ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO
Avenida Professor Luciano Gualberto, travessa 3 nº 158 CEP 05508-900 São Paulo SP
Telefone: (011) 818-5583 Fax (011) 818-5294
Departamento de Engenharia de Computação e Sistemas Digitais

PCS 2428 E PCS 2059

INTELIGÊNCIA ARTIFICIAL

RACIOCÍNIO PROBABILÍSTICO AO LONGO DO TEMPO

ALUNOS: **GRUPO 8**

ALEXANDRE SHIROMA **5174737**
CELSO DE ALMEIDA SAAD **5123393**
MÁRCIO YUDI SATO **5179979**

CONTEÚDO

1 INTRODUÇÃO

Agentes em ambientes incertos devem ter a capacidade de manter o estado atual do ambiente, tal como agentes lógicos. Esta tarefa é dificultada pela incerteza sobre como o ambiente muda ao longo do tempo. No melhor dos casos, o agente poderá obter um valor probabilístico acerca da situação atual. O mundo muda, é necessário controlar e prever essas mudanças.

2 TEMPO E INCERTEZA

Exemplos: Tratar um paciente diabético (onde o nível de sangue no sangue, e a quantidade de insulina com que é feito tratamento, muda a cada instante de tempo), monitoramento do trânsito (onde o volume de veículos muda com o passar do tempo).

Processo de mudança visto como uma série de fotografias (ou *time slices*), onde cada uma delas descreve o estado do mundo em um determinado instante de tempo.

2.1 ESTADOS E OBSERVAÇÕES

Cada *time slice* possui um conjunto de variáveis aleatórias:

- X_t – conjunto de variáveis de estado (não-observadas) no instante t.
- E_t – conjunto de variáveis de evidência (observadas) no instante t.

2.1.1 EXEMPLO

Um guarda da segurança está no interior de um edifício e quer saber se está chovendo, mas o seu único acesso ao mundo exterior acontece cada manhã, quando ele vê o diretor entrar com ou sem guarda-chuva.

Por cada dia t:

– $\mathbf{E_t}$ contém a variável de evidência U_t (se o guarda-chuva aparece)

– $\mathbf{X_t}$ contém a variável de estado R_t (se está chovendo)

A notação $\mathbf{X_{a:b}}$ denota o conjunto de variáveis de $\mathbf{X_a}$ a $\mathbf{X_b}$

3 PROCESSOS ESTACIONÁRIOS E A SUPOSIÇÃO DE MARKOV

Podemos colocar as variáveis pela ordem temporal natural e fazer questões sobre a independência condicional dos predecessores, dado algum conjunto de pais.

O conjunto de variáveis é ilimitado porque inclui as variáveis de estado e de evidência para cada *time-slice*. Isso cria dois tipos de problemas:

- Podemos ter que especificar um número ilimitado de tabelas de probabilidades condicionadas para cada variável em cada *time-slice*.
- Cada *slice* pode envolver um número ilimitado de pais.

Os processos são *estacionários.* Não confundir com *estáticos*. Um processo estático é aquele que é fixo, que não muda com o passar do tempo, se não houver nenhuma ação. Já um processo estacionário é aquele que muda com o passar do tempo, mas que segue uma regra fixa. Neste caso é possível prever o seu comportamento. Com isso é possível contornar o primeiro problema.

Para o segundo problema, levaremos em conta a Suposição de Markov. A **Suposição de Markov** nos diz que o estado atual depende apenas de um número finito (n) de estados anteriores. Ou seja, $\mathbf{X_t}$ depende de um nº limitado de subconjuntos de $\mathbf{X_{0:t-1}}$

Os processos que satisfazem esta suposição são definidos como processos de Markov. Existem várias ordens do tipo n, onde o n representa o número de estados da qual irá depender o estado atual.

