac{1}{e}} \exp\big[-(n - |w|)\, \frac{p(wa)^2 C}{|w| + 2}\big]. \tag{4.16}$$

Por outro lado, usando a expressão (4.7) e o fato que

$$p(a|w) = \frac{p(wa)}{p(w)},$$

temos, multiplicando e dividindo por $n - |w|$ a expressão acima que

$$\mathbb{P}\big(|\hat{p}_n(a|w) - p(a|w)| > t, N_n(w\cdot) \geq 1\big) \leq \mathbb{P}\big(\,|\frac{N_n(wa)}{N_n(w\cdot)} - \frac{(n - |w|)p(wa)}{(n - |w|)p(w)}| \,>\, t\big). \tag{4.17}$$

Somando e subtraindo o termo $\frac{N_n(wa)}{(n-|w|)p(w)}$ obtemos

$$\begin{aligned}|\frac{N_n(wa)}{N_n(w\cdot)} - \frac{(n - |w|)p(wa)}{(n - |w|)p(w)}| \;\leq\; & \frac{N_n(wa)}{N_n(w\cdot)(n - |w|)p(w)}\, \big|(n - |w|)p(w) - N_n(w\cdot)\big| + \\ & \frac{1}{(n - |w|)p(w)}\, \big|N_n(wa) - (n - |w|)p(wa)\big|.\end{aligned}$$

Desta forma, usando que $\frac{N_n(wa)}{N_n(w\cdot)} \leq 1$ e $N_n(w\cdot) = N_{n-1}(w)$, o lado direito em (4.17) pode ser limitado superiormente pela soma

$$\begin{aligned}\mathbb{P}\big(\,|N_{n-1}(w) - (n - |w|)p(w)| \,>\, \frac{t(n - |w|)p(w)}{2}\big) \;+ \\ \mathbb{P}\big(\,|N_n(wa) - (n - |w|)p(wa)| \,>\, \frac{t(n - |w|)p(w)}{2}\big).\end{aligned}$$

Usando o Teorema 4.3.1 podemos limitar superiormente a última expressão por

$$2\, e^{\frac{1}{e}}\, \exp\big[-(n - |w|)\, \frac{t^2 p(wa)^2 C}{4(|w| + 2)}\big]. \tag{4.18}$$

Assim, somando (4.16) e (4.18) obtemos (4.15). □

4.4 Velocidade de convergência do algoritmo PST

O principal resultado deste capítulo é o seguinte

Teorema 4.4.1. *Seja (τ, p) uma árvore probabilística de sufixos que satisfaz a Condição 4.1.6 e seja $x_1, x_2, \ldots, x_n$ uma realização de (τ, p). Então para qualquer k, qualquer $\delta < D_k$ e para todo $n > k$ temos que*

$$\mathbb{P}(\hat{\tau}_n^k = \tau^k) \;\geq\; 1 \;-\; 6\,|A|^{k+1}\, e^{\frac{1}{e}}\, \exp\big[-(n-k)\,\frac{\min(D_k-\delta,\delta)^2\epsilon_{k+1}^2 C}{16(k+2)}\big],$$

onde

$$C = \frac{1-\beta_0}{8e(1+\beta)}.$$

Demonstração. Definamos

$$O_n(uw) = \{\Delta_n(uw) > \delta\}\,,\;\; O_n^k = \bigcup_{\substack{w\in\tau^k \\ uw\in\hat{\tau}_n^k}} O_n(uw),$$

e

$$U_n(uw) = \{\Delta_n(uw) \leq \delta\}\,,\;\; U_n^k = \bigcup_{\substack{uw\in\tau^k \\ w\in\hat{\tau}_n^k}} U_n(uw).$$

Assim, se $k < n$ então

$$\{\hat{\tau}_n^k \neq \tau^k\} = O_n^k \cup U_n^k.$$

O resultado segue de uma sucessão de lemas.

Lema 4.4.2. *Para qualquer $w \in \tau^k$, $uw \in \hat{\tau}_n^k$ temos que*

$$\mathbb{P}(O_n(uw)) \;\leq\; 6\,|A|\, e^{\frac{1}{e}}\, \exp\big[-(n-k)\,\frac{\delta^2\epsilon_{k+1}^2 C}{16(k+2)}\big].$$

Demonstração. Lembremos que

$$\Delta_n(uw) = \max_{a\in A} |\hat{p}_n(a|uw) - \hat{p}_n(a|\mathrm{suf}(uw))|.$$

Note que se $uw \in \hat{\tau}_n^k$ então $|w| < k$. Isto implica que $w \in \tau$. Logo, para qualquer seqüência finita u e qualquer símbolo $a \in A$ temos que $p(a|w) = p(a|uw)$. Portanto,

$$\mathbb{P}(\Delta_n(uw) > \delta) \;\leq\; \sum_{a\in A} \big[\,\mathbb{P}\big(|\hat{p}_n(a|w) - p(a|w)| > \frac{\delta}{2}\big) + \mathbb{P}\big(|\hat{p}_n(a|uw) - p(a|uw)| > \frac{\delta}{2}\big)\big].$$

Usando o Corolário 4.3.4 podemos limitar superiormente o termo direito da desigualdade acima por

$$6\,|A|\, e^{\frac{1}{e}}\, \exp\big[-(n-k)\,\frac{\delta^2 p(uwa)^2 C}{16(|uw|+2)}\big].$$

Lembremos que, por definição, $|uw| \leq k$ e $p(uwa) \geq \epsilon_{k+1}$. Isso conclui a demonstração. □

Lema 4.4.3. *Para qualquer $uw \in \tau^k$ e $w \in \hat{\tau}_n^k$ temos que*

$$\mathbb{P}(U_n(uw)) \;\leq\; 6\, e^{\frac{1}{e}}\, \exp\big[-(n-k)\,\frac{(D_k-\delta)^2\epsilon_{k+1}^2 C}{16(k+2)}\big].$$

Demonstração. Começamos observando que para qualquer $a \in A$,

$$|\hat{p}_n(a|\mathrm{suf}(uw)) - \hat{p}_n(a|uw)| \;\geq\; |p(a|\mathrm{suf}(uw)) - p(a|uw)| - |\hat{p}_n(a|\mathrm{suf}(uw)) - p(a|\mathrm{suf}(uw))| - |\hat{p}_n(a|uw) - p(a|uw)|.$$

Assim, para todo $a \in A$ temos que

$$\Delta_n(uw) \;\geq\; D_k - |\hat{p}_n(a|\mathrm{suf}(uw)) - p(a|\mathrm{suf}(uw))| - |\hat{p}_n(a|uw) - p_\tau(a|uw)|.$$

Portanto,

$$\begin{aligned}\mathbb{P}(\Delta_n(uw) \leq \delta) \;\;\leq\;\; & \mathbb{P}\big(\bigcap_{a\in A}\{\,|\hat{p}_n(a|\mathrm{suf}(uw)) - p(a|\mathrm{suf}(uw))| \geq \frac{D_k-\delta}{2}\,\}\big) \\ & + \mathbb{P}\big(\bigcap_{a\in A}\{\,|\hat{p}_n(a|uw) - p(a|uw)| \geq \frac{D_k-\delta}{2}\,\}\big).\end{aligned}$$

Como $\delta < D_k$ podemos usar o Corolário 4.3.4 como antes para limitar superiormente o lado direito dessa desigualdade por

$$6\, e^{\frac{1}{e}}\, \exp\big[-(n-k)\,\frac{(D_k-\delta)^2\epsilon_{k+1}^2 C}{16(k+2)}\big].$$

Isso conclui a demonstração. □

Agora podemos finalizar a demonstração do Teorema 4.4.1. Temos que

$$\mathbb{P}(\hat{\tau}_n^k \neq \tau^k) = \mathbb{P}(O_n^k) + \mathbb{P}(U_n^k).$$

Segue da definição de O_n^k e U_n^k que

$$\mathbb{P}(\hat{\tau}_n^k \neq \tau^k) \leq \sum_{\substack{w\in\tau^k \\ uw\in\hat{\tau}_n^k}} \mathbb{P}(O_n(uw)) + \sum_{\substack{uw\in\tau^k \\ w\in\hat{\tau}_n^k}} \mathbb{P}(U_n(uw)).$$

Usando o Lema 4.4.2 e o Lema 4.4.3 obtemos a desigualdade

$$\mathbb{P}(\hat{\tau}_n^k \neq \tau^k) \;\leq\; 6\,|A|^{k+1}\, e^{\frac{1}{e}}\, \exp\big[-(n-k)\,\frac{\min(D_k-\delta,\delta)^2\epsilon_{k+1}^2 C}{16(k+2)}\big].$$

Com isso concluimos a demonstração do Teorema 4.4.1. □

Uma primeira conseqüência do teorema é o seguinte resultado de consistência forte para as árvores probabilísticas de sufixos empíricas, truncadas a um nível k.

