生命情報学 （5）

隠れマルコフモデル

阿久津　達也

京都大学　化学研究所
バイオインフォマティクスセンター

内容

- 配列モチーフ
- 最尤推定、ベイズ推定、MAP推定
- 隠れマルコフモデル(HMM)
- Viterbiアルゴリズム
- EMアルゴリズム
- Baum-Welchアルゴリズム
  - 前向きアルゴリズム、後向きアルゴリズム
- プロファイルHMM

配列モチーフ

モチーフ発見

- 配列モチーフ： 同じ機能を持つ遺伝子配列などに見られる共通の文字列パターン

正規表現など文法表現を用いるもの

例： ロイシンジッパーモチーフ

L-x(6)-L-x(6)-L-x(6)-L

ジンクフィンガーモチーフ

C-x(2,4)-C-x(3)-[LIVMFYWC]-x(8)-H-x(3,5)-H

人間にとってわかりやすいが表現力が弱い

確率的な表現法を用いるもの

重み行列（プロファイル）

HMM (隠れマルコフモデル)

人間にとってわかりにくいが
一般に表現力は高い

A	3.8	-3.5	1.2	2.3
C	1.5	1.3	-0.3	-4.6
G	-1.5	-2.9	4.2	3.1
T	0.2	-4.1	3.7	-1.3

A	A	C	G	G	C

score= 3.8 + 1.3 + 4.2 + 3.1

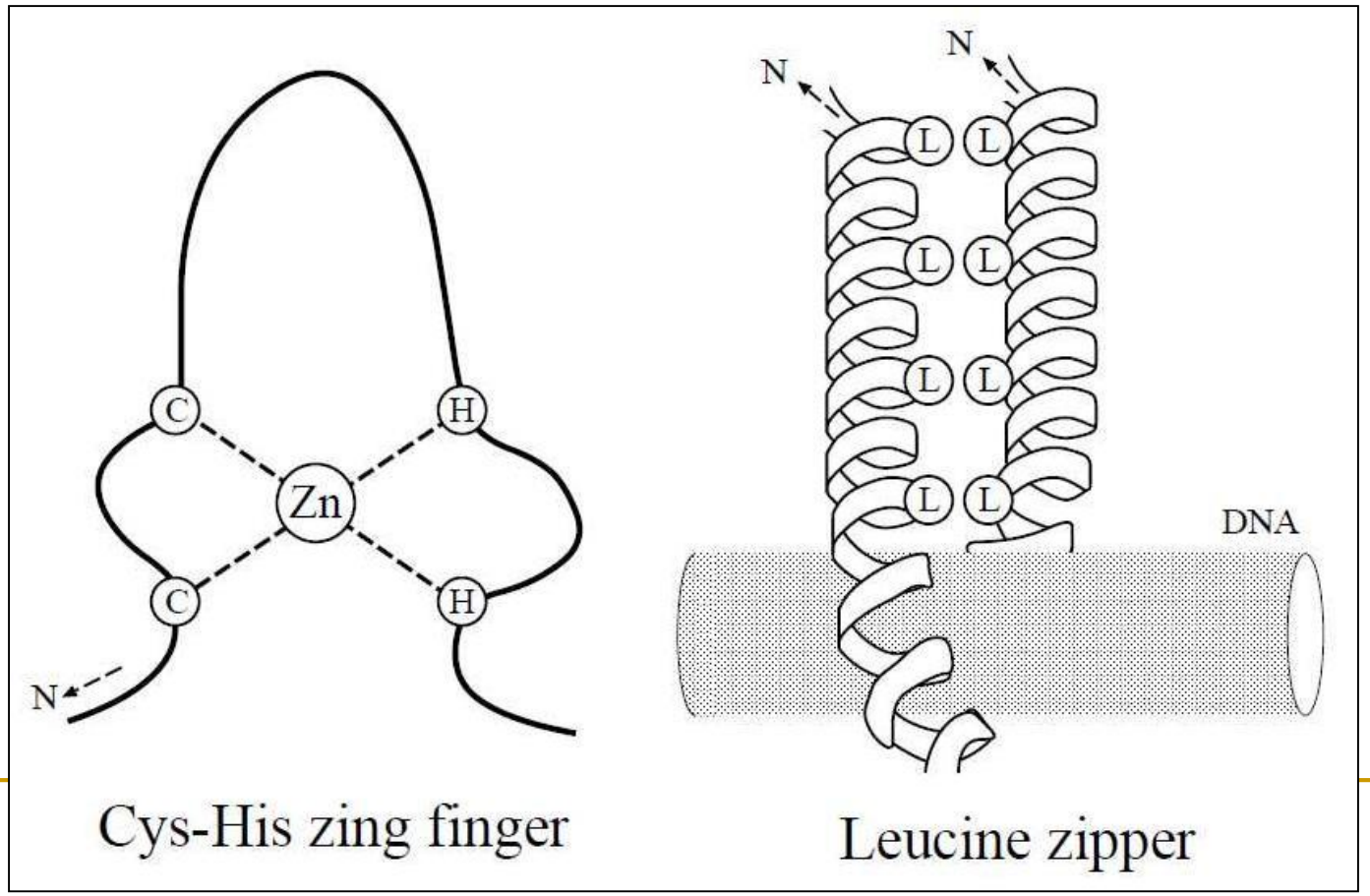
モチーフの例

- ジンクフィンガーモチーフ

  C-x(2,4)-C-x(3)-[LIVMFYWC]-x(8)-H-x(3,5)-H

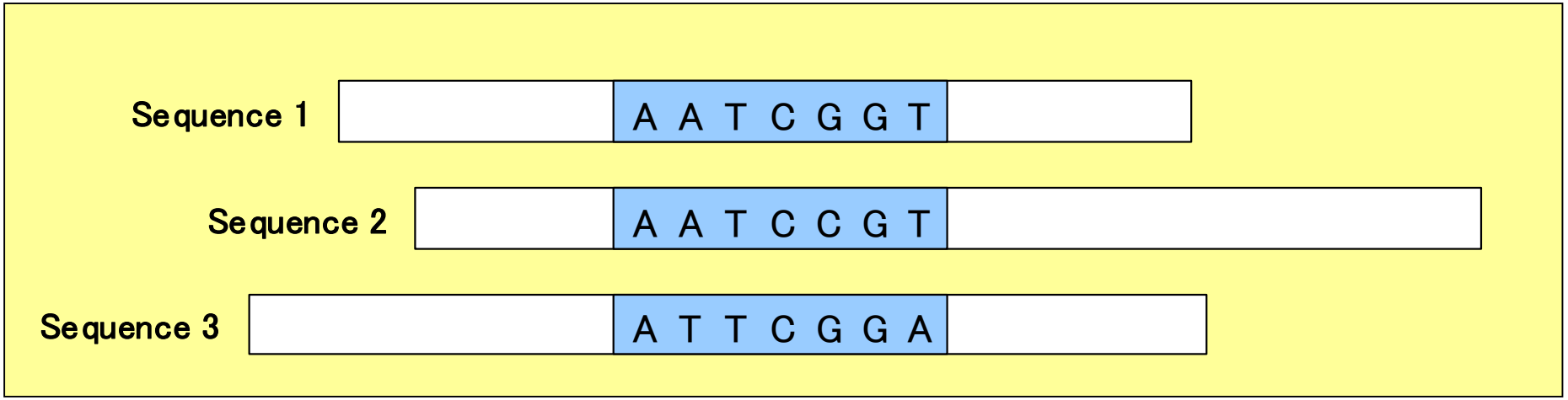
- ロイシンジッパーモチーフ

  L-x(6)-L-x(6)-L-x(6)-L



局所マルチプルアラインメント

- 複数配列と長さ L が与えられた時、スコア最大となるように各配列から長さ L の部分列を抽出
- モチーフ発見などに有用



相対エントロピースコアのもとでの局所マルチプルアラインメント

- 相対エントロピースコアの定義
  - $f_j(a)$: （モチーフ領域の）j列目におけるaの出現頻度
  - $p(a)$: aの出現頻度（事前確率）
  - L: モチーフ領域の長さ

$$score = \sum_{j=1}^{L} \sum_{a} f_j(a) \log \frac{f_j(a)}{p(a)}$$

