Tema 3

Modelos probabilísticos para linguagens regulares

Professora:
Ariane Machado Lima

Vídeo 1

Gramáticas estocásticas

Professora:
Ariane Machado Lima

Contextualização

- Na aula passada:
  - Hierarquia de Chomksy
  - Linguagens regulares
    - Gramáticas regulares
    - Autômatos finitos (determinísticos e não-determinísticos)
- Hoje:
  - Introdução às versões probabilísticas (gramáticas estocásticas e HMMs)
- Na próxima aula:
  - Treinamento/aprendizado (inferência gramatical)
  - Estimação de desempenho

Gramáticas Estocásticas

- Definição: uma gramática estocática G é uma quíntupla (V, Σ, S, P, ρ), onde
  - V é o conjunto de símbolos não-terminais (variáveis)
  - Σ é o conjunto de símbolos terminais
  - S é o símbolo inicial
  - P é o conjunto de produções da forma

    $(\Sigma \cup V)^* \, V \, (\Sigma \cup V)^* \rightarrow (\Sigma \cup V)^*$
  - ρ é o conjunto de distribuições de probabilidades sobre as produções de mesmo lado esquerdo

    $\sum_i \rho(\alpha \rightarrow \beta_i) = 1$

Exemplo

```
atga
atgg
atta
aaga
cgag
```

$G = \{V, \Sigma, S, P\}$

$V = \{S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8, S_9, S_{10}\}$

$\Sigma = \{a, c, g, t\}$

$S = S_1$

$P = \{$
$S_1 \rightarrow aS_2 \mid cS_3$
$S_2 \rightarrow tS_4 \mid aS_5$
$S_4 \rightarrow gS_6 \mid tS_7$
$S_6 \rightarrow a \mid g$
$S_7 \rightarrow a$
$S_5 \rightarrow gS_8$
$S_8 \rightarrow a$
$S_3 \rightarrow gS_9$
$S_9 \rightarrow aS_{10}$
$S_{10} \rightarrow g$
$\}$

Exemplo

atga
atgg
atta
aaga
cgag

$G = \{V, \Sigma, S, P\}$

$V = \{S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8, S_9, S_{10}\}$

$\Sigma = \{a, c, g, t\}$

$S = S_1$

$P = \{ S_1 \rightarrow aS_2$ [0.80] $\mid cS_3$ [0.20]
$S_2 \rightarrow tS_4 \mid aS_5$
$S_4 \rightarrow gS_6 \mid tS_7$
$S_6 \rightarrow a \mid g$
$S_7 \rightarrow a$
$S_5 \rightarrow gS_8$
$S_8 \rightarrow a$
$S_3 \rightarrow gS_9$
$S_9 \rightarrow aS_{10}$
$S_{10} \rightarrow g$
$\}$

Exemplo

atga
atgg
atta
aaga
cgag

$G = \{V, \Sigma, S, P\}$

$V = \{S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8, S_9, S_{10}\}$

$\Sigma = \{a, c, g, t\}$

$S = S_1$

$P = \{\ S_1 \rightarrow aS_2\ [0.80] \mid cS_3\ [0.20]$

$S_2 \rightarrow tS_4\ [0.75] \mid aS_5\ [0.25]$

$S_4 \rightarrow gS_6 \mid tS_7$

$S_6 \rightarrow a \mid g$

$S_7 \rightarrow a$

$S_5 \rightarrow gS_8$

$S_8 \rightarrow a$

$S_3 \rightarrow gS_9$

$S_9 \rightarrow aS_{10}$

$S_{10} \rightarrow g$

$\}$

Exemplo

atga
atgg
atta
aaga
cgag

$G = \{V, \Sigma, S, P\}$

$V = \{S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8, S_9, S_{10}\}$

$\Sigma = \{a, c, g, t\}$

$S = S_1$

$P = \{ S_1 \rightarrow aS_2$ [0.80] | cS_3 [0.20]

$S_2 \rightarrow tS_4$ [0.75] | aS_5 [0.25]

$S_4 \rightarrow gS_6$ [0.66666] | tS_7 [0.33333]

$S_6 \rightarrow a$ | g

$S_7 \rightarrow a$

$S_5 \rightarrow gS_8$

$S_8 \rightarrow a$

$S_3 \rightarrow gS_9$

$S_9 \rightarrow aS_{10}$

$S_{10} \rightarrow g$

$\}$

Exemplo

```
atga
atgg
atta
aaga
cgag
```

$G = \{V, \Sigma, S, P\}$

$V = \{S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8, S_9, S_{10}\}$

$\Sigma = \{a, c, g, t\}$

$S = S_1$

$P = \{$
$S_1 \rightarrow aS_2$ [0.80] | cS_3 [0.20]
$S_2 \rightarrow tS_4$ [0.75] | aS_5 [0.25]
$S_4 \rightarrow gS_6$ [0.66666] | tS_7 [0.33333]
$S_6 \rightarrow a$ [0.50] | g [0.50]
$S_7 \rightarrow a$
$S_5 \rightarrow gS_8$
$S_8 \rightarrow a$
$S_3 \rightarrow gS_9$
$S_9 \rightarrow aS_{10}$
$S_{10} \rightarrow g$
$\}$

Exemplo

```
atga
atgg
atta
aaga
cgag
```

$G = \{V, \Sigma, S, P\}$

$V = \{S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8, S_9, S_{10}\}$

$\Sigma = \{a, c, g, t\}$

$S = S_1$

$P = \{$ $S_1 \rightarrow aS_2$ [0.80] | cS_3 [0.20]

$S_2 \rightarrow tS_4$ [0.75] | aS_5 [0.25]

$S_4 \rightarrow gS_6$ [0.66666] | tS_7 [0.33333]

$S_6 \rightarrow a$ [0.50] | g [0.50]

$S_7 \rightarrow a$ [1.00]

$S_5 \rightarrow gS_8$ [1.00]

$S_8 \rightarrow a$ [1.00]

$S_3 \rightarrow gS_9$ [1.00]

$S_9 \rightarrow aS_{10}$ [1.00]

$S_{10} \rightarrow g$ [1.00]

$\}$

Exemplo

```
atga
atgg
atta
aaga
cgag
```

$G = \{V, \Sigma, S, P\}$

$V = \{S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8, S_9, S_{10}\}$

$\Sigma = \{a, c, g, t\}$

$S = S_1$

$P = \{$
$S_1 \rightarrow aS_2$ [0.80] | cS_3 [0.20]
$S_2 \rightarrow tS_4$ [0.75] | aS_5 [0.25]
$S_4 \rightarrow gS_6$ [0.66666] | tS_7 [0.33333]
$S_6 \rightarrow a$ [0.50] | g [0.50]
$S_7 \rightarrow a$ [1.00]
$S_5 \rightarrow gS_8$ [1.00]
$S_8 \rightarrow a$ [1.00]
$S_3 \rightarrow gS_9$ [1.00]
$S_9 \rightarrow aS_{10}$ [1.00]
$S_{10} \rightarrow g$ [1.00]
$\}$

P (x | G) = produto das probabilidades das produções usadas na derivação de x (dadas pela gramática G)

Exemplo

```
atga
atgg
atta
aaga
cgag
```

$G = \{V, \Sigma, S, P\}$

$V = \{S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8, S_9, S_{10}\}$

$\Sigma = \{a, c, g, t\}$

$S = S_1$

$P = \{ S_1 \rightarrow aS_2\ [0.80] \mid cS_3\ [0.20]$

$S_2 \rightarrow tS_4\ [0.75] \mid aS_5\ [0.25]$

$S_4 \rightarrow gS_6\ [0.66666] \mid tS_7\ [0.33333]$

$S_6 \rightarrow a\ [0.50] \mid g\ [0.50]$

$S_7 \rightarrow a\ [1.00]$

$S_5 \rightarrow gS_8\ [1.00]$

$S_8 \rightarrow a\ [1.00]$

$S_3 \rightarrow gS_9\ [1.00]$

$S_9 \rightarrow aS_{10}\ [1.00]$

$S_{10} \rightarrow g\ [1.00]$

$\}$

P(atga | G) = 0.8 * 0.75 * 0.66666 * 0.5 = 0.2

P (x | G) = produto das probabilidades das produções usadas na derivação de x (dadas pela gramática G)

Exemplo

atga
atgg
atta
aaga
cgag

$G = \{V, \Sigma, S, P\}$

$V = \{S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8, S_9, S_{10}\}$

$\Sigma = \{a, c, g, t\}$

$S = S_1$

$P = \{$
$S_1 \rightarrow aS_2$ [0.80] | cS_3 [0.20]
$S_2 \rightarrow tS_4$ [0.75] | aS_5 [0.25]
$S_4 \rightarrow gS_6$ [0.66666] | tS_7 [0.33333]
$S_6 \rightarrow a$ [0.50] | g [0.50]
$S_7 \rightarrow a$ [1.00]
$S_5 \rightarrow gS_8$ [1.00]
$S_8 \rightarrow a$ [1.00]
$S_3 \rightarrow gS_9$ [1.00]
$S_9 \rightarrow aS_{10}$ [1.00]
$S_{10} \rightarrow g$ [1.00]
$\}$