Corolário 4.4.4. *Seja $k \in \mathbb{N}$ uma constante arbitrária fixa. Para qualquer $\delta \in (0, D_k)$, qualquer $\kappa > 0$ e para quase toda realização infinita $x_1, x_2, \ldots$ de (τ, p) existe um $\bar{n}$ tal que para todo $n \geq \bar{n}$ temos que*

1. $\hat{\tau}_n^k = \tau^k$;
2. $\max\limits_{w \in \tau^k,\, a \in A} |\hat{p}_n(a|w) - p(a|w)| < \kappa$.

A demonstração utiliza o Lema de Borel-Cantelli, um resultado clássico da teoria das probabilidades. A continuação, apresentamos o enunciado desse lema, cuja demonstração pode ser consultada em Stroock (1993).

Lema 4.4.5 (Primeiro Lema de Borel-Cantelli)**.** *Seja $(B_n)_{n=1}^{\infty}$ uma seqüência de eventos tais que $\sum_{n=1}^{\infty} \mathbb{P}(B_n) < +\infty$. Logo,*

$$\mathbb{P}(\cap_{n=1}^{\infty} \cup_{i=n}^{\infty} B_i) = 0.$$

Demonstração do Corolário 4.4.4. Sabemos, pela consistência forte dos estimadores de máxima verossimilhança das probabilidades de transição que $\hat{p}_n(a|w) \to p(a|w)$ quase certamente, para toda seqüência $w \in A^*$ e todo símbolo $a \in A$. Portanto, para cada $w \in \tau^k$ e $a \in A$ existe um inteiro $\bar{n}(w, a)$ tal que, para todo $n \geq \bar{n}(w, a)$, vale que

$$|\hat{p}_n(a|w) - p(a|w)| < \kappa.$$

Por outro lado, definimos os eventos

$$B_n = \{\hat{\tau}_n^k \neq \tau^k\}.$$

Usando o Teorema 4.4.1 e o Lema 4.4.5 concluímos que

$$\mathbb{P}(\cap_{n=1}^{\infty} \cup_{i=n}^{\infty} B_i) = 0.$$

Por tanto, existe quase certamente um $\bar{n}(\tau)$ tal que, para todo $n \geq \bar{n}(\tau)$, vale que

$$\hat{\tau}_n^k = \tau^k.$$

Tomando $\bar{n}$ igual ao máximo entre os $n(w, a)$, com $w \in \tau^k$, $a \in A$ e $n(\tau)$ concluímos a demonstração do Corolário 4.4.4. □

Capítulo 5

Estimação de árvores por máxima verossimilhança penalizada

Neste capítulo apresentamos a demonstração da velocidade de convergência de uma abordagem alternativa para a estimação de árvores probabilísticas. Essa abordagem está baseada na maximização do logaritmo da verossimilhança, penalizada com um termo que depende do tamanho da árvore e do tamanho da amostra. Uma forma particular desse tipo de critério, no caso em que a penalização cresce como o logaritmo do tamanho da amostra, é conhecida como Critério da Informação Bayesiana (BIC), ou Critério de Schwarz.

O BIC foi primeiramente utilizado para estimar a ordem de uma cadeia de Markov, que constitui um caso particular das cadeias estocásticas de memória variável. A consistência quase certamente do BIC nesse caso, sem assumir nenhuma limitante superior sobre a ordem da cadeia, foi provada por Csiszár & Shields (2000). Recentemente, foi provada também a consistência quase certamente desse estimador no caso de cadeias estocásticas de memória variável, com memória não necessariamente limitada (Csiszár & Talata, 2006), mas deixando crescer a ordem da cadeia como o logaritmo do tamanho da amostra. Em nenhum dos casos citados era conhecido até o momento um resultado de velocidade de convergência para o BIC.

5.1 Critério da Informação Bayesiana generalizado

Como no caso do capítulo anterior, assumiremos que o processo X_t, sobre o alfabeto finito A, tem árvore probabilística de sufixos (τ, p) e satisfaz a Condição 4.1.6. Isto é,

1. (τ, p) é do tipo-A, com taxa de continuidade somável e $\beta_0 < 1$.

2. $D_k > 0$ para todo $k \geq 1$.

Dada uma árvore τ', denotaremos com $\mathrm{Int}(\tau')$ o conjunto de todos os sufixos próprios de seqüências $w \in \tau'$. Para $w \in \mathrm{Int}(\tau^k)$ seja

$$\delta_k(w) = \sum_{uw \in \tau^k} \sum_{a \in A} p(uwa) \log p(a|uw) - \sum_{a \in A} p(wa) \log p(a|w). \tag{5.1}$$

Pelo Lema 1.2.1 temos que

$$\sum_{a\in A} p(wa)\log p(a|w) < \sum_{a\in A}\sum_{b\in A} p(bwa)\log p(a|bw),$$

já que $D_k > 0$ para todo $k \geq 1$, em particular para $k = |w|+1$. Iterando o mesmo procedimento para cada $b \in A$ na soma anterior concluímos que para cada $w \in \mathrm{Int}(\tau^k)$ vale que $\delta_k(w) > 0$. Denotaremos com

$$\delta_k = \min_{w\in \mathrm{Int}(\tau^k)} \{\, \delta_k(w) \,\} \tag{5.2}$$

e com

$$p_k = \min_{w\in\tau^k,\, a\in A} \{\, p(a|w) \colon\ p(a|w) > 0 \,\}. \tag{5.3}$$

Seja $x_1, \ldots, x_n$ uma realização da árvore probabilística de sufixos (τ, p). Para qualquer inteiro $k < n$, qualquer seqüência w, com $|w| \leq k$, e qualquer símbolo $a \in A$ definimos os contadores

$$N_n^k(w,a) = \sum_{t=k+1}^{n} \mathbf{1}\{x_{t-|w|}^{t-1} = w, x_t = a\}, \tag{5.4}$$

e

$$N_n^k(w,\cdot) = \sum_{b\in A} N_n(w,b)\,. \tag{5.5}$$

Note que essa definição é sensivelmente diferente da dada nos capítulos anteriores. Isso se deve ao fato que no caso dos estimadores apresentados neste capítulo são comparadas globalmente as verossimilhanças de todos os modelos envolvidos. Portanto, devemos começar a contagem de ocorrência das seqüências a partir da memória máxima possível, para não desfavorecer erroneamente os modelos de memória menor, já que a verossimilhança diminui com valores maiores de $N_n^k(\cdot,\cdot)$.

Desta forma, a verossimilhança da amostra no modelo dado pela árvore τ' está definida por

$$\hat{\mathbb{P}}_{\mathrm{ML},\tau'}(x_1^n) = \prod_{w\in\tau'}\prod_{a\in A} \hat{p}_n^k(a|w)^{N_n^k(w,a)}, \tag{5.6}$$

onde as probabilidades empíricas $\hat{p}_n^k(a|w)$ estão dadas por

$$\hat{p}_n^k(a|w) = \frac{N_n^k(w,a)}{N_n^k(w,\cdot)},$$

se $N_n^k(w,\cdot) \geq 1$, e $\hat{p}_n^k(a|w) = \frac{1}{|A|}$ se $N_n^k(w,\cdot) = 0$. Além disso usamos a convenção $0^0 = 1$ quando $N_n^k(w,a) = 0$ na expressão (5.6).

A seguir, definimos a noção de *árvore viável*, que constituirá o espaço de busca do estimador de máxima verossimilhança penalizada.