Um **Modelo de transição** é a lei que descreve como o estado muda ao longo do tempo:

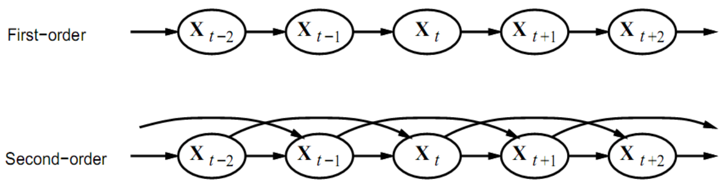
$\mathbf{P(X_t|X_{0:t-1})} = \mathbf{P(X_t|X_\alpha)}$ onde $\alpha \subseteq \{1\ldots t-1\}$

- Processo de Markov de 1ª ordem:

$\mathbf{P(X_t|X_{0:t-1})} = \mathbf{P(X_t|X_{t-1})}$

- Processo de Markov de 2ª ordem:

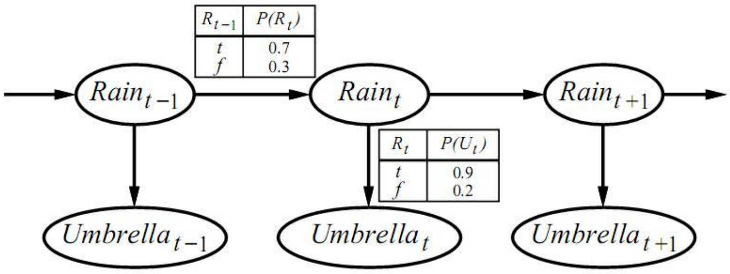
$\mathbf{P(X_t|X_{0:t-1})} = \mathbf{P(X_t|X_{t-2},X_{t-1})}$



É assumido que as variáveis de evidência no instante **t** dependem apenas do estado atual.

O **Modelo Sensor** é a lei que descreve como as variáveis de evidência (sensores) vão ser afetadas pelo estado atual do mundo:

$$\mathbf{P(E_t | X_{0:t}, E_{0:t-1}) = P(E_t | X_t)}$$



- O modelo de transição: $P(Rain_t | Rain_{t-1})$.
- O modelo sensor: $P(Umbrella_t | Rain_t)$.
- Processo de Markov de ordem um.

Aproximação de ordem um nem sempre é a melhor para o mundo real. Duas tentativas de resolução:

1. Aumentar a ordem do processo Markov.
   - **Exemplo:** modelo de ordem 2 adicionando $Rain_{t-2}$ como pai de $Rain_t$ → pode dar previsões mais precisas.
2. Aumentar o conjunto de variáveis de estado.
   - **Exemplo:** adicionar *Temperatura*$_t$, *Umidade*$_t$ e *Pressão*$_t$ → permite o uso de um modelo físico para as condições de chuva.

4 INFERÊNCIA EM MODELOS TEMPORAIS

Uma vez obtida a estrutura de um modelo temporal, as seguintes inferências podem ser realizadas:

- ***Filtragem ou monitoramento***: computar o estado de crença atual, dada toda a evidência até ao momento, ou seja, computar $\mathbf{P(X_t|e_{1:t})}$.
  Exemplo: Qual é a probabilidade de chuva hoje, dadas todas as observações guarda-chuvas até hoje?

- ***Previsão***: computar distribuição posterior sobre algum estado futuro, dada toda a evidência até ao momento, ou seja, computar $\mathbf{P(X_{t+k}|e_{1:t})}$ para k>0.
  Exemplo: Qual é a probabilidade de que irá chover daqui a três dias, dadas todas as observações de guarda-chuvas até hoje?

- ***Suavização***: computar a distribuição posterior sobre algum estado passado, dada toda a evidência até ao momento, ou seja, computar $\mathbf{P(X_k|e_{1:t})}$ para $0 \le k < t$.
  Exemplo: Qual é a probabilidade de ter chovido ontem, dadas todas as observações de guarda-chuvas até hoje?

- ***Explicação mais provável***: encontrar a seqüência de estados que mais provavelmente geraram a seqüência de observações, ou seja, computar $\mathbf{argmax}_{x1:t}\ \mathbf{P(x_{1:t}|e_{1:t})}$
  Exemplo: se o guarda-chuva apareceu nos primeiros três dias, mas não no quarto, então a explicação mais provável é que choveu nos primeiros três dias e não choveu no quarto dia.

