Łańcuchy Markowa i ukryte modele Markowa

Wyspy CpG

W ludzkim genomie zachodzi proces chemiczny, w którym nukleotyd *C* z dinukleotydu CpG ulega metylacji i jest mutowany do nukleotydu *T*. W wyniku tego, w DNA mamy rzadziej do czynienia z dinukleotydami CpG niż wynikałoby to z ogólnej ilości występowania nukleotydów C oraz G. Jednakże z pewnych biologicznych względów, istnieją w genomie krótkie sekwencje, w których powyższy proces nie zachodzi. Owe krótkie sekwencje w DNA noszą nazwę **wysp CpG**.

Problemy:

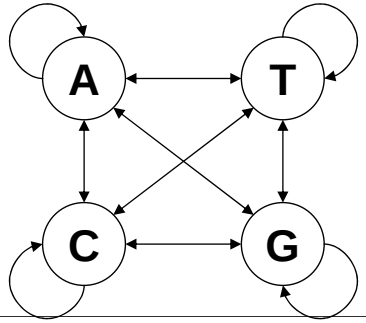
- mając krótki kawałek DNA sprawdzić czy należy on do wyspy CpG,
- mając długi kawałek DNA znaleźć wyspy CpG.

Łańcuchy Markowa

Dobry i prosty model probabilistyczny opisujący problem 1.

Łańcuch markowa: ciąg zdarzeń zdefiniowany na dyskretnej przestrzeni stanów, w którym prawdopodobieństwo wystąpienia każdego zdarzenia zależy tylko od zdarzenia poprzedniego.

Łańcuch Markowa dla DNA może być przedstawiony jako pełny digraf z pętlami:



Z każdym łukiem związane jest prawdopodobieństwo przejścia ze stanu *s* do stanu *t* oznaczane jako a_{st}:

$$a_{st} = P(\pi_i = t \mid \pi_{i-1} = s).$$

Prawdopodobieństwo wystąpienia danej sekwencji π:

$$P(\pi) = P(\pi_1,\pi_2,\ldots,\pi_L) = \ldots = P(\pi_1) \prod_{i=2}^{L} P(\pi_i|\pi_{i-1})$$

Posiadając rozkład prawdopodobieństwa stanu początkowego $P(\pi_1)$ możemy obliczyć prawdopodobieństwo wystąpienia danej sekwencji.

Przykład użycia:
Z grupy ludzkiego genomu wyodrębniono 48 domniemanych wysp CpG oraz przekształcono w dwa modele łańcuchów Markowa:

- Model + dla regionów oznaczonych jako wyspy CpG
- Model – dla reszty sekwencji

Każda komórka w tabeli odpowiada wartości $a_{st} = \frac{c_{st}}{\sum_{t'} c_{st'}}$, gdzie c_{st} jest obserwowaną ilością przypadków, w którym stan *s* poprzedza stan *t* w regionie CpG.

+	A	C	G	T
A	0,180	0,724	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

-	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

Ukryte modele Markowa

Ukryty model Markowa (HMM): zbiór stanów, w którym z każdym z nich skojarzony jest rozkład prawdopodobieństwa emisji symbolu z alfabetu Σ. Stan generuje wartości obserwowane zgodnie z posiadanym rozkładem prawdopodobieństwa. Przejścia pomiędzy stanami odbywają się jak w łańcuchu Markowa.

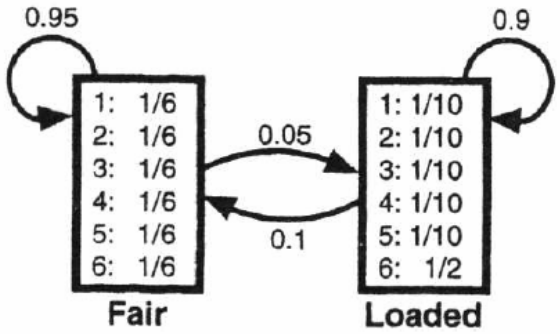
Prawdopodobieństwo przejścia ze stanu *k* do stanu *l*:
$a_{kl} = P(\pi_i = l \mid \pi_{i-1} = k)$, gdzie π oznacza sekwencję stanów.

Stan początkowy oznaczamy jako 0 zaś a_{0k} oznacza prawdopodobieństwo, że sekwencja rozpoczyna się od stanu *k*.