Definição 5.1.1. Dada uma amostra $x_1, \ldots, x_n$ e um inteiro $k < n$, diremos que a árvore τ' é *viável* se $d(\tau') \leq k$ e $N_n^k(w, \cdot) \geq 1$ para todo $w \in \tau'$. Além disso, se w' é tal que $N_n^k(w', \cdot) \geq 1$, então w' é um sufixo de algum $w \in \tau'$ ou tem um sufixo w que pertence a τ'.

Denotaremos com $\mathcal{T}^k(x_1^n)$ ao conjunto de todas as árvores viáveis.

Dada a amostra $x_1, \ldots, x_n$, a definição clássica do *Critério da Informação Bayesiana* (BIC), proposta por Schwarz (1978), está dada pela árvore τ' que minimiza a expressão

$$-\log \hat{\mathbb{P}}_{\mathrm{ML},\tau'}(x_1^n) + \frac{(|A|-1)|\tau'|}{2} \log n.$$

Nessa definição, a constante $(|A|-1)|\tau'|$ representa o número de parâmetros que devem ser estimados no modelo dado pela árvore τ'. A definição utilizada neste capítulo será um pouco mais geral, com o objetivo de estudar diferentes termos de penalização e de simplificar as constantes. Nesse caso, o BIC para a árvore τ' estará dado por

$$\mathrm{BIC}_{\tau'}(x_1^n) = -\log \hat{\mathbb{P}}_{\mathrm{ML},\tau'}(x_1^n) + |\tau'| f(n), \tag{5.7}$$

onde $f(n)$ é uma função positiva tal que $f(n) \to +\infty$, quando $n \to +\infty$, e $n^{-1} f(n) \to 0$, quando $n \to +\infty$.

Seguindo o mesmo caminho que em Csiszár & Talata (2006), consideraremos como espaço de busca o conjunto de árvores viáveis $\mathcal{T}^k(x_1^n)$. Assim,

Definição 5.1.2. Dada a amostra $x_1, \ldots, x_n$ e o inteiro $k < n$, o estimador $\hat{\tau}_{\mathrm{BIC}}^k(x_1^n)$ está definido por

$$\hat{\tau}_{\mathrm{BIC}}^k(x_1^n) = \underset{\tau' \in \mathcal{T}^k(x_1^n)}{\arg\min} \{ \mathrm{BIC}_{\tau'}(x_1^n) \},$$

onde $\mathrm{BIC}_{\tau'}(x_1^n)$ está dado por (5.7).

A consistência quase certamente do estimador $\hat{\tau}_{\mathrm{BIC}}^k(x_1^n)$ foi demonstrada recentemente em (Csiszár & Talata, 2006), para o caso de árvores ilimitadas e usando o termo de penalização $f(n) = \frac{(|A|-1)}{2} \log n$. Além da velocidade de convergência, que era desconhecida até o momento, tanto nesse caso quanto no caso de estimação da ordem de uma cadeia de Markov, existiam algumas outras perguntas ainda em aberto que podem ser respondida a partir dos resultados deste capítulo. Um exemplo disso é a consistência do estimador para termos de penalização com crescimento menor que $\log n$, como é o caso de $f(n) = \log \log n$.

5.2 Velocidade de convergência do BIC

Nesta seção apresentamos a demonstração do resultado de velocidade de convergência do estimador $\hat{\tau}^k_{\mathrm{BIC}}(x_1^n)$, definido na seção anterior. Antes disso, provaremos um resultado que apresenta uma fórmula recursiva para calcular uma limitante superior do tamanho do espaço de estados $\mathcal{T}^k(x_1^n)$, que é uma das constantes que aparecem no teorema de velocidade de convergência. Também provaremos um outro resultado básico que encontra uma limitante superior para a probabilidade de que o valor absoluto do logaritmo da razão entre as probabilidades de transição empírica e teórica seja maior que um certo valor positivo. Esse último resultado deriva-se das desigualdades exponenciais demonstradas no Capítulo 4, e será utilizado na demonstração do teorema de velocidade de convergência.

Lema 5.2.1. *A seqüência* $\{\, |\mathcal{T}^k(x_1^n)| \colon k = 1, 2, \dots \}$ *satisfaz as seguintes propriedades*

1. $|\mathcal{T}^1(x_1^n)| = 1$.
2. *Para todo* $k \geq 2$, $|\mathcal{T}^k(x_1^n)| \;\leq\; \big[1 + |\mathcal{T}^{k-1}(x_1^n)|\,\big]^{|A|}$.

Demonstração. A única árvore $\tau_1 \in \mathcal{T}^1(x_1^n)$ está dada pelo conjunto de seqüências

$$\tau_1 = \{a \in A : N_n^k(a, \cdot) \geq 1\}.$$

Isso prova 1. Por outro lado, para qualquer árvore $\tau_k \in \mathcal{T}^k(x_1^n)$ existe um conjunto de símbolos $A(\tau_k) = \tau_1 \cap \mathrm{Int}(\tau_k)$ e, para cada $a \in A(\tau_k)$ existe uma árvore $\tau_{k-1}(a) \in \mathcal{T}^{k-1}(x_1^n)$, tal que

$$\tau_k = \cup_{a \in A(\tau_k)} \{ua \colon u \in \tau_{k-1}(a)\} \cup (\tau_1 \setminus A(\tau_k)).$$

Portanto, para *gerar* todas as árvores $\tau_k \in \mathcal{T}^k(x_1^n)$ é suficiente tomar uma combinação de j símbolos de τ_1, para $j = 0, \dots, |\tau_1|$, e *sortear* (com reposição) j árvores de $\mathcal{T}^{k-1}(x_1^n)$. Assim, temos que

$$|\mathcal{T}^k(x_1^n)| \;=\; \sum_{j=0}^{|\tau_1|} \binom{|\tau_1|}{j} |\mathcal{T}^{k-1}(x_1^n)|^j \;=\; [\,1 + |\mathcal{T}^{k-1}(x_1^n)|\,]^{|\tau_1|} \;\leq\; [\,1 + |\mathcal{T}^{k-1}(x_1^n)|\,]^{|A|}.$$

□

Lema 5.2.2. *Para qualquer seqüência finita* w *e qualquer símbolo* $a \in A$ *tal que* $p(wa) > 0$ *e para qualquer* $t > 0$ *vale a seguinte desigualdade*

$$\mathbb{P}\Big[\, \big|\log \frac{\hat{p}_n^k(a|w)}{p(a|w)}\big| > t \,\Big] \;\leq\; 6\, e^{\frac{1}{e}} \exp\big[-(n-k)\, \frac{\min(t^2, 1) p(wa)^2 p(a|w)^2 C}{16(|w|+2)}\big], \tag{5.8}$$

onde

$$C = \frac{1 - \beta_0}{8e(1+\beta)}.$$

Demonstração. Usando a desigualdade (4.13) do capítulo anterior vemos que $|\log x| \leq 2|x-1|$ para x tal que $|x-1| \leq \frac{1}{2}$. Assim, temos que

$$\begin{aligned}\mathbb{P}\Big[\,\big|\log\frac{\hat{p}_n^k(a|w)}{p(a|w)}\big| > t\Big] \;&\leq\; \mathbb{P}\Big[\,\big|\frac{\hat{p}_n^k(a|w)}{p(a|w)} - 1\big| > \frac{t}{2}\Big] \;+\; \mathbb{P}\Big[\,\big|\frac{\hat{p}_n^k(a|w)}{p(a|w)} - 1\big| > \frac{1}{2}\Big] \\ &\leq\; 2\,\mathbb{P}\Big[\,\big|\hat{p}_n^k(a|w) - p(a|w)\big| > \frac{\min(t,1)p(a|w)}{2}\Big].\end{aligned}$$

Desta forma, usando o Corolário 4.3.4 podemos limitar superiormente a última expressão por

$$6\,e^{\frac{1}{e}}\,\exp\big[-(n-k)\,\frac{\min(t^2,1)p(wa)^2p(a|w)^2C}{16(|w|+2)}\big].$$

□

A seguir, apresentamos o resultado principal deste capítulo, a demonstração da velocidade de convergência do estimador $\hat{\tau}^k_{\mathrm{BIC}}(x_1^n)$.