4.1 FILTRAGEM OU MONITORAMENTO

O objetivo é obter uma estimação recursiva. Dado o resultado de filtragem até o instante *t*, pode-se computar o resultado para *t+1* utilizando a evidência e_{t+1}. Assim, para alguma função *f*:

$$P(X_{t+1}|e_{1:t+1}) = f(e_{t+1}, P(X_t|e_{1:t}))$$

Este processo é normalmente chamado de *estimação recursiva*. O estado atual da distribuição é projetado de *t* para *t+1*, depois é atualizado utilizando uma nova evidência e_{t+1}:

$$\begin{aligned} P(X_{t+1}|e_{1:t+1}) &= P(X_{t+1}|e_{1:t}, e_{t+1}) \text{ (dividindo pela evidência)} \\ &= \alpha\, P(e_{t+1}|X_{t+1}, e_{1:t})\, P(X_{t+1}|e_{1:t}) \text{ (usando a Regra de Bayes)} \\ &= \alpha\, P(e_{t+1}|X_{t+1}) P(X_{t+1}|e_{1:t}) \text{ (evidência de Markov)} \end{aligned}$$

Na equação acima, α é a constante de normalização. O termo $P(X_{t+1}|e_{1:t})$ representa a predição do estado seguinte e o termo $P(e_{t+1}|X_{t+1})$, que pode ser obtido diretamente do modelo sensor. Condicionando a variável $X_{t:}$

$$P(X_{t+1}|e_{1:t+1}) \quad = \quad \alpha\, P(e_{t+1}|X_{t+1}) \sum_{xt} P(X_{t+1}|x_t, e_{1:t})\, P(x_t|e_{1:t})$$

$$= \quad \alpha\, P(e_{t+1}|X_{t+1}) \sum_{xt} P(X_{t+1}|x_t)\, P(x_t|e_{1:t})$$

Dentro da somatória, o primeiro termo é o modelo de transição e o segundo termo é o estado atual da distribuição. Pode-se notar a recursividade na equação, pois há uma relação direta entre $P(X_{t+1}|e_{1:t+1})$ e $P(x_t|e_{1:t})$.

O termo $P(x_t|e_{1:t})$ pode ser visto como uma mensagem $f_{1:t}$ que é propagada para a frente ao longo da seqüência, modificada por cada transição e atualizada a cada nova observação. Assim:

$$f_{1:t+1} = \alpha \text{ FORWARD } (f_{1:t}, e_{t+1}) ,$$

onde FORWARD implementa a atualização descrita na equação.

4.1.1 Exemplo

Utilizando o mesmo exemplo do guarda-chuva, vamos supor que o guarda de segurança possui uma crença inicial sobre o fato de ter chovido no dia 0 e que esta crença seja de <0.5, 0.5>.

Aplicando o modelo de transição para o dia 1:

$$P(R_1) = \sum_{r0} P(R_1|r_0)\, P(r_0) = \langle 0.7,0.3\rangle \times 0.5 + \langle 0.3,0.7\rangle \times 0.5 = \langle 0.5,0.5\rangle$$

Neste dia, o guarda-chuva aparece (U_1=true). Atualizando com esta evidência:

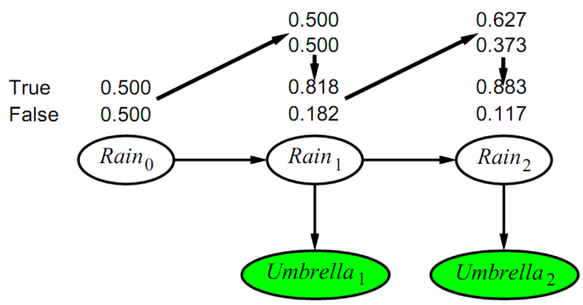
$$P(R_1|u_1) = \alpha\, P(u_1|R_1)\, P(R_1) = \alpha \langle 0.9,0.2\rangle\langle 0.5,0.5\rangle = \langle 0.818,0.182\rangle$$

Aplicando o modelo de transição para o dia 2:

$$P(R_2|u_1) = \sum_{r1} P(R_2|r_1)\, P(r_1|u_1) = \langle 0.7,0.3\rangle \times 0.818 + \langle 0.3,0.7\rangle \times 0.182 = \langle 0.627,0.373\rangle$$

No dia 2, o guarda-chuva aparece novamente (U_2=true). Atualizando com esta evidência:

$$P(R_2|u_{1,}\, u_2) = \alpha\, P(u_2|R_2)\, P(R_1|u_1) = \alpha \langle 0.9,0.2\rangle\langle 0.627,0.373\rangle = \langle 0.883,0.117\rangle$$



4.2 Previsão

Semelhante a filtragem, mas sem a adição de uma nova evidência.