P(**atga** | G) = 0.8 * 0.75 * 0.66666 * 0.5 = 0.2
P(**atgg** | G) = 0.8 * 0.75 * 0.66666 * 0.5 = 0.2
P(**atta** | G) = 0.8 * 0.75 * 0.33333 * 1 = 0.2
P(**aaga** | G) = 0.8 * 0.25 * 1 * 1 = 0.2
P(**cgag** | G) = 0.2 * 1 * 1 * 1 = 0.2
P(**attg** | G) = 0

P (x | G) = produto das probabilidades das produções usadas na derivação de x (dadas pela gramática G)

Exemplo

Estimação por Máxima Verossimilhança

$= \{ S_1 \rightarrow aS_2$ [0.80] | cS_3 [0.20]

$S_2 \rightarrow tS_4$ [0.75] | aS_5 [0.25]

$S_4 \rightarrow gS_6$ [0.66666] | tS_7 [0.33333]

$S_6 \rightarrow a$ [0.50] | g [0.50]

$S_7 \rightarrow a$ [1.00]

$S_5 \rightarrow gS_8$ [1.00]

$S_8 \rightarrow a$ [1.00]

$S_3 \rightarrow gS_9$ [1.00]

$S_9 \rightarrow aS_{10}$ [1.00]

$S_{10} \rightarrow$ g [1.00]

}

$S = S_1$

P(**atga** | G) = 0.8 * 0.75 * 0.66666 * 0.5 = 0.2
P(**atgg** | G) = 0.8 * 0.75 * 0.66666 * 0.5 = 0.2
P(**atta** | G) = 0.8 * 0.75 * 0.33333 * 1 = 0.2
P(**aaga** | G) = 0.8 * 0.25 * 1 * 1 = 0.2
P(**cgag** | G) = 0.2 * 1 * 1 * 1 = 0.2
P(**attg** | G) = 0

P (x | G) = produto das probabilidades das produções usadas na derivação de x (dadas pela gramática G)

Exemplo

E qual a vantagem???

atga
atgg
atta
aaga
cgag

$P = \{ S_1 \rightarrow aS_2$ [0.80] $\mid cS_3$ [0.20]

$S_2 \rightarrow tS_4$ [0.75] $\mid aS_5$ [0.25]

$S_4 \rightarrow gS_6$ [0.66666] $\mid tS_7$ [0.33333]

$S_6 \rightarrow a$ [0.50] $\mid$ g [0.50]

$S_7 \rightarrow a$ [1.00]

$S_5 \rightarrow gS_8$ [1.00]

$S_8 \rightarrow a$ [1.00]

$S_3 \rightarrow gS_9$ [1.00]

$S_9 \rightarrow aS_{10}$ [1.00]

$S_{10} \rightarrow g$ [1.00] }

P(**atga** | G) = 0.8 * 0.75 * 0.66666 * 0.5 = 0.2
P(**atgg** | G) = 0.8 * 0.75 * 0.66666 * 0.5 = 0.2
P(**atta** | G) = 0.8 * 0.75 * 0.33333 * 1 = 0.2
P(**aaga** | G) = 0.8 * 0.25 * 1 * 1 = 0.2
P(**cgag** | G) = 0.2 * 1 * 1 * 1 = 0.2
P(**attg** | G) = 0

P (x | G) = produto das probabilidades das produções usadas na derivação de x (dadas pela gramática G)

Exemplo

`atga`
`atgg`
`atta`
`aaga`
`cgag`

E qual a vantagem???

Conseguir atribuir uma probabilidade a cadeias que não pertencem ao treinamento (generalização)

$P = \{ S_1 \rightarrow aS_2 \ [0.80] \mid cS_3 \ [0.20]$

$S_2 \rightarrow tS_4 \ [0.75] \mid aS_5 \ [0.25]$

$S_4 \rightarrow gS_6 \ [0.66666] \mid tS_7 \ [0.33333]$

$S_6 \rightarrow a \ [0.50] \mid g \ [0.50]$

$S_7 \rightarrow a \ [1.00]$

$S_5 \rightarrow gS_8 \ [1.00]$

$S_8 \rightarrow a \ [1.00]$

$S_3 \rightarrow gS_9 \ [1.00]$

$S_9 \rightarrow aS_{10} \ [1.00]$

$S_{10} \rightarrow g \ [1.00]$

}

P(**atga** | G) = 0.8 * 0.75 * 0.66666 * 0.5 = 0.2
P(**atgg** | G) = 0.8 * 0.75 * 0.66666 * 0.5 = 0.2
P(**atta** | G) = 0.8 * 0.75 * 0.33333 * 1 = 0.2
P(**aaga** | G) = 0.8 * 0.25 * 1 * 1 = 0.2
P(**cgag** | G) = 0.2 * 1 * 1 * 1 = 0.2
P(**attg** | G) = 0

P (x | G) = produto das probabilidades das produções usadas na derivação de x (dadas pela gramática G)

Exemplo

`atga`
`atgg`
`atta`
`aaga`
`cgag`

E qual a vantagem???

Conseguir atribuir uma probabilidade a cadeias que não pertencem ao treinamento (generalização)

$P = \{ S_1 \rightarrow aS_2$ [0.80] | cS_3 [0.20]

$S_2 \rightarrow tS_4$ [0.75] | aS_5 [0.25]

$S_4 \rightarrow gS_6$ [0.66666] | tS_7 [0.33333]

$S_6 \rightarrow a$ [0.50] | g [0.50]

$S_7 \rightarrow a$ [1.00]

$S_5 \rightarrow gS_8$ [1.00]

$S_8 \rightarrow a$ [1.00]

$S_3 \rightarrow gS_9$ [1.00]

$S_9 \rightarrow aS_{10}$ [1.00]

$S_{10} \rightarrow g$ [1.00]

}

P(**atga** | G) = 0.8 * 0.75 * 0.66666 * 0.5 = 0.2
P(**atgg** | G) = 0.8 * 0.75 * 0.66666 * 0.5 = 0.2
P(**atta** | G) = 0.8 * 0.75 * 0.33333 * 1 = 0.2
P(**aaga** | G) = 0.8 * 0.25 * 1 * 1 = 0.2
P(**cgag** | G) = 0.2 * 1 * 1 * 1 = 0.2
P(**attg** | G) = 0

P (x | G) = produto das probabilidades das produções usadas na derivação de x (dadas pela gramática G)

Exemplo

$P = \{ S_1 \rightarrow aS_2 \ [0.80] \mid cS_3 \ [0.20]$

$S_2 \rightarrow tS_4 \ [0.75] \mid aS_5 \ [0.25]$

$S_4 \rightarrow gS_6 \ [0.66666] \mid tS_7 \ [0.33333]$

$S_6 \rightarrow a \ [0.50] \mid g \ [0.50]$

$S_7 \rightarrow a \ [0.90] \mid g \ [0.10]$

$S_5 \rightarrow gS_8 \ [1.00]$

$S_8 \rightarrow a \ [1.00]$

$S_3 \rightarrow gS_9 \ [1.00]$

$S_9 \rightarrow aS_{10} \ [1.00]$

$S_{10} \rightarrow g \ [1.00]$

$\}$

atga
atgg
atta
aaga
cgag

E qual a vantagem???

Conseguir atribuir uma probabilidade a cadeias que não pertencem ao treinamento (generalização)

P(**atga** | G) = 0.8 * 0.75 * 0.66666 * 0.5 = 0.2
P(**atgg** | G) = 0.8 * 0.75 * 0.66666 * 0.5 = 0.2
P(**atta** | G) = 0.8 * 0.75 * 0.33333 * 0.9 = 0.18
P(**aaga** | G) = 0.8 * 0.25 * 1 * 1 = 0.2
P(**cgag** | G) = 0.2 * 1 * 1 * 1 = 0.2
P(**attg** | G) = 0.8 * 0.75 * 0.33333 * 0.1 = 0.02

P (x | G) = produto das probabilidades das produções usadas na derivação de x (dadas pela gramática G)

E os autômatos?

- Podem ser
  - Determinísticos estocásticos
  - Não-determinísticos estocásticos
- Para cada estado, há uma distribuição de probabilidades sobre as transições que dele partem

```
atga
atgg
atta
aaga
cgag
```

Exemplo

gramática regular (linear à direita)

