バイオプログラミング第２

慶應義塾大学理工学部
生命情報学科
榊原康文、佐藤健吾

ゲノム配列の決定

- 2003年４月１４日
  およそ30億塩基対からなるヒトゲノムの解読が完了した

```
  1 agatggcggc gctgaggggt cttgggggct ctaggccggc cacctactgg tttgcagcgg
 61 agacgacgca tggggcctgc gcaataggag tacgctgcct gggaggcgtg actagaagcg
121 gaagtagttg tgggcgcctt tgcaaccgcc tgggacgccg ccgagtggtc tgtgcaggtt
181 cgcgggtcgc tggcgggggt cgtgagggag tgcgccggga gcggagatat ggagggagat
241 ggttcagacc cagagcctcc agatgccggg gaggacagca agtccgagaa tggggagaat
301 gcgcccatct actgcatctg ccgcaaaccg gacatcaact gcttcatgat cgggtgtgac
361 aactgcaatg agtggttcca tggggactgc atccggatca ctgagaagat ggccaaggcc
421 atccgggagt ggtactgtcg ggagtgcaga gagaaagacc ccaagctaga gattcgctat
481 cggcacaaga agtcacggga gcgggatggc aatgagcggg acagcagtga gccccgggat
541 gagggtggag ggcgcaagag gcctgtccct gatccagacc tgcagcgccg ggcagggtca
601 gggacagggg ttggggccat gcttgctcgg ggctctgctt cgccccacaa atcctctccg
661 cagcccttgg tggccacacc cagccagcat caccagcagc agcagcagca gatcaaacgg
721 tcagcccgca tgtgtggtga gtgtgaggca tgtcggcgca ctgaggactg tggtcactgt
```

遺伝子配列解析

- 遺伝子配列のアラインメント
- 遺伝子領域の探索、発見、予測
- 遺伝子の構造予測、機能予測

配列を比較する

- 配列が似ていれば遺伝子としての機能も似ているに違いない！

↓

- 複数の配列を比較し、それらに含まれている遺伝情報を抽出する
- 遺伝子配列を比較することで似ている部分、違う部分を発見する

配列のアラインメント

- ２つの配列に対して、適切な位置にギャップ記号を挿入することで、配列中の同じ位置にある文字が並ぶようにする

入力
GTCAGA
AGCGTAG

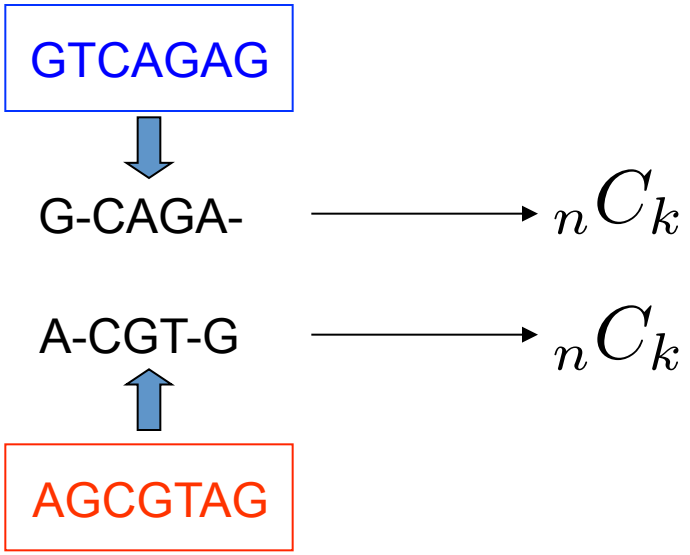
↓

出力
-GTCAG-A-
AG-C-GTAG

大域アラインメントの数

- 長さ *n* 本の2本の配列に対して:

$$ {}_{2n}C_n = \frac{(2n)!}{(n!)(n!)} \simeq \frac{2^{2n}}{\sqrt{\pi n}} $$



$$ {}_nC_k $$

$$ {}_nC_k $$

$$ \sum_{k=1}^{n} {}_nC_k \times {}_nC_k = {}_{2n}C_n $$

大域アラインメントの数

- 長さ *n* 本の2本の配列に対して:

$$ {}_{2n}C_n = \frac{(2n)!}{(n!)(n!)} \simeq \frac{2^{2n}}{\sqrt{\pi n}} $$

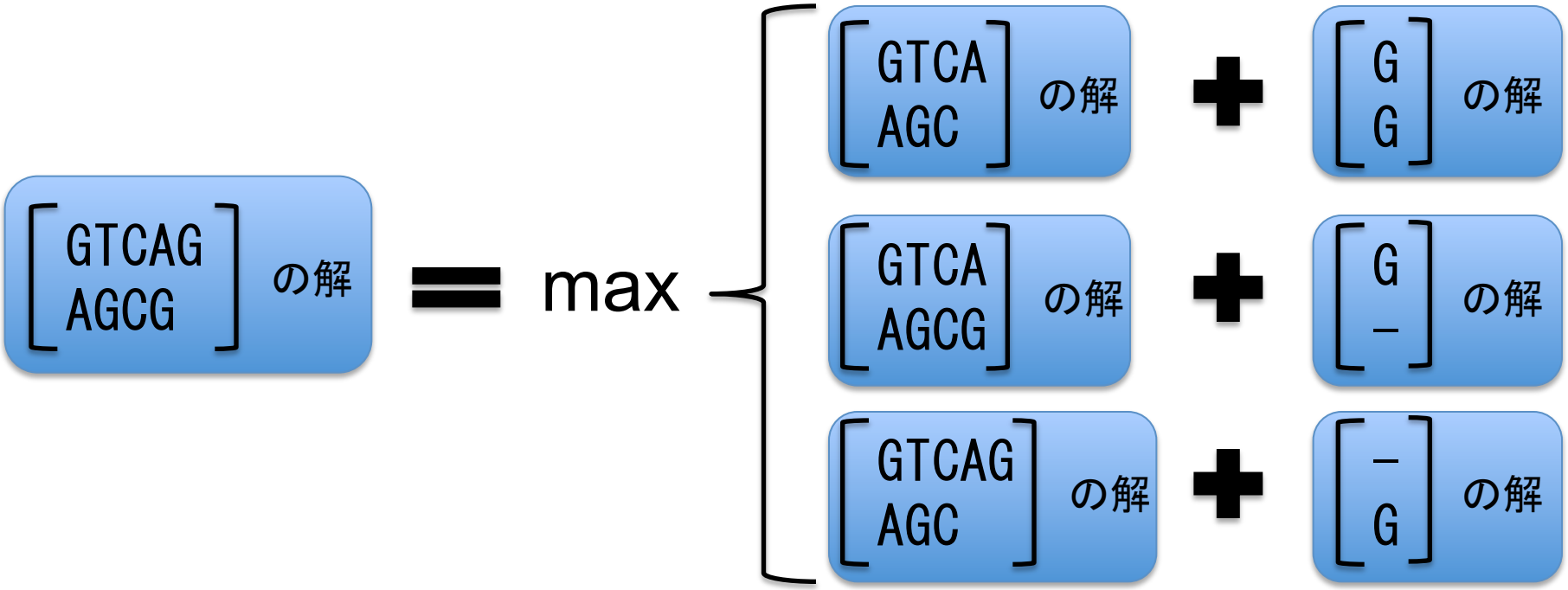
n=10の時 $$ \frac{2^{2\times 10}}{\sqrt{10\pi}} \simeq \frac{(10^3)^2}{5.6} \simeq 1.8 \times 10^5 $$

n=100の時 $$ \frac{2^{2\times 100}}{\sqrt{100\pi}} \simeq \frac{(10^3)^{20}}{17.7} \simeq 5.6 \times 10^{58} $$

数え上げるのは無理

分割統治

- 問題をいくつかの小問題に分割して、それぞれを解く。



動的計画法によるアラインメント

		0	1	2	3	4	5	6
			G	T	C	A	G	A
0								
1	A							
2	G							
3	C					2	2	
4	G					2	3	
5	T							
6	A							
7	G							

$D(i,j)$: $x_1x_2\ldots x_i$と$y_1y_2\ldots y_j$の最適アラインメントのスコア

動的計画法によるアラインメント

- 初期化

$$D(0,0) = 0$$

$$D(i,0) = D(i-1,0) + d \quad (i > 0)$$

$$D(0,j) = D(0,j-1) + d \quad (j > 0)$$

- 再帰式

マッチスコア

$$D(i,j) = \max \begin{cases} D(i-1,j-1) + s(x_i, y_j) & \text{case (1)} \\ D(i-1,j) + d & \text{case (2)} \\ D(i,j-1) + d & \text{case (3)} \end{cases}$$