- 実用的アルゴリズム
  - Gibbsサンプリング, EMアルゴリズム

Gibbs サンプリング

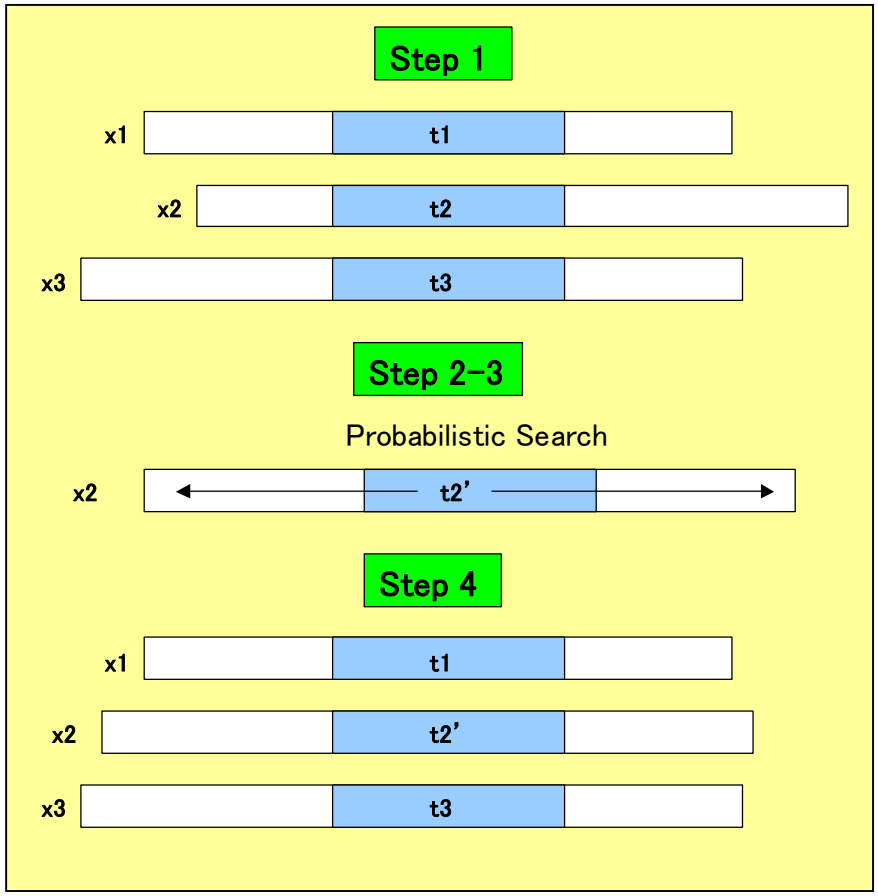
1. 各配列 x_j からランダムに部分配列 t_j を選ぶ
2. 1個の配列 x_i をランダムに選ぶ
3. x_i の部分列 t_i' を

$$\prod_{j=1}^{L} \frac{f_j(t'_i[j])}{p(t'_i[j])}$$

に比例する確率で選ぶ

4. t_i をt_i' でおきかえる
5. ステップ2-4を十分な回数だけ繰り返す

($t_i[j]$: 部分列t_i のj列目の文字)



最尤推定、ベイズ推定、MAP推定

最尤推定

- $P(D|\theta)$ （尤度）
  - モデルパラメータ θ のもとでのデータ D の出現確率
- 最尤法
  - $P(D|\theta)$ を最大化する θ を選ぶ
- 例
  - コインを5回投げて、表が3回出た後、裏が2回出た
  - p(表)=a, p(裏)=1-a とすると、$P(D|\theta)=a^3(1-a)^2$
  - a=3/5の時、$P(D|\theta)$ は最大
  - 一般に表が出る頻度を f とすると $a=f$ で尤度は最大

ベイズ推定とMAP推定

- ベイズ推定：尤度とモデル（パラメータ）の事前確率から、ベイズの定理により、事後確率を推定

$$P(\theta \mid D) = \frac{P(D \mid \theta)P(\theta)}{P(D)}$$

ただし、$P(D) = \int_{\theta'} P(D \mid \theta')P(\theta')$　（θが連続値の時）

- 最大事後確率（MAP）推定
  - $P(D|\theta)P(\theta)$ を最大化する θ を計算
  - $P(\theta)$ が一様分布なら最尤推定と同じ

不正サイコロのベイズ推定

- 公正サイコロと不正サイコロ
  - 公正 : $P(i|\text{公正})=1/6$
  - 不正 : $P(6|\text{不正})=1/2$, $P(i|\text{不正})=1/10$ for $i\neq 6$
  - $P(\text{公正})=0.99$, $P(\text{不正})=0.01$