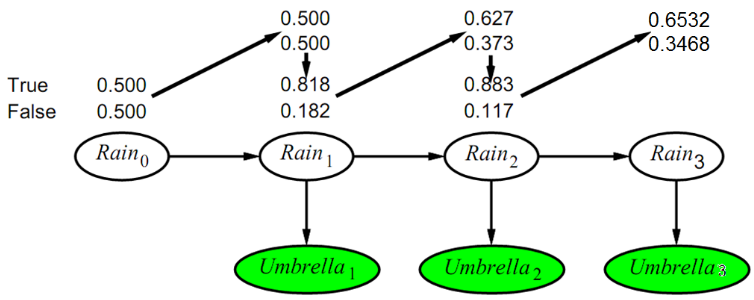
$$P(X_{t+k+1}|\ e_{1:t}) = \Sigma_{xt+k} P(X_{t+k+1}|x_{t+k})\, P(x_{t+k}|e_{1:t})$$

Esta computação envolve apenas o modelo de transição e não o modelo sensor.

4.2.1 EXEMPLO

Supondo o mesmo problema do guarda-chuva já apresentado. Vamos supor que o guarda de segurança queira saber se no dia 3 irá chover. Assim, aplicando o modelo de transição para o dia 3:

$$P(R_3|u_2,u_1) = \sum_{r2} P(R_3|r_2)\, P(r_2|u_2,u_1) = \langle 0.7,0.3\rangle \times 0.883 + \langle 0.3,0.7\rangle \times 0.117 = \langle 0.6532,0.3468\rangle$$



4.3 SUAVIZAÇÃO

Suavização é o processo de computar a distribuição em algum instante de tempo passado k, dada a seqüência completa de observações de 1 a t, ou seja, $P(X_k|e_{1:t})$, para $1 \leq k < t$. Isto é feito mais convenientemente em duas etapas: as evidências até o instante *k* e as evidências de *k+1* até *t*:

$$P(X_k|e_{1:t}) = P(X_k|e_{1:k},e_{k+1:t})$$

$$= \alpha\, P(X_k|e_{1:k})\, P(e_{k+1:t}|X_k,e_{1:k}) \text{ (Regra de Bayes) } =$$

$$\alpha\, P(X_k|e_{1:k})\, P(e_{k+1:t}|X_k)$$

$$= \alpha\, f_{1:k} b_{k+1:t}$$

onde a mensagem *backward* $b_{k+1:t}$, análoga à mensagem *forward*, é computada por um processo recursivo que anda para trás a partir de t.

Abrindo o termo $b_{k+1:t} = P(e_{k+1:t}|X_k)$:

$$P(e_{k+1:t}|\, X_k) = \sum_{xk+1} P(e_{k+1:t}|\, X_k,x_{k+1})\, P(x_{k+1}|\, X_k)$$

$$= \sum_{xk+1} P(e_{k+1:t,}\, e_{k+2:t}\,|\, x_{k+1})\, P(x_{k+1}|\, X_k)$$

$$= \sum_{xk+1} P(e_{k+1}|x_{k+1})\, P(e_{k+2:t}|x_{k+1}) P(x_{k+1}|\, X_k)$$

Dentro da somatória, o primeiro termo é o modelo sensor e o terceito termo é o modelo de transição. O segundo termo é que dá a recursividade na equação, pois há uma relação direta entre $P(e_{k+1:t}|X_k)$ e $P(e_{k+2:t}|x_{k+1})$. Utilizando a notação de mensagem, temos:

$$b_{k+1:t} = \text{BACKWARD}\, (b_{k+2:t},e_{k+1:t})$$

Assim, na equação $\alpha\, f_{1:k}b_{k+1:t}$, os dois termos podem ser computados através da recursividade, um partindo de *1* até *k*, utilizando a equação da filtragem, e o outro, de *t* até *k+1* utilizando a equação encontrada.