Do każdego stanu jest przypisany parametr: prawdopodobieństwo emisji $e_k(b)$. Dla stanu *k* i symbolu $b \in \Sigma$:

$$e_k(b) = P(x_i = b | \pi_i = k) \text{ więc } \sum_{b \in \Sigma} e_k(b) = 1$$

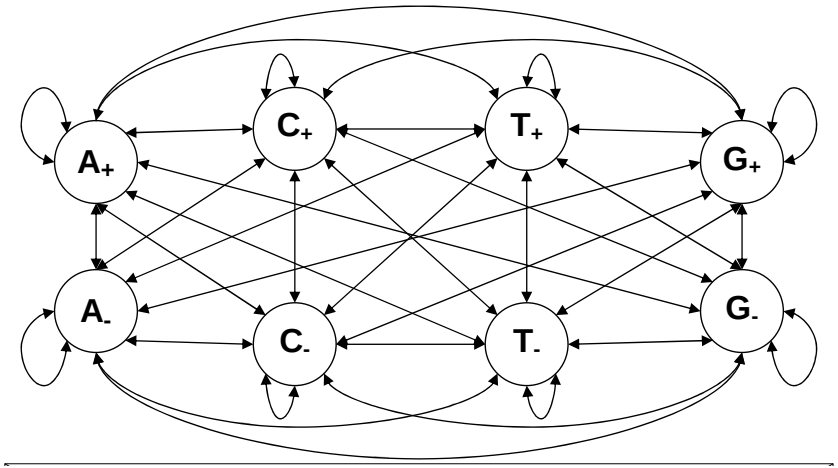
Przykład: nieuczciwe kasyno.
Wyobraźmy sobie kasyno, w którym rzucamy kostką 6-ścienną. W normalnym przypadku prawdopodobieństwo wyrzucenia każdej wartości wynosi 1/6. Jednak w kasynie zwykła kostka jest czasami podmieniana na kostkę oszukaną, a kostka oszukana podmieniana z powrotem na zwykła. Prawdopodobieństwo wyrzucania kostką konkretnych wartości oraz prawd. podmieniania kostek jest przedstawione na poniższym modelu:



Łuki odpowiadają prawd. przejścia a_{kl} a prawd. emisji $e_k(b)$ są umieszczone w prostokątach. Patrząc na sekwencję wyrzuconych oczek (wyemitowane symbole) nie jesteśmy w stanie stwierdzić czy wynik otrzymano rzucając kostką zwykłą czy oszukaną. Stany modelu są ukryte przed obserwatorem.

HMM umożliwia zamodelowanie sekwencji, w której znajdują się zarówno wyspy CpG jak i pozostała część genomu.

W poniższym modelu nukleotydy należące do wysp CpG są oznaczone znakiem ‘+’.



Przyjmujemy $e_{x_\pm}(Y) = 1 \Leftrightarrow X = Y$ dla $X \in \Sigma = \{A, C, T, G\}$ i 0 w przeciwnym wypadku.

Algorytm Viterbi

Korzystając z modelu HMM nie jesteśmy w stanie jednoznacznie określić jaki stan modelu odpowiada obserwowanemu symbolowi. Algorytm Viterbi jest stosowany do znalezienia najbardziej prawdopodobnej sekwencji stanów w danym HMM na podstawie obserwowanego słowa $x \in \Sigma^+$.

Przykład: w modelu CpG sekwencje stanów (C_+, G_+, C_+, G_+), (C_-, G_-, C_-, G_-) oraz (C_+, G_-, C_+, G_-) wygenerują słowo *CGCG*.

Niech $P(x, \pi)$ oznacza łączne prawdopodobieństwo dla słowa $x=x_1 \ldots x_L$ i sekwencji stanów $\pi = \pi_1 \ldots \pi_L$:

$$P(x,\pi) = a_{0\pi_1} \prod_{i=1}^{L-1} e_{\pi_i}(x_i) a_{\pi_i \pi_{i+1}}$$

Obserwując sekwencję *x* możemy spróbować wywnioskować, jaka mogłaby być dla niej sekwencja stanów π. Sekwencja taka może nam wskazać w jakich miejscach występują wyspy CpG. Jako kandydata możemy wybrać ścieżkę dla której $P(x, \pi)$ jest największe:

$$\pi^* = \text{argmax}_\pi P(x, \pi)$$

Najbardziej prawdopodobną ścieżką π^* wyznaczamy rekursywnie. Załóżmy, że prawdopodobieństwo $v_k(i)$ najbardziej prawdopodobnej ścieżki kończącej się stanem $\pi_i=k$, która wygenerowała słowo $x_1 \ldots x_i$ jest znane dla wszystkich stanów *k*. Wówczas prawdopodobieństwo dla stanu *l* i symbolu x_{i+1} możemy zapisać jako:

$$v_l(i+1) = e_l(x_{i+1}) \max_k (v_k(i) a_{kl})$$

Kroki algorytmu:

Inicjalizacja $i = 0$

$v_0(0) = 1$;
$v_k(0) = 0$ dla $k > 0$;

Dla $i = 1,\ldots,L$

$v_l(i) = e_l(x_i) \max_k (v_k(i\text{-}1) a_{kl})$;
$\text{ptr}_i(l) = \text{argmax}_k (v_k(i\text{-}1) a_{kl})$;

Zakończenie

$P(x,\pi^*) = \max_k (v_k(L))$;
$\pi_L^* = \text{argmax}_k (v_k(L))$;

Powrót $i = L,\ldots,1$

$\pi^*_{i-1} = \text{ptr}_i(\pi^*_i)$

Poniższa tabela pokazuje wartości *v* dla sekwencji *CGCG* przy wykorzystaniu modelu dla CpG. Najbardziej prawdopodobna ścieżka jest zaznaczona na szaro.

Stan		C	G	C	G
β	1	0	0	0	0
A_+	0	0	0	0	0
C_+	0	0,13	0	0,012	0
G_+	0	0	0,034	0	0,0032
T_+	0	0	0	0	0
A_-	0	0	0	0	0
C_-	0	0,13	0	0,0026	0
G_-	0	0	0,010	0	0,00021
T_-	0	0	0	0	0

Algorytm „prefiksowy”

Algorytm umożliwia wyznaczenie prawdopodobieństwa emisji sekwencji $P(x)$. Możemy je wyrazić następująco:

$$P(x) = \sum_{\pi} P(x, \pi)$$

Powyższy wzór jest jednak niepraktyczny (liczba możliwych ścieżek π rośnie wykładniczo względem długości sekwencji). Oznaczmy prawdopodobieństwo łączne sekwencji kończącej się na x_i oraz stanu $\pi_i = k$ jako $f_k(i)$:

$$f_k(i) = P(x_1, \ldots, x_i, \pi_i = k)$$

Rekursywny wzór można zapisać następująco:

$$f_l(i+1) = e_l(x_{i+1}) \sum_k f_k(i) a_{kl}$$

Kroki algorytmu:

Inicjalizacja, $i = 0$

$$f_0(0) = 1;$$
$$f_k(0) = 0 \text{ dla } k>0;$$

Dla $i = 1 \ldots L$

$$f_l(i) = e_l(x_i) \sum_k f_k(i\text{-}1) a_{kl};$$

Zakończenie

$$P(x) = \sum_k f_k(L);$$

Algorytm „sufiksowy”

Niech $b_k(i)$ będzie prawdopodobieństwem wyemitowania końcówki $x_{i+1} \ldots x_L$ przez układ, który w chwili *i* był w stanie *k*:

$$b_k(i) = P(x_{i+1} \ldots x_L \mid \pi_i = k)$$

Algorytm działa podobnie do „prefiksowego”.
Kroki algorytmu:
Inicjalizacja $i = L$

$$b_k(L) = 1 \text{ dla wszystkich } k;$$

Dla $i = L\text{-}1,\ldots,1$

$$b_k(i) = \sum_l a_{kl} e_l(x_{i+1}) b_l(i+1)$$

Zakończenie

$$P(x) = \sum_l a_{0l} e_l(x_1) b_l(1)$$

Przejdźmy teraz do obliczenia $P(\pi_i=k|x)$, które jest prawdopodobieństwem aposteriori wystąpienia stanu *k* na pozycji *i* w sekwencji stanów, która wyemitowała słowo $x \in \Sigma^+$:

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

$$= \frac{P(x_1 \ldots x_i, \pi_i = k)\, P(x_{i+1} \ldots x_L \mid \pi_i = k)}{P(x)} = \frac{f_k(i) b_k(i)}{P(x)}$$

Zauważmy, że ciąg stanów π o największym prawd. warunkowym, gdzie π_i' = argmax$_k P(\pi_i = k|x)$ nie musi być legalną ścieżką, w przypadku gdy nie wszystkie przejścia pomiędzy stanami są dozwolone.