$G = \{V, \Sigma, S, P\}$

$V = \{S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8, S_9, S_{10}\}$

$\Sigma = \{a, c, g, t\}$

$S = S_1$

$P = \{ S_1 \rightarrow aS_2 \mid cS_3$

$S_2 \rightarrow tS_4 \mid aS_5$

$S_4 \rightarrow gS_6 \mid tS_7$

$S_6 \rightarrow a \mid g$

$S_7 \rightarrow a$

$S_5 \rightarrow gS_8$

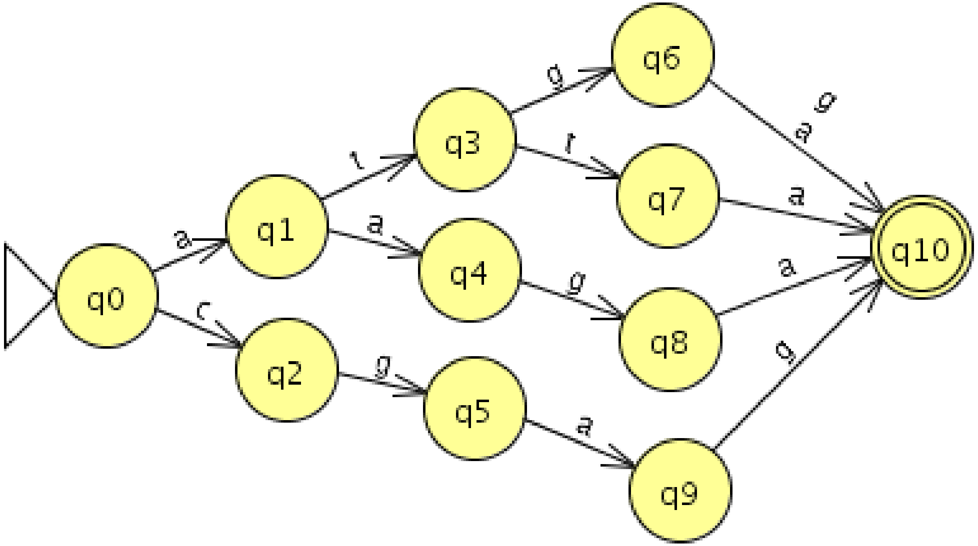
$S_8 \rightarrow a$

$S_3 \rightarrow gS_9$

$S_9 \rightarrow aS_{10}$

$S_{10} \rightarrow g$

$\}$



AFN

atga
atgg
atta
aaga
cgag

Exemplo

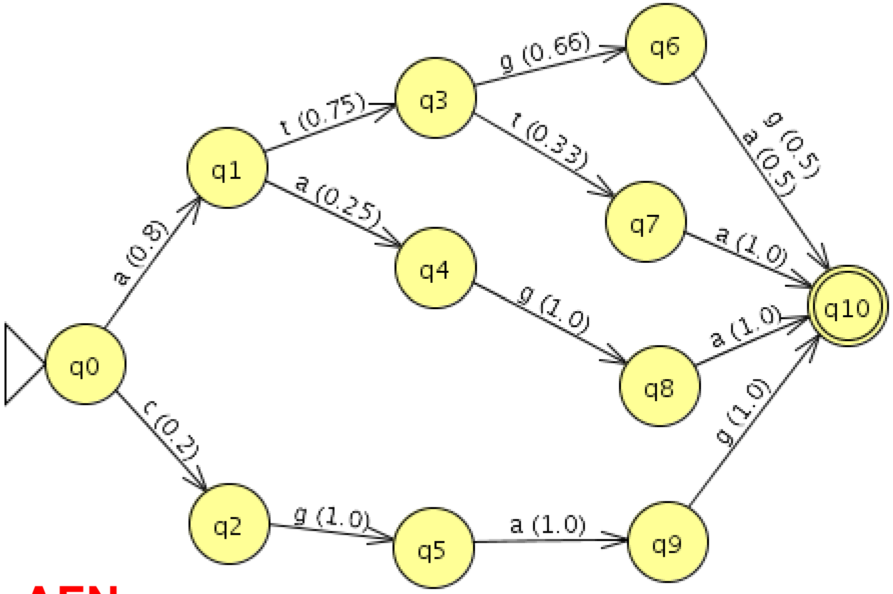
$G = \{V, \Sigma, S, P\}$

$V = \{S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8, S_9, S_{10}\}$

$\Sigma = \{a, c, g, t\}$

$S = S_1$

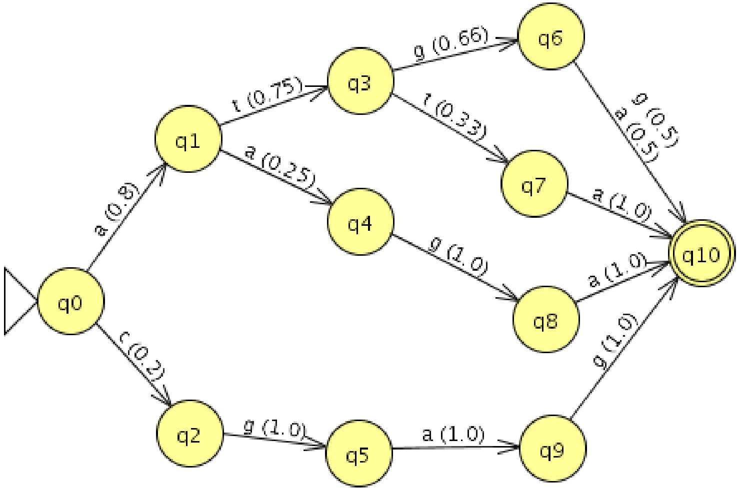
$P = \{$
$S_1 \rightarrow aS_2$ [0.80] | cS_3 [0.20]
$S_2 \rightarrow tS_4$ [0.75] | aS_5 [0.25]
$S_4 \rightarrow gS_6$ [0.66666] | tS_7 [0.33333]
$S_6 \rightarrow a$ [0.50] | g [0.50]
$S_7 \rightarrow a$ [1.00]
$S_5 \rightarrow gS_8$ [1.00]
$S_8 \rightarrow a$ [1.00]
$S_3 \rightarrow gS_9$ [1.00]
$S_9 \rightarrow aS_{10}$ [1.00]
$S_{10} \rightarrow g$ [1.00]
$\}$



AFN

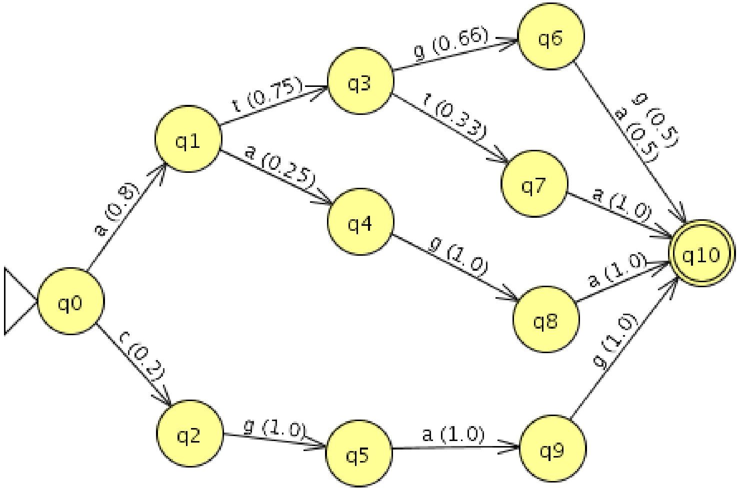
Autômatos probabilísticos (ou estocásticos)

- À sua definição (AFD ou AFD) acrescente:
  - p: Q x Σ X Q → [0,1], de tal forma que
  - para todo q_{atual} є Q, $\Sigma_{a \in \Sigma,\, q_{prox} \in Q}\ p(q_{atual}, a, q_{prox}) = 1$



Autômatos probabilísticos (ou estocásticos)

- À sua definição (AFD ou AFN) acrescente:
  - p: $Q \times \Sigma \times Q \rightarrow [0,1]$, de tal forma que
  - para todo $q_{atual} \in Q$, $\Sigma_{a \in \Sigma,\, q_{prox} \in Q}\; p(q_{atual}, a, q_{prox}) = 1$
  - Em algumas formulações tenho ainda uma distribuição inicial π sobre os estados $\Sigma_i = \pi(q_i) = 1$



Fim do vídeo 1

Gramáticas estocásticas

Professora:
Ariane Machado Lima

Vídeo 2

HMMs (parte 1)

Professora:
Ariane Machado Lima

AFN Probabilístico e HMM

- AFNs probabilísticos são equivalentes a HMMs

Modelo Oculto de Markov
Hidden Markov Model (HMM)

- Primeiro um caso mais simples: imagine que eu tenho uma urna de bolas coloridas (ou seja, a urna tem uma distribuição de probabilidades sobre essas cores)

- Alguém tem que adivinhar qual a próxima cor. Como isso poderia ser feito?

Modelo Oculto de Markov
Hidden Markov Model (HMM)

- Primeiro um caso mais simples: imagine que eu tenho uma urna de bolas coloridas (ou seja, a urna tem uma distribuição de probabilidades sobre essas cores)

- Alguém tem que adivinhar qual a próxima cor. Como isso poderia ser feito?

  Chutando a cor de maior probabilidade

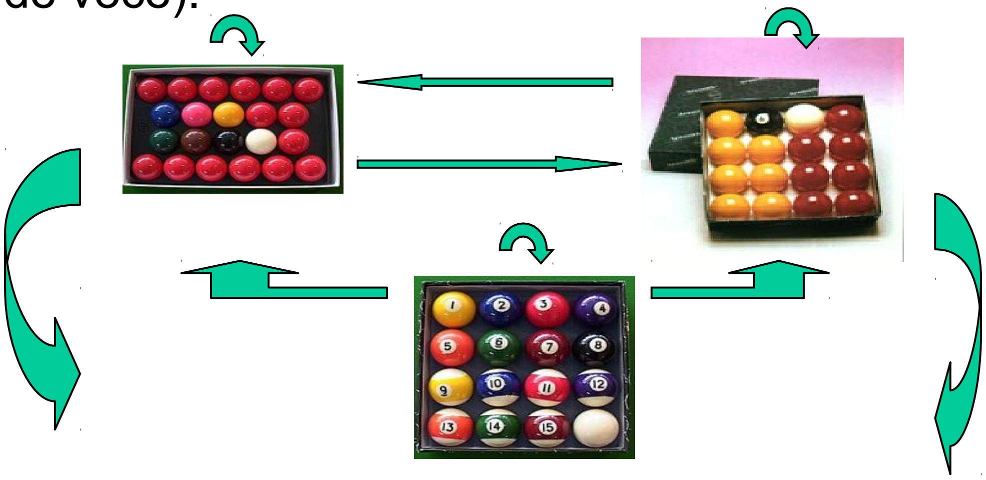
Modelo Oculto de Markov
Hidden Markov Model (HMM)

- AGORA imagine que eu tenho várias urnas de bolas coloridas (ou seja, cada urna tem uma distribuição de probabilidades sobre essas cores, potencialmente diferente).