Notar que a fase de retrocesso é inicializada com $b_{t+1:t}$ = $P(e_{t+1:t}|X_t)$ = 1. (Devido ao fato de $e_{t+1:t}$ ser uma seqüência vazia, a probabilidade de observação é 1).

4.3.1 EXEMPLO

Utilizando o mesmo exemplo anterior, vamos computar a probabilidade de chover no instante *t = 1*, dadas as observações de guarda-chuvas nos dias 1 e 2.

$$P(R_1|\ u_{1,}u_2) = \alpha\ P(R_1|u_1)\ P(u_2|R_1)$$

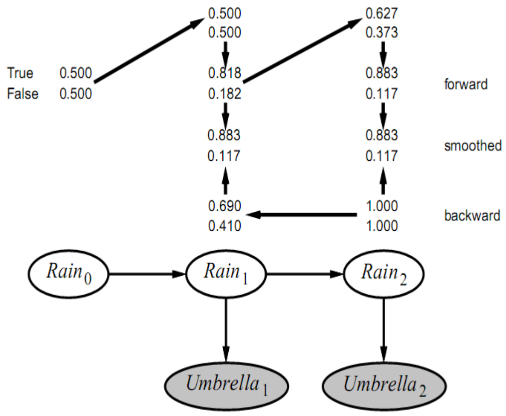
O primeiro termo já foi calculado anteriormente, no processo de filtragem, onde foi obtido o valor <0.818, 0.182>. O segundo termo pode ser computado aplicando o algoritmo de *backward*:

$$P(u_2|\ R_1) = \sum_{r2} P(u_2|r_2)\ P(|r_2)\ P(r_2|R_1)$$

$$= (0.9 \times 1 \times <0.7,0.3>) + (0.2 \times 1 \times <0.3,0.7>) = <0.69,0.41>$$

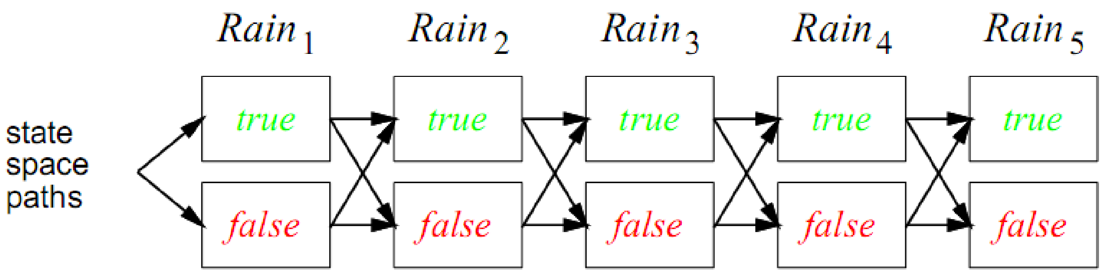
Assim:

$$P(R_1|\ u_{1,}u_2) = \alpha <0.818,0.182> \times <0.69,0.41> = <0.883,0.117>$$



4.4 EXPLICAÇÃO MAIS PROVÁVEL

O algoritmo de explicação mais provável tem como objetivo traçar o a seqüência mais provável. Para tal, ao invés de essa seqüência se calculada como suavizações sobre cada espaço de tempo, o cálculo é feito assumindo cada seqüência como um caminho sobre um grafo cujos nós são possíveis estados a cada timeslice.