Teorema 5.2.3. *Seja $x_1, x_2, \ldots$ uma realização de uma árvore probabilística (τ, p), que satisfaz a Condição 4.1.6, e seja k um inteiro. Então, existe um $\bar{n} \in \mathbb{N}$ tal que, para todo $n \geq \bar{n}$ vale que*

$$\begin{aligned}\mathbb{P}\big[\,\hat{\tau}^k_{\mathrm{BIC}}(x_1^n) \neq \tau^k\big] \leq e^{\frac{1}{e}}|A|^k \exp\big[-(n-k)\frac{\epsilon_k^2 C}{k+1}\big] + e^{\frac{1}{e}}|A|^{k+1}|\mathcal{T}^k(x_1^n)|\Big[\,3\exp\big[-\frac{f(n)\epsilon_{k+1}^2 p_k C}{4|A|^{k+1}(k+2)}\big] \\ +\;13\exp\big[-(n-k)\,\frac{[\delta_k - \frac{|A|^k f(n)}{n-k}]^2\epsilon_{k+1}^2 p_k^2 C}{144|\log p_k|^2|A|^{2(k+1)}(k+2)}\big]\,\Big], \qquad (5.9)\end{aligned}$$

onde τ^k é a árvore τ truncada ao nível k. Além disso,

$$\bar{n} \;=\; \inf_{m>k}\{\,\frac{|A|^k f(i)}{i-k} < \delta_k\,,\ \textit{para todo } i \geq m\}$$

e

$$C = \frac{1-\beta_0}{8e(1+\beta)}.$$

Demonstração. Primeiro notemos que

$$\begin{aligned}\mathbb{P}\big[\,\hat{\tau}^k_{\mathrm{BIC}}(x_1^n) \neq \tau^k\big] \;&\leq\; \mathbb{P}\Big[\bigcup_{\substack{\tau' \in \mathcal{T}^k(x_1^n)\\ \tau' \neq \tau^k}} \{\,\mathrm{BIC}_{\tau'}(x_1^n) < \mathrm{BIC}_{\tau^k}(x_1^n),\, \tau^k \in \mathcal{T}^k(x_1^n)\}\,\Big] + \mathbb{P}\big[\tau^k \notin \mathcal{T}^k(x_1^n)\big] \\ &\leq \sum_{\substack{\tau' \in \mathcal{T}^k(x_1^n)\\ \tau' \neq \tau^k}} \mathbb{P}\big[\,\mathrm{BIC}_{\tau'}(x_1^n) < \mathrm{BIC}_{\tau^k}(x_1^n),\, \tau^k \in \mathcal{T}^k(x_1^n)\,\big] + \mathbb{P}\big[\tau^k \notin \mathcal{T}^k(x_1^n)\big].\end{aligned}$$

Usando o Corolário 4.3.3 do capítulo anterior temos que

$$\mathbb{P}\big[\tau^k \notin \mathcal{T}^k(x_1^n)\big] \;\leq\; \sum_{w\in\tau^k} \mathbb{P}[N_n^k(w,\cdot)=0] \;\leq\; |\tau^k|\, e^{\frac{1}{e}}\, \exp\big[-(n-k)\,\frac{\epsilon_k^2 C}{k+1}\big],$$

onde

$$C = \frac{1-\beta_0}{8e(1+\beta)}.$$

Essa desigualdade, limitando $|\tau^k|$ com $|A|^k$, constitui o primeiro termo do lado direito em (5.9). Por outro lado, temos que

$$\begin{aligned}\sum_{\substack{\tau'\in\mathcal{T}^k(x_1^n)\\ \tau'\neq\tau^k}} \mathbb{P}\big[\,\mathrm{BIC}_{\tau'}(x_1^n) <\, \mathrm{BIC}_{\tau^k}(x_1^n),\, \tau^k\in\mathcal{T}^k(x_1^n)\,\big] = \\ \sum_{\substack{\tau'\in\mathcal{T}^k(x_1^n)\\ \tau'\neq\tau^k}} \mathbf{1}_{\{\tau'\cap\mathrm{Int}(\tau^k)=\emptyset\}}\mathbb{P}\big[\,\mathrm{BIC}_{\tau'}(x_1^n) < \mathrm{BIC}_{\tau^k}(x_1^n),\, \tau^k\in\mathcal{T}^k(x_1^n)\,\big] \;+ \\ \sum_{\substack{\tau'\in\mathcal{T}^k(x_1^n)\\ \tau'\neq\tau^k}} \mathbf{1}_{\{\tau'\cap\mathrm{Int}(\tau^k)\neq\emptyset\}}\mathbb{P}\big[\,\mathrm{BIC}_{\tau'}(x_1^n) < \mathrm{BIC}_{\tau^k}(x_1^n),\, \tau^k\in\mathcal{T}^k(x_1^n)\,\big].\end{aligned} \tag{5.10}$$

Para w tal que $N_n^k(w,\cdot)\geq 1$ vamos denotar por

$$\hat{\mathbb{P}}_{\mathrm{ML},w}(x_1^n) = \prod_{a\in A} \hat{p}_n^k(a|w)^{N_n^k(w,a)},$$

onde, como antes, usaremos a convenção $0^0=1$ no caso $N_n^k(w,a)=0$. Assim, temos que

$$\log\hat{\mathbb{P}}_{\mathrm{ML},w}(x_1^n) = \sum_{a\in A} N_n^k(w,a)\log\hat{p}_n^k(a|w)$$

e, para toda árvore $\tau'\in\mathcal{T}^k(x_1^n)$

$$\mathrm{BIC}_{\tau'}(x_1^n) = -\sum_{w\in\tau'}\log\hat{\mathbb{P}}_{\mathrm{ML},w}(x_1^n) + |\tau'|f(n).$$

Desta forma,

$$\begin{aligned}\mathbb{P}\big[\,\mathrm{BIC}_{\tau'}(x_1^n) \;<\; \mathrm{BIC}_{\tau^k}(x_1^n),\, \tau^k\in\mathcal{T}^k(x_1^n)\big] \leq \\ \mathbb{P}\big[\sum_{w\in\tau^k}\log\hat{\mathbb{P}}_{\mathrm{ML},w}(x_1^n) - \sum_{w'\in\tau'}\log\hat{\mathbb{P}}_{\mathrm{ML},w'}(x_1^n) < (|\tau^k|-|\tau'|)f(n)\big].\end{aligned} \tag{5.11}$$

A expressão

$$\sum_{w\in\tau^k} \log \hat{\mathbb{P}}_{\mathrm{ML},w}(x_1^n) - \sum_{w'\in\tau'} \log \hat{\mathbb{P}}_{\mathrm{ML},w'}(x_1^n)$$

pode ser reescrita como

$$\sum_{\substack{w\in\tau^k \\ w\in \mathrm{Int}(\tau')}} [\log \hat{\mathbb{P}}_{\mathrm{ML},w}(x_1^n) - \sum_{uw\in\tau'} \log \hat{\mathbb{P}}_{\mathrm{ML},uw}(x_1^n)] + \sum_{\substack{w\in\tau' \\ w\in \mathrm{Int}(\tau^k)}} [\sum_{uw\in\tau^k} \log \hat{\mathbb{P}}_{\mathrm{ML},uw}(x_1^n) - \log \hat{\mathbb{P}}_{\mathrm{ML},w}(x_1^n)]. \quad (5.12)$$

A demonstração segue dos seguintes lemas.