-

Modelo Oculto de Markov
Hidden Markov Model (HMM)

- AGORA imagine que eu tenho várias urnas de bolas coloridas (ou seja, cada urna tem uma distribuição de probabilidades sobre essas cores, potencialmente diferente). A cada sorteio de bola eu posso trocar de urna (troca de urnas é um processo que eu faço escondido de você).



- Alguém tem que adivinhar qual a próxima cor. Como isso poderia ser feito?

Modelo Oculto de Markov
Hidden Markov Model (HMM)

- Símbolos de emissão (cores de bola)
- Estados ocultos (urnas)
- Uma distribuição de probabilidades de emissão de símbolos associada a cada estado
- Probabilidade de transição entre estados
- Distribuição de probabilidades do estado inicial

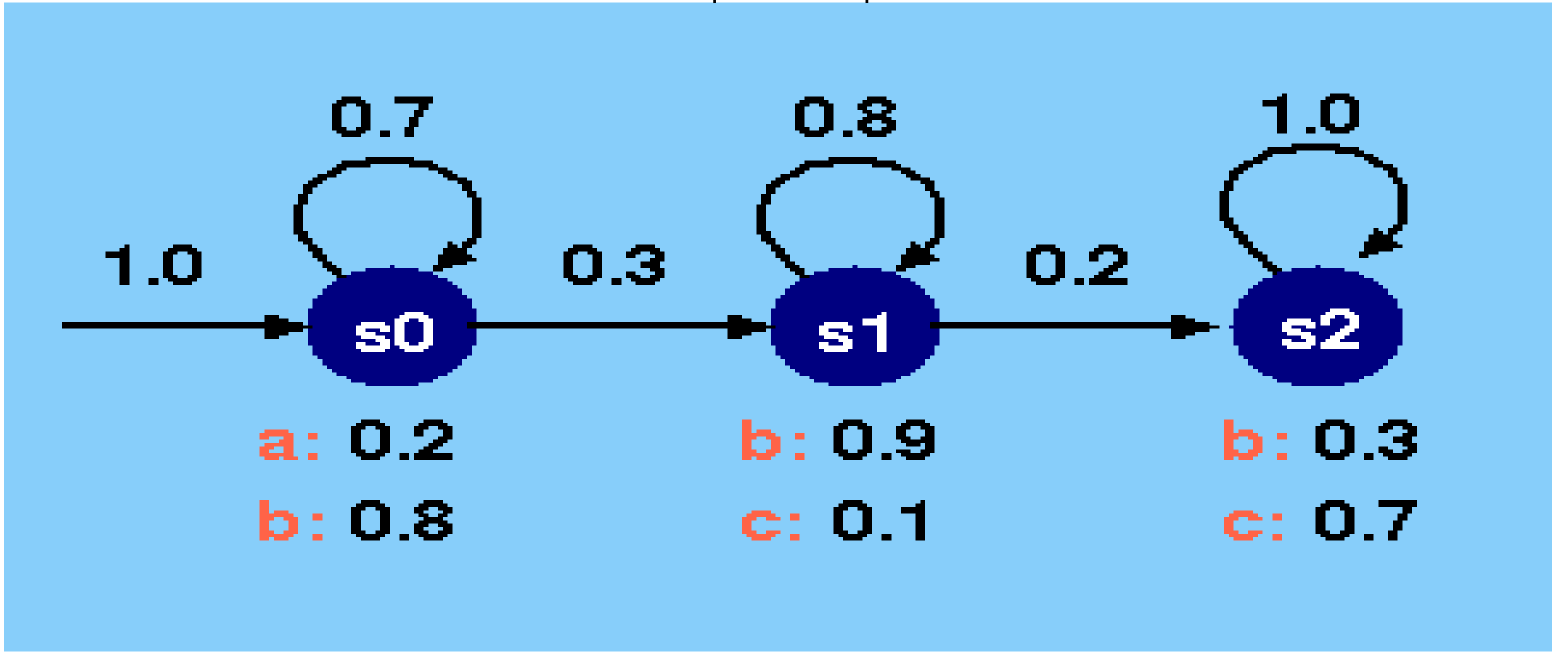
Modelo Oculto de Markov
Hidden Markov Model (HMM)

Q: conjunto de estados ocultos
Σ: conjunto de símbolos de emissão

Probabilidades **iniciais** $\pi: Q \rightarrow [0,1]$
$\Sigma_i = \pi(q_i) = 1$

Probabilidades de **transições**
$t_{ij} = P(q_j \mid q_i)$
$\Sigma_j\, t_{ij} = 1$

Probabilidades de **emissões**
$e_i\,(a) = P(a \mid q_i)$
$\Sigma_{a \in \Sigma}\, e_i(a) = 1$



Modelo Oculto de Markov
Hidden Markov Model (HMM)

Q: conjunto de estados ocultos

Σ: conjunto de símbolos de emissão

Probabilidades **iniciais** π: Q → [0,1]

$\Sigma_i = \pi(q_i) = 1$

Probabilidades de **transições**

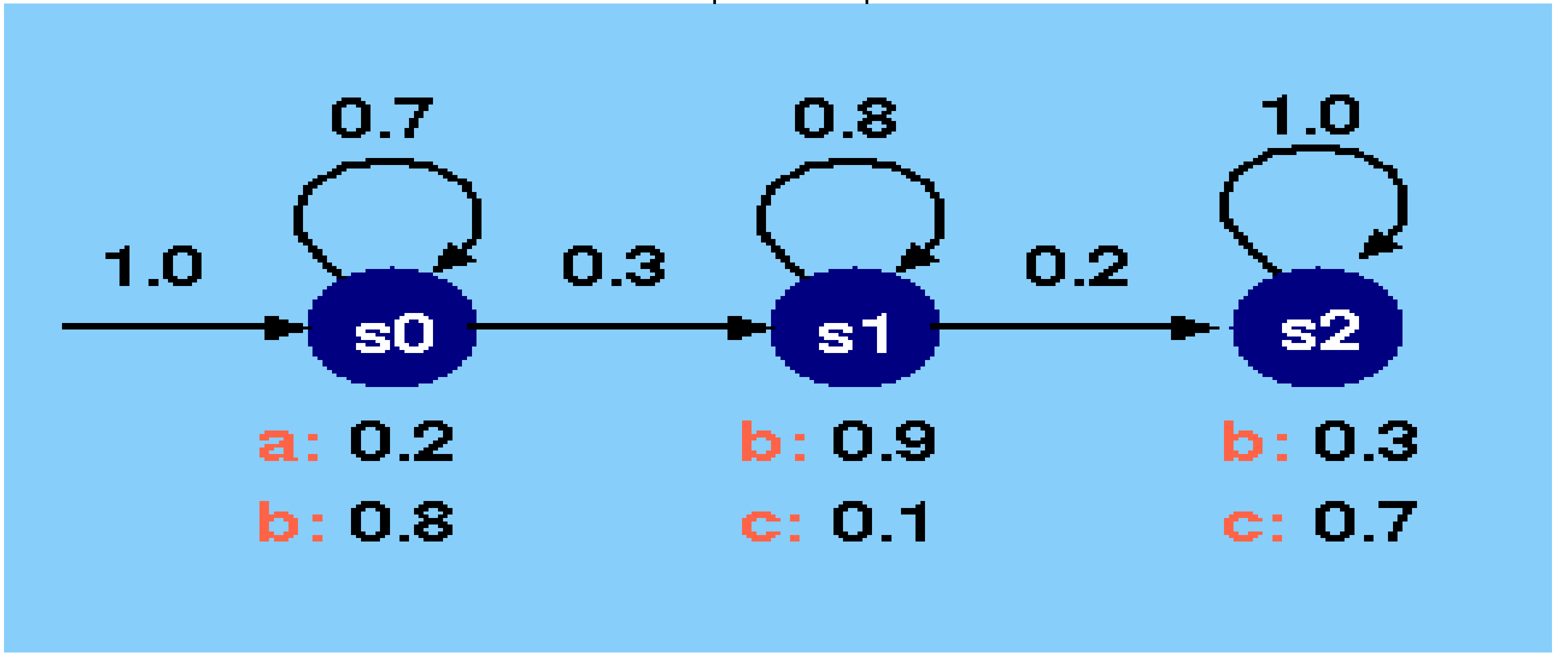
$t_{ij} = P(q_j \mid q_i)$

$\Sigma_j\, t_{ij} = 1$

Probabilidades de **emissões**

$e_i\,(a) = P(a \mid q_i)$

$\Sigma_{a \in \Sigma}\, e_i(a) = 1$

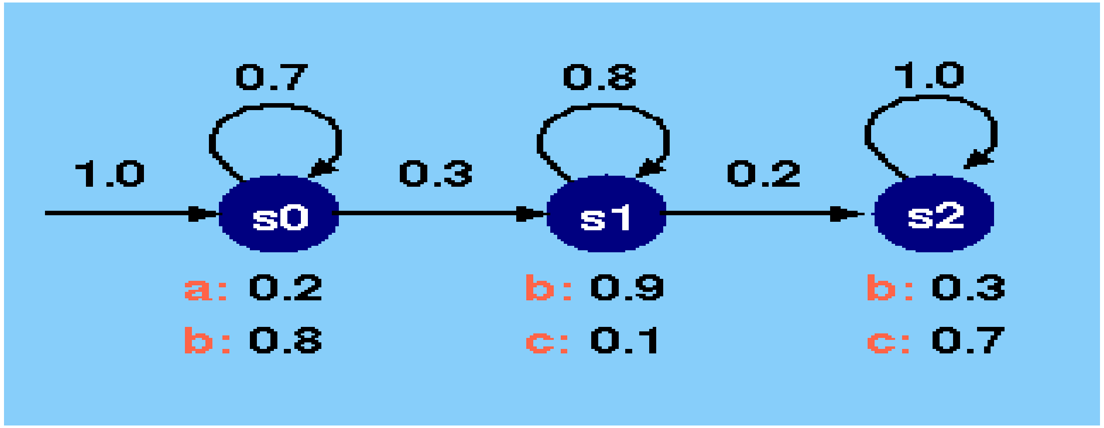


Semelhança com algo?