- 6が3回続けて出た場合の事後確率

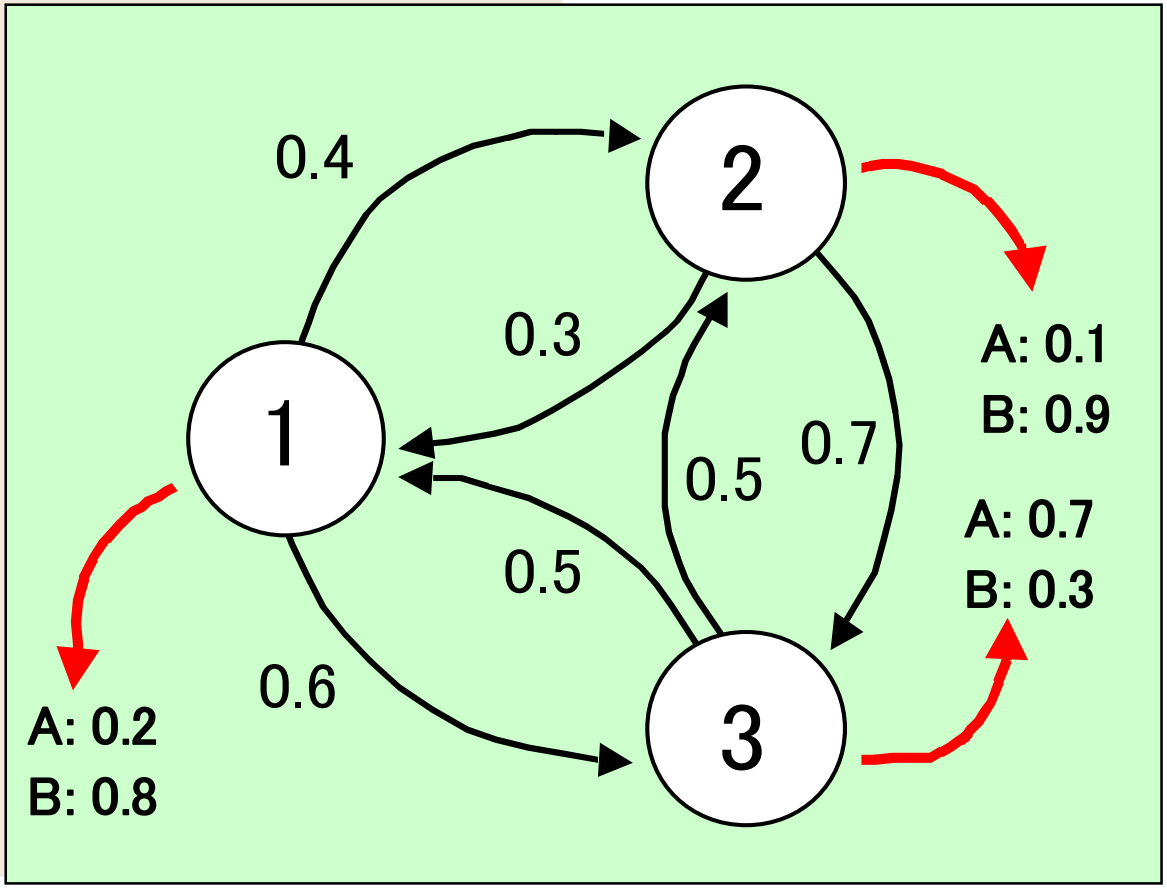
$$P(\text{不正}|666)=\frac{P(666|\text{不正})P(\text{不正})}{P(666)}$$

$$=\frac{(0.5)^3(0.01)}{(0.5)^3(0.01)+(\frac{1}{6})^3(0.99)}=0.21$$

隠れマルコフモデル

隠れマルコフモデル(HMM)

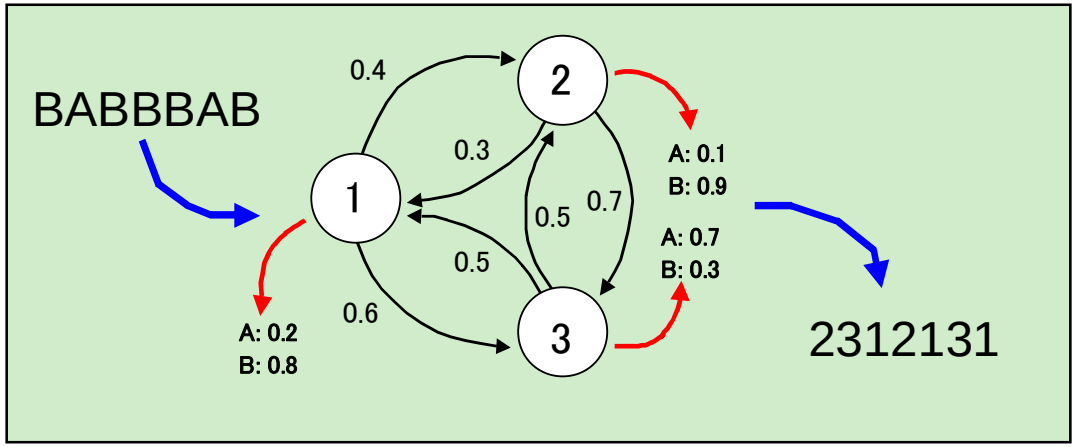
- HMM≒有限オートマトン＋確率
- 定義
  - 出力記号集合Σ
  - 状態集合
    $S=\{1,2,\ldots n\}$
  - 遷移確率($k\rightarrow l$)
    a_{kl}
  - 出力確率
    $e_k(b)$
  - (開始状態=
    終了状態= 0)