Lema 5.2.4. *Se $\tau' \neq \tau^k$ e $\tau' \cap \mathrm{Int}(\tau^k) = \emptyset$ então*

$$\mathbb{P}\big[\,\mathrm{BIC}_{\tau'}(x_1^n) < \mathrm{BIC}_{\tau^k}(x_1^n),\, \tau^k \in \mathcal{T}^k(x_1^n)\big] \;\leq\; 3\, e^{\frac{1}{e}}\, |A|^{k+1} \exp\big[-\frac{f(n)\,\epsilon_{k+1}^2\, p_k\, C}{4|A|^{k+1}(k+2)}\big], \quad (5.13)$$

onde

$$C = \frac{1-\beta_0}{8e(1+\beta)}.$$

Demonstração. Usando que $\tau' \cap \mathrm{Int}(\tau^k) = \emptyset$ na expressão (5.12) temos que

$$\mathbb{P}\big[\,\mathrm{BIC}_{\tau'}(x_1^n) \;<\; \mathrm{BIC}_{\tau^k}(x_1^n),\, \tau^k \in \mathcal{T}^k(x_1^n)\big] \;\leq$$

$$\mathbb{P}\big[\sum_{\substack{w\in\tau^k \\ w\in \mathrm{Int}(\tau')}} [\,\log \hat{\mathbb{P}}_{\mathrm{ML},w}(x_1^n) - \sum_{uw\in\tau'} \log \hat{\mathbb{P}}_{\mathrm{ML},uw}(x_1^n)\,] < (|\tau^k| - |\tau'|) f(n)\big]. \quad (5.14)$$

Sabemos, pelo estimador de máxima verossimilhança das probabilidades de transição, que

$$\hat{\mathbb{P}}_{\mathrm{ML},w}(x_1^n) \;\geq\; \prod_{a\in A} p(a|w)^{N_n^k(w,a)}. \quad (5.15)$$

Logo, podemos limitar superiormente o lado direito da desigualdade (5.14) por

$$\begin{aligned}
&\mathbb{P}\big[\sum_{\substack{w\in\tau^k \\ w\in \mathrm{Int}(\tau')}} [\sum_{a\in A} N_n^k(w,a) \log p(a|w) - \sum_{uw\in\tau'} \log \hat{\mathbb{P}}_{\mathrm{ML},uw}(x_1^n)\,] < (|\tau^k| - |\tau'|) f(n)\big] \\
&= \mathbb{P}\big[\sum_{\substack{w\in\tau^k \\ w\in \mathrm{Int}(\tau')}} [\sum_{a\in A} \sum_{uw\in\tau'} N_n^k(uw,a) \log p(a|uw) - \sum_{uw\in\tau'} \log \hat{\mathbb{P}}_{\mathrm{ML},uw}(x_1^n)\,] < (|\tau^k| - |\tau'|) f(n)\big] \\
&= \mathbb{P}\big[\sum_{\substack{w\in\tau^k \\ w\in \mathrm{Int}(\tau')}} [\sum_{a\in A} \sum_{uw\in\tau'} N_n^k(uw,a) \log \frac{p(a|uw)}{\hat{p}_n^k(a|uw)}\,] < (|\tau^k| - |\tau'|) f(n)\big].
\end{aligned}$$

A primeira igualdade acima segue de substituir $N_n^k(w,a)$ pela expressão $\sum_{uw\in\tau'} N_n^k(uw,a)$ e o fato que $p(a|uw) = p(a|w)$ para todo $uw \in \tau'$, já que $|w| < k$ implica que $w \in \tau$. Observemos que

$$\begin{aligned}\sum_{a\in A}\sum_{uw\in\tau'} N_n^k(uw,a)\log\frac{p(a|uw)}{\hat{p}_n^k(a|uw)} &= \sum_{uw\in\tau'} N_n^k(uw,\cdot)\sum_{a\in A}\hat{p}_n^k(a|uw)\log\frac{p(a|uw)}{\hat{p}_n^k(a|uw)}\\ &= -\sum_{uw\in\tau'} N_n^k(uw,\cdot)\, D\big(\hat{p}_n^k(\cdot|uw)\,\|\,p(\cdot|uw)\big),\end{aligned}$$

onde D é a divergência entre as distribuições $\hat{p}_n^k(\cdot|uw)$ e $p(\cdot|uw)$, apresentada na Seção 1.2. É fácil provar que a divergência satisfaz

$$D(p\,\|\,q) \le \sum_{a\in\mathcal{A}}\frac{(p(a)-q(a))^2}{q(a)}\,, \tag{5.16}$$

para duas distribuições de probabilidade p e q quaisquer sobre A. Essa demonstração pode ser encontrada em Csiszár & Talata (2006, Lemma 6.3). Assim, usando (5.16) e dividindo por $n-k$ temos que

$$\begin{aligned}&\mathbb{P}\big[\sum_{\substack{w\in\tau^k\\ w\in\mathrm{Int}(\tau')}}\big[-\sum_{uw\in\tau'} N_n^k(uw,\cdot)\,D\big(\hat{p}_n^k(\cdot|uw)\,\|\,p(\cdot|uw)\big)\big] < (|\tau^k|-|\tau'|)f(n)\big]\\ &\le\ \mathbb{P}\big[\sum_{\substack{w\in\tau^k\\ w\in\mathrm{Int}(\tau')}}\big[-\sum_{uw\in\tau'}\frac{N_n^k(uw,\cdot)}{n-k}\sum_{a\in A}\frac{[\hat{p}_n^k(a|uw)-p(a|uw)]^2}{p(a|uw)}\big] < \frac{(|\tau^k|-|\tau'|)f(n)}{n-k}\big].\end{aligned}$$

Por ser $\tau' \neq \tau^k$ e $\tau' \cap \mathrm{Int}(\tau^k) = \emptyset$ temos que a árvore τ' é estritamente maior que a árvore τ^k, assim $|\tau^k| - |\tau'| \le -1$. Por outro lado, $N_n^k(uw,\cdot) \le n-k$ e $f(n) > 0$. Desta forma podemos limitar superiormente o lado direito da expressão anterior por

$$\sum_{\substack{w\in\tau^k\\ w\in\mathrm{Int}(\tau')}}\sum_{uw\in\tau'}\sum_{a\in A}\mathbb{P}\Big[\,|\hat{p}_n^k(a|uw)-p(a|uw)| > \sqrt{\frac{f(n)p(a|uw)}{(n-k)|A|^{k+1}}}\,\Big].$$

Assim, usando o Corolário 4.3.4 podemos limitar superiormente a última expressão por

$$3\,e^{\frac{1}{e}}\,|A|^{k+1}\exp\big[-\frac{f(n)\,\epsilon_{k+1}^2\,p_k\,C}{4|A|^{k+1}(k+2)}\big],$$

onde

$$C = \frac{1-\beta_0}{8e(1+\beta)}.$$

Desta forma obtemos (5.13). □

Lema 5.2.5. *Se $\tau' \neq \tau^k$ e $\tau' \cap \mathrm{Int}(\tau^k) \neq \emptyset$ então*

$$\mathbb{P}\big[\,\mathrm{BIC}_{\tau'}(x_1^n) < \mathrm{BIC}_{\tau^k}(x_1^n)\,, \tau^k \in \mathcal{T}^k(x_1^n)\big] \;\leq$$

$$13\, e^{\frac{1}{e}}\, |A|^{k+1}\, \exp\big[-(n-k)\, \frac{[\delta_k - \frac{|A|^k f(n)}{n-k}]^2 \epsilon_{k+1}^2 p_k^2 C}{144 |\log p_k|^2 |A|^{2(k+1)}(k+2)}\big], \quad (5.17)$$

onde

$$C = \frac{1-\beta_0}{8e(1+\beta)}.$$

Demonstração. Dividindo por $n-k$ em ambos lados da desigualdade em (5.11) temos que

$$\mathbb{P}\big[\,\mathrm{BIC}_{\tau'}(x_1^n) < \mathrm{BIC}_{\tau^k}(x_1^n)\,, \tau^k \in \mathcal{T}^k(x_1^n)\big] \;\leq$$

$$\mathbb{P}\big[\sum_{w\in\tau^k} \frac{1}{n-k} \log \hat{\mathbb{P}}_{\mathrm{ML},w}(x_1^n) - \sum_{w'\in\tau'} \frac{1}{n-k} \log \hat{\mathbb{P}}_{\mathrm{ML},w'}(x_1^n) \;<\; \frac{(|\tau^k| - |\tau'|) f(n)}{n-k}\big].$$