Modelo Oculto de Markov
Hidden Markov Model (HMM)

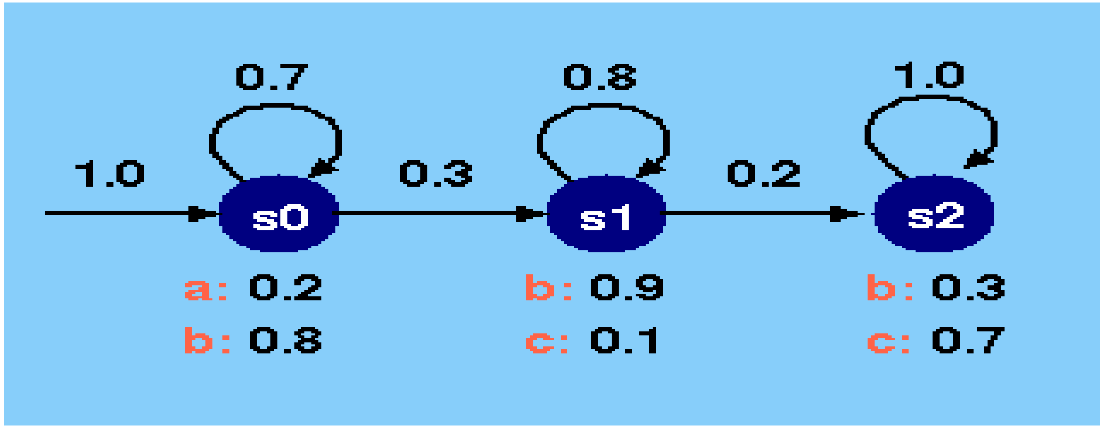
- Autômatos finitos (não determinísticos) probabilísticos são equivalentes a modelos ocultos de Markov

Modelo Oculto de Markov
Hidden Markov Model (HMM)



- Como transformo essa HMM em um AFN probabilístico?

Modelo Oculto de Markov
Hidden Markov Model (HMM)



- Como transformo essa HMM em um AFN probabilístico?

P_{AFN}(transição do estado qi para o estado qj lendo o símbolo k) =

$P_{AFN}(qi, qj, k) = t_{ij} * e_i(k)$

Máquinas de estados

- Máquinas de Mealy: aceitam símbolos na transição (autômatos tradicionais)

- Máquinas de Moore: aceitam símbolos nos estados (HMM – modelo oculto de Markóv)

Problemas relacionados a HMM

1) Dados um HMM e uma cadeia, calcular a probabilidade dessa cadeia
   - Soma das probabilidades de cada caminho
   - Probabilidade de cada caminho: produto das probabilidades do caminho (transição e emissão)
   - Algoritmo forward ou backword

Problemas relacionados a HMM

2) Dados um HMM e uma cadeia, calcular o caminho mais provável dessa cadeia

- Algoritmo viterbi

3) Dados um HMM e conjunto de cadeias (treinamento), estimar os parâmetros (probabilidades de emissão e transição)

- Algoritmo Baum-Welch

Problemas relacionados a HMM

4) Projetar a topologia de uma HMM

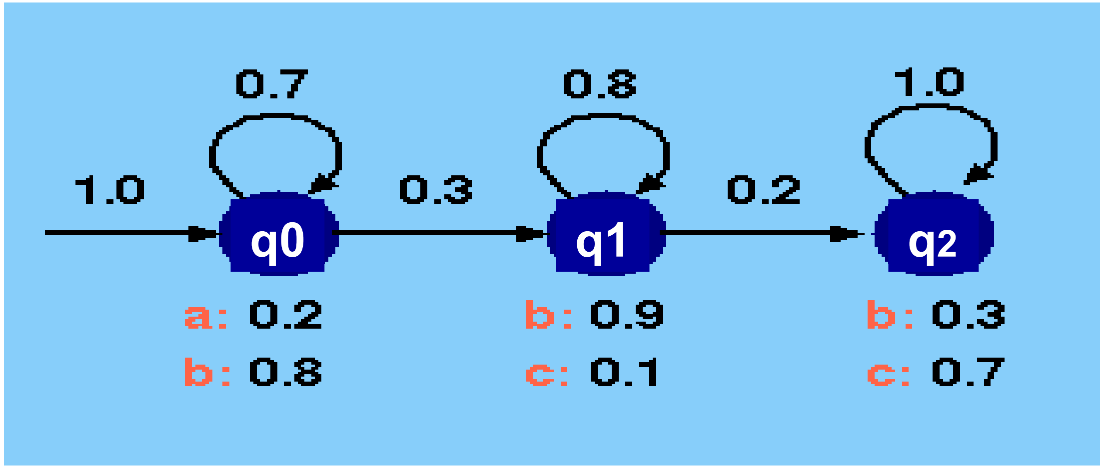
Problemas relacionados a HMM (hoje)

1) **Dados um HMM e uma cadeia, calcular a probabilidade dessa cadeia**

2) **Dados um HMM e uma cadeia, calcular o caminho mais provável dessa cadeia**

Como calcular P(x | HMM)

- Uma cadeia x pode percorrer diversos caminhos por um HMM



Ex: **aabbb**
Caminhos possíveis (c):

c1 = q0, q0, q0, q0, q0
c2 = q0, q0, q0, q0, q1
c3 = q0, q0, q0, q1, q1
c4 = q0, q0, q1, q1, q1
c5 = q0, q0, q1, q1, q2
c6 = q0, q0, q1, q2, q2

- Probabilidade conjunta de x e um caminho específico c:

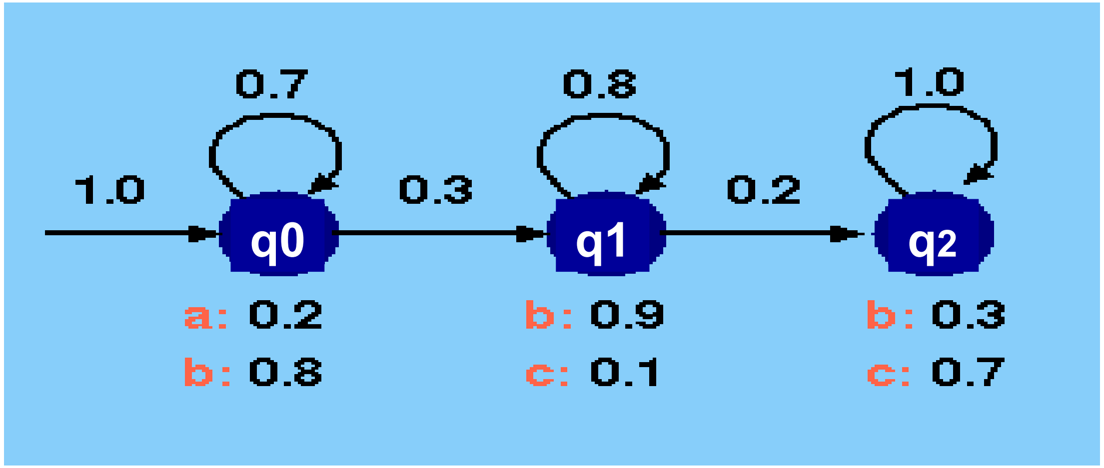
$P(x,c|HMM) = P(x=x_1x_2...x_n, c=r_1r_2...r_n |HMM) = \pi(r_1) * e_{r1}(x_1) * \Pi_{i=2..n}\ t_{(ri-1)ri} * e_{ri}(x_i)$

Ex: $P(x=aabbb, c6=q_0q_0q_1q_2q_2) = \pi(q_0)*e_{q0}(a) * t_{q0q0} *e_{q0}(a) * t_{q0q1} *e_{q1}(b)* t_{q1q2} *e_{q2}(b)* t_{q2q2} * e_{q2}(b)$

$= 1 * 0.2 * 0.7 * 0.2 * 0.3 * 0.9 * 0.2 * 0.3 * 1 * 0.3$ = **0,0001**

Como calcular P(x | HMM)