ギャップペナルティ

動的計画法によるアラインメント

		0	1	2	3	4	5	6
			G	T	C	A	G	A
0		0	0 ←	0 ←	0 ←	0 ←	0 ←	0 ←
1	A	0 ↑						
2	G	0 ↑						
3	C	0 ↑						
4	G	0 ↑						
5	T	0 ↑						
6	A	0 ↑						
7	G	0 ↑						

動的計画法によるアラインメント

		0	1	2	3	4	5	6
			G	T	C	A	G	A
0		0	0 ←	0 ←	0 ←	0 ←	0 ←	0 ←
1	A	0 ↑	0 ↑	0 ↑	0 ↑	1 ↖	1 ←	1 ↖
2	G	0 ↑	1 ↖	1 ←	1 ←	1 ↑	2 ↖	2 ←
3	C	0 ↑	1 ↑	1 ↑	2 ↖	2 ←	2 ↑	2 ↑
4	G	0 ↑	1 ↖	1 ↑	2 ↑	2 ↑	3 ↖	3 ←
5	T	0 ↑	1 ↑	2 ↖	2 ↑	2 ↑	3 ↑	3 ↑
6	A	0 ↑	1 ↑	2 ↑	2 ↑	3 ↖	3 ↑	4 ↖
7	G	0 ↑	1 ↖	2 ↑	2 ↑	3 ↑	4 ↖	4 ↑

動的計画法によるアラインメント

		0	1	2	3	4	5	6
			G	T	C	A	G	A
0		0	0 ←	0 ←	0 ←	0 ←	0 ←	0 ←
1	A	0 ↑	0 ↑	0 ↑	0 ↑	1 ↖	1 ←	1 ↖
2	G	0 ↑	1 ↖	1 ←	1 ←	1 ↑	2 ↖	2 ←
3	C	0 ↑	1 ↑	1 ↑	2 ↖	2 ←	2 ↑	2 ↑
4	G	0 ↑	1 ↖	1 ↑	2 ↑	2 ↑	3 ↖	3 ←
5	T	0 ↑	1 ↑	2 ↖	2 ↑	2 ↑	3 ↑	3 ↑
6	A	0 ↑	1 ↑	2 ↑	2 ↑	3 ↖	3 ↑	4 ↖
7	G	0 ↑	1 ↖	2 ↑	2 ↑	3 ↑	4 ↖	4 ↑

－ G T C A G － A －

A G － C － G T A G

動的計画法によるアラインメント

- 初期化

$$T(i,0) = \text{“}\leftarrow\text{”} \quad (i > 0)$$
$$T(0,j) = \text{“}\uparrow\text{”} \quad (j > 0)$$

- 再帰式

$$T(i,j) = \begin{cases} \text{“}\nwarrow\text{”} & \text{case } (1) \\ \text{“}\leftarrow\text{”} & \text{case } (2) \\ \text{“}\uparrow\text{”} & \text{case } (3) \end{cases}$$

できれば今日中にやること

- ダウンロードした雛形プログラムに以下の関数を追加する。
  - align: 配列間の類似度を計算する
  - traceback: アラインメントを出力する
  - scoref: マッチスコアを返す

http://www.dna.bio.keio.ac.jp/lecture/progen2/data/dp.c

- 簡単な配列で実行してみる。
  - http://www.dna.bio.keio.ac.jp/lecture/progen2/data/example-sequence1.data
  - http://www.dna.bio.keio.ac.jp/lecture/progen2/data/example-sequence2.data

float align(int len_x, int len_y)

- 表（*D*, *T*）を埋めて、最大スコアを返す。

		0	1	2	3	4	5	6
			G	T	C	A	G	A
0		0	0 ←	0 ←	0 ←	0 ←	0 ←	0 ←
1	A	0 ↑	0 ↑	0 ↑	0 ↑	1 ↖	1 ←	1 ↖
2	G	0 ↑	1 ↖	1 ←	1 ←	1 ↑	2 ↖	2 ←
3	C	0 ↑	1 ↑	1 ↑	2 ↖	2 ←	2 ↑	2 ↑
4	G	0 ↑	1 ↖	1 ↑	2 ↑	2 ↑	3 ↖	3 ←
5	T	0 ↑	1 ↑	2 ↖	2 ↑	2 ↑	3 ↑	3 ↑
6	A	0 ↑	1 ↑	2 ↑	2 ↑	3 ↖	3 ↑	4 ↖
7	G	0 ↑	1 ↖	2 ↑	2 ↑	3 ↑	4 ↖	4 ↑

int traceback(int i, int j)

