

3 BLAST and FASTA

This lecture is based on the following papers, which are all recommended reading:

- D.J. Lipman and W.R. Pearson, Rapid and Sensitive Protein Similarity Searches. *Science* 227, 1435-1441 (1985).
- Pearson, W.R. and Lipman, D.J. Improved tools for biological sequence comparison. *PNAS USA* 85, 2444-2448 (1988).
- S.F. Altschul, W. Gish, W. Miller, E.W. Myers and D.J. Lipman. *Basic local alignment search tool*, *J. Molecular Biology*, 215:403-410 (1990).
- <http://www.ncbi.nlm.nih.gov/BLAST/tutorial/Altschul-1.html>, accessed April 2016.
- D. Gusfield, Algorithms on strings, trees and sequences, pg. 376–381, 1997.

3.1 BLAST

Pairwise alignment is used to detect homologies between different protein or DNA sequences, either as global or local alignments.

This can be solved using dynamic programming in time proportional to the product of the lengths of the two sequences being compared.

However, this is too slow for searching current databases and in practice algorithms are used that run much faster, at the expense of possibly missing some significant hits due to the heuristics employed. Such algorithms usually *seed and extend* approaches in which first small exact matches are found, which are then extended to obtain long inexact ones.

3.1.1 BLAST terminology

BLAST, the Basic Local Alignment Search Tool, is perhaps the most widely used bioinformatics tool ever written. It is an alignment heuristic that determines local alignments between a *query* and a *database*.

Let q be the query and d the database. A *segment* is simply a substring s of q or d .

A *segment-pair* (s, t) consists of two segments, one in q and one d , of the same length.

```

      x x x x x V A L L A R   x x x
...x x x x x x P A M M A R   x x x x x x...

```

We think of s and t as being aligned without gaps and *score* this alignment using a substitution score matrix, such as BLOSUM62.

The alignment score for (s, t) is denoted by $\sigma(s, t)$.

Definition 3.1.1 (HSP) Let C be a given minimum score threshold. A segment pair (s, t) is called a high-scoring segment pair (HSP), if it is locally maximal and $\sigma(s, t) \geq C$.

Locally maximal is defined by the X-drop algorithm, described below.

Finally, a *word* is simply a short substring.

The goal of BLAST is to compute all HSPs between two sequences (or a query and a database), for a given minimum score threshold C .

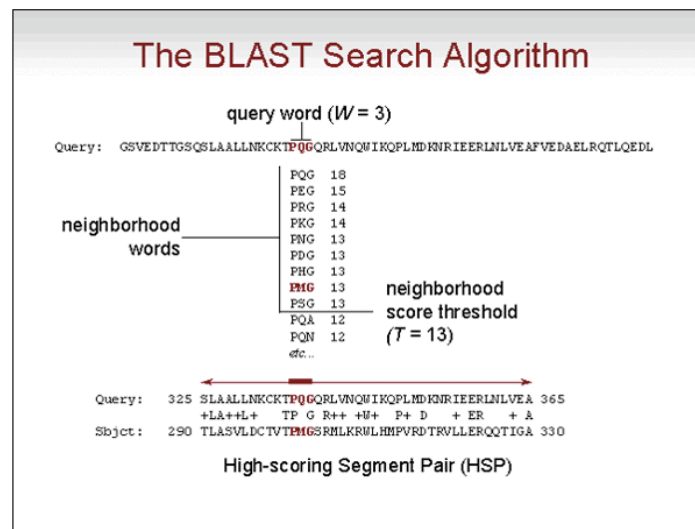
3.1.2 The BLAST algorithm

The BLAST algorithm has three parameters: The *word size* W , *word similarity threshold* T and *minimum match score* C .

For *protein* sequences, BLAST operates as follows:

1. The list ℓ of all words of length W that have similarity $\geq T$ to some word in the query sequence q is generated.
2. The database sequence d is scanned to detect each hit t of a word s from the list ℓ .
3. Each such pair (s, t) (called a *seed*) is extended in either direction until the running score drops X below the best score seen so far. (This is called the *X-drop algorithm*.)
4. For each seed match, the best extension is reported, if it has score $\geq C$.

In practice: W is 2 – 4 for proteins.



(<https://www.cs.umd.edu/class/fall2011/cmsc858s/BLAST.pdf>, accessed April 2016)

With care, the list of all words of length W that have similarity $\geq T$ to some word in the query sequence q can be produced in time proportional to the number of words in the list.

These are placed in a “keyword tree” and then, for each word in the tree, all exact locations of the word in the database d are detected in time proportional to the length of d .

The original version of BLAST did not allow “indels” (insertions or deletions), making hit extension very fast.

Note that the use of seeds of length W and the X -drop algorithm speed up the algorithm, but imply that BLAST is a heuristic that is not guaranteed to find all high-scoring alignments.

For *DNA* sequences, BLAST operates as follows:

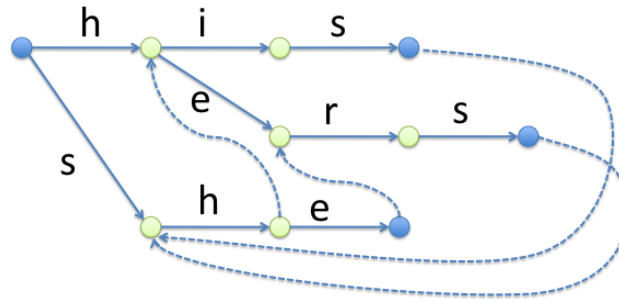
- The list L of all words of length W in the query sequence q is generated.
- The database d is scanned for all hits of words in L . Blast uses a two-bit encoding for DNA. This saves space and also search time, as four bases are encoded per byte.

In practice, W is around 12 for DNA.

3.1.3 Keyword tree

Assume that we want find occurrences of the words “his”, “hers” and “she” in the text “shishers”.

We can use a *keyword tree* to perform the search, which is a finite state automaton used in pattern matching:



As we read the text from left to right, we move through the graph following arrows with matching labels, if we can, or jumping back to the root, if not. Whenever we complete a word in the graph, we print out its text location. The dashed arrows are followed to find overlapping matches.

Occurrences: “his” at position 2, “hers” at position 5, “she” at position 4.

3.1.4 The BLAST family

There are a number of different variants of the BLAST program:

- BLASTN: compares a DNA query sequence to a DNA sequence database,
- BLASTP: compares a protein query sequence to a protein sequence database,
- TBLASTN: compares a protein query sequence to a DNA sequence database (6 frames translation),
- BLASTX: compares a DNA query sequence (6 frames translation) to a protein sequence database, and
- TBLASTX: compares a DNA query sequence (6 frames translation) to a DNA sequence database (6 frames translation).

Input:

```
>000078_0077_0290 length=107
GGGTTGAGGCTGTCGGTAAACACCTGGGCGCGGGTGATATGGCCTTTTCAACGTCGAAAT
GCAGTTCACGCGCCAGGTAAAGCGTTCATCCAGCAGATGCGAGAA
```

Example of BLASTN result:

```
BLASTN 2.2.13 [Nov-27-2005]

Query= 000078_0077_0290 length=107 (107 letters)

Database: EcoliK12.fna 1 sequences; 4,639,675 total letters

>gi|49175990|ref|NC_000913.2| Escherichia coli K12, complete genome
Length = 4639675

Score = 172 bits (87), Expect = 4e-44
Identities = 106/110 (96%), Gaps = 3/110 (2%)
Strand = Plus / Plus

Query: 1      gggttgaggctgtcggtaaacacctgggcgcggggtgatatggcc-ttttcaacgtcgaaa 59
            |||
Sbjct: 4621278 gggttgaggctgtcggtaaacacctgggcgcggggtgatatggccttttcaacgtcgaaa 4621337

Query: 60      tgcagttccacgc--gcccaggtaaacggttcacagcagatgcgagaa 107
            |||
Sbjct: 4621338 tgcagttccacgcgcccaggtaaacggttcacagcagatgcgagaa 4621387
```

Example of BLASTX result:

```

BLASTX 2.2.13 [Nov-27-2005]

Query= 000078_0077_0290 length=107 (107 letters)

Database: nr 3,044,223 sequences; 1,047,289,308 total letters

>gi|16132203|ref|NP_418803.1| lipoate-protein ligase A [Escherichia coli K12]
lipoate-protein ligase A [Escherichia coli K12]
yjjf [Escherichia coli] Lipoate-protein ligase A lipoate-protein ligase A
      Length = 338

Score = 50.4 bits (119), Expect = 2e-05
Identities = 27/36 (75%), Positives = 27/36 (75%), Gaps = 1/36 (2%)
Frame = -1

Query: 107 FSHLLDERFTWARGTA-FRR*KGHITRAQVFTDSL 3
          FSHLLDERFTW      F KGHITRAQVFTDSL
Sbjct: 252 FSHLLDERFTWGGVELHFDVEKKGHITRAQVFTDSL 287

```

3.1.5 The BLAST E-value

Question: Suppose we have computed an HSP (s, t) with score S , how significant is this match?

In the following, we assume that the length m and n of the query and database are sufficiently large. HSP scores depend on two parameters, K and λ .

Definition 3.1.2 (E-value) *The number of HSPs with score $\geq S$ that one can expect to see by random chance is known as the E-value, which equals:*

$$E = K m n e^{-\lambda S}.$$

The parameters K and λ depend on the background probabilities of the symbols and on the employed scoring matrix. Essentially, K and λ are scaling-factors for the search space and for the scoring scheme, respectively.

3.1.6 The BLAST bit score

In addition to the E-value, BLAST also reports a so-called *bit score*.

For a given HSP (s, t) we transform the *raw* score $S = \sigma(s, t)$ into a *bit score* thus:

$$S' = \frac{\lambda S - \ln K}{\ln 2}.$$

Such bit scores make it easier to compare between different BLAST searches.

The bit score hides the two parameters K and λ . Given the bit score of a match, one can easily compute the E-value without knowing the parameters K and λ , using only the length of the query and database. The E-value is obtain from a bit score S' as follows:

$$E = m n 2^{-S'}.$$

To see this, first solve for S in the equation for S' above and then plug the result into the original E-value equation.

3.1.7 The BLAST P-value

The number of HSPs (s, t) with $\sigma(s, t) \geq S$ that one obtains when comparing against a database of *random sequence* is given by a Poisson distribution. So, the probability of finding exactly k HSPs with a score $\geq S$ is given by

$$P(k) = e^{-E} \times \frac{E^k}{k!},$$

where E is the E -value for S .¹

Definition 3.1.3 (P-value) *The probability of finding at least one HSP with a score $\geq S$ “by chance” is*

$$P(k \geq 1) = 1 - P(0) = 1 - e^{-E},$$

called the P-value.

BLAST reports E -values rather than P -values because it is easier, for example, to interpret the difference between an E -value of 5 and 10, than to interpret the difference between a P -value of 0.993 and 0.99995.

3.2 FASTA

FASTA² (pronounced fast-ay) is a heuristic for finding significant matches between a query string q and a database string d .

FASTA's general strategy is to quickly find the most significant diagonals in the dynamic programming matrix.

The performance of the algorithm is influenced by a *word-size* parameter k , usually 6 for DNA and 2 for amino acids.

The first step of the algorithm is to determine all exact matches of length k between the two sequences, called *hot-spots*.

To find these exact matches quickly, a hash table is built that consists of all words of length k that are contained in the query sequence. Exact matches are then found by look-up of each word of length k contained in the database.

A *hot-spot* is given by (i, j) , where i and j are the *locations* (i.e., start positions) of an exact match of length k .

Any such hot-spot (i, j) lies on the diagonal $(i - j)$ of the dynamic programming matrix.

Using this scheme, the main diagonal has number 0, whereas diagonals above the main one have positive numbers, the ones below negative.

A *diagonal run* is a set of hot-spots that lie in a consecutive sequence on the same diagonal. It corresponds to a gapless local alignment.

A score is assigned to each diagonal run. This is done by giving a positive score to each match (using e.g. the PAM250 match score matrix in the case of proteins) and a negative score for gaps in the run.

The algorithm then locates the ten best diagonal runs.

Each of the ten diagonal runs is re-scored using the match score matrix and the best-scoring sub-alignment of each is extracted.

The next step is to combine high scoring sub-alignments into a single larger alignment, allowing the introduction of gaps into the alignment.

Finally, a *banded* Smith-Waterman dynamic program is used to produce an optimal local alignment along the best matched regions.

In this way, FASTA determines a highest scoring region, not all high scoring alignments between two sequences. Hence, FASTA may miss instances of repeats or multiple domains shared by two proteins.

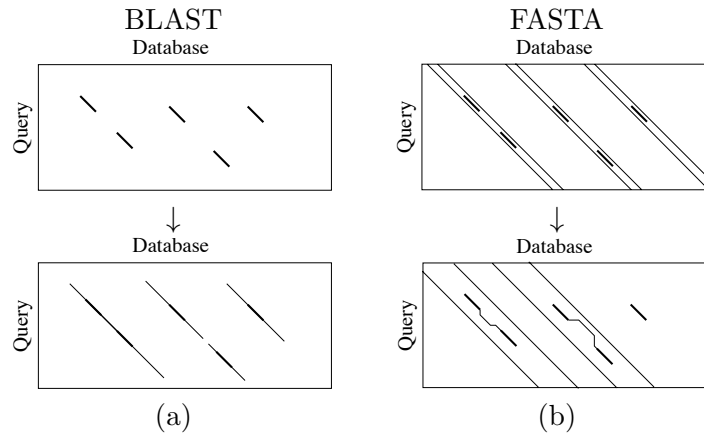
¹ Recall **Poisson distribution**: The probability that exactly k events occurs in a fixed time period, given that the expected number of events is λ in that time period, is given by:

$$P(k, \lambda) = \frac{\lambda^k e^{-\lambda}}{k!}.$$

Example: Assume that a call center gets 10 calls per hour, on average. The probability that it will get 100 calls in a given hour is $P(100, 10)$.

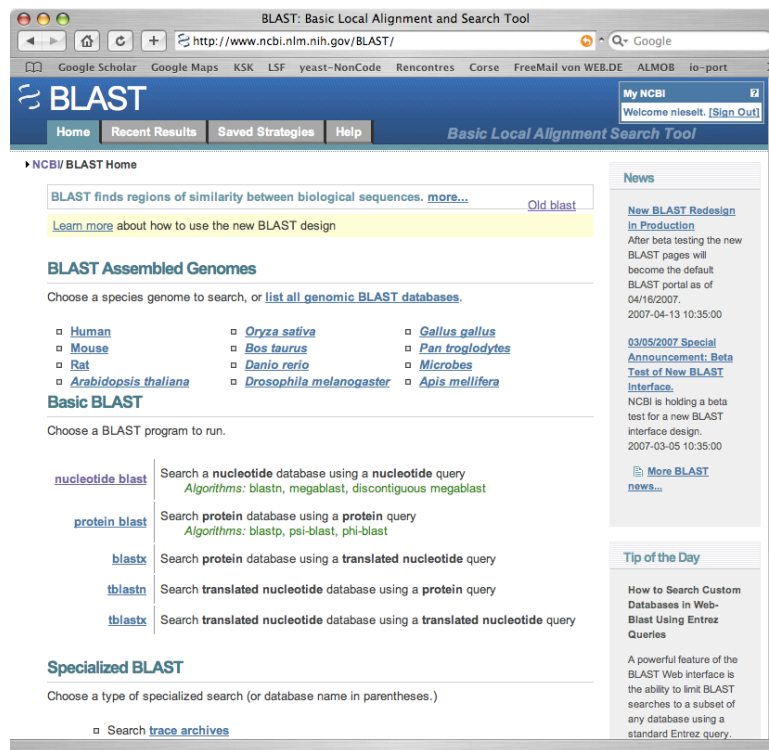
²Lipman, DJ; Pearson, WR (1985). Rapid and sensitive protein similarity searches. Science 227 (4693): 1435-41

3.3 BLAST and FASTA



(a) In BLAST, individual seeds are found and then extended without indels. (b) In FASTA, individual seeds contained in the same diagonal are merged and the resulting segments are then connected using a banded Smith-Waterman alignment.

3.4 BLAST as a web service



<http://www.ncbi.nlm.nih.gov/BLAST/>

3.4.1 BLAST run example

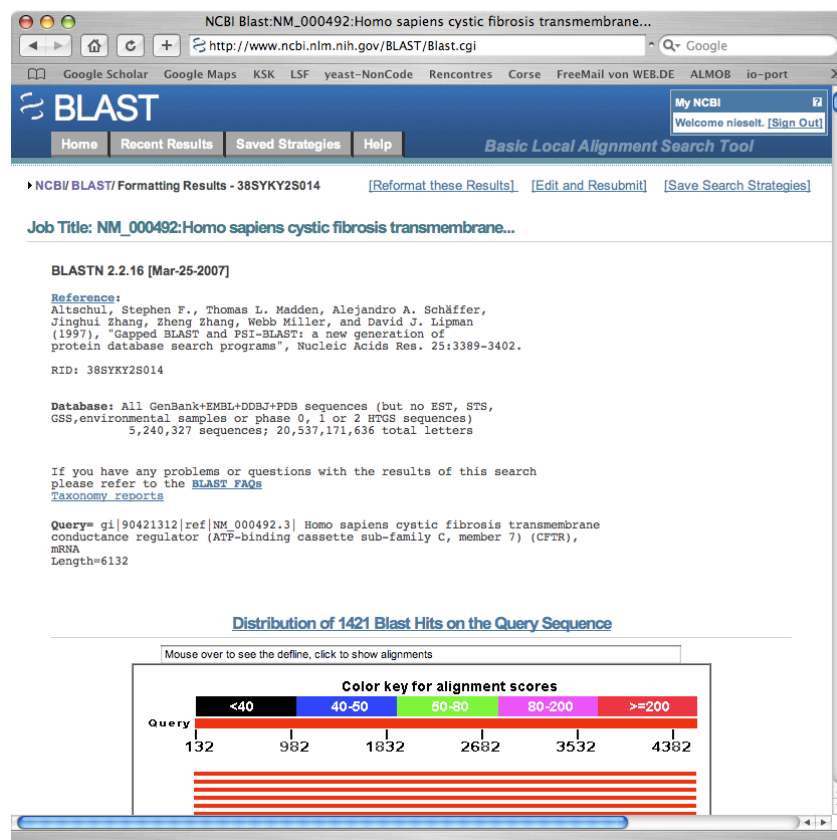
The image displays two screenshots of the NCBI BLAST web interface, showing the steps to configure a BLAST search.

Top Screenshot: Enter Query Sequence and Choose Search Set

- Enter Query Sequence:**
 - Enter accession number, gi, or FASTA sequence: 90421312
 - Query subrange: From 133, To 4575
 - Or, upload file: Choose File (no file selected)
 - Job Title: NM_000492:Homo sapiens cystic fibrosis transmembrane...
- Choose Search Set:**
 - Database: Nucleotide collection (nr/nt)
 - Organism: Any (selected), Human, A. thaliana, Mouse, Custom...
 - Entrez Query: (empty)
- Program Selection:**
 - Optimize for: Somewhat similar sequences (blastn) (selected)
 - Choose a BLAST algorithm: (dropdown)

Bottom Screenshot: Algorithm parameters

- BLAST:** Search database nr using Blastn (Optimize for somewhat similar sequences). Show results in a new window (checked).
- Algorithm parameters:**
 - General Parameters:**
 - Max target sequences: 100
 - Short queries: Automatically adjust parameters for short input sequences (checked)
 - Expect threshold: 1e-07
 - Word size: 7
 - Scoring Parameters:**
 - Match/Mismatch Scores: 2,-3
 - Gap Costs: Existence: 5, Extension: 2
 - Filters and Masking:**
 - Filter: Low complexity regions (checked), Species-specific repeats for: Human
 - Mask: Mask for lookup table only (checked), Mask lower case letters
- BLAST:** Search database nr using Blastn (Optimize for somewhat similar sequences). Show results in a new window (checked).



3.4.2 BLAST run output

NCBI BLAST:NM_000492:Homo sapiens cystic fibrosis transmembrane...

http://www.ncbi.nlm.nih.gov/BLAST/Blast.cgi

Google Scholar Google Maps KSK LSF yeast-NonCode Rencontres Corse FreeMail von WEB.DE ALMOB io-port Entrez PubMed NCBI BLAST

Distance tree of results **NEW**

Legend for links to other resources: [U](#) UniGene [E](#) GEO [G](#) Gene [S](#) Structure [M](#) Map Viewer

Sequences producing significant alignments:
(Click headers to sort columns)

Accession	Description	Max score	Total score	Query coverage	E value	Max ident	Links
NM_000492.3	Homo sapiens cystic fibrosis transmembrane conductance	8013	8013	72%	0.0	100%	U G
M28668.1	Human cystic fibrosis mRNA, encoding a presumed transmembrane	7999	7999	72%	0.0	99%	U G
XM_519330.2	PREDICTED: Pan troglodytes cystic fibrosis transmembrane	7950	7950	72%	0.0	99%	U G
NM_001032938.1	Macaca mulatta cystic fibrosis transmembrane conductance	7705	7705	72%	0.0	98%	U G
AY608405.1	Cysteine-free hCFTR in pGEMHE, complete sequence	7643	7643	72%	0.0	98%	U G
AY585334.1	Sus scrofa cystic fibrosis transmembrane conductance reg	6163	6163	72%	0.0	90%	U G
NM_001009781.1	Ovis aries cystic fibrosis transmembrane conductance reg	6123	6123	72%	0.0	90%	U G
NM_174018.2	Bos taurus cystic fibrosis