- Uma cadeia x pode percorrer diversos caminhos por um HMM



Ex: **aabbb**

Caminhos possíveis (c):

c1 = q0, q0, q0, q0, q0
c2 = q0, q0, q0, q0, q1
c3 = q0, q0, q0, q1, q1
c4 = q0, q0, q1, q1, q1
c5 = q0, q0, q1, q1, q2
c6 = q0, q0, q1, q2, q2

- Probabilidade conjunta de x e um caminho específico c:

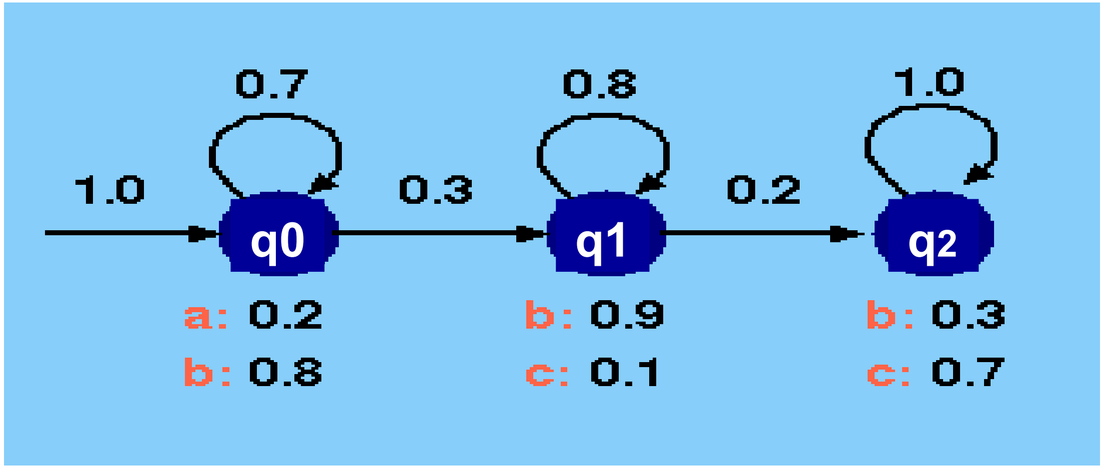
$$P(x,c|HMM) = P(x=x_1x_2...x_n, c=r_1r_2...r_n\,|HMM) = \pi(r_1) * e_{r1}(x_1) * \Pi_{i=2..n}\ t_{(ri-1)ri} * e_{ri}(x_i)$$

- Probabilidade de x:

$$P(x\ |\ HMM) = \Sigma_c\ P(x,c\ |HMM)$$

Como calcular P(x | HMM)

- Uma cadeia x pode percorrer diversos caminhos por um HMM



Ex: **aabbb**
Caminhos possíveis (c):

c1 = q0, q0, q0, q0, q0
c2 = q0, q0, q0, q0, q1
c3 = q0, q0, q0, q1, q1
c4 = q0, q0, q1, q1, q1
c5 = q0, q0, q1, q1, q2
c6 = q0, q0, q1, q2, q2

- Probabilidade conjunta de x e um caminho específico c:

$$P(x,c|HMM) = P(x=x_1x_2\ldots x_n, c=r_1r_2\ldots r_n\,|HMM) = \pi(r_1) * e_{r1}(x_1) * \Pi_{i=2..n}\, t_{(ri-1)ri} * e_{ri}(x_i)$$

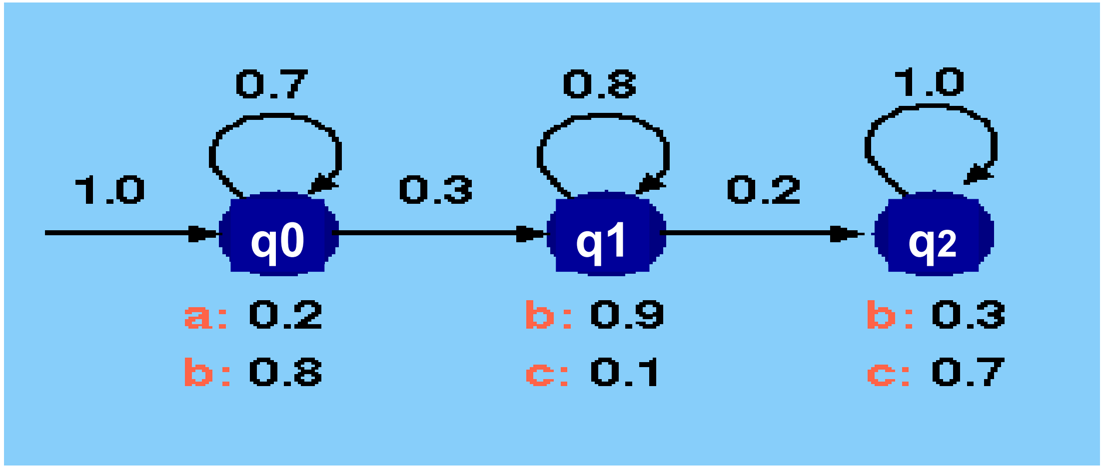
- Probabilidade de x:

$$P(x \mid HMM) = \Sigma_c\, P(x,c\,|HMM)$$

OPS! O número de caminhos cresce exponencialmente com o tamanho da cadeia…

Como calcular P(x | HMM)

- Uma cadeia x pode percorrer diversos caminhos por um HMM



Ex: **aabbb**
Caminhos possíveis (c):

c1 = q0, q0, q0, q0, q0
c2 = q0, q0, q0, q0, q1
c3 = q0, q0, q0, q1, q1
c4 = q0, q0, q1, q1, q1
c5 = q0, q0, q1, q1, q2
c6 = q0, q0, q1, q2, q2

- Probabilidade conjunta de x e um caminho específico c:

$$P(x,c|HMM) = P(x=x_1x_2...x_n,\ c=r_1r_2...r_n\ |HMM) = \pi(r_1) * e_{r1}(x_1) * \Pi_{i=2..n}\ t_{(ri-1)ri} * e_{ri}(x_i)$$

- Probabilidade de x:

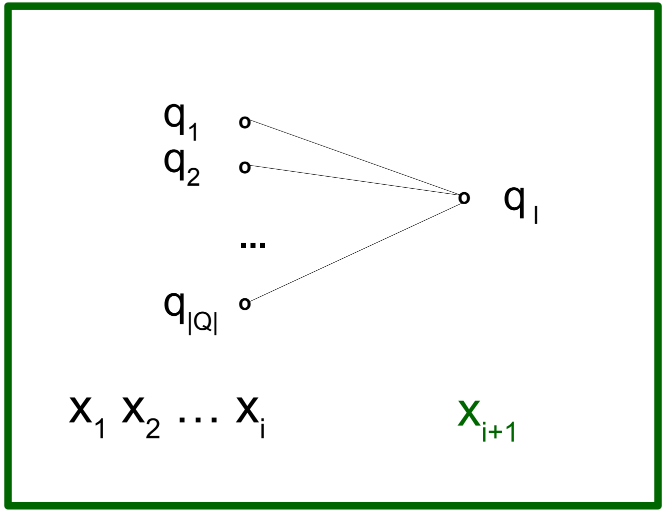
$$P(x \mid HMM) = \Sigma_c\ P(x,c\ |HMM)$$

OPS! O número de caminhos cresce exponencialmente com o tamanho da cadeia…

Solução: Programação Dinâmica!!!

Algoritmo **forward** - P(x | HMM θ)

- Cadeia a ser analisada: $x_1 \ldots x_n$
- Variável $f_k(i) = P(x_1 x_2 \ldots x_i, q_k \mid \theta)$: probabilidade da subcadeia $x_1 \ldots x_i$ e estar no estado q_k no "tempo" i (quando x_i é emitido)
- Lendo mais um símbolo (x_{i+1}) e estar no estado q_l:
  - $f_l(i+1) = e_{ql}(x_{i+1}) * \Sigma_k f_k(i) * t_{qk\ ql}$



- Algoritmo: preenchimento de uma matriz f de dimensões |Q| X n

// Inicialização (emissão do primeiro símbolo no "primeiro" estado)

para k = 1 até |Q| f [k][1] = $\pi(q_k) * e_{qk}(x_1)$ // $f_k(1)$

// Indução:

para i = 1 até n-1

para l = 1 até |Q| f [l][i] = $e_{ql}(x_{i+1}) * \Sigma_k$ f [k][i]* $t_{qk\ ql}$ // $f_l(i+1)$

//Finalização:

$P(x \mid \theta) = \Sigma_{k \in Q}$ f[k][n]

Algoritmo **backward** - P(x | HMM θ)

- Cadeia a ser analisada: $x_1 \ldots x_n$
- Variável $b_k(i) = P(\mathbf{x_{i+1} x_{i+2} \ldots x_n} \mid r_i = q_k, \theta)$: probabilidade da subcadeia $x_{i+1} \ldots x_n$ e estar no estado q_k no "tempo" i (a próxima transição irá para um estado no qual será emitido o símbolo x_{i+1})
- Calculando indo para trás na cadeia (estando no estado q_l):
  - $\mathbf{b}_l\,(\mathbf{i}) = \Sigma_k\, t_{ql\,qk} * e_{qk}(x_{i+1}) * b_k(i+1)$
- Algoritmo: preenchimento de uma matriz f de dimensões |Q| X n

