Alignment Scores and PSI-Blast

These notes can be found near the bottom of the page:

`http://globin.cse.psu.edu/`

```
Query: 59  FSFLKDSAGVQDSPKLQAHAGKVFGMVRDSAAQLRATGGVVLGDATLGAIHIQNGVVDP-
           F  L     V  +PK++AH  KV G   D  A L    G     ATL  +H     VDP
Sbjct: 46  FGDLSTPDAVMGNPKVKAHGKKVLGAFSDGLAHLDNLKGTF---ATLSELHCDKLHVDPE


Query: 118 HFVVVKEALLKTIKESSGDKWSEELSTAWEVAYDALATAI 157
           +F ++   L+  +    G +++  +  A++     +A A+
Sbjct: 103 NFRLLGNVLVCVLAHHFGKEFTPPVQAAYQKVVAGVANAL 142
```

Is this alignment correct? Are the two sequences related, or is this just an alignment of unrelated sequences? Can I find a better alignment of those two sequences?

(Actually, the sequences are a plant globin and a human globin. More on them later.)

What is an Alignment?

What is meant by an “alignment” of two given sequences? In particular, what is a *local* alignment?

Define “Alignment”

An alignment of two sequences (frequently called a *local alignment*) can be obtained as follows.

1. extract a segment from each sequence
2. add dashes (gap symbols) to each segment to create equal-length sequences
3. place one padded segment over the other

For example:

```
AACC-GTACTTG
A-CAGGTGG-TG
```

Alignment Scores

We need to differentiate good alignments from poor ones. We use a rule that assigns a numerical score to any alignment; the higher the score, the better the alignment.

For any proposed rule for scoring an alignment, there are two questions:

1. Given any alignment, can we compute its score?
2. Given two sequences, can we automatically find a local alignment of highest possible score?

For some rules, the second answer is “No”.

Simple Rule for Scoring Alignments

We give a score to each possible column, then add scores of an alignment's columns.

Let a match (column with identical symbols) score 1 and each other column score -1. For example:

```
AACC-GTACTTG
A-CAGGTGC-TG
+-+--++-+-++
```

Total score is 2.

Optimal Alignments

With this scoring method, for any two sequences we can compute a highest-scoring local alignment (in time proportional to the product of the two sequence lengths, using “dynamic programming”).

Needleman and Wunsch (1970); Smith and Waterman (1981)

Unusable Rule for Scoring Alignments

Again, each mismatch scores -1. A match column scores $n/(n+1)$, where n is the number of match columns for that same letter (thus the n identical matches total $n^2/(n+1)$).

```
AACC-GTACTTT
A-CAGGTGC-TT
+-+--++-+-++
```

There are 5 mismatch columns (score -5), 1 A-over-A (score $1/2$), 2 C-over-C (score $4/3$), 1 G-over-G (score $1/2$) and 3 T-over-T (score $9/4$). Total score is $-5/12$.

But given two sequences, can we find an alignment that maximizes this score?

More General Substitution Scores?

How about the following *substitution-score matrix*?

	A	C	G	T
A	91	-114	-31	-123
C	-114	100	-125	-31
G	-31	-125	100	-114
T	-123	-31	-114	91

Optimal alignments under an arbitrary substitution-score matrix can be computed at essentially no penalty in computational time.

More General Substitution Scores?

How about a *position-specific scoring matrix*, that depends on the first sequence being aligned? For ACCTGAT we might want:

	A	C	C	T	G	A	T
A	91	-114	-63	-123	-31	33	27
C	-114	100	55	-31	-125	-42	-29
G	-31	-125	-81	-114	100	-8	-29
T	-123	-31	-112	91	-114	-49	27

Optimal alignments under these scores can be computed at essentially no penalty in computational time.

More General Gap Penalties?

Which alignment is preferable? (They have the same set of columns.)

```
ACAAT
A-A-T
```

or

```
ACAAT
A--AT
```

Gap Penalties (continued)

Let's subtract, say, 1 for each *gap*, i.e., run of consecutive dashes. Thus,

```
ACAAT
A-A-T
```

scores 1 less than does:

```
ACAAT
A--AT
```

Using such a *gap open penalty* roughly doubles the time for computing a highest-scoring alignment.

More General Alignment Scores?

Which alignment is preferable? (They have the same set of columns.)

```
ACTTCTCGAGGA...
||||||
ACTTCTTTTTTT...
```

or

```
ACCGTATGCGTA...
| | | | | |
ATCTTTTTCTTT...
```

What scoring rule makes the right distinction?