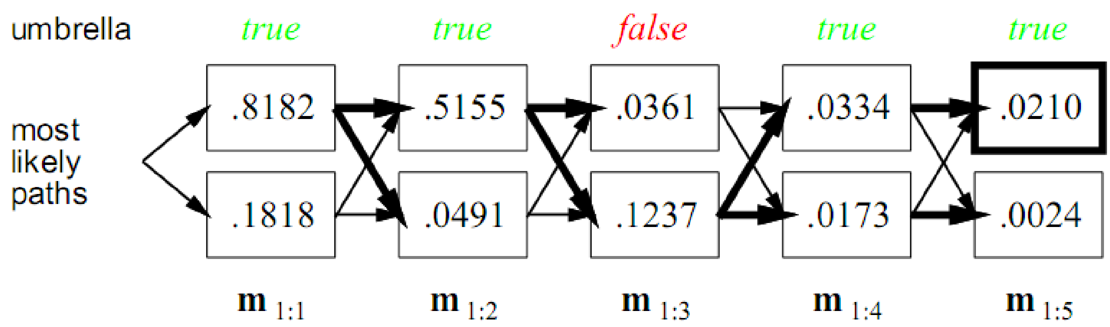
O algoritmo consiste na premissa de que há um relacionamento recursivo entre os caminhos mais prováveis para cada estado x_{t+1} e os caminhos mais prováveis para cada estado x_t.

$$\max_{\mathbf{x}_1 \ldots \mathbf{x}_t} \mathbf{P}(\mathbf{x}_1, \ldots, \mathbf{x}_t, \mathbf{X}_{t+1} | \mathbf{e}_{1:t+1})$$
$$= \alpha \mathbf{P}(\mathbf{e}_{t+1} | \mathbf{X}_{t+1}) \max_{\mathbf{x}_t} \left(\mathbf{P}(\mathbf{X}_{t+1} | \mathbf{x}_t) \max_{\mathbf{x}_1 \ldots \mathbf{x}_{t-1}} P(\mathbf{x}_1, \ldots, \mathbf{x}_{t-1}, \mathbf{x}_t | \mathbf{e}_{1:t}) \right)$$

No caso desta equação, a mensagem (analogamente aos algoritmos anteriores) é:

$$\mathbf{m}_{1:t} = \max_{\mathbf{x}_1 \ldots \mathbf{x}_{t-1}} \mathbf{P}(\mathbf{x}_1, \ldots, \mathbf{x}_{t-1}, \mathbf{X}_t | \mathbf{e}_{1:t})$$

O algoritmo necessita armazenar ponteiros que identificam a seqüência escolhida , por isso a complexidade espacial é igual a *t,* a complexidade temporal é igual ao tamanho da seqüência, *t*. Como, dependendo do problema, a solução pode ter um tamanho muito grande para a memória disponível à infra-estrutura, pode ser definida uma heurística que delimita a profundidade da busca.



A equação do algoritmo de Viterbi é:

$m_{1:t+1} = P(e_{t+1}|X_{t+1}) \max_{X_t} (P(X_{t+1}| x_t) m_{1:t})$, aimplementação é mostrada abaixo, nela fica explícita a recursão caracterizada pela passagem das mensagens.

$$m_t[x] = \max_{x_{1:t-1}} P(x_{1:t-1}, x)$$
$$= \max_{x_{1:t-1}} P(x_{1:t-1}) P(x | x_{t-1})$$
$$= \max_{x_{t-1}} P(x_t | x_{t-1}) \max_{x_{1:t-2}} P(x_{1:t-1})$$
$$= \max_{x_{t-1}} P(x_t | x_{t-1}) m_{t-1}[x]$$

$$m_1[x] = P(x_1)$$

5 MODELO DE MARKOV OCULTO

O Modelo de Markov Oculto (HMM – Hidden Markov Model) é um modelo probabilístico temporal no qual o estado do processo é descrito por uma única variável. Variáveis de estado podem ser adicionadas a um HMM se forem combinadas formando uma "megavariável" cujos valores são todas as possíveis tuplas de valores das variáveis de estado individuais.

O HMM possibilita que os algoritmos vistos acima sejam implementados de uma forma elegante em forma de matrizes e seu uso possibilita a aplicação dos algoritmos em reconhecimento de padrões, como em reconhecimento de fala ou de escrita. O uso nesses casos é justificado pois não são sabidos todos os estados do problema, apesar de saber-se que o problema segue uma cadeia de Markov.