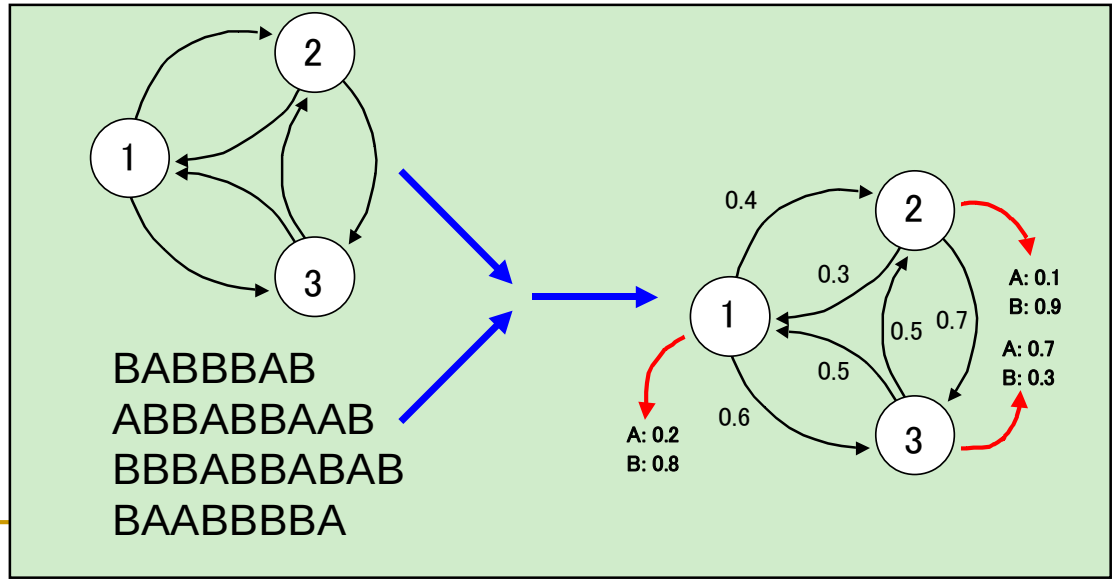


HMMにおける基本アルゴリズム

- Viterbiアルゴリズム
  - 出力記号列から状態列を推定
  - 構文解析
- Baum-Welchアルゴリズム

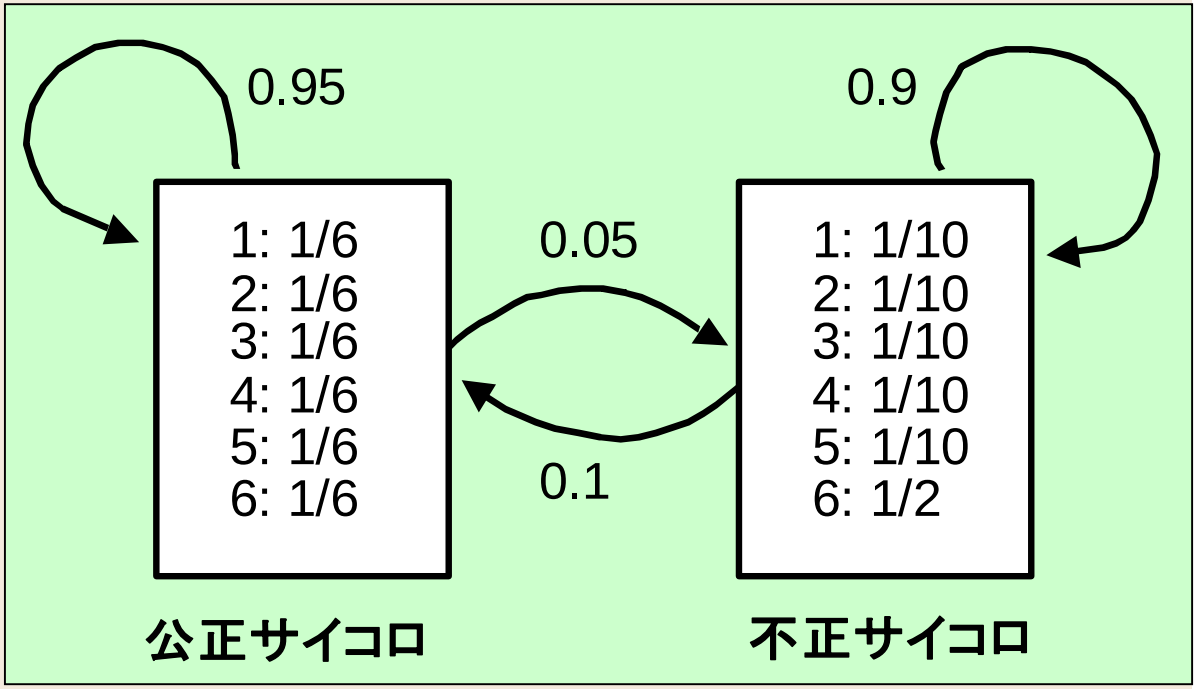
  (EMアルゴリズム)
  - 出力記号列からパラメータを推定
  - 学習





時々いかさまをするカジノ

- サイコロの出目だけが観測可能、どちらのサイコロを振っているかは観測不可能
- サイコロの出目から、どちらのサイコロを振っているかを推定
- 6,2,6,6,3,6,6,6,
  4,6,5,3,6,6,1,2
  →不正サイコロ
- 6,1,5,3,2,4,6,3,
  2,2,5,4,1,6,3,4
  →公正サイコロ
- 6,6,3,6,5,6,6,1,
  5,4,2,3,6,1,5,2
  →途中で公正サイコロに交換



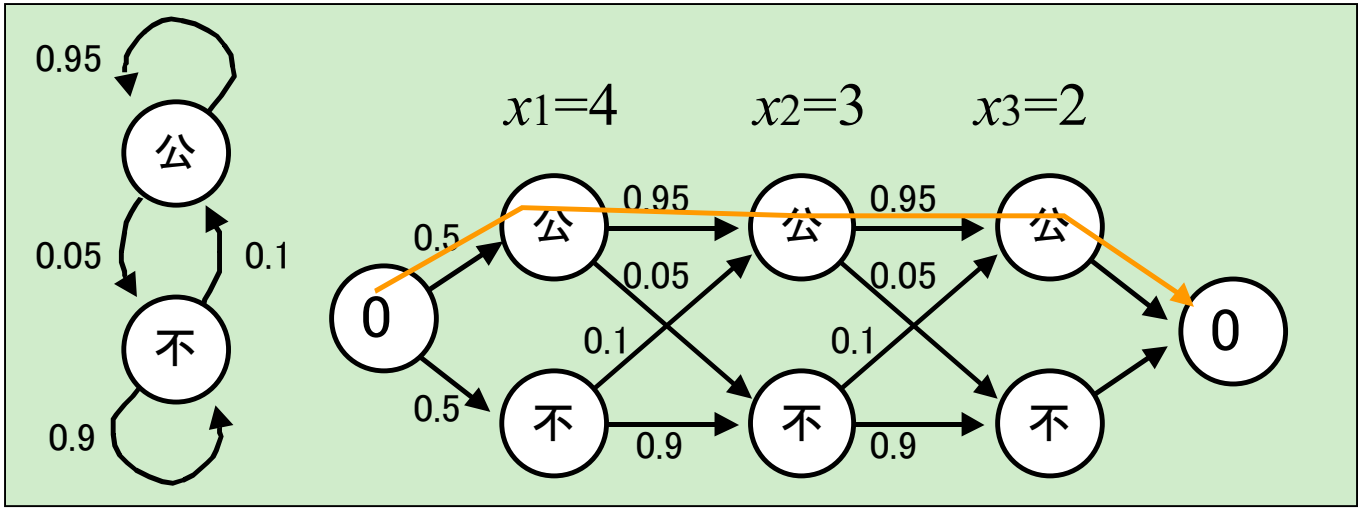
Viterbiアルゴリズム

Viterbiアルゴリズム(1)

- 観測列(出力配列データ) $x=x_1\ldots x_L$と状態列$\pi=\pi_1\ldots\pi_L$が与えられた時、その同時確率は

$$P(x,\pi)=a_{0\,\pi_1}\Pi e_{\pi i}(x_i)a_{\pi i\pi i+1} \qquad \text{但し、}\pi_{L+1}=0$$