Let’s add 1 for each match that immediately follows another match. Thus,

```
ACTTCTCGAGGA...
||||||
ACTTCTTTTTTT...
```

scores 5 more than does:

```
ACCGTATGCGTA...
| | | | | |
ATCTTTTTCTTT...
```

Optimal alignments under these scores can be computed at only a small (say 10%) penalty in computational time.

More General Alignment Scores?

Which alignment is preferable? (Both have 12 matches.)

```
ACACACACACAC
ACACACACACAC
```

or

```
ACCGTATGCGTA
ACCGTATGCGTA
```

What scoring rule makes the right distinction?

Substitution Scores for Protein Sequences

Which alignment column should be given the higher score?

```
A
A
```

or

```
W
W
```

The point is that A occurs substantially more frequently in protein sequences than does W.

BLOSUM62 Substitution Scores

	A	R	N	I	L	W	Y
A	5	−2	−1	−1	−2	−3	−2
R	−2	7	−1	−4	−3	−3	−1
N	−1	−1	7	−3	−4	−4	−2
I	−1	−4	−3	5	2	−3	−1
L	−2	−3	−4	2	5	−2	−1
W	−3	−3	−4	−3	−2	15	2
Y	−2	−1	−2	−1	−1	2	8

(20 columns)

(20 rows)

Which Substitution Scores Should I Use?

Blastp (for protein sequences) uses Blosum62 by default, but offers other scores (BLOSUM80, BLOSUM45, PAM30, PAM70) as options. In theory, BLOSUM80, PAM30 and PAM70 are tuned to work better for detecting relatively similar sequences using shorter matches. BLOSUM45 might be useful for identifying extremely distant matches.

A reasonable rule of thumb is to completely ignore these alternative scoring systems. There is, however, a more radically different way to score alignments that is frequently useful.

A Leghemoglobin Sequence

For the following example, we use the following plant globin sequence, which is a distant relative of animal globins:

```
>CAA38024.1 alfalfa leghemoglobin
MQIQIAKQKQKNKKRNMGFTEKQEALVNSSFE
SFKQNPGYSVLFYTIILEKAPAAKGMFSFLKD
SAGVQDSPKLQAHAGKVFGMVRDSAAQLRATG
GVVLGDATLGAIHIQNGVVDPHFVVVKEALLK
TIKESSGDKWSEELSTAWEVAYDALATAIKKAMS
```

PSI-Blast (Protein Sequences Only)

Searching with the plant globin sequence, Blastp gives 388 hits; number 166 is with human beta globin:

```
NP_000509.1 hemoglobin, beta ...  0.094
```

The E-value 0.094 means that a match of this score with an unrelated sequence would occur about 10% of the time. Results of PSI-Blast iteration 1 (391 hits) include:

```
NP_000509.1 hemoglobin, beta ...  0.068
```

Results of PSI-Blast iteration 2 (965 hits) include:

```
NP_000509.1 hemoglobin, beta ...  0.004
```

Results of PSI-Blast iteration 3 (1660 hits) include:

```
NP_000509.1 hemoglobin, beta ...  2e-33
```

Which Positions are Critical?

Consider some trustworthy Blastp alignments of the plant globin to some fairly distant relatives.

```
NP_435895.1 putative flavohemo..  0.0004
BAA81644.1 bacterial hemoglob...  0.001
```

Look at positions 106-125 of the leghemoglobin sequence:

```
alfalfa:      GAIHIQNGVVDPHFVVVKEA
flavohemo:    AHKHASLGVRPEQYPIVGEH
bacterial:    GVIHCNAKVQPEHYPIVGKH
                 H    V        V
```

PSI-Blast Learns and Uses Position-Specific Scores

PSI-Blast learned this about positions 106-125 of the leghemoglobin sequence:

```
alfalfa:      GAIHIQNGVVDPHFVVVKEA
flavohemo:    AHKHASLGVRPEQYPIVGEH
bacterial:    GVIHCNAKVQPEHYPIVGKH
                 H    V         V
```

Original Blastp run:

```
alfalfa:      GAIHIQNGVVDP-HFVVVKEA
human:        SELHCDKLHVDPENFRLLGNV
                 H     VDP  F
```

After third iteration of PSI-Blast:

```
alfalfa       GAIHI-QNGVVDPHFVVVKEA
human         SELHCDKLHVDPENFRLLGNV
                 H     V    F
```

When Is PSI-Blast Better Than Blastp?

PSI-Blast can beat Blastp **if Blastp finds some reliable alignments to database sequences.** (Moderately distant matches are particularly useful.) Then, PSI-Blast (which starts by running Blastp) can determine which positions in the query sequence are conserved during evolution and devise an appropriate **Position-Specific Scoring Matrix**, which can be used to identify relatives at a further evolutionary distance.

If the original Blastp run cannot find any reliable alignment, PSI-Blast is powerless.