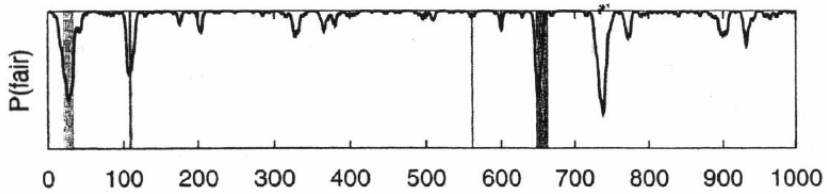
Niech $g(k) = 1$ gdy stan k należy do interesującego nas zbioru stanów a w przeciwnym wypadku niech $g(k) = 0$. Funkcja przedstawiona poniżej opisuje prawdopodobieństwo aposteriori tego, że symbol x_i należał do tego zbioru np. na pozycji i mamy wyspę CpG.

$$G(i|x) = \sum_k P(\pi_i = k|x) g(k)$$

```
Rolls   315116246446644245311321631164152133625144543631656626566666
Die     FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
Viterbi FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Rolls   651166453132651245636664631636663162326455236266666625151631
Die     LLLLLLFFFFFFFFFFFFLLLLLLLLLLLLLLLLFFFLLLLLLLLLLLLLFFFFFFFFFF
Viterbi LLLLLLFFFFFFFFFFFFLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLFFFFFFFFF
```

Nieuczciwe kasyno (alg. Viterbi): 'Die' – rodzaj kostki, F – prawdziwa, L – oszukana.



Nieuczciwe kasyno (alg. aposteriori): oś X – numer rzutu, ciemne pola – kostka oszukana.

Szacowanie parametrów HMM

W przypadku gdy znamy sekwencję stanów π i odpowiadające jej słowo x, możemy policzyć ile razy występują określone przejścia i emisje a na tej podstawie szacować prawdopodobieństwa.

Niech A_{kl} oznacza ilość przejść ze stanu k do l w naszym wzorcowym zbiorze sekwencji oraz $E_k(b)$ będzie analogiczną liczbą emisji symboli:

$$a_{kl} = \frac{A_{kl}}{\sum_{l'} A_{kl'}}$$

$$e_k(b) = \frac{E_k(b)}{\sum_{b'} E_k(b')} \qquad (*)$$

Gdy nie znamy sekwencji stanów π odpowiadającej obserwacji x możemy zastosować metodę Baum'a – Welch'a.

Niech $x^1, \ldots, x^n \in \Sigma^+$ oznacza zbiór zaobserwowanych sekwencji uczących. W metodzie tej będziemy się starać znaleźć zbiór parametrów modelu θ, który maksymalizuje następujące wyrażenie (logarytm prawdopodobieństwa):

$$l(x^1, \ldots, x^n|\theta) = \log P(x^1, \ldots, x^n|\theta) = \sum_{j=1}^{n} \log P(x^j|\theta)$$

Metoda kolejnych przybliżeń Baum'a – Welch'a polega na wyznaczeniu A_{kl} oraz $E_k(b)$ jako oczekiwanych ilości przejść i emisji występujących we wzorcowych sekwencjach. Ponieważ

$$P(\pi_i = k, \pi_{i+1} = l \mid x, \theta) = \frac{f_k(i) a_{kl} e_l(x_{i+1}) b_l(i+1)}{P(x)}$$

wartości te możemy znaleźć korzystając z zależności:

$$A_{kl} = \sum_{j=1}^{n} \frac{1}{P(x^j)} \sum_{i=1}^{|x^j|-1} f_k^j(i) a_{kl} e_l(x_{i+1}^j) b_l^j(i+1) \qquad (**)$$

$$E_k(b) = \sum_{j=1}^{n} \frac{1}{P(x^j)} \sum_{i:x_i^j=b} f_k^j(i) b_k^j(i) \qquad (***)$$

Kroki algorytmu:

Inicjalizacja:

Wybierz zbiór stanów i początkowe prawd. a_{kl} i $e_k(b)$.

while not ($l(x^1,\ldots,x^n|\theta)$ większy niż wybrany próg lub osiągnięto maksymalną liczbę iteracji)

for j:=1 **to** n **do**

begin

Oblicz $f_k(i)$ dla sekwencji x^j używając procedury prefiksowej.

