Curs 12.

Modele Markov cu stări invizibile (Hidden Markov Models)

Biblio: cap. 11 din “An introduction to Bioinformatics algorithms”, N.Jones, P. Pevzner

Cap. 3 din “Biological sequence analysis”, R. Durbin at al.

Modele Markov cu stari invizibile

- Motivație
- Modele Markov
- Problema identificării secvențelor CG și problema monedei false
- Structura unui model Markov cu stări invizibile
- Algoritm de decodificare
- Algoritmul Forward-Backward

Motivatie

In problemele studiate în cursurile anterioare s-a utilizat următoarea ipoteză simplificatoare:

- Frecvența de apariție a nucleotidei x pe poziția i este independentă de frecvența de apariție a nucleotidei y pe poziția (i+1):

$$P(s(i)=x,s(i+1)=y)=P(s(i)=x)P(s(i+1)=y)$$

- In realitate evenimentele corespunzătoare unor poziții succesive în secvența ADN nu sunt independente:

$$P(s(i)=x,s(i+1)=y)=P(s(i+1)=y|s(i)=x)P(s(i)=x)$$

- Aceasta dependență poate fi descrisă printr-un model Markov simplu

Modelul Markov simplu

- Se folosește pentru a descrie succesiuni de evenimente (stări) dependente caracterizate prin faptul ca starea de la momentul (i+1) este influențată de starea de la momentul i (distribuția de probabilitate a stării corespunzătoare momentului (i+1) depinde doar de starea s_i):

$$P(s_{i+1}=x_{i+1}|s_i=x_i,s_{i-1}=x_{i-1},\ldots,s_0=x_0)=P(s_{i+1}=x_{i+1}|s_i=x_i)$$

- Probabilitatea să se observe o secvența $x_0x_1\ldots x_L$ este

$$P(x_0x_1\ldots x_L)=P(x_L|x_{L-1})P(x_{L-1}|x_{L-2})\ldots P(x_1|x_0)P(x_0)$$

- Pentru a putea calcula probabilitatea de apariție a unei secvențe este suficient să fie cunoscute probabilitățile de tranziție între oricare două stări (acestea formează așa numita matrice de tranziție).

Modelul Markov simplu

Matrice de tranziție.

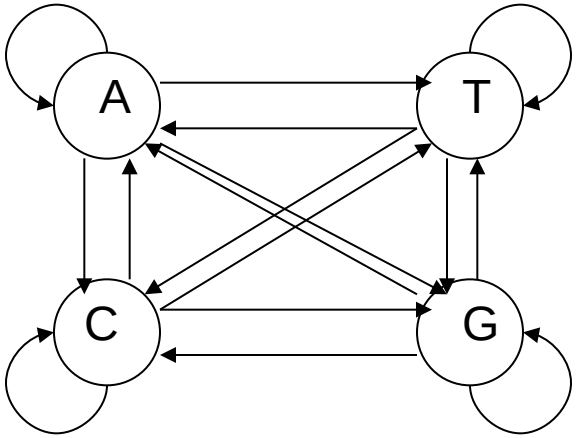
- Se presupune că probabilitatea de tranziție dintr-o stare în alta nu depinde de momentul tranziției (modelul este omogen)
- Intrucât mulțimea stărilor este finită se poate considera ca starea este descrisă prin indicele ei în mulțimea de valori
- Pe linia k și coloana l a matricii de tranziție se află probabilitatea de tranziție din starea k în starea l:

$$a_{kl}=P(s_{i+1}=l|s_i=k) \text{ pentru un i arbitrar}$$

- Suma elementelor de pe fiecare linie a matricii este egală cu 1 (valorile de pe fiecare linie formează o distribuție de probabilitate)
- Daca probabilitatea de tranziție ar depinde de i atunci modelul nu ar fi omogen și nu ar fi suficientă o singură matrice de tranziție

Modelul Markov simplu

Exemplu: Presupunem ca mulțimea stărilor posibile este {A,C,G,T}. Trecerea dintr-o stare în alta poate fi ilustrată printr-o diagramă de stare având arcele etichetate cu probabilități



	A	C	G	T
A	0.180	0.274	0.426	0.120
C	0.171	0.368	0.274	0.188
G	0.161	0.339	0.375	0.125
T	0.079	0.355	0.384	0.182

Matrice de tranziție

Problema insulelor de tip CG

"Insula de tip CG" = secvență în care perechea CG este frecventă

- In cazul în care fiecare nucleotidă are probabilitatea de apariție 1/4, și se consideră că nucleortidele consecutive sunt realizări ale unor variabile aleatoare independente, probabilitatea de apariție a unei perechi de nucleotide este circa 1/16
- In realitate frecvența de apariție a perechilor de nucleotide poate fi mult diferită de 1/16
- De exemplu perechea CG apare rar, frecvența ei fiind mai mica de 1/16. Principala cauză: C se transforma ușor în T prin metilare
- Totuși procesul de transformare a lui C în T este inhibat în vecinătatea genelor (în porțiunile promotoare ale genelor), unde pot apare așa-numitele insule de CG
- Identificarea acestor insule este o problemă importantă. Dificultatea constă în faptul că nu se cunoaște structura secvenței căutate ci doar faptul că distribuția simbolurilor (perechi CG) ce apar este diferită față de restul secvenței

Problema insulelor de tip CG

- Porțiunile în care perechile de tip CG sunt frecvente se caracterizează prin matrici de tranziție diferite de celelalte porțiuni
- Fiecare porțiune se caracterizează printr-un alt model
- Problema: cum se poate identifica care model este mai potrivit ?
- Exemplu: probabilități de tranziție estimate pornind de la 60000 de nucleotide

Insula CG

M1	A	C	G	T
A	0.180	0.274	0.426	0.120
C	0.171	0.368	0.274	0.188
G	0.161	0.339	0.375	0.125
T	0.079	0.355	0.384	0.182

M2	A	C	G	T
A	0.300	0.205	0.285	0.210
C	0.322	0.298	0.078	0.302
G	0.248	0.246	0.298	0.208
T	0.177	0.239	0.292	0.292

Regiune diferită de o insulă CG

Problema insulelor de tip CG

- Cum se poate decide dacă o anumită porțiune corespunde unei insule CG sau nu ?
- Caz particular: se cunoaște lungimea, L, a secvenței corespunzătoare unei insule CG
- Fie x o secvență de nucleotide pentru care se pune problema dacă face parte sau nu dintr-o insulă CG
- Se calculează logaritmul raportului șanselor (log odds ratio) corespunzătoare celor două modele (M1 și M2):

$$S(x) = \log \frac{P(x \mid M1)}{P(x \mid M2)} = \sum_{l=1}^{L} \log \frac{a^1_{x_{l-1}x_l}}{a^2_{x_{l-1}x_l}}$$

Element din matricea de tranziție corespunzătoare modelului M1

Problema insulelor de tip CG

Log odds ratio:

$$S(x) = \log \frac{P(x \mid M1)}{P(x \mid M2)} = \sum_{l=1}^{L} \log \frac{a^1_{x_{l-1}x_l}}{a^2_{x_{l-1}x_l}}$$

- Interpretare: daca S(x)<0 atunci probabilitatea ca x să corespundă modelului M2 este mai mare decât cea aferentă modelului M1; dacă S(x)>0 este mai mare șansa ca x să fi fost generată de modelul M1

- Dificultate: nu se cunoaște lungimea insulei CG și relația acesteia cu lungimea secvenței de analizat astfel că secvența de analizat poate fi constituită din subsecvențe corespunzătoare unor modele diferite.

- Problemă similară: problema monedei false

Problema monedei false

- Problema identificării insulelor CG este similară cu cea a detectării utilizarii unei monede false prin analiza rezultatelor mai multor aruncări

- Se aruncă succesiv o monedă, rezultatul înregistrat fiind cap (H-head) sau pajură (T-Tail)
- Dacă se folosește o monedă corectă atunci probabilitățile de a obține fiecare dintre variante sunt egale cu 0.5

- Daca se folosește monedă falsă atunci probabilitățile sunt diferite (de exemplu se obține cap cu probabilitatea ¾ și pajură cu probabilitatea ¼)
- La fiecare aruncare crupierul poate schimba moneda (inlocuiește moneda corectă cu una falsă sau invers) cu probabilitatea 0.1 – aceasta curespunde cu schimbarea modelului

- Scopul urmărit: analizând rezultatele obținute să se poată decide pentru care aruncări a fost folosită moneda corectă și pentru care aruncări s-a folosit moneda falsă.

Problema monedei false

- **Intrare:** o secvență $x = x_1x_2x_3\ldots x_n$ de rezultate obținute în n aruncări succesive ale unei monede (corecte sau false)

- **Iesire:** o secvența $\pi = \pi_1\ \pi_2\ \pi_3\ldots\ \pi_n$, unde fiecare π_i *este F (fair – moneda corecta) sau B (biased – moneda falsa)*

- **Dificultate:** orice secvență de rezultate poate fi obținută pornind de la oricare dintre secvențele de stări; apare problema identificării secvenței de stări care conduce la cea mai mare probabilitate de a observa secvența de valori obținute

Problema monedei false

Ipoteza 1: nu se schimbă moneda (modelul) de-a lungul secvenței de n aruncări

Moneda corecta

$P(x_1 \ldots x_n|F)=(1/2)^n$

Moneda falsă

$P(x_1 \ldots x_n|B)=(3/4)^k\ (1/4)^{(n-k)}=3^k/4^n$

k=număr situații în care s-a obtinut cap

Obs: Daca $k=n/\log_2 3$ atunci $P(x_1 \ldots x_n|F)= P(x_1 \ldots x_n|B)$ (aceeași probabilitate de observare pentru ambele tipuri de monede)

Pentru a obtine o informatie despre tipul monedei se poate folosi "log odds ratio":

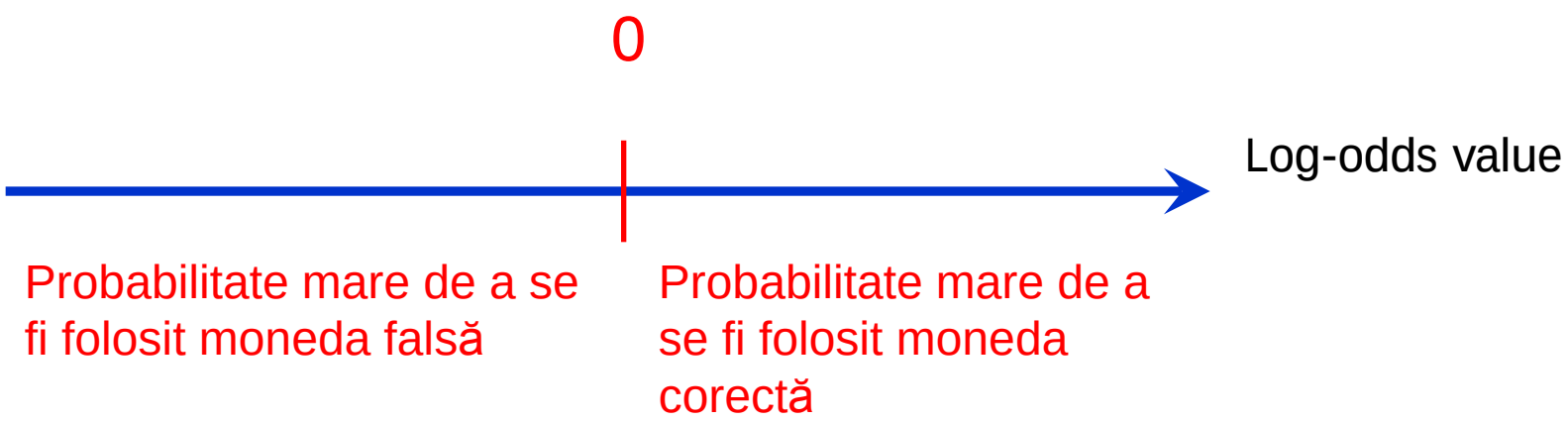
$$\log_2(P(x|F) / P(x|B)) = \Sigma^n_{i=1} \log_2(p(x_i|F)/ p(x_i|B)) = n - k \log_2 3$$

Problema monedei false

Ipoteza 2: se schimbă moneda (modelul) de-a lungul secvenței de n aruncări

Se calculeaza “log-odds ratio” pentru subsecvențe (folosind o “fereastră” mobilă)

$$x_1 x_2 \boxed{x_3 x_4 x_5 x_6 x_7} x_8 \ldots x_n$$



Dezavantaje:

- Lungimea porțiunii corespunzătoare unui model (de exemplu a insulei CG nu este cunoscuta)
- Ferestre de lungimi diferite pot conduce la rezultate diferite pentru aceeași poziție

Model Markov cu stari invizibile

Soluție: utilizarea unui model care să ia în considerare stările ascunse ale sistemului (cele care specifică tipul de monedă utilizat) : model Markov cu stări ascunse (invizibile)

Poate fi interpretat ca fiind un automat abstract cu:

- k stări ascunse (starea curentă este necunoscută)
- un alfabet Σ cu simboluri pe care le emite (simbolul generat este vizibil)

La fiecare etapă de evoluție automatul ia două decizii:

- Care este următoarea stare? (exista probabilități de tranziție între stări)

- Ce simbol trebuie generat? (fiecare stare are asociată o distribuție de probabilitate pentru generarea simbolurilor vizibile)

Model Markov cu stari invizibile

Obs:
- stările sunt invizibile
- sunt vizibile doar simbolurile generate
- în cazul lanțurilor Markov stările sunt vizibile întrucât coincid cu simbolurile observate

Notații și exemplu (problema monedei false):

Σ: mulțimea de simboluri.

Ex.: Σ = {H(ead), T(ail)} (problema monedei false)

Q: mulțimea de stări ascunse

Ex: Q={F(air),B(iased)} (problema monedei false)

Model Markov cu stari invizibile

Parametrii modelului:

A = (a_{kl}): matrice |Q| x |Q| cu probabilitățile de trecere din starea *k* in starea *l.*

$a_{FF} = 0.9$ $a_{FB} = 0.1$
$a_{BF} = 0.1$ $a_{BB} = 0.9$

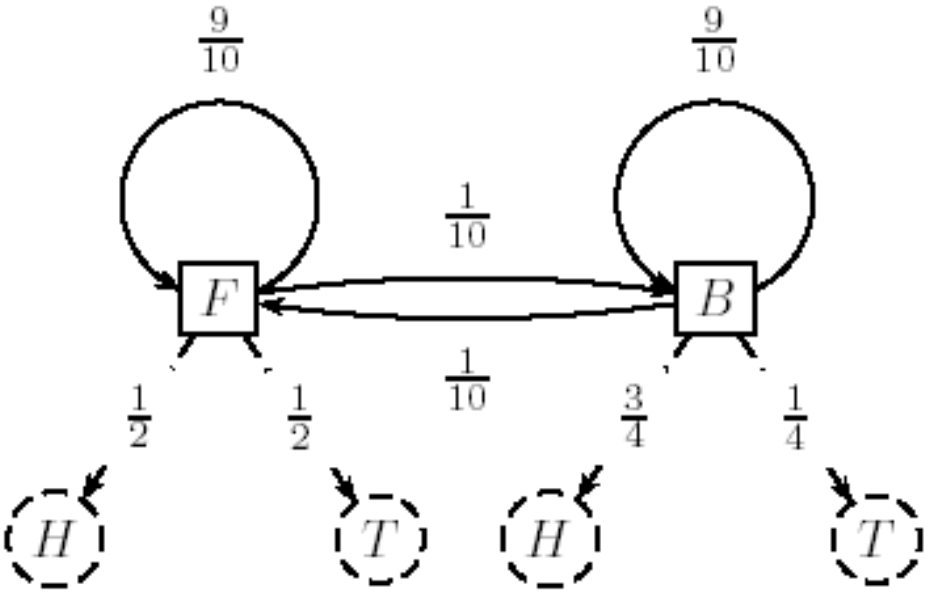
E = ($e_k(b)$): matrice |Q| x |Σ| ce conține probabilități de emitere a unui simbol (b) într-o anumită stare (k)

$e_F(T) = ½$ $e_F(H) = ½$

$e_B(T) = ¼$ $e_B(H) = ¾$

Model Markov cu stari invizibile

Diagrama de stări asociată problemei monedei false



Model Markov cu stari invizibile

- O cale $\pi = \pi_1 \ldots \pi_n$ în model este o secvență de stări
- Exemplu:
  - Cale: π = FFFBBBBBFFF
  - Secvența de simboluri generate: x = 01011101001 (Head=1, Tail=0)

Probabilitatea de producere a simbolului x_i din starea π_i

x =	0	1	0	1	1	1	0	1	0	0	1
π =	F	F	F	B	B	B	B	B	F	F	F
$P(x_i\|\pi_i)$ =	½	½	½	¾	¾	¾	¼	¾	½	½	½
$P(\pi_{i-1} \rightarrow \pi_i)$ =	½	9/10	9/10	1/10	9/10	9/10	9/10	9/10	1/10	9/10	9/10

Probabilitatea de tranziție din starea π_{i-1} în starea π_i

Model Markov cu stari invizibile

Probleme:

1. Determinarea celei mai probabile secvențe de stări (calea) care a condus la generarea unei secvențe de simboluri (problema decodificării)

2. Determinarea distribuției de probabilitate a stărilor corespunzătoare observării unui anumit simbol

Model Markov cu stari invizibile

- P(x|π): Probabilitatea ca secvența x să fi fost generată de calea π:

Scop: fiind dată succesiunea simbolurilor produse se caută secvența de stări (calea) optimală (de probabilitate maximă).

Input: Secvența de simboluri $x = x_1 \ldots x_n$ generate de un model $M(\Sigma, Q, A, E)$

Output: O cale π care maximizează $P(x|\pi)$

Determinarea căii = decodificarea modelului

Probabilitatea de trecere din starea π_i in starea π_{i+1}

$$P(x \mid \pi) = a_{o\pi_1} \prod_{i=1}^{n} e_{\pi_i}(x_i) a_{\pi_i \pi_{i+1}}$$

Probabilitatea stării inițiale

Probabilitatea de generare a simbolului x_i când sistemul e în starea π_i

Algoritm de decodificare a modelului

Algoritmul Viterbi (1969)

Idee de bază: determinarea căii optimale este similară cu identificarea unei căi in problema “turistului in Manhattan”

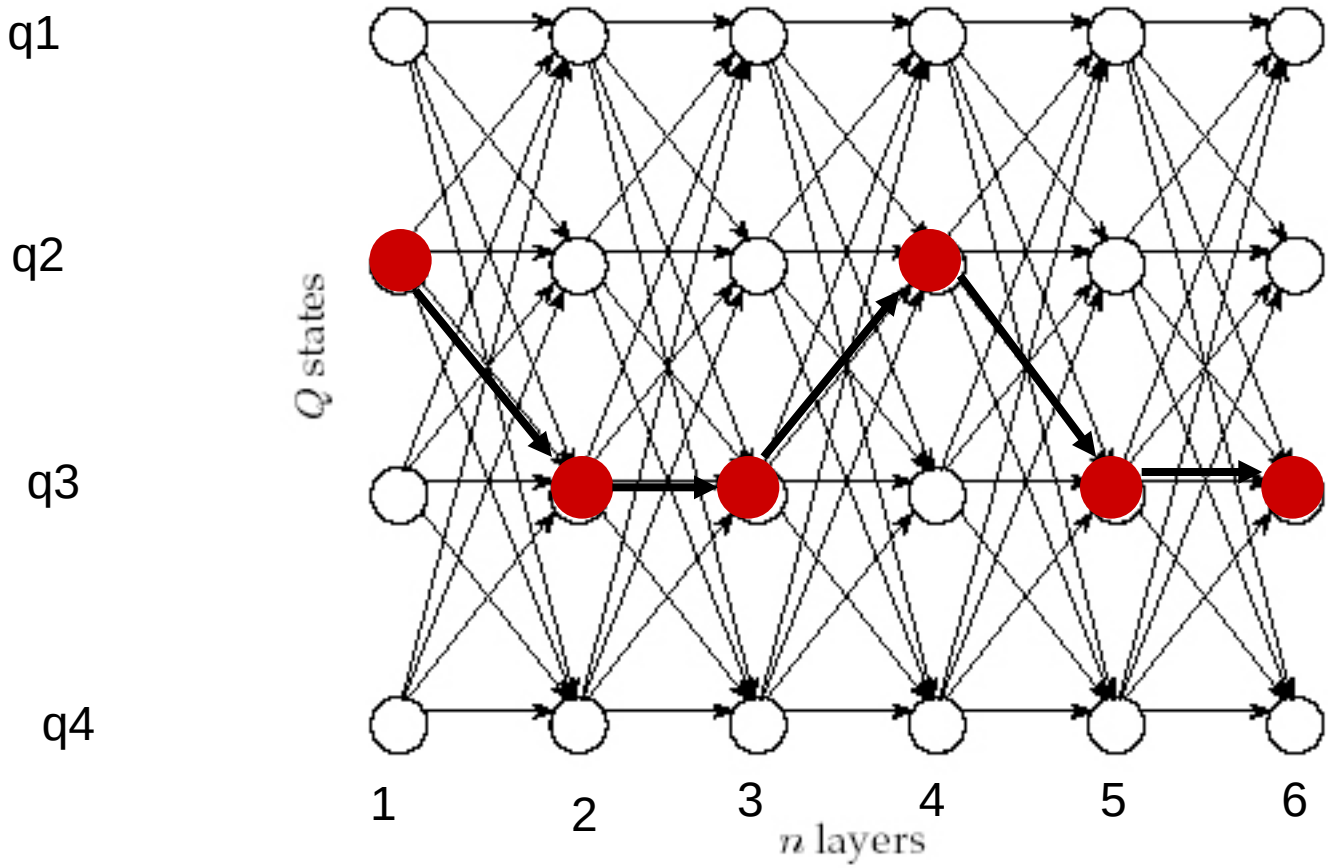
Graf cu |Q|n noduri (corespunzătoare stărilor fiecăreia dintre cele n etape) $|Q|^{2(n-1)}$ muchii (corespunzătoare tranzițiilor dintr-o stare în alta: este posibilă trecerea între oricare două stări iar din fiecare stare este posibilă trecerea către fiecare dintre simbolurile de ieșire)

Fiecare secvență de stări $\pi = \pi_1 \ldots \pi_n$ corespunde unui drum in graf

Direcțiile de deplasare validă în graf: către dreapta (orizontal sau pe diagonală)

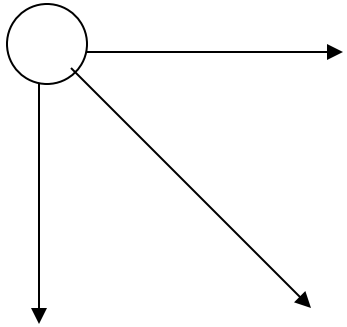
Algoritmul Viterbi

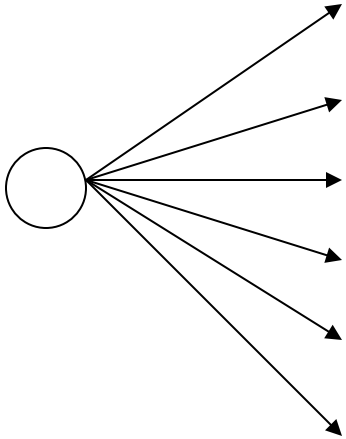
- Structura grafului



Algoritmul Viterbi

- Direcții de deplasare în graf:





- Varianta clasică a problemei turistului (folosită la problema alinierii)
- Identificarea căii în modelul Markov cu stări ascunse

Algoritmul Viterbi

Fiecare cale în graf are asociat un scor reprezentat de probabilitatea $P(x|\pi)$ (probabilitatea sa se fi observat secventa x la parcurgerea căii π).

Algoritmul Viterbi identifică calea ce maximizează $P(x|\pi)$ folosind principiul programării dinamice

Este necesară stabilirea unui scor pentru fiecare tranziție din (k,i) în (l,i+1). Acest scor este produsul dintre probabilitatea de trecere din starea k in starea l și probabilitatea de a emite simbolul x_{i+1} atunci când sistemul se afla în starea l:

$$w_{ki,l(i+1)}=e_l(x_{i+1}).\ a_{kl}$$

Algoritmul Viterbi

Relația de recurență corespunzătoare problemei generice (în care se ajunge în starea l la etapa i+1):

$s_{l,i+1} = \max_{k \in Q} \{s_{k,i} \cdot w_{ki,l(i+1)}\} =$

$$\max_{k \in Q} \{s_{k,i} \cdot a_{kl} \cdot e_l(x_{i+1})\} = e_l(x_{i+1}) \cdot \max_{k \in Q} \{s_{k,i} \cdot a_{kl}\}$$

Cazuri particulare:

$s_{begin,0} = 1$

$s_{k,0} = 0$ for $k \neq begin$.

(begin este o stare fictivă corespunzătoare momentului inițial)

Obs: pe prima coloană se va afla valoarea 1 doar pe linia corespunzătoare stării fictive

In implementari se folosește varianta logaritmată (S=log (s)):

$$S_{l,i+1} = \log(e_l(x_{i+1})) + \max_{k \in Q} \{S_{k,i} + \log(a_{kl})\},$$

Algoritmul Viterbi

Construirea căii:

- se identifică maximul aflat pe ultima coloană (indicele de linie al maximului indică starea finală)

- se parcurge matricea de la ultima coloană către prima alegând la fiecare etapă linia corespunzătoare (cea pe care se află valoarea ce determină obținerea maximului)

Obs. Pentru a ușura construirea căii se poate reține indicele de linie al elementului pentru care se realizează maximul pe fiecare coloană.

Algoritmul Forward-Backward

Problema: determinarea probabilității unei stări atunci când a fost observată o anumită secvență de simboluri ($P(\pi_i = k|x)$)

Intrare: secvența de simboluri produse de un model Markov (ex: succesiune de rezultate obținute prin aruncarea monedei).

Iesire: probabilitatea ca sistemul să se fi aflat într-o anumită stare la un moment dat (de exemplu, probabilitatea ca la o anumită aruncare să se fi utilizat o moneda falsă)

Algoritmul Forward-Backward

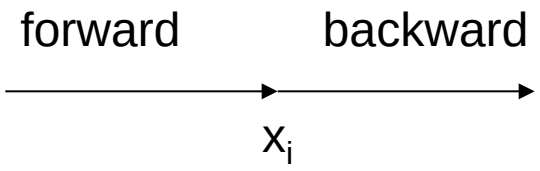
Notatii:

$f_{k,i}$ (*forward probability*) = probabilitatea de a se fi generat secvența de simboluri $x_1 \ldots x_i$ și de a se fi atins starea $\pi_i = k$.

$b_{k,i}$ (*backward probability*) = probabilitatea de a se fi generat secvența de simboluri $x_{i+1} \ldots x_n$ după ce a fost atinsă starea $\pi_i = k$.

Probabilitatea ca la momentul i sistemul să fie în starea k

$$P(\pi_i = k \mid x) = \frac{P(x, \pi_i = k)}{P(x)} = \frac{f_{k,i} b_{k,i}}{P(x)}$$



Algoritmul Forward-Backward

- Relația de recurență pentru etapa Forward

$$f_{k,i} = e_k(x_i)\sum_{l \in Q} f_{l,i-1}a_{lk}$$

$$f_{0,0} = 1, \quad f_{k,0} = 0 \text{ pentru } k \neq 0$$

$$P(x) = \sum_{k \in Q} f_{k,L}a_{k0}$$

- Relația de recurență pentru etapa Backward

$$b_{k,i} = \sum_{l \in Q} e_l(x_{i+1})b_{l,i+1}a_{kl}$$

$$b_{k,L} = a_{k0}$$

Obs: stările fictive (begin și end) sunt ambele notate cu 0

Estimarea parametrilor

- Input: secvența de observații
- Output: matricea de tranziție între stările ascunse (A)

  matricea cu probabilitățile de generare a simbolurilor de ieșire (E)
- Idee: se folosește un set de observații (set de antrenare): $x^1,\ldots,x^m$ și se determină A și E care maximizează:

$$\prod_{i=1}^{m} P(x^i \mid A, E)$$

- Observație: Dacă s-ar cunoaște o secvență de stări corespunzătoare secvenței de simboluri observate atunci s-ar putea estima elementele lui A și E pe baza frecvențelor de tranziție între două stări și a frecvențelor de emitere a simbolurilor

Estimarea parametrilor

- Abordare euristică (antrenare Viterbi)

Pas 1: se pornește de la o secvență aproximativă de stări

Pas 2: se estimează A și E pe baza acestei secvențe de stări și a secvenței de simboluri observate

Pas 3: folosind A și E estimate la pasul anterior și secvența de simboluri observate se determină o nouă aproximare a secvenței de stări (folosind algoritmul Viterbi de decodificare)

Pas 4: se reia de la Pasul 2 până când valorile parametrilor se stabilizează