Usando a expressão (5.12) e subtraindo a soma

$$\sum_{\substack{w\in\tau' \\ w\in\mathrm{Int}(\tau^k)}} \delta_k(w),$$

onde $\delta_k(w)$ está dado por (5.1), podemos limitar superiormente o lado direito da última desigualdade por

$$\mathbb{P}\big[\sum_{\substack{w\in\tau^k \\ w\in\mathrm{Int}(\tau')}} [\frac{1}{n-k} \log \hat{\mathbb{P}}_{\mathrm{ML},w}(x_1^n) - \sum_{uw\in\tau'} \frac{1}{n-k} \log \hat{\mathbb{P}}_{\mathrm{ML},uw}(x_1^n)] \;+ \quad (5.18)$$

$$\sum_{\substack{w\in\tau' \\ w\in\mathrm{Int}(\tau^k)}} [\sum_{uw\in\tau^k} \frac{1}{n-k} \log \hat{\mathbb{P}}_{\mathrm{ML},uw}(x_1^n) - \frac{1}{n-k} \log \hat{\mathbb{P}}_{\mathrm{ML},w}(x_1^n) - \delta_k(w)] < (|\tau^k| - |\tau'|) \frac{f(n)}{n-k} - \delta_k\big].$$

Para cada $w \in \tau^k \cap \mathrm{Int}(\tau')$, o termo

$$\frac{1}{n-k} \log \hat{\mathbb{P}}_{\mathrm{ML},w}(x_1^n) - \sum_{uw\in\tau'} \frac{1}{n-k} \log \hat{\mathbb{P}}_{\mathrm{ML},uw}(x_1^n)$$

pode ser limitado inferiormente, usando a desigualdade (5.15) como no caso do Lema 5.2.4, pela expressão

$$-\sum_{uw\in\tau'} \frac{N_n^k(uw,\cdot)}{n-k} \sum_{a\in A} \frac{[\hat{p}_n^k(a|uw) - p(a|uw)]^2}{p(a|uw)}.$$

Por outro lado, para cada $w \in \tau' \cap \mathrm{Int}(\tau^k)$, podemos reescrever o termo

$$\sum_{uw \in \tau^k} \frac{1}{n-k} \log \hat{\mathbb{P}}_{\mathrm{ML},uw}(x_1^n) - \frac{1}{n-k} \log \hat{\mathbb{P}}_{\mathrm{ML},w}(x_1^n) - \delta_k(w)$$

como

$$\sum_{uw \in \tau^k} \sum_{a \in A} p(uwa) \log \frac{p(a|w)}{p(a|uw)} - \frac{N_n^k(uwa)}{n-k} \log \frac{\hat{p}_n^k(a|w)}{\hat{p}_n^k(a|uw)},$$

usando que $p(wa) = \sum_{uw \in \tau^k} p(uwa)$ e $N_n^k(w,\cdot) = \sum_{uw \in \tau^k} \sum_{a \in A} N_n^k(uwa)$. Desta forma, podemos limitar superiormente a expressão (5.18) por

$$\mathbb{P}\big[- \sum_{\substack{w \in \tau^k \\ w \in \mathrm{Int}(\tau')}} \sum_{uw \in \tau'} \sum_{a \in A} \frac{N_n^k(uw,\cdot)}{n-k} \frac{[\hat{p}_n^k(a|uw) - p(a|uw)]^2}{p(a|uw)} \; + \; \sum_{\substack{w \in \tau' \\ w \in \mathrm{Int}(\tau^k)}} \sum_{uw \in \tau^k} \sum_{a \in A} p(uwa) \log \frac{p(a|w)}{p(a|uw)} - \frac{N_n^k(uwa)}{n-k} \log \frac{\hat{p}_n^k(a|w)}{\hat{p}_n^k(a|uw)} < (|\tau^k| - |\tau'|) \frac{f(n)}{n-k} - \delta_k \big].$$

No total, a soma do lado esquerdo dentro da probabilidade tem como máximo $|A|^{k+1}$ termos. Portanto, esta probabilidade é menor ou igual a

$$\sum_{\substack{w \in \tau^k \\ w \in \mathrm{Int}(\tau')}} \sum_{uw \in \tau'} \sum_{a \in A} \mathbb{P}\big[- \frac{N_n^k(uw,\cdot)}{n-k} \frac{[\hat{p}_n^k(a|uw) - p(a|uw)]^2}{p(a|uw)} < \frac{(|\tau^k| - |\tau'|)\frac{f(n)}{n-k} - \delta_k}{|A|^{k+1}} \big] \; + \tag{5.19}$$

$$\sum_{\substack{w \in \tau' \\ w \in \mathrm{Int}(\tau^k)}} \sum_{uw \in \tau^k} \sum_{a \in A} \mathbb{P}\big[p(uwa) \log \frac{p(a|w)}{p(a|uw)} - \frac{N_n^k(uwa)}{n-k} \log \frac{\hat{p}_n^k(a|w)}{\hat{p}_n^k(a|uw)} < \frac{(|\tau^k| - |\tau'|)\frac{f(n)}{n-k} - \delta_k}{|A|^{k+1}} \big].$$

Temos que $|\tau^k| - |\tau'| < |A|^k$. Logo, para todo n tal que

$$\frac{|A|^k f(n)}{n-k} \; < \; \delta_k$$

obtemos

$$\frac{(|\tau^k| - |\tau'|)\frac{f(n)}{n-k} - \delta_k}{|A|^{k+1}} \; < \; \frac{\frac{|A|^k f(n)}{(n-k)} - \delta_k}{|A|^{k+1}} \; < \; 0\,.$$

Usando que $\frac{N_n^k(uw,a)}{n-k} \leq 1$ e o Corolário 4.3.4 como no caso do Lema 5.2.4 concluímos que

$$\begin{aligned}
\mathbb{P}\big[-\frac{N_n^k(uw,\cdot)}{n-k}\, \frac{[\hat{p}_n^k(a|uw) - p(a|uw)]^2}{p(a|uw)} < \frac{(|\tau^k| - |\tau'|)\frac{f(n)}{n-k} - \delta_k}{|A|^{k+1}} \big] \\
\leq\ \mathbb{P}\big[\, |\hat{p}_n^k(a|uw) - p(a|uw)| > \sqrt{[\delta_k - \frac{|A|^k f(n)}{n-k}]\, p(a|uw)\, |A|^{-(k+1)}}\, \big] \\
\leq\ 3\, e^{\frac{1}{e}}\, \exp\big[-(n-k)\, \frac{[\delta_k - \frac{|A|^k f(n)}{n-k}] p(uwa)^2\, p(a|uw)\, C}{4|A|^{k+1}(|uw|+2)}\big], \qquad (5.20)
\end{aligned}$$

onde

$$C = \frac{1-\beta_0}{8e(1+\beta)}.$$

Por outro lado, somando e subtraindo o termo

$$\frac{N_n(uw,a)}{n-k} \log \frac{p(a|w)}{p(a|uw)}$$

da expressão

$$p(uwa) \log \frac{p(a|w)}{p(a|uw)} - \frac{N_n^k(uwa)}{n-k} \log \frac{\hat{p}_n^k(a|w)}{\hat{p}_n^k(a|uw)}$$

e usando novamente que $|\tau^k| - |\tau'| < |A|^k$ obtemos

$$\begin{aligned}
\mathbb{P}\big[\, p(uwa) \log \frac{p(a|w)}{p(a|uw)} - \frac{N_n^k(uwa)}{n-k} \log \frac{\hat{p}_n^k(a|w)}{\hat{p}_n^k(a|uw)} < \frac{(|\tau^k| - |\tau'|)\frac{f(n)}{n-k} - \delta_k}{|A|^{k+1}} \big] \\
\leq\ \mathbb{P}\big[\log \frac{p(a|w)}{p(a|uw)} \big[p(uwa) - \frac{N_n^k(uw,a)}{n-k} \big] \ + \qquad (5.21) \\
\frac{N_n^k(uw,a)}{n-k} \big[\log \frac{p_n(a|w)}{\hat{p}_n^k(a|w)} + \log \frac{\hat{p}_n(a|uw)}{p(a|uw)} \big] < \frac{\frac{|A|^k f(n)}{n-k} - \delta_k}{|A|^{k+1}} \big].
\end{aligned}$$