// Inicialização (já li toda a cadeia – inicialização "matemática")

para k = 1 até |Q|　　b[k][n] = 1　　// $b_k(n)$

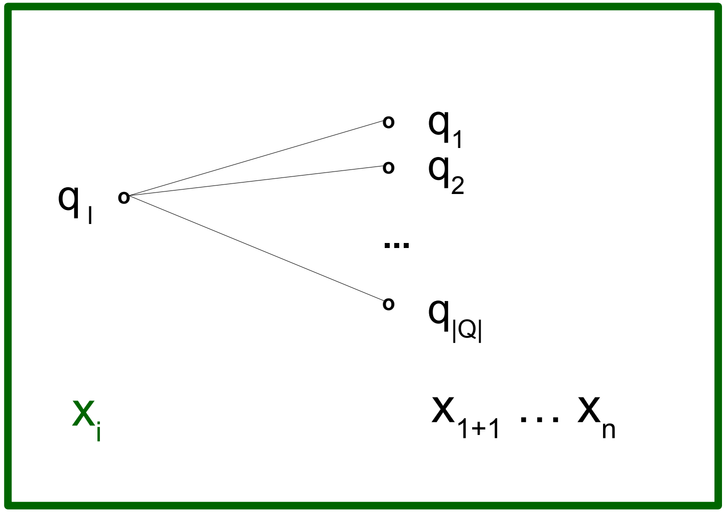
// Indução:

para i = n-1 até 1

　　para l = 1 até |Q|　　$b[l][i] = \Sigma_k\, t_{ql\,qk} * e_{qk}(x_{i+1}) * b[k][i+1]$　　// $b_l(i)$

//Finalização:

$P(x \mid \theta) = \Sigma_{k \in Q}\, \pi(q_k) * e_{qk}(x_1) * b[k][1]$



Problemas relacionados a HMM

2) Dados um HMM e uma cadeia, calcular o caminho mais provável dessa cadeia
   - Algoritmo viterbi

3) Dados um HMM e conjunto de cadeias (treinamento), estimar os parâmetros (probabilidades de emissão e transição)
   - Algoritmo Baum-Welch

Algoritmo **forward** - $P(x \mid \theta)$

- Cadeia a ser analisada: $x_1 \dots x_n$
- Variável $f_k(i) = P(x_1 x_2 \dots x_i, q_k \mid \theta)$: probabilidade da subcadeia $x_1 \dots x_i$ e estar no estado q_k no "tempo" i (quando x_i é emitido)
- Lendo mais um símbolo (x_{i+1}) e estar no estado q_l:
  - $f_l(i+1) = e_{ql}(x_{i+1}) * \Sigma_k f_k(i) * t_{qk\ ql}$
- Algoritmo: preenchimento de uma matriz f de dimensões |Q| X n

// Inicialização (emissão do primeiro símbolo no "primeiro" estado)

para k = 1 até |Q| $\quad$ $f[k][1] = \pi(q_k) * e_{qk}(x_1)$ $\quad$ // $f_k(1)$

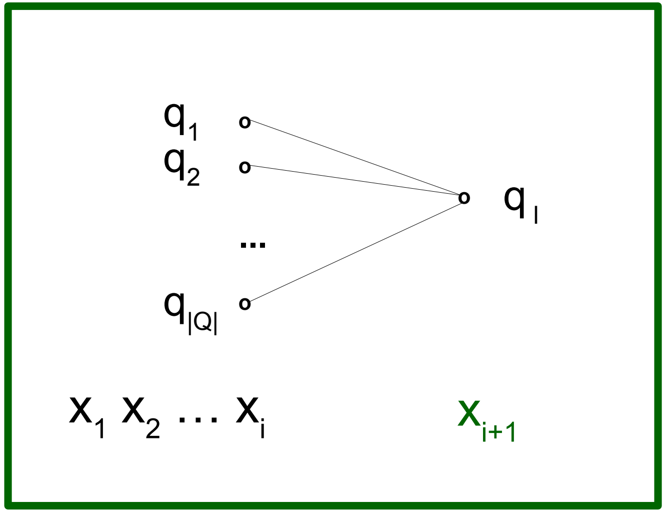
// Indução:

para i = 1 até n-1

para l = 1 até |Q| $\quad$ $f[l][i] = e_{ql}(x_{i+1}) * \Sigma_k f[k][i] * t_{qk\ ql}$ $\quad$ // $f_l(i+1)$

//Finalização:

$P(x \mid \theta) = \Sigma_{k \in Q} f[k][n]$



Algoritmo **Viterbi:** $c = \text{argmax}_c P(x,c \mid \theta)$

- Cadeia a ser analisada: $x_1 \ldots x_n$
- Variável $f_k(i) = P(x_1 x_2 \ldots x_i, q_k \mid \theta)$: probabilidade da subcadeia $x_1 \ldots x_i$ e estar no estado q_k no “tempo” i (quando x_i é emitido)
- Lendo mais um símbolo (x_{i+1}) e estar no estado q_l:
  - $f_l(i+1) = e_{ql}(x_{i+1}) * \Sigma_k f_k(i) * t_{qk\ ql}$
- Algoritmo: preenchimento de uma matriz f de dimensões |Q| X n

// Inicialização (emissão do primeiro símbolo no “primeiro” estado)

para k = 1 até |Q| $\quad f[k][1] = \pi(q_k) * e_{qk}(x_1)$ // $f_k(1)$

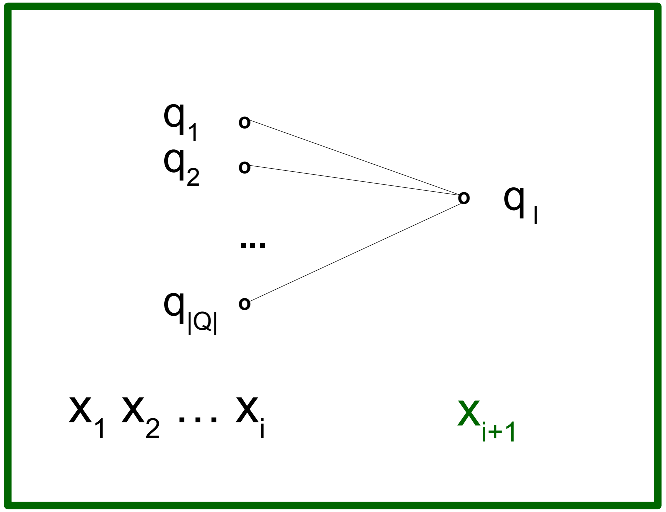
// Indução:

para i = 1 até n-1

para l = 1 até |Q| $\quad f[l][i] = e_{ql}(x_{i+1}) * \Sigma_k f[k][i] * t_{qk\ ql}$ // $f_l(i+1)$

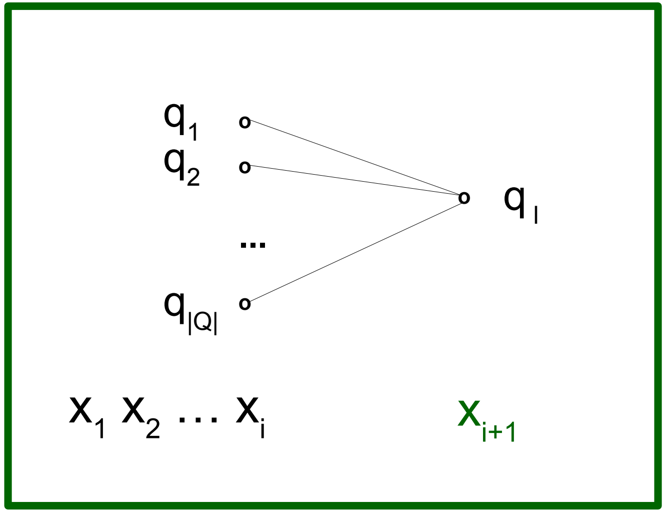
//Finalização:

$P(x \mid \theta) = \Sigma_{k \in Q} f[k][n]$



Algoritmo **Viterbi:** $c = \text{argmax}_c P(x,c \mid \theta)$

- Cadeia a ser analisada: $x_1 \dots x_n$
- Variável $v_k(i) = P(x_1 x_2 \dots x_i,\ q_k\ ,\ c_{max} \mid \theta)$: probabilidade da subcadeia $x_1 \dots x_i$ e estar no estado q_k no "tempo" i (quando x_i é emitido) no caminho mais provável
- Lendo mais um símbolo (x_{i+1}) e estar no estado q_l:
  - $f_l(i+1) = e_{ql}(x_{i+1}) * \Sigma_k f_k(i) * t_{qk\ ql}$
- Algoritmo: preenchimento de uma matriz f de dimensões |Q| X n



// Inicialização (emissão do primeiro símbolo no "primeiro" estado)

para k = 1 até |Q| $\quad f[k][1] = \pi(q_k) * e_{qk}(x_1) \quad // \ f_k(1)$

// Indução:

para i = 1 até n-1

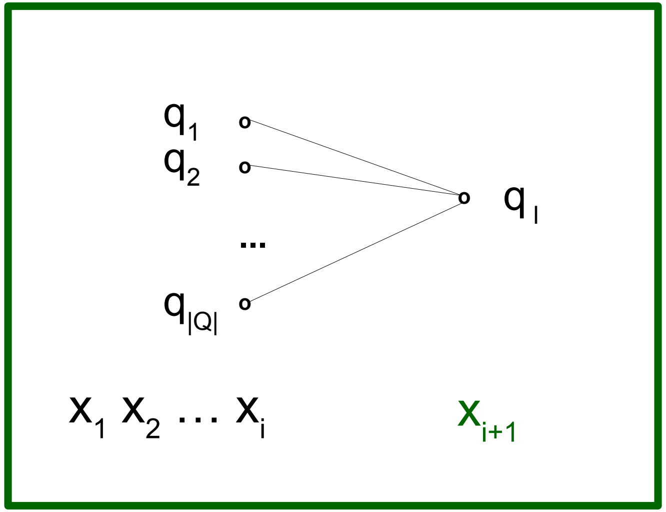
para l = 1 até |Q| $\quad f[l][i] = e_{ql}(x_{i+1}) * \Sigma_k f[k][i] * t_{qk\ ql} \quad // \ f_l(i+1)$