Oblicz $b_k(i)$ dla sekwencji x^j używając procedury sufiksowej.

Uaktualnij A oraz E na podstawie (**) i (***).

end

Oblicz nowe parametry a_{kl} i $e_k(b)$ modelu (*).

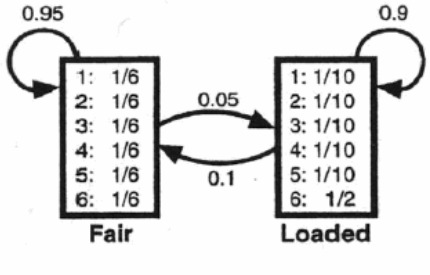
Oblicz nowe $l(x^1,\ldots,x^n|\theta)$

End

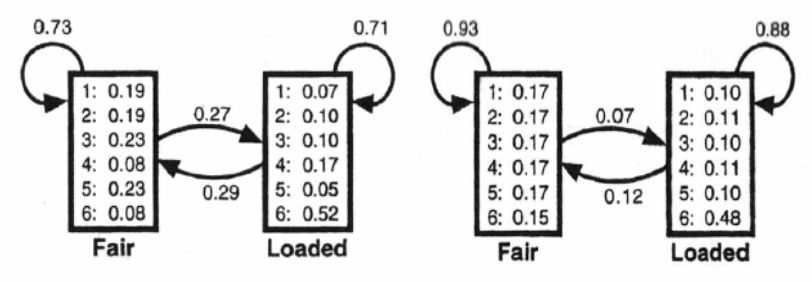
Metoda Baum'a – Welch'a z każda iteracją nie zmniejsza $l(x^1,..., x^n|\theta)$ ale nie gwarantuje znalezienie maksimum globalnego.

- Wielokrotne uruchamianie poszukiwań dla różnych parametrów początkowych
- Wprowadzenie drobnych, losowych zaburzeń podczas korzystania z formuł (*) – (***).

Przykład „douczania" się parametrów dla nieuczciwego kasyna. Różne długości ciągów uczących:



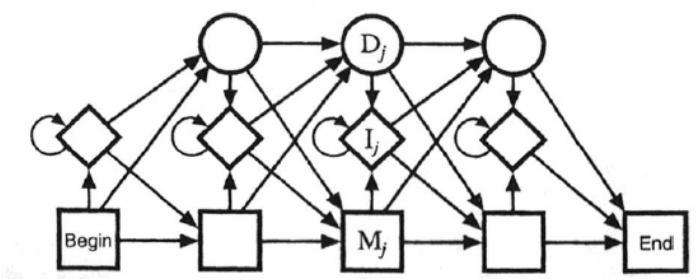
Wzorzec.



Dla ciągu o długości 300 oraz 30 000.

Modelowanie profili sekwencji przy użyciu HMM

Celem modelowania profili sekwencji jest znalezienie HMM (topologii i parametrów), który zgromadzi informacje dotyczące charakterystycznych cech danej rodziny sekwencji. Za pomocą takiego modelu można określić jak blisko inna sekwencja jest spokrewniona z rodziną bazową.



Opis modelu:

- *M* – stany (zgodne z definicją HMM). Kolejne M_i odpowiadają kolumnom profilu.
- *I* – insercja, daje możliwość umieszczania symboli, które nie pasują do profilu. Przykładowo: możliwe jest przejście ze stanu M_j do I_j, wielokrotne przechodzenie w pętli z I_j do I_j dla insercji oraz przejście do stanu M_{i+1}.
- *D* – delecja, tak zwane „ciche” stany. Nie zachodzi w nich emisja symboli.
- Prawdopodobieństwo emisji słowa *x* na drodze Begin→End jest miarą podobieństwa *x* do profilu.
- Najbardziej prawdopodobna ścieżka Begin→End emitująca *x* wyznacza dopasowanie słowa do profilu.