Usando que $\frac{N_n^k(uw,a)}{n-k} \leq 1$, $|\log \frac{p(a|w)}{p(a|uw)}| \leq 2|\log p_k|$, e para todo n tal que

$$\frac{|A|^k f(n)}{n-k} \ < \ \delta_k,$$

podemos limitar superiormente o termo direito da expressão (5.21) por

$$\begin{aligned}
\mathbb{P}\big[\, |p(uwa) - \frac{N_n^k(uw,a)}{n-k}| > \frac{\delta_k - \frac{|A|^k f(n)}{n-k}}{6|\log p_k||A|^{k+1}} \big] \ + \ \mathbb{P}\big[\, \big|\log \frac{p_n(a|w)}{\hat{p}_n^k(a|w)}\big| > \frac{\delta_k - \frac{|A|^k f(n)}{n-k}}{3|A|^{k+1}} \big] \\
+ \ \mathbb{P}\big[\, \big|\log \frac{\hat{p}_n^k(a|uw)}{p(a|uw)}\big| > \frac{\delta_k - \frac{|A|^k f(n)}{n-k}}{3|A|^{k+1}} \big].
\end{aligned}$$

Aplicando o Teorema 4.3.1 do capítulo anterior temos que

$$\begin{aligned}\mathbb{P}\big[\,|p(uwa)-\frac{N_n^k(uw,a)}{n-k}| > \frac{\delta_k - \frac{|A|^k f(n)}{n-k}}{6|\log p_k||A|^{k+1}}\big] \\ \leq\ e^{\frac{1}{e}} \exp\big[-(n-k)\ \frac{[\delta_k - \frac{|A|^k f(n)}{n-k}]^2 C}{36|\log p_k|^2|A|^{2(k+1)}(k+2)}\big],\end{aligned} \tag{5.22}$$

onde, como antes,

$$C = \frac{1-\beta_0}{8e(1+\beta)}.$$

Por outro lado, usando o Lema 5.2.2 obtemos

$$\begin{aligned}\mathbb{P}\big[\,\big|\log\frac{p_n(a|w)}{\hat{p}_n^k(a|w)}\big| > \frac{\delta_k - \frac{|A|^k f(n)}{n-k}}{3|A|^{k+1}}\big]\ +\ \mathbb{P}\big[\,\big|\log\frac{\hat{p}_n^k(a|uw)}{p(a|uw)}\big| > \frac{\delta_k - \frac{|A|^k f(n)}{n-k}}{3|A|^{k+1}}\big] \\ \leq\ 12\, e^{\frac{1}{e}}\ \exp\big[-(n-k)\ \frac{[\delta_k - \frac{|A^k| f(n)}{n-k}]^2 \epsilon_{k+1}^2 p_k^2 C}{144|A|^{2(k+1)}(k+2)}\big].\end{aligned} \tag{5.23}$$

Desta forma, juntando (5.22) e (5.23) podemos limitar superiormente (5.21), obtendo

$$\begin{aligned}\mathbb{P}\big[\,p(uwa)\log\frac{p(a|w)}{p(a|uw)} - \frac{N_n^k(uwa)}{n-k}\log\frac{\hat{p}_n^k(a|w)}{\hat{p}_n^k(a|uw)} < \frac{(|\tau^k| - |\tau'|)\frac{f(n)}{n} - \delta_k}{|A|^{k+1}}\,\big] \\ \leq\ 13\, e^{\frac{1}{e}} \exp\big[-(n-k)\ \frac{[\delta_k - \frac{|A|^k f(n)}{n-k}]^2 \epsilon_{k+1}^2 p_k^2 C}{144|\log p_k|^2|A|^{2(k+1)}(k+2)}\big].\end{aligned}$$

Assim, juntando a última desigualdade com a desigualdade em (5.20) podemos limitar superiormente (5.19) com

$$13\, e^{\frac{1}{e}}\ |A|^{k+1}\ \exp\big[-(n-k)\ \frac{[\delta_k - \frac{|A|^k f(n)}{n-k}]^2 \epsilon_{k+1}^2 p_k^2 C}{144|\log p_k|^2|A|^{2(k+1)}(k+2)}\big].$$

Desta forma concluímos a demonstração do Lema 5.2.5. □

Para finalizar a demonstração do Teorema 5.2.3 basta utilizar os resultados dos Lemas 5.2.4 e 5.2.5 para limitar superiormente cada termo da soma (5.10). □

Uma primeira conseqüência do Teorema 5.2.3 é o seguinte resultado de convergência em probabilidade para o estimador $\hat{\tau}_{\mathrm{BIC}}^k(x_1^n)$, para qualquer termo de penalização $f(n)$ satisfazendo as condições em (5.7).

Corolário 5.2.6. *Seja $f(n)$ qualquer função positiva tal que $f(n) \to +\infty$, quando $n \to +\infty$, e $n^{-1}f(n) \to 0$, quando $n \to +\infty$. Logo, para todo inteiro k e para quase toda realização infinita $x_1, x_2, \ldots,$ da árvore probabilística de sufixos (τ, p) temos que*

$$\lim_{n\to+\infty} \mathbb{P}\big[\, \hat{\tau}^k_{\mathrm{BIC}}(x_1^n) \neq \tau^k \,\big] = 0.$$

Demonstração. Direta, a partir do Teorema 5.2.3. □

Uma outra conseqüência do Teorema 5.2.3 é o seguinte resultado de consistência forte, no caso $f(n) \sim n^\alpha$, com $0 < \alpha < 1$.

Corolário 5.2.7. *Seja $f(n) = cn^\alpha$, com c uma constante positiva e α tal que $0 < \alpha < 1$. Logo, para qualquer inteiro k e qualquer $\kappa > 0$ existe um $\bar{n}$ tal que, para todo $n \geq \bar{n}$ temos que*

1. $\hat{\tau}^k_{\mathrm{BIC}}(x_1^n) = \tau^k$;
2. $\max\limits_{w\in\tau^k,\, a\in A} |\hat{p}^k_n(a|w) - p(a|w)| < \kappa$.

Demonstração. Análoga à demonstração do Corolário 4.4.4, substituindo $\hat{\tau}^k_n$ por $\hat{\tau}^k_{\mathrm{BIC}}(x_1^n)$ e utilizando a desigualdade do Teorema 5.2.3. □

Conclusão

Nesta tese estudamos um novo tipo de modelo de cadeias estocásticas; a saber, as cadeias estocásticas parcimoniosas. Para estimar os parâmetros dessas cadeias introduzimos um novo algoritmo, chamado de SPST. Esse algoritmo está relacionado com o algoritmo PST, utilizado para estimar os parâmetros de uma cadeia estocástica de memória variável.

O algoritmo SPST foi utilizado para estudar dois importantes problemas da genômica. O primeiro problema é o da classificação de proteínas em famílias, através da identificação de padrões na sua cadeia de aminoácidos. Nesse caso, mostramos que o algoritmo SPST, e a sua variação F-SPST, conseguem classificar corretamente mais seqüências do que o algoritmo PST. Também mostramos que alguns nós das árvores de contextos estimadas com o algoritmo SPST se correspondem exatamente com grupos de aminoácidos, obtidos a partir de suas propriedades físico-químicas. O segundo problema estudado através da modelagem com cadeias estocásticas parcimoniosas foi a análise filogenética de seqüências relacionadas. Nesse caso, calculamos uma distância entre todos os pares de árvores de contextos associadas com as seqüências e obtivemos dessa forma uma matriz de distâncias. Com essa matriz construimos uma árvore filogenética, através da utilização de pacotes específicos para esse fim. Esse método foi aplicado a dois conjuntos de dados. No caso do primeiro conjunto, integrado por seqüências da família FGF em humanos, foi possível reconstruir os subgrupos de seqüências já obtidos com outros métodos filogenéticos e publicados na literatura específica. No caso do segundo conjunto de dados, integrado por seqüências pertencentes à família das globinas, foram comparadas as árvores filogenéticas construídas com a matriz de distâncias baseada nas árvores de contextos e a matriz obtida através da distância PAM em seqüências alinhadas. Nesse caso foi possível concluir que essas duas abordagens identificam basicamente as mesmas relações filogenéticas entre as seqüências. Uma vantagem da distância baseada em árvores de contextos é que não é necessário obter preliminarmente um alinhamento múltiplo das seqüências.