5.1 DEFINIÇÃO:

Um HMM é a tripla $M = (\Sigma, Q, \Theta)$,na qual:

- Σ é um alfabeto de símbolos.
- Q é um conjunto finite de estados, capaz de emitir símbolos do alfabeto Σ.
- Θ é o conjunto de probabilibdades, constituído de:
  - – Probabilidades de transição de estado, denotadas por a_{kl} para cada k, $l \in Q$.
  - – Probabilidades de emissão, denotadas por $e_k(b)$ para cada $k \in Q$ e $b \in \Sigma$.

Um caminho $\Pi = (\pi 1, \ldots, \pi L)$ no modelo M é uma seqüência de estados. O estado segue uma cadeia de Markov simples, então, a probabilidade de se mover a um dado estado depende apenas do estado anterior. Como no modelo de cadeia de Markov, é possível definer as probaboçodades de transição de estado em termos de Π :

$$a_{kl} = P(\pi i = l | \pi i{-}1 = k)$$

Entretanto, em um HMM não há uma correspondência um para um entre os estados e os símbolos. Logo, em um HMM, introduz-se um novo conjunto de parâmetros, $e_k(b)$, chamado emissão de probabilidades. Dada uma seqüêcia $X = (x1, \ldots, xL) \in \Sigma*$ define-se :

$$ek(b) = P(xi = b | \pi i = k)$$

$ek(b)$ é a probabilidade de que o símbolo b seja visto quando está em um estado k.

Logo, a probabilidade de que a sequencia X foi gerada pelo modelo M dado o caminho Π é:

$$P(X, \Pi) = a_{\pi_1,\pi_2} \cdot \prod_{i=1}^{L} e_{\pi_i}(x_i) \cdot a_{\pi_i,\pi_{i+1}}$$

Na qual denota-se $\pi 0$ = *início* e $\pi L+1$ = *fim*.

5.2 ALGORITMOS MATRICIAIS SIMPLIFICADOS

Modelo Transacional:

O modelo $P(X_t|X_{t-1})$ se torna uma matriz T, SxS na qual:

$T_{ij}=P(X_t|X_{t-1}=i)$

Ou seja, T_{ij} é a probabilidade da transação do estado i ao j. No caso do exemplo do guarda-chuva:

$$\mathbf{T} = \mathbf{P}(X_t|X_{t-1}) = \begin{pmatrix} 0.7 & 0.3 \\ 0.3 & 0.7 \end{pmatrix}$$

Modelo do Sensor:

Para cada passo de tempo t, é construída uma matriz O_t cujas entradas diagonais são dadas pelo valor $P(e_t|X_t=i)$ e cujas outras entradas são zero. No exemplo do guarda chuva no qual U_1=true

$$\mathbf{O}_1 = \begin{pmatrix} 0.9 & 0 \\ 0 & 0.2 \end{pmatrix}$$

Forward/Backward:

A partir das definições anteriores, chega-se nos cálculos das mensagens de forward e de backward utilizadas nos algoritmos de filtragem, predição e suavização:

$$f_{1:t+1} = \alpha O_{t+1} T^T f_{1:t}$$

$$b_{k+1:t} = T O_{k+1} b_{k+2:t}$$

Que tem complexidade temporal de $O(S^2t)$, dado que matrizes são multiplicadas a cada passo e requer armazenagem de $O(St)$, t vetores de tamanho S.

Online Smoothing:

A representação matricial permite que seja facilitado cálculo da suavização, dado um intervalo entre os time slices cuja diferença de intervalos apresente um lag do qual o cálculo de suavização seja independente. Dado um lag d, o estado atual sendo t e será suavizado o time slice t-d, sem se alongar muito nos cálculos:

$$\mathbf{b}_{t-d+2:t+1} = \left(\prod_{i=t-d+2} \mathbf{TO}_i \right) \mathbf{b}_{t+2:t+1} = \mathbf{B}_{t-d+2:t+1}\mathbf{1}$$

$$\mathbf{B}_{t-d+2:t+1} = \mathbf{O}_{t-d+1}^{-1}\mathbf{T}^{-1}\mathbf{B}_{t-d+1:t}\mathbf{T}\mathbf{O}_{t+1}$$

As equações acima provêem uma maneira recursiva para o cálculo da suavização independente do tamanho do lag.