- x が与えられた時、最も尤もらしい状態列は $\pi^*=\mathrm{argmax}_\pi\ P(x,\pi)$
- 例:どちらのサイコロがいつ使われたかを推定



$$\max_\pi P(x_1x_2x_3,\pi)=P(x_1x_2x_3,\text{公公公})=0.5\cdot\tfrac{1}{6}\cdot0.95\cdot\tfrac{1}{6}\cdot0.95\cdot\tfrac{1}{6}$$

Viterbiアルゴリズム(2)

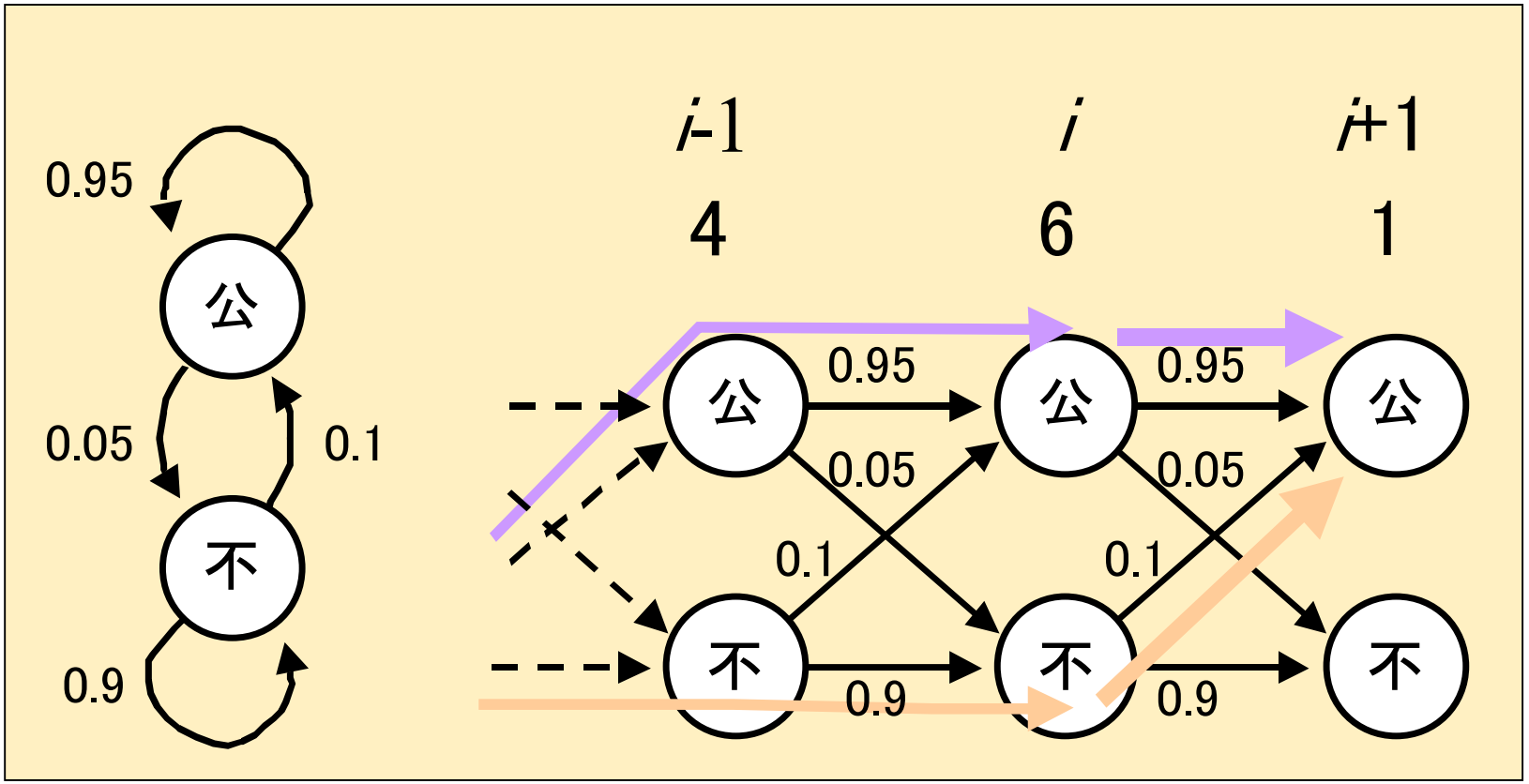
- x から、$\pi^* = \mathrm{argmax}_\pi \; P(x,\pi)$ を計算
- そのためには $x_1 \ldots x_i$ を出力し、状態 k に至る確率最大の状態列の確率 $v_k(i)$ を計算

$$v_k(i) = \max_{\pi} \left\{ \prod_{j=1}^{i} e_{\pi_j}(x_j) a_{\pi_{j-1}\pi_j} \right\}$$

- $v_k(i)$は以下の式に基づき動的計画法で計算

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

Viterbiアルゴリズム(3)



$$v_{公}(i+1) = \max\{\ e_{公}(1)\cdot 0.95\cdot v_{公}(i),\ e_{公}(1)\cdot 0.1\cdot v_{不}(i)\ \}$$

EMアルゴリズム

EM(Expectation Maximization)アルゴリズム

- 「欠けているデータ」のある場合の最尤推定のための一般的アルゴリズム

x:観測データ、y:欠けているデータ、

θ:パラメータ集合

目標：$\log P(x|\theta) = \log \sum_{y} P(x,y|\theta)$ の最大化

- 最大化は困難であるので、反復により尤度を単調増加させる（θ^tよりθ^{t+1}を計算）
- HMMの場合、「欠けているデータ」は状態列

EMアルゴリズムの導出

$$\log P(x|\theta) = \log P(x,y|\theta) - \log P(y|x,\theta)$$

両辺に$P(y|x,\theta^t)$をかけてyについての和をとり、

$$\log P(x|\theta) = \sum_y P(y|x,\theta^t)\log P(x,y|\theta) - \sum_y P(y|x,\theta^t)\log P(y|x,\theta)$$

右辺第１項を$Q(\theta|\theta^t)$とおくと、

$$\log P(x|\theta) - \log P(x|\theta^t) =$$

$$Q(\theta|\theta^t) - Q(\theta^t|\theta^t) + \sum_y P(y|x,\theta^t)\log\frac{P(y|x,\theta^t)}{P(y|x,\theta)}$$

最後の項は相対エントロピーで常に正なので、

$$\log P(x|\theta) - \log P(x|\theta^t) \geq Q(\theta|\theta^t) - Q(\theta^t|\theta^t)$$

よって、$\theta^{t+1} = \arg\max_\theta Q(\theta|\theta^t)$ とすれば尤度は増大

EMアルゴリズムの一般形

1. 初期パラメータ Θ^0 を決定。t=0とする
2. $Q(\theta|\theta^t)=\sum P(y|x, \theta^t) \log P(x,y|\theta)$ を計算
3. $Q(\theta|\theta^t)$を最大化するθ^*を計算し、$\theta^{t+1} = \theta^*$ とする。t=t+1とする
4. Qが増大しなくなるまで、２，３を繰り返す