Por outro lado, demonstramos alguns resultados teóricos relacionados com a estimação das árvores de contextos de cadeias estocásticas de memória variável. Em primeiro lugar, generalizamos um resultado prévio de velocidade exponencial de convergência do algoritmo PST, no caso de cadeias com memória não necessariamente limitada. Em segundo lugar,

obtivemos limitantes superiores da velocidade de convergência de uma abordagem alternativa para a estimação das árvores de contextos. Essa abordagem está baseada na escolha da árvore que maximiza a verossimilhança, penalizada com um termo que depende do tamanho da árvore e do tamanho da amostra. O caso mais conhecido desse tipo de abordagem é chamado de Critério da Informação Bayesiana, ou Critério de Schwarz. Nesse caso, a penalização está dada pela quantidade de parâmetros que devem ser estimados, multiplicado pelo logaritmo do tamanho da amostra. Nossos resultados valem para qualquer termo de penalização, sob certas condições de crescimento.

Referências

Altschul, S., Madden, T., Schaffer, A., Zhang, J., Zhang, Z., Miller, W. & Lipman, D. (1997). Gapped BLAST and PSI-BLAST: a new generation of protein database search programs, *Nucl. Acids Res.* **25**(17): 3389–3402.

Apostolico, A. & Bejerano, G. (2000). Optimal amnesic probabilistic automata or how to learn and classify proteins in linear time and space, *Proc. Int'l Conf. Computational Molecular Biology*, Vol. 4, pp. 25–32.

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, *Nucl. Acids Res.* **32**(90001): D138–141.

Bejerano, G. (2003). *Automata learning and stochastic modeling for biosequence analysis*, PhD thesis, Hebrew University.

Bejerano, G., Seldin, Y., Margalit, H. & Tishby, N. (2001). Markovian domain fingerprinting: statistical segmentation of protein sequences, *Bioinformatics* **17**(10): 927–934.

Bejerano, G. & Yona, G. (2001). Variations on probabilistic suffix trees: statistical modeling and prediction of protein families, *Bioinformatics* **17**(1): 23–43.

Billingsley, P. (1961). *Statistical inference for markov processes*, The University of Chicago Press.

Boeckmann, B., Bairoch, A., Apweiler, R., Blatter, M.-C., Estreicher, A., Gasteiger, E., Martin, M. J., Michoud, K., O'Donovan, C., Phan, I., Pilbout, S. & Schneider, M. (2004). The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003, *Nucleic Acids Res.* **31**(1): 365–370.

Bourguignon, P.-Y. & Robelin, D. (2004). Modèles de Markov parcimonieux: sélection de modèle et estimation. JOBIM 2004, Montreal.

Bressaud, X., Fernández, R. & Galves, A. (1999). Decay of correlations for non Hölderian dynamics. a coupling approach, *Elect. J. Prob.* **4**: 161–173.

Bülhmann, P. & Wyner, A. J. (1999). Variable length Markov chains, *Annals of Stat.* **27**: 480–513.

Csiszár, I. & Shields, P. C. (2000). The consistency of the BIC Markov order estimator, *Ann. Statist.* **28**(6): 1601–1619.

Csiszár, I. & Talata, Z. (2006). Context tree estimation for not necessarily finite memory processes, via bic and mdl, *Information Theory, IEEE Transactions on* **52**(3): 1007–1016.

Dayhoff, M. O., Schwartz, R. M. & Orcutt, B. C. (1978). *A model for evolutionary change in proteins*, Vol. 5 of *Margaret O. Dayhoff, editor, Atlas of Protein Sequence and Structure*, National Biochemical Research Foundation, Washington DC, pp. 345–352.

Dedecker, J. & Doukhan, P. (2003). A new covariance inequality and applications, *Stochastic Process. Appl.* **106**(1): 63–80.

Dedecker, J. & Prieur, C. (2005). New dependence coefficients. examples and applications to statistics, *Prob. Theory and Relat. Fields* **132**: 203–236.

Duarte, D., Galves, A. & Garcia, N. (2006). Markov approximation and consistent estimation of unbounded probabilistic suffix trees, *Bull. Braz. Math. Soc.* p. Aceito.

Eskin, E., Grundy, W. N. & Singer, Y. (2000). Protein family classification using sparse markov transducers, *Proc. Int'l Conf. Intell. Syst. Mol. Biol.*, Vol. 8, pp. 134–145.

Felsenstein, J. (2004). Phylip (phylogeny inference package) version 3.6. Distributed by the author. Department of Genome Sciences, University of Washington, Seattle.

Fernández, R. & Galves, A. (2002). Markov approximations of chains of infinite order, *Bull. Braz. Math. Soc.* **33**(3): 295–306.

Ferrari, F. & Wyner, A. (2003). Estimation of general stationary processes by variable length Markov chains, *Scand. J. Statist.* **30**(3): 459–480.

Galves, A., Maume-Deschamps, V. & Schmitt, B. (2006). Exponential inequalities for VLMC empirical trees. Submetido.

Guttorp, P. (1995). *Stochastic modeling of scientific data*, Chapman & Hall.

Itoh, N. & Ornitz, D. (2004). Evolution of the Fgf and Fgfr gene families, *TRENDS in Genetics* **20**(11): 563–569.

Karp, R. (2002). Mathematical challenges from genomics and molecular biology, *Notices Amer. Math. Soc.* **49**(5): 544–553.

Leonardi, F. (2005). Probabilistic tree based phylogenetics of protein families. Cartaz em X-meeting 2005. Caxambu, MG, Brasil. Disponível em `http://www.ime.usp.br/~leonardi/articles/x-meeting.pdf`.

Leonardi, F. (2006). A generalization of the PST algorithm: modeling the sparse nature of protein sequences, *Bioinformatics* **22**(11): 1302–1307.

Leonardi, F. & Galves, A. (2005). Sequence motif identification and protein family classification using probabilistic trees, *Advances in Bioinformatics and Computational Biology. Proc. BSB 2005.*, Vol. LNBI 3594, pp. 190–193.

Leonardi, F., Matioli, S., Armelin, H. & Galves, A. (2006). A distance between contexts trees for alignment-free sequence comparison. Manuscrito. Disponível em `http://www.ime.usp.br/~leonardi/articles/phyl-spst.pdf`.

Murzin, A. G., Brenner, S. E., Hubbard, T. & Chothia, C. (1995). SCOP: a structural classification of proteins database for the investigation of sequences and structures, *J. Mol. Biol.* **247**: 536–540.

Pearson, W. (1995). Comparison of methods for searching protein sequence databases, *Protein Sci.* **4**: 1145–1160.

Pearson, W. (2000). Flexible sequence similarity searching with the FASTA3 program package, *Methods Mol. Biol.* **132**: 185–219.

Rabiner, L. (1986). Tutorial on hidden Markov models and selected applications in speech recognition, *Proc. IEEE* **77**: 257–286.

Rissanen, J. (1983). A universal data compression system, *IEEE Trans. Inform. Theory* **29**(5): 656–664.

Ron, D., Singer, Y. & Tishby, N. (1996). The power of amnesia: Learning probabilistic automata with variable memory length, *Machine Learning* **25**(2-3): 117–149.

Rust, A. G., Mongin, E. & Birney, E. (2002). Genome annotation techniques: new approaches and challenges, *Drug Discovery Today* **7**(11): 70–76.

Schwarz, G. (1978). Estimating the dimension of a model, *The Annals of Statistics* **6**: 461–464.

Simovici, D. & Szymon, J. (2002). An axiomatization of partition entropy, *IEEE Transactions on Information Theory* **48**(7): 2138–2142.

Simovici, D. & Szymon, J. (2006). A new metric splitting criterion for decision trees, *Journal of Parallel, Emerging and Distributed Computing* **21**(4): 239–256.

Smith, T. & Waterman, M. (1981). Identification of common molecular subsequences, *J. Mol. Biol.* **147**(1): 195–197.

Stroock, D. (1993). *Probability theory, an analytic view*, Cambridge University Press.

Sujatha, S., Balaji, S. & Srinivasan, N. (2001). PALI: a database of alignments and phylogeny of homologous protein structures, *Bioinformatics* **17**(4): 375–376.

Taylor, W. (1986). The classification of amino acid conservation, *Journal of Theoretical Biology* **119**(2): 205–218.

Thompson, J., Higgins, D. & Gibson, T. (1994). CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, positions-specific gap penalties and weight matrix choice, *Nucleic Acids Research* **22**: 4673–4680.