//Finalização:

$P(x \mid \theta) = \Sigma_{k \epsilon Q} f[k][n]$

Algoritmo **Viterbi:** $c = \text{argmax}_c P(x,c \mid \theta)$

- Cadeia a ser analisada: $x_1 \ldots x_n$
- Variável $v_k(i) = P(x_1 x_2 \ldots x_i, q_k, c_{max} \mid \theta)$: probabilidade da subcadeia $x_1 \ldots x_i$ e estar no estado q_k no "tempo" i (quando x_i é emitido) no caminho mais provável
- Lendo mais um símbolo (x_{i+1}) e estar no estado q_l:
  - $v_l(i+1) = e_{ql}(x_{i+1}) * \max_k v_k(i) * t_{qk\, ql}$
- Algoritmo: preenchimento de uma matriz f de dimensões |Q| X n



// Inicialização (emissão do primeiro símbolo no "primeiro" estado)

para k = 1 até |Q| $\quad$ $v[k][1] = \pi(q_k) * e_{qk}(x_1)$ $\quad$ // $b_k(1)$

// Indução:

para i = 1 até n-1

$\quad$ para l = 1 até |Q| $\quad$ $v[l][i] = e_{ql}(x_{i+1}) * \max_k v[k][i] * t_{qk\, ql}$ $\quad$ // $v_l(i+1)$

//Finalização:

$P(x, c_{max} \mid \theta) = \max_{k \in Q} v[k][n]$

Algoritmo **Viterbi:** $c = \text{argmax}_c P(x,c \mid \theta)$

- Cadeia a ser analisada: $x_1 \ldots x_n$
- Variável $v_k(i) = P(x_1 x_2 \ldots x_i, q_k, c_{max} \mid \theta)$: probabilidade da subcadeia $x_1 \ldots x_i$ e estar no estado q_k no "tempo" i (quando x_i é emitido) no caminho mais provável
- Lendo mais um símbolo (x_{i+1}) e estar no estado q_l:
  - $v_l(i+1) = e_{ql}(x_{i+1}) * \max_k v_k(i) * t_{qk\,ql}$
- Algoritmo: preenchimento de uma matriz f de dimensões |Q| X n

// Inicialização (emissão do primeiro símbolo no "primeiro" estado)

para k = 1 até |Q| $\quad v[k][1] = \pi(q_k) * e_{qk}(x_1)$ // $b_k(1)$

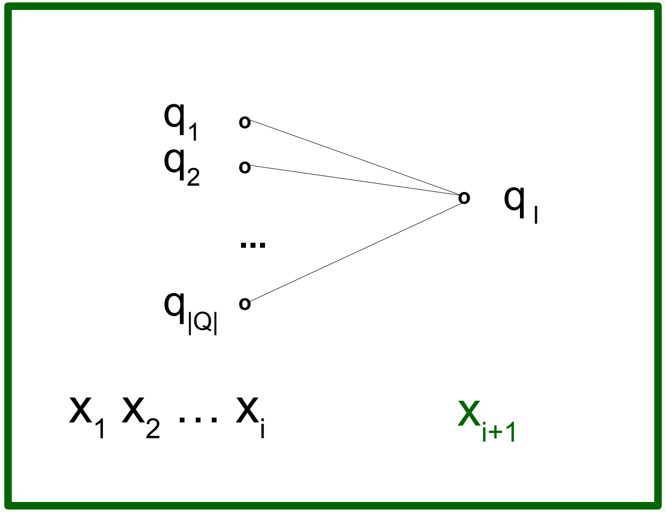
// Indução:

para i = 1 até n-1

para l = 1 até |Q| $\quad v[l][i] = e_{ql}(x_{i+1}) * \max_k v[k][i] * t_{qk\,ql}$ // $v_l(i+1)$

//Finalização:

$P(x, c_{max} \mid \theta) = \max_{k \in Q} v[k][n]$



Para eu conhecer cmax, basta sempre armazenar o argmax_k

Próxima aula (tema 4)

3) Dados um HMM e conjunto de cadeias (treinamento), estimar os parâmetros (probabilidades de emissão e transição)
   - Algoritmo Baum-Welch

4) Topologia

Fim do vídeo 2

HMMs (parte 1)

Professora:
Ariane Machado Lima

Vídeo 3

Score log-odd e posteriori

Professora:
Ariane Machado Lima

Questões que agora aparecem

1) Quando uma probabilidade $P(x|\theta)$ (θ sendo de uma gramática G ou de uma HMM) é suficiente para eu dizer que x deveria ser classificada como sendo da classe representada por θ? (limiar de classificação)

2) Como eu avalio θ e esse limiar de classificação?

Questões que agora aparecem

1) Quando uma probabilidade $P(x|\theta)$ (θ sendo de uma gramática G ou de uma HMM) é suficiente para eu dizer que x deveria ser classificada como sendo da classe representada por θ? (limiar de classificação)

2) Como eu avalio θ e esse limiar de classificação?

Vou chamar esse θ de G no restante dessa aula...

Quando uma probabilidade P(x |G) é suficiente?

- Algumas alternativas:
  - Testar vários limiares (avaliar **medidas de desempenho** para cada um e escolher o mais adequado) – que precisa da reposta da questão nr 2!
  - Razão log-odd

Razão log-odd

- Em um contexto de classificação binária (duas classes $\rightarrow$ duas gramáticas G_1 e G_2) :
- Classifico como sendo da classe 1 se:

$$P(x|G_1) > P(x|G_2)$$

$$\Rightarrow P(x|G_1) / P(x|G_2) > 1$$ (Razão de verossimilhança)

$$\Rightarrow \log P(x|G_1) - \log P(x|G_2) > 0$$ **(log-odd)**

Razão log-odd (com modelo nulo)

- Comparação de G com um modelo N bem mais simples, geral, aleatório (nulo)
- $P(x|G) > P(x|N)$ se x realmente é “da classe de G”
- => $P(x|G) / P(x|N) > 1$
- **Score log-odd $S^G(x) = \log P(x|G) - \log P(x|N) > 0$**

Limiar de classificação baseado no log-odd score

Razão log-odd (com modelo nulo)

- Comparação de G com um modelo N bem mais simples, geral, aleatório (nulo)
- $P(x|G) > P(x|N)$ se x realmente é "da classe de G"
- $\Rightarrow P(x|G) / P(x|N) > 1$
- **Score log-odd $S^G(x) = \log P(x|G) - \log P(x|N) > 0$**

Limiar de classificação baseado no log-odd score
Quanto maior o limiar, mais CONSERVADORA é a classificação

Modelo nulo

- Geralmente posição independente
- Exemplos:
  - Distribuição uniforme sobre os símbolos
    - Ex: P(a) = P(c) = P(g) = P(t) = 0.25

      $P(x|N) = 0.25^{|x|}$
  - Distribuição de background
    - Ex: frequência de cada nucleotídeo no genoma sendo analisado
    - Ex: frequência de pixels 0, 1, no banco de imagens PB sendo analisado

(MACHADO-LIMA, 2010) para mais exemplos e análises

Classificação Bayesiana

- Em um contexto de classificação binária (duas classes $\rightarrow$ duas gramáticas G_1 e G_2) :
- Classifico como sendo da classe 1 se:

$$P(G_1 | x) > P(G_2 | x)$$ POSTERIORI

$$\Rightarrow P(x|G_1)\, P(G_1) > P(x|G_2)\, P(G_2)$$

Classificação Bayesiana

- Em um contexto de classificação binária (duas classes $\rightarrow$ duas gramáticas G_1 e G_2) :
- Classifico como sendo da classe 1 se:

$$P(G_1 | x) > P(G_2 | x)$$

POSTERIORI

$$\Rightarrow P(x|G_1)\, P(G_1) > P(x|G_2)\, P(G_2)$$

- Assistam os vídeos sobre classificação Bayesiana:
    - Vídeo 1: revisão de probabilidade
    - Vídeo 2: O que isso tem a ver com classificação
    - Vídeo 3: Generalizando (Teoria da Decisão)

Fim do vídeo 3

Score log-odd e posteriori

Professora:
Ariane Machado Lima

Referências

DURBIN, R.; EDDY, S. R.; KROGH, A. **Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids.** Cambridge University Press, 2002. Cap 3 a 6

MACHADO-LIMA, A.; KASHIWABARA, A. Y.; DURHAM, A. M. Decreasing the number of false positives in sequence classification. **BMC Genomics** 11 (Suppl 5):S10, 2010.

RABINER, L. R. A tutorial on hidden Markov models and selected applications in speech recognition. **Proceedings of the IEEE**, v. 77, n. 2, p. 257-286 1989