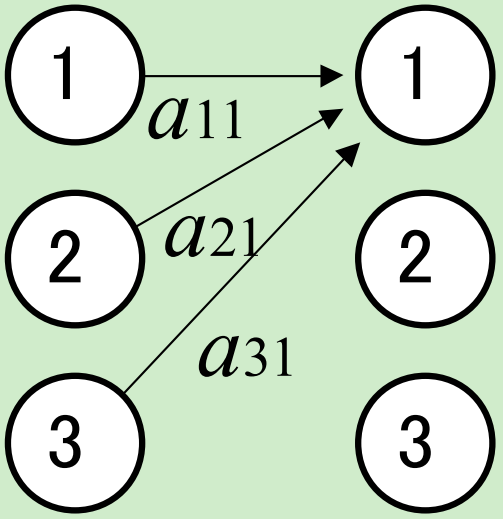
前向きアルゴリズム

- 配列 x の生成確率 $P(x)=\sum P(x,\pi)$ を計算
- Viterbiアルゴリズムと類似
- $f_k(i)=P(x_1 \ldots x_i,\pi_i=k)$ をDPにより計算

$$f_0(0)=1,\ f_k(0)=0$$

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

$$P(x)=\sum_k f_k(L)a_{k0}$$



$$f_1(i)=e_1(x_i)(f_1(i-1)a_{11}+f_2(i-1)a_{21}+f_3(i-1)a_{31})$$

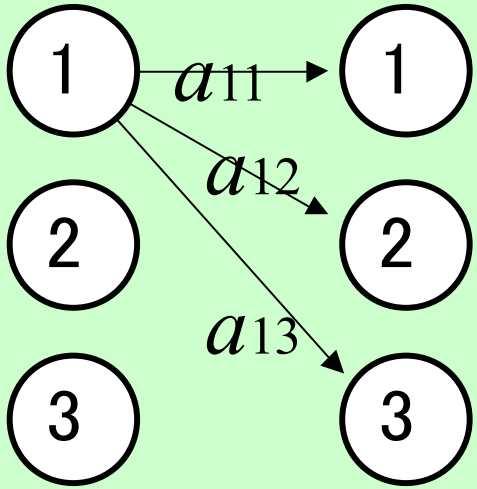
後向きアルゴリズム

- $b_k(i)=$
  $P(x_{i+1}\ldots x_L|\pi_i=k)$
  をDPにより計算
- $P(\pi_i=k|x) =$
  $f_k(i)b_k(i)/P(x)$

$$b_k(L) = a_{k0}$$

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

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



$$b_1(i) =$$
$$(a_{11}e_1(x_{i+1})b_1(i+1) +$$
$$a_{12}e_2(x_{i+1})b_2(i+1) +$$
$$a_{13}e_3(x_{i+1})b_3(i+1))$$

Viterbi と前向きアルゴリズムの比較

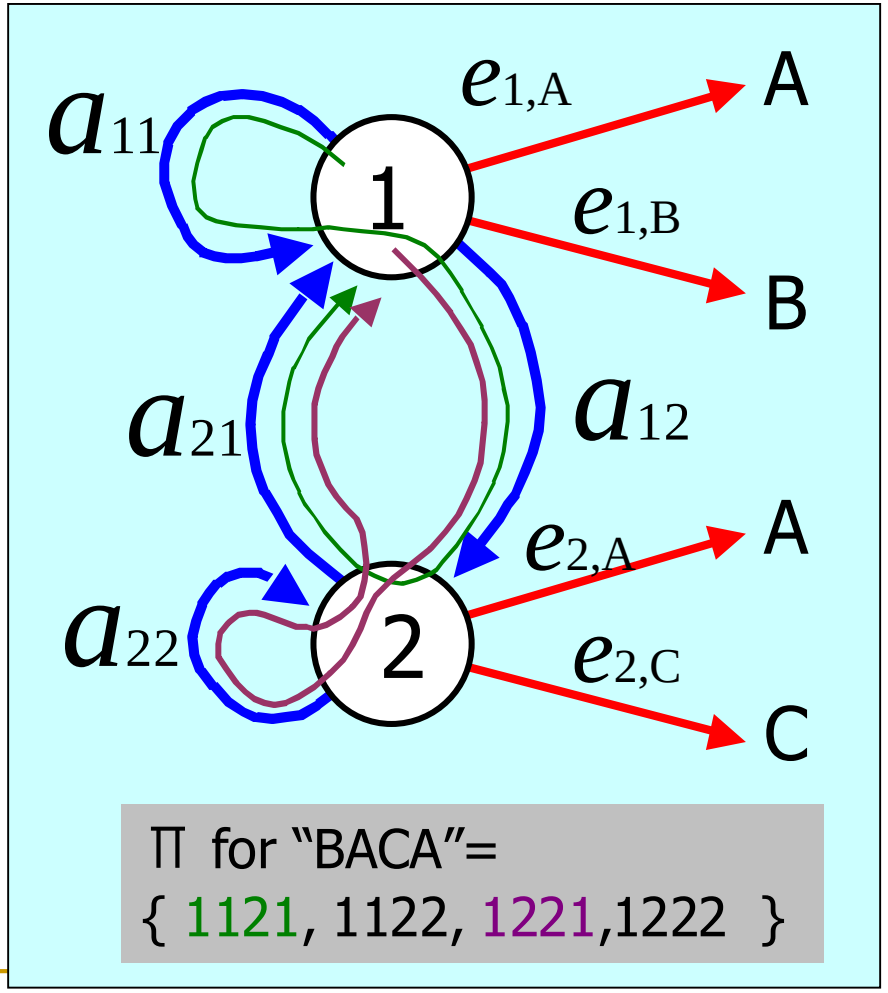
$$P(x,\pi|\theta) = \prod_{i=1}^{n} a_{\pi_{i-1}\pi_i} e_{\pi_i, x_i}$$

- Viterbiアルゴリズム

$$\max_{\pi} \{ P(x,\pi|\theta) \}$$

- Forwardアルゴリズム

$$\sum_{\pi} \{ P(x,\pi|\theta) \}$$



HMMに対するEMアルゴリズム（Baum-Welchアルゴリズム）

A_{kl}: a_{kl}が使われる回数の期待値　　x^j : j番目の配列

$E_k(b)$: 文字bが状態kから現れる回数の期待値

$$A_{kl} = \sum_j \frac{1}{P(x^j)} \sum_i f_k^j(i)\, a_{kl}\, e_l(x_{i+1}^j)\, b_l^j(i+1)$$

$$E_k(b) = \sum_j \frac{1}{P(x^j)} \sum_{\{i \mid x_j^j = b\}} f_k^j(i)\, b_k^j(i)$$

パラメータの更新式

$$\hat{a}_{kl} = \frac{A_{kl}}{\sum_{l'} A_{kl'}} \qquad \hat{e}_k(b) = \frac{E_k(b)}{\sum_{b'} E_k(b')}$$

Baum-WelchのEMによる解釈

$$P(x,\pi\mid\theta)=\prod_{k=1}^{M}\prod_{b}\left[e_k(b)\right]^{E_k(b,\pi)}\prod_{k=0}^{M}\prod_{l=1}^{M}a_{kl}^{A_{kl}(\pi)}$$ および

$$Q(\theta\mid\theta^t)=\sum_{\pi}P(\pi\mid x,\theta^t)\log P(x,\pi\mid\theta)$$ より、

$$Q(\theta\mid\theta^t)=\sum_{\pi}P(\pi\mid x,\theta^t)\times\left[\sum_{k=1}^{M}\sum_{b}E_k(b,\pi)\log e_k(b)+\sum_{k=0}^{M}\sum_{l=1}^{M}A_{kl}(\pi)\log a_{kl}\right]$$

$$=\sum_{k=1}^{M}\sum_{b}E_k(b)\log e_k(b)+\sum_{k=0}^{M}\sum_{l=1}^{M}A_{kl}\log a_{kl}$$

ここで$\sum_i p_i\log q_i$は$q_i=p_i$の時、最大より、

$$e_k(b)=\frac{E_k(b)}{\sum_{b'}E_k(b')},\quad a_{kl}=\frac{A_{kl}}{\sum_{l'}A_{kl}}$$

プロファイルHMM

配列アラインメント

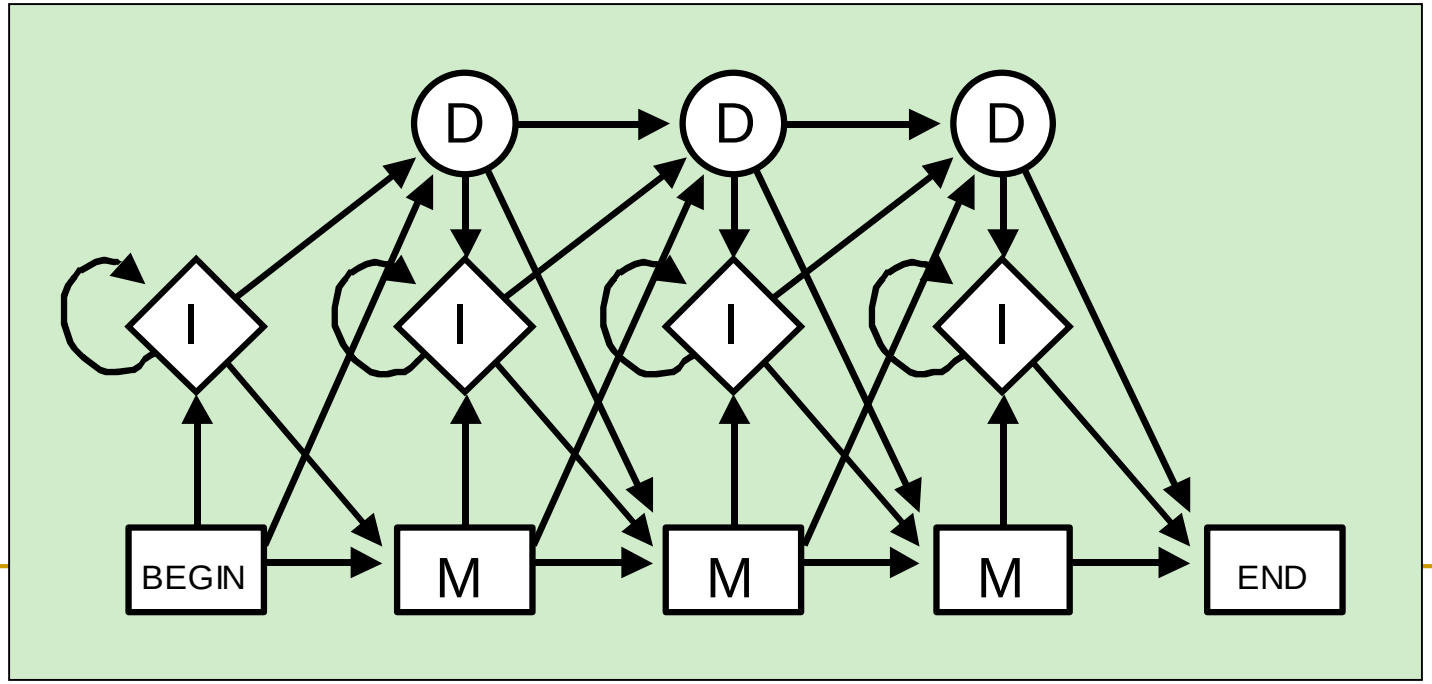
- 2個もしくは3個以上の配列の類似性の判定に利用
  - 2個の場合：ペアワイズアラインメント
  - 3個以上の場合：マルチプルアラインメント
- 文字間の最適な対応関係を求める（最適化問題）
- 配列長が同じになるよう、ギャップ記号を挿入

```
HBA_HUMAN     VGAHAGEY
HBB_HUMAN     VNVDEV
MYG_PHYCA     VEADVAGH
GLB5_PETMA    VYSTYETA
LGB2_LUPLU    FNANIPKH
GLB1_GLYDI    IAGADNGAGV
```

```
HBA_HUMAN     V G A - - H A G E Y
HBB_HUMAN     V - - - - N V D E V
MYG_PHYCA     V E A - - D V A G H
GLB5_PETMA    V Y S - - T Y E T A
LGB2_LUPLU    F N A - - N I P K H
GLB1_GLYDI    I A G A D N G A G V
```

プロファイルHMM (1)

- 配列をアラインメントするためのHMM
- タンパク質配列分類やドメイン予測などに有用
  - 例:ドメインの種類ごとにHMMを作る
    - PFAM(http://pfam.wustl.edu/)
- 一致状態(M)、欠失状態(D)、挿入状態(I)を持つ

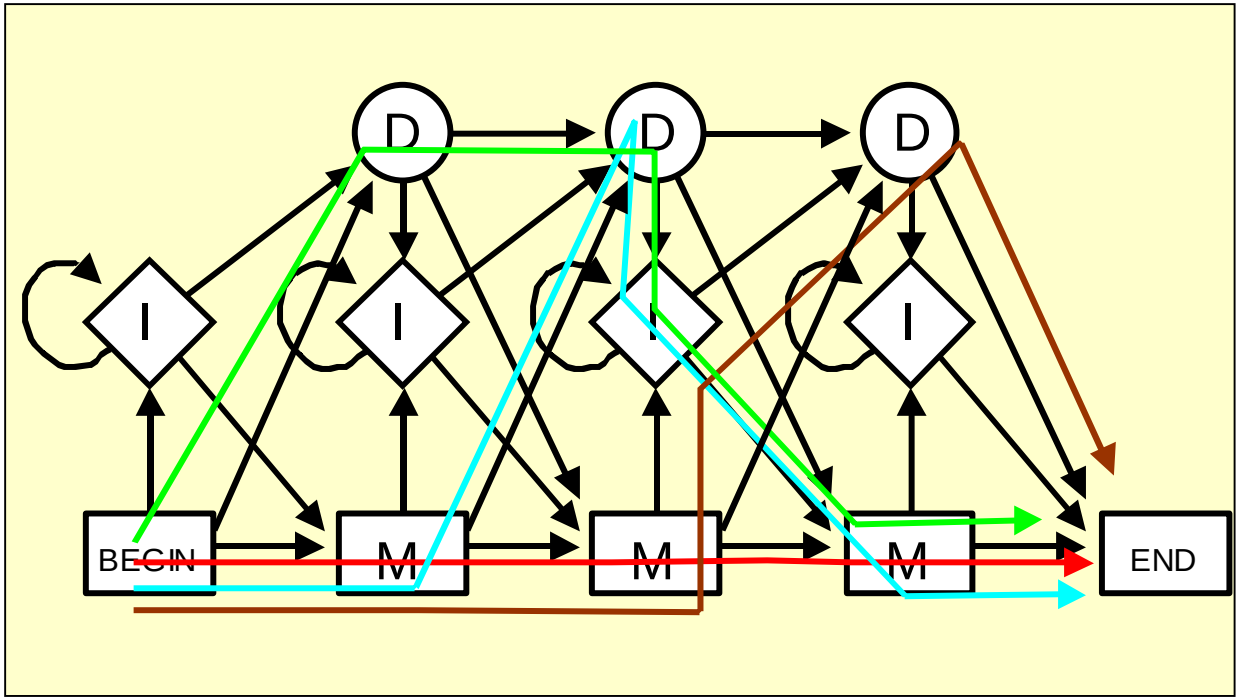


プロファイルHMM (2)

マルチプル
アラインメント

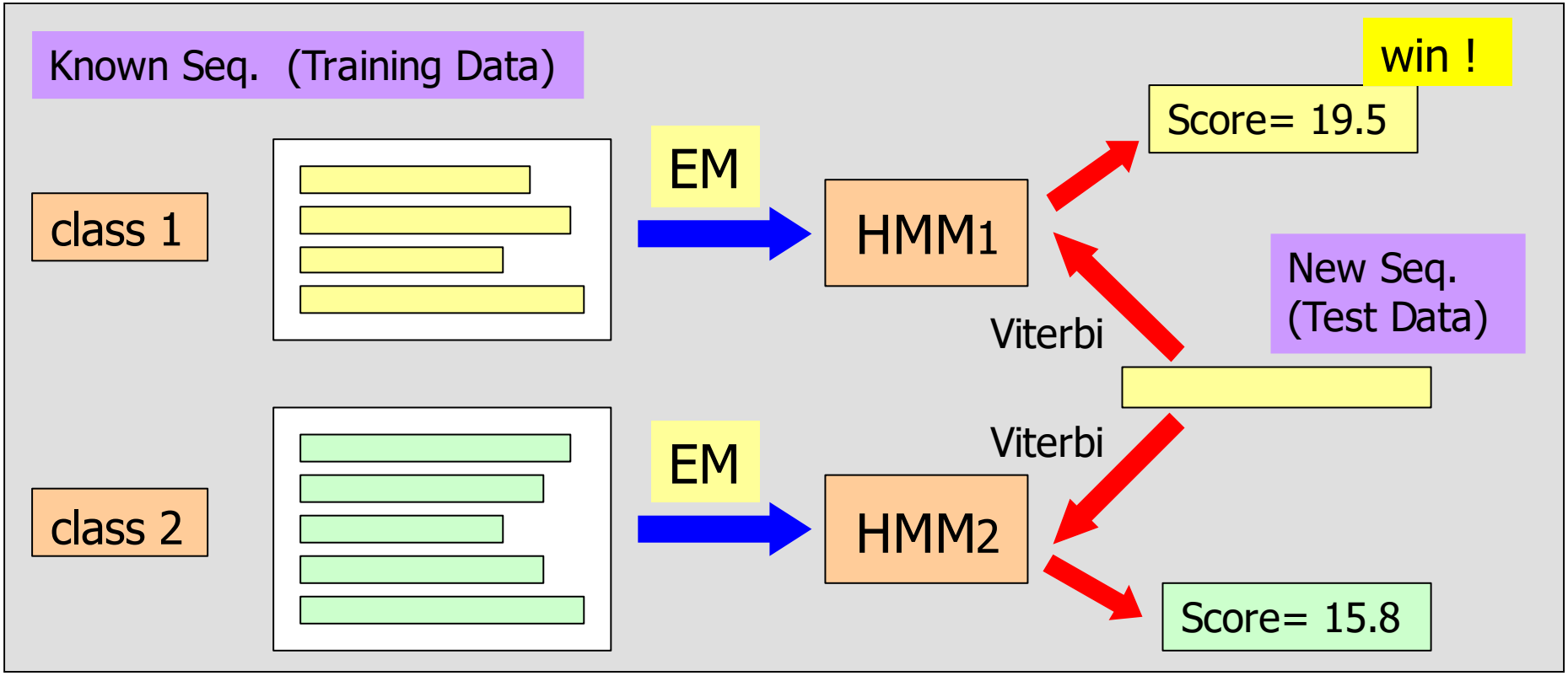
プロファイル
HMM

	M	M	・	・	・	M
こうもり	A	G	—	—	—	C
ラット	A	—	A	G	—	C
ネコ	A	G	—	A	A	—
ハエ	—	—	A	A	A	C
ヤギ	A	G	—	—	—	C



プロファイルHMM (3)

- 各配列ファミリーごとに HMM を作成
- スコア最大のHMMのファミリーに属すると予測



まとめ

- 配列モチーフ
  - 局所マルチプルアラインメント
  - Gibbsサンプリング
- HMMによる配列解析
  - 最尤推定、ベイズ推定、MAP推定
  - 隠れマルコフモデル(HMM)
  - Viterbiアルゴリズム
  - Baum-Welchアルゴリズム
    - EMアルゴリズムに基づく
    - 前向きアルゴリズム、後向きアルゴリズム
  - プロファイルHMM