- *T*の矢印を辿りアラインメントを作る。

		0	1	2	3	4	5	6
			G	T	C	A	G	A
0		0	0 ←	0 ←	0 ←	0 ←	0 ←	0 ←
1	A	0 ↑	0 ↑	0 ↑	0 ↑	1 ↖	1 ←	1 ↖
2	G	0 ↑	1 ↖	1 ←	1 ←	1 ↑	2 ↖	2 ←
3	C	0 ↑	1 ↑	1 ↑	2 ↖	2 ←	2 ↑	2 ↑
4	G	0 ↑	1 ↖	1 ↑	2 ↑	2 ↑	3 ↖	3 ←
5	T	0 ↑	1 ↑	2 ↖	2 ↑	2 ↑	3 ↑	3 ↑
6	A	0 ↑	1 ↑	2 ↑	2 ↑	3 ↖	3 ↑	4 ↖
7	G	0 ↑	1 ↖	2 ↑	2 ↑	3 ↑	4 ↖	4 ↑

－ G T C A G － A －
A G － C － G T A G

float scoref(char x, char y)

- 塩基xと塩基yがマッチした時のスコアを返す。
  - DNA配列の場合
    - +1　(x==y)
    - -1　(x!=y)
  - アミノ酸配列の場合
    - BLOSUM（後述）に従う

実行例

```
sun% ./a.out example-sequence1.data example-sequence2.data
score 21.000000
GAGGTTATCAA-AA-GCTACTAGTC-CA
GAGG--AT-AACAAGGCTACTA-TCACA
```

提出までにやること

- scorefがアミノ酸の置換スコアBLOSUMを返すように変更する。

http://www.dna.bio.keio.ac.jp/lecture/progen2/data/score-matrix.c

  – ギャップスコアを-8に変更する。
- ４生物種のヘモグロビンのアミノ酸配列を比較する。

http://www.dna.bio.keio.ac.jp/lecture/progen2/data/hemoglobin-human.data
http://www.dna.bio.keio.ac.jp/lecture/progen2/data/hemoglobin-gorilla.data
http://www.dna.bio.keio.ac.jp/lecture/progen2/data/hemoglobin-chicken.data
http://www.dna.bio.keio.ac.jp/lecture/progen2/data/hemoglobin-horse.data

実行例

```
sun% ./a.out hemoglobin-human.data hemoglobin-chicken.data
score 706.000000
MVHLTPEEKSAVTALWGKVNVDEVGGEALGRLLVVYPWTQRFFESFGDLSTPDAVMGNPKVKAHGK
KVLGAFSDGLAHLDNLKGTFATLSELHCDKLHVDPENFRLLGNVLVCVLAHHFGKEFTPPVQAAYQ
KVVAGVANALAHKYH
MVHWTAEEKQLITGLWGKVNVAECGAEALARLLIVYPWTQRFFASFGNLSSPTAILGNPMVRAHGK
KVLTSFGDAVKNLDNIKNTFSQLSELHCDKLHVDPENFRLLGDILIIVLAAHFSKDFTPECQAAWQ
KLVRVVAHALARKYH
sun% ./a.out hemoglobin-human.data hemoglobin-gorilla.data
score 987.000000
MVHLTPEEKSAVTALWGKVNVDEVGGEALGRLLVVYPWTQRFFESFGDLSTPDAVMGNPKVKAHGK
KVLGAFSDGLAHLDNLKGTFATLSELHCDKLHVDPENFRLLGNVLVCVLAHHFGKEFTPPVQAAYQ
KVVAGVANALAHKYH
MVHLTPEEKSAVTALWGKVNVDEVGGEALGRLLVVYPWTQRFFESFGDLSTPDAVMGNPKVKAHGK
KVLGAFSDGLAHLDNLKGTFATLSELHCDKLHVDPENFKLLGNVLVCVLAHHFGKEFTPPVQAAYQ
KVVAGVANALAHKYH
```

課題の提出

- 今回の課題番号は「13」
- 締切：１月２０日（月）