

Strojno učenje

Markovljevi modeli

Tomislav Šmuc
Svibanj, 2012

Bayesian Learning => Hidden Markov Models

Chris Bishop: Pattern Recognition and Machine Learning

Chapter 12: Sequential Models

L.R. Rabiner: "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition"

Bayesov pristup učenju modela sekvenci

- Markovljevi modeli sekvenci (lanaca)
(Markov chain models)
- Skriveni Markovljevi modeli
(Hidden Markov Models)

Opis sekvence

- Enumerirana stanja (=uočena stanja)
 $1, 2, \dots, N$
- Uočene/mjerene („Observations”) sekvence
 $q_1, q_2, q_3, \dots, q_T$

Markovljev lanac 1. reda

Pretpostavka I:

$$P(q_t = j | q_{t-1} = i, q_{t-2} = k, \dots, q_1 = m) = P(q_t = j | q_{t-1} = i)$$

Pretpostavka II: Stacionarnost

$$P(q_t = j | q_{t-1} = i) = P(q_{t+l} = j | q_{t+l-1} = i)$$

Matrica prijelaza stanja (state transition matrix)

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

Gdje je

$$a_{ij} = P(q_t = j | q_{t-1} = i)$$

Uvjeti na a_{ij} :

$$\begin{aligned} a_{ij} &\geq 0, & \forall i, j \\ \sum_j^N a_{ij} &\geq 0, & \forall i \end{aligned}$$

Primjer:

3 stanja:

sunčano, kišno, oblačno (s,k,o)

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} ss & sk & so \\ ks & kk & ko \\ os & ok & oo \end{bmatrix}$$

Pitanje:

Izračunati vjerojatnost sekvence *sskkos*?

Kolika je vjerojatnost da tjedan dana bude sunčano ?

Bayesovo pravilo

$$P(A, B) = P(A|B) * P(B)$$

Pravilo Markov. Lanca

$$P(\text{opservirana sekvenca}|\text{model}) = \frac{P(\text{opservirana sekvenca})}{P(\text{model})}$$

$$P(\text{opservirana sekvenca}|\text{model}) \sim P(\text{opservirana sekvenca})$$

$$= P(s, s, k, k, o, s)$$

$$P(s, s, k, k, o, s) = P(s)P(s|s)P(s, k, k, o, s)$$

$$P(s, s, k, k, o, s) = P(s)P(s|s)P(s|k)P(k, k, o, s)$$

$$\mathbf{P(s, s, k, k, o, s) = P(s)P(s|s)P(s|k) \dots P(o|s)}$$

$$P(s) = \pi_s - \text{apriorna vjerojatnost}$$

Kolika je **vjerojatnost da tjedan dana** bude sunčano ?
(sekvenca „ostaje” u istom **stanju i** točno **n vremenskih jedinica**)

$$\begin{aligned} p_i(n) &= P(q_1 = i, q_2 = i, \dots, q_n = i, q_{n+1} \neq i) = \\ &= \pi_i (a_{ii})^{n-1} (1 - a_{ii}) \end{aligned}$$

(za $\pi_i = 0.3$, i $a_{ii} = ss = 0.5$; $p_s(7) = ?$)

Koja je očekivana vrijednost trajanja neprekinutih
sunčanih perioda \bar{n} ?

$$\bar{n} = \sum_{n=1}^{\infty} n \cdot p_i(n) = \sum_{n=1}^{\infty} n \cdot (a_{ii})^{n-1} (1 - a_{ii}) = \frac{1}{1 - a_{ii}}$$

Diskretni Markovljevi modeli – svojstva

- **Homogenost** - Vjerojatnosti prijelaza se ne mijenjaju tijekom vremena
- **Ergodičnost** – Ako u bilo kojem momentui možemo prijeći iz bilo kojeg stanja u bilo koje stanje $P_t(a, b) > 0, \forall t, a, b$

Skriveni Markovljevi modeli (HMM)

- Stanja u kojima se sustav nalazi nisu ono što opserviramo
- Opservacije su vjerojatnosne funkcije stanja (uvjetne vjerojatnosti)
- Prijelazi stanja su uvjetne vjerojatnosti

Primjeri:

- Bacanje dvije kocke (jedna namještena, druga ispravna)
- Vađenje kockica različitih boja (C) iz različitih šešira (N)
 - svaki šešir ima drugu distribuciju boja C

Zašto “Hidden Markov Models”?

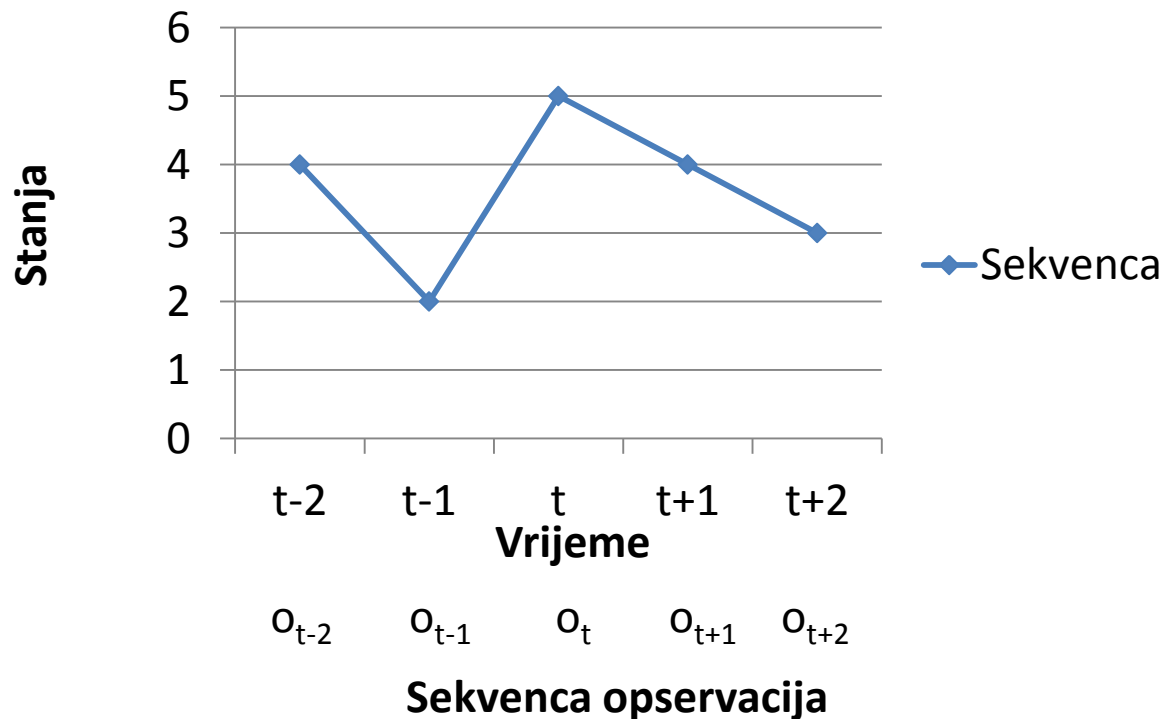
- Promatrač vidi „emitirane simbole” (opservacije) ali ne zna u kojem se stanju HMM trenutno nalazi !
 - Praktični primjeri:
 - Prepoznavanje govora (fonemi=emitirani simboli; stanja=slovo, riječ, pauza)
 - Prepoznavanje gena u DNA (A,G,C,T=emitirani simboli; stanja=exon; intron)
- Osnovni zadatak je zaključiti koja su najvjerojatnija stanja HMM na bazi sekvenc(e)i emitiranih simbola.
 - Maximum likelihood estimation (maksimalna izglednost)

Skriveni Markovljevi modeli (HMM)

Bacanje dvije kocke (jedna namještena, druga ispravna)

1. Odaberi slučajno jednu od dvije kocke (odabir stanja)
2. Baci kocku i upamti broj (opservacija!)
3. Ponavlja (1. i 2.) T puta

Trellis graf - sekvenca



Skriveni Markovljevi modeli (HMM)

Dijelovi HMM:

- S – skup stanja $S = \{1, 2, \dots, N_s\}$
- N_s - broj stanja N_s
- O – skup simbola (opservacije) $O = \{1, 2, \dots, M_o\}$
- A – matrica vjerojatnosti prijelaza stanja

$$a_{ij} = P(q_{t+1} = j | q_t = i) \quad 1 \leq i, j \leq N_s$$

- B – Distribucija vjerojatnosti emisije opservacija

$$b_j(k) = P(o_t = k | q_t = j) \quad 1 \leq k \leq M_o$$

- π – inicijalna vjerojatnost stanja

$$\pi_i = P(q_1 = i) \quad 1 \leq i \leq N_s$$

- θ – oznaka čitavog modela $\theta = (A, B, \pi)$

Osnovni problemi koje želimo rješavati:

1. Dana sekvenca $O = \{o_1, o_2, o_3, \dots, o_T\}$ i model $\Theta = (A, B, \pi)$ - izračunati $P(O | \Theta)$ (Forward alg.)
2. Dana sekvenca $O = \{o_1, o_2, o_3, \dots, o_T\}$ i model $\Theta = (A, B, \pi)$ - izračunaj optimalnu sekvenču skrivenih stanja $q = \{q_1, q_1, q_3, \dots, q_T\}$ (Viterbi alg.)
3. Uz zadan skup sekvenci $O_i = \{o_1, o_2, o_3, \dots, o_T\}_i, i = 1, 2, \dots, S$ pronaći $\Theta = (A, B, \pi)$ model koji maksimizira $P(O | \Theta)$
Baum-Welch alg. (Forward-Backward)

Osnovni problemi koje želimo rješavati:

1. Dana sekvenca $O = \{o_1, o_2, o_3, \dots, o_T\}$ i model $\Theta = (A, B, \pi)$ - izračunati $P(O | \Theta)$
 - Skrivena stanja kompliciraju evaluaciju
 - Računanje izglednosti $P(O | \Theta)$ može pomoći u određivanju najboljeg modela Θ !

- Problem: Izračunati $P(O | \Theta)$
- Algoritam
 - Neka je $q = \{q_1, q_1, q_3, \dots, q_T\}$ sekvenca stanja
 - Pretpostavka – opservacije su nezavisne

$$P(O | q, \Theta) = \prod_{i=1}^T P(o_i | q_i, \Theta) = b_{q_1}(o_1)b_{q_2}(o_2) \dots b_T(o_T)$$

- Vjerojatnost neke sekvence stanja

$$P(q | \Theta) = \prod_{i=1}^T \pi_{q_1} a_{q_1 q_2} a_{q_2 q_3} \dots a_{q_{T-1} q_T}$$

isto tako vrijedi

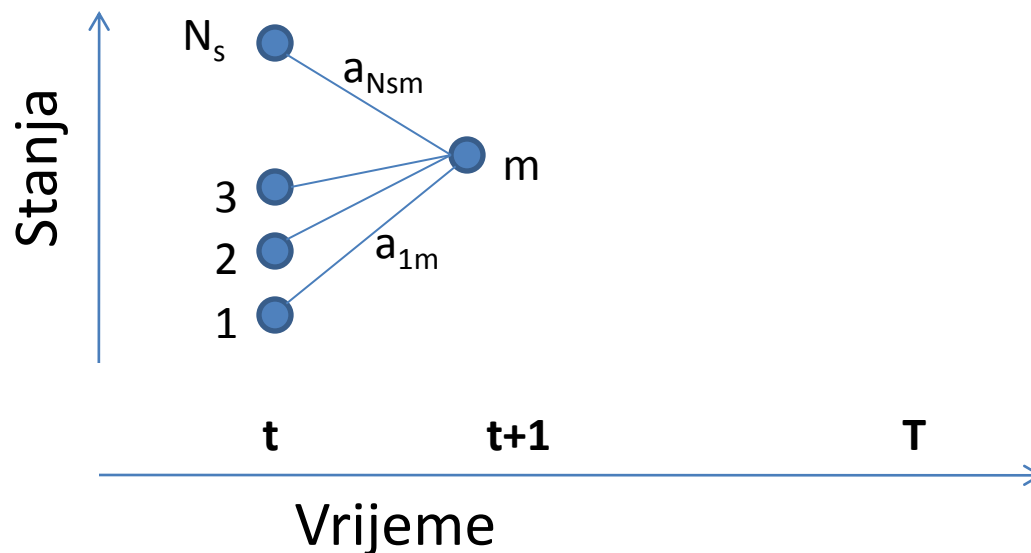
$$P(O, q | \Theta) = P(O | q, \Theta) \cdot P(q | \Theta)$$

- Za sve puteve preko stanja sumiraj vjerojatnosti

$$P(O | \Theta) = \sum_q P(O | q, \Theta) \cdot P(q, \Theta)$$

- Kompleksnost:
 - N_s^T sekvenci stanja i $O(T)$ proračuna $\sim O(TN_s^T)$

Procedura izračuna unaprijed (“Forward algorithm”)



Procedura izračuna unaprijed (“Forward algorithm”)

- Definicija α (forward probability)

$$\alpha_t(i) = P(o_1, o_2, o_3, \dots, o_t, q_t = i \mid \Theta)$$

- $\alpha_t(i)$ - vjerojatnost ostvarivanja parcijalne sekvence $(o_1, o_2, o_3, \dots, o_t)$ uz to da se HMM nalazi u t u stanju $q_t = i$

Algoritam

1 Inicijalizacija $\alpha_1(i) = \pi_i b_i(o_1)$

2 Indukcija

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N_s} \alpha_t(i) \cdot a_{ij} \right] \cdot b_j(o_{t+1})$$

•Kraj

$$P(O \mid \Theta) = \sum_{i=1}^{N_s} \alpha_T(i)$$

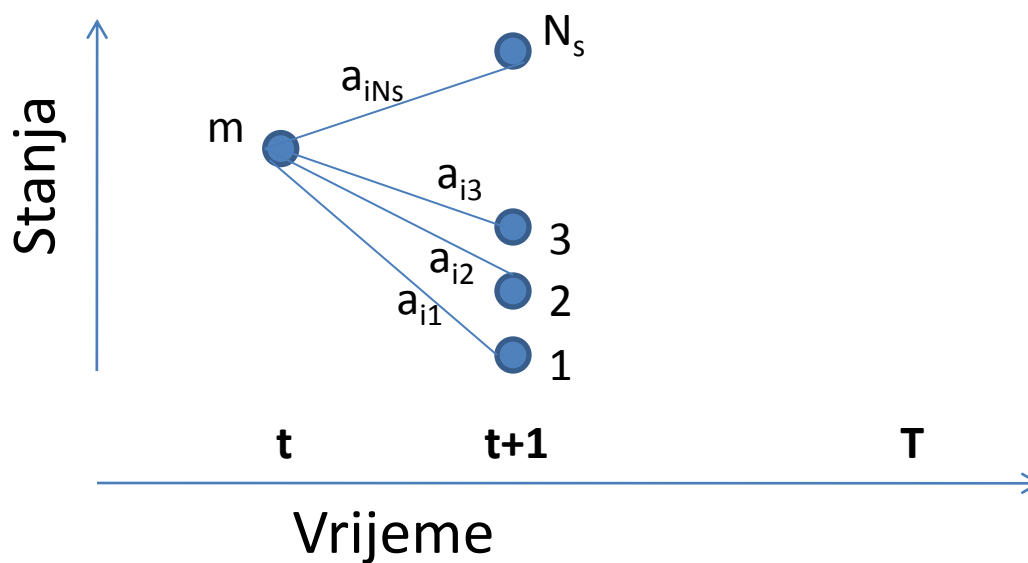
Vježba

$P(O q)$	Stanja	
	Kocka 1	Kocka 2
$P(\text{Glava} \text{Kocka})$	0.5	0.33
$P(\text{Pismo} \text{Kocka})$	0.5	0.67
a_{ij}	Stanja	
	Kocka 1	Kocka 2
Kocka 1	0.5	0.5
Kocka 2	0.5	0.5
Inicijalne vjerojatnosti	Stanja	
	Kocka 1	Kocka 2
Π_i	0.5	0.5

Uočena sekvenca simbola $O=(G,G,G,G,P,P,G,G,P,P,P,P)$

- Koja sekvenca stanja je najvjerojatnija ?
- Koja je združena vjerojatnost $P(O,q | \Theta)$
- Koja je vjerojatnost O ako bacamo samo Kocku 1 odnosno samo Kocku 2 ?

Procedura izračuna unatrag (“Backward algorithm”)



Procedura izračuna unatrag (“Backward algorithm”)

- Definicija β (backward probability)

$$\beta_t(i) = P(o_{t+1}, o_{t+2}, o_{t+3}, \dots, o_T \mid q_t = i, \Theta)$$

- $\beta_t(i)$ je vjerojatnost ostvarivanja parcijalne sekvence $(o_{t+1}, o_{t+2}, o_{t+3}, \dots, o_T)$ uz to da se HMM nalazi u t u stanju $q_t = i$

- Algoritam

1 Inicijalizacija $\beta_T(i) = 1$

2 Indukcija

$$\beta_t(i) = \sum_{j=1}^{N_s} a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)$$

$$1 \leq i \leq N_s,$$

$$t = T-1, T-2, \dots, 1$$

Rješavanje problema 2

Uz zadanu sekvencu $O = \{o_1, o_2, o_3, \dots, o_T\}$ i model $\Theta = (A, B, \pi)$

\Rightarrow izračunaj sekvencu skrivenih stanja $q = \{q_1, q_1, q_3, \dots, q_T\}$
koja maksimizira *izglednost* („Likelihood“)

$$P(q_1, q_1, q_3, \dots, q_T \mid O, \Theta)$$

\Rightarrow Rješavanje - dinamičkim programiranjem

$$\delta_t(i) = \max_{q_1, q_2, \dots, q_{t-1}} P(q_1, q_1, q_3, \dots, q_t = i, o_1, o_2, \dots, o_t \mid \Theta)$$

$\delta_t(i)$ – put s najvećom vjerojatnosti koji završava u stanju i

Indukcijom se dobiva:
$$\delta_{t+1}(i) = \max_j [\delta_t(j) a_{ji}] \cdot b_i(o_{t+1})$$

Viterbi algoritam

- Inicijalizacija $\delta_1(i) = \pi_i b_i(o_1)$
 $\Psi_1(i) = 0$
- Rekurzija $\delta_t(j) = \max_{1 \leq i \leq N_s} [\delta_{t-1}(i) a_{ij}] \cdot b_j(o_t)$
 $\Psi_t(j) = \arg \max_{1 \leq i \leq N_s} [\delta_{t-1}(i) a_{ij}]$
 $2 \leq t \leq T, \quad 1 \leq j \leq N_s$
- Prekid $P^{\max}(q | O, \Theta) = \max_{1 \leq i \leq N_s} [\delta_T(i)]$
 $q_T^{\max} = \arg \max_{1 \leq i \leq N_s} [\delta_T(i)]$
- Izlaz – sekvenca (backtracking)
 $q_t^{\max} = \Psi_{t+1}(q_{t+1}^{\max}) \quad t = T-1, T-2, \dots, 1$

Problem određivanja modela HMM - $\theta = (A, B, \pi)$

- Iterativno rješenje – Baum-Welch algoritam

- 1 Neka je inicijalni model θ_0

- 2 Izračunaj novi θ na osnovu O i θ_0

- 3 Ako je $\log[P(O|\theta)] - \log[P(O|\theta_0)] < \Delta \Rightarrow \text{STOP}$

- 4 Inače $\theta_0 \leftarrow \theta$; vrati se na 2.

Baum-Welch algoritam

Definicije:

- Vjerojatnost da se HMM nalazi u stanju i u trenutku t i u stanju j u trenutku $t+1$:

$$\begin{aligned}\omega_t(i, j) &= \frac{\alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}{P(O | \Theta)} \\ &= \frac{\alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^{N_s} \sum_{j=1}^{N_s} \alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}\end{aligned}$$

- Vjerojatnost da se HMM nalazi u stanju i u trenutku t , uz danu sekvencu O :

$$\gamma_t(i) = \sum_{j=1}^{N_s} \omega_t(i, j)$$

- $\sum_{t=1}^T \gamma_t(i)$ – očekivani broj posjeta stanju i
- $\sum_{t=1}^{T-1} \omega_t(i, j)$ – očekivani broj prijelaza iz stanja i u stanje j

Baum-Welch algoritam

Podsjetnik:

$$\alpha_t(i) = P(o_1, o_2, o_3, \dots, o_t, q_t = i \mid \Theta) \qquad \alpha_{t+1}(j) = \left[\sum_{i=1}^{N_s} \alpha_t(i) \cdot a_{ij} \right] \cdot b_j(o_{t+1})$$

$\alpha_t(i)$ - vjerojatnost ostvarivanja parcijalne sekvence $(o_1, o_2, o_3, \dots, o_t)$ uz to da se HMM nalazi u t u stanju $q_t = i$; (Forward probabilities)

$$\beta_t(i) = P(o_{t+1}, o_{t+2}, o_{t+3}, \dots, o_T \mid q_t = i, \Theta) \qquad \beta_t(i) = \sum_{j=1}^{N_s} a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)$$

$1 \leq i \leq N_s,$
 $t = T-1, T-2, \dots, 1$

$\beta_t(i)$ - vjerojatnost ostvarivanja parcijalne sekvence $(o_{t+1}, o_{t+2}, o_{t+3}, \dots, o_T)$ uz to da se HMM nalazi u t u stanju $q_t = i$; (Backward probabilities)

Baum-Welch algoritam

Definicije (nastavak) => pravila obnavljanja a_{ij} i b_j :

- $\bar{\pi}_i$ – očekivani broj posjeta stanju i ($= \gamma_1(i)$)

- Očekivani # $i \rightarrow j$ / očekivani # $i \rightarrow (j = 1, N_s)$

$$\bar{a}_{ij} = \frac{\sum \omega_t(i, j)}{\sum \gamma_t(i)}$$

- Očekivani # u stanju j uz simbol k / očekivani # u stanju j

$$\bar{b}_j(k) = \frac{\sum_{t, o_t=k} \gamma_t(j)}{\sum_t \gamma_t(j)}$$

Baum-Welch algoritam

(EM - Expectation Maximization algoritam)

1. $\mathbf{O}=\{o_1, o_2, o_3, \dots, o_T\}$
2. Inicijaliziraj $\theta_0(A, B, \pi)$, $k=0$

3. $k=k+1$

4. Uz \mathbf{O} i θ_k izračunaj: (E-step)
 $\gamma_t(i), \quad \omega_t(ij)$

5. Izračunaj:
 $\sum_{t=1}^T \gamma_t(i); \sum_{t=1}^{T-1} \omega_t(i, j)$

6. Izračunaj nove vrijednosti za θ_{k+1} : (M-step)
 $\bar{a}_{ij}, \bar{b}_j, \pi_i$

7. Ukoliko nije postignuta konvergencija => Korak3

E-step (expectation):

Kada imamo procjenu θ_k onda možemo procijeniti očekivane vrijednosti:

- Očekivani broj puta u stanju i
- Očekivani broj prijelaza iz i u j

M-step (maximization):

Ako znamo:

- Očekivani broj puta u stanju i
- Očekivani broj prijelaza iz i u j

Onda možemo izračunati maksimalno vjerojatne vrijednosti (ML - Max Likelihood) – novog θ_{k+1} :

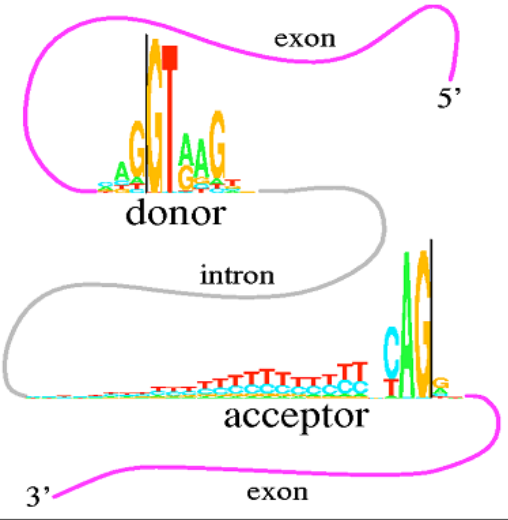
$$\theta_{k+1} = \{\bar{a}_{ij}\}, \{\bar{b}_j\}, \{\pi_i\}$$

Skriveni Markovljevi modeli (HMM)

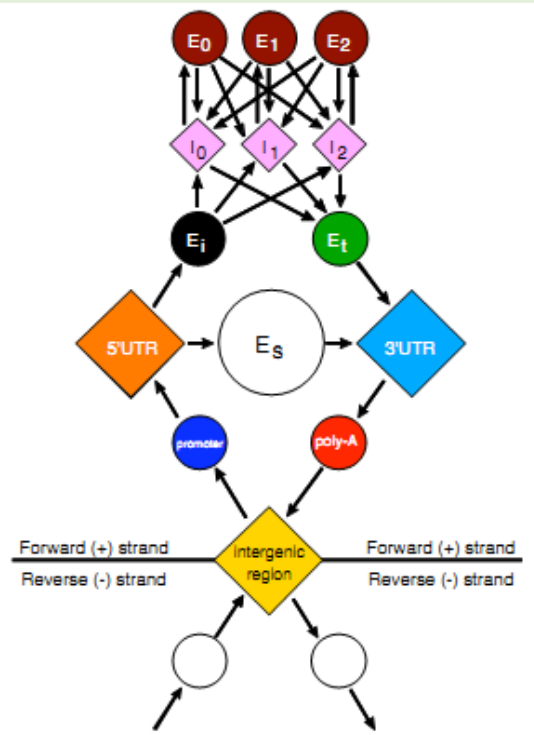
Neki poznati HMM modeli:

GENSCAN

- Traženje gena u DNA



GENSCAN (Burge & Karlin)



62001	AGGACAGGTA CGGCTGTCTAT CACTTAGACC TCACCCCTGTG GAGCCACACC
62051	CTAGGGTTGG CCAATCTACT CCGAGGAGCA GGGAGGSCAG GAGCCAGGGC
62101	TGGGCATAAA AGTCAGGSCA GAGCATCTA TTGCTACAT TTGCTCTGA
62151	CACAACTGTG TTCACTAGCA ACCTCAAACA GACACC
62201	
62251	GGT TGGTAACAAG GTACAAGAC
62301	AGGTTTAAGG AGACCAATAG AAACCTGGCA TGTGGAGACA GAGAAGCTC
62351	TTGGGTTTCT GATAGGCACT GACTCTCTCT GCTATTGGT CTATTTCCG
62401	ACCTTAGGCT TGTGTGTGT CTACCTCTGG ACCGAGAGGT TCTTTGAGTC
62451	CTTTGGGAT CTGTCACTT CTGATGCTT TATGGCAAC CTAAGGTGA
62501	AGGCTCATGG CAAGAAAGTG CTCGGTGCTT TTAGTATGG CCTGGCTCAC
62551	TGGGACAACC TCAAGGGCAC CTTGGCACA CCAAGTGAGC TGCACGTGA
62601	CAAGTGTGAC TTGGATCTGT AGAATCTGAG GGTGAGCTTA TGGGACCTTT
62651	GATGTTTCTT TTCCCTCTCT TTCTATGGT TAAGTTCATG TCATAGGAAG
62701	GGGAGAAGTA ACAGGGTACA GTTTAGAATG GGAACAGAC GAATGATTGC
62751	ATCAGTGTGG AAGTCTCAGG ATCGTTTTCG TTCTCTTAT TTGCTGTCA
62801	TAACAATTGT TTCTTTTGT TTAATCTCTG CTCTTCTTTT TTCTCTCTG
62851	CGCAATTTT ACTATTATAC TTAATGCTT AACATTGTGT ATACCAAAAG
62901	GAAATATCTC TGAGATACAT TAAATACTT AAAAAAAAC TTACACAGT
62951	CTGGCTAGTA CATTACTATT TGGAAATAT GTGTGCTTAT TTGCATATTC
63001	ATAATCTCCC TACTTTATTT TCTTTTATTT TTAATTGATA CATAATCATT
63051	ATACATATTT ATGGGTTAAA GTGTAATGTT TTAATAATG TACACATATT
63101	GACCAATCA GGGTAATTTT GCATTGTGAA TTTTAAAAA TGCTTCTTC
63151	TTTTAATATA CTTTTTGTG TATCTTATTT CTAATACCTT CCTTAATCTC
63201	TTCTTTTCAG GGCAATAATG ATACAATGTA TCATGCTCTT TTGACCAATT
63251	CTAAGAATA ACAGTGATAA TTTCTGGGTT AAGGCAATAG CAATATTCTT
63301	GCATATAAAT ATTTCTGCAT ATAAATGTA ACTGATGTAA GAGGTTTCAT
63351	ATTGCTAATA GCAGCTACAA TCCAGTACAC ATTCTGCTTT TATTTTATGG
63401	TTGGGATAAG CTCGGATTAT TCTGAGTCCA AGCTAGGCCC TTTGCTTAAT
63451	CATGTTTATA CCTCTTATCT TCTCTCCACA GGGGAGGAG AAGTCTCTG
63501	CTGTGTTGCT GTCCTATTAC TTTGGAAAG AATTCAGCCC ACCAGTGCAG
63551	CTGTCTTATC AGAAAGTGTG GCTGTTGTTG SCTAATGCCC TGGCTCAACA
63601	GTATCACTAA GCTCGCTTTC TTGCTGTCCA ATTTCTATTA AAGGTCTCTT
63651	TGTTCCCTAA GTCCACTAC TAAACTGGGG GATATTATGA AGGGCTTGA
63701	GCATCTGGAT TCTGCTTAT AAAAAACATT TATTTTCATT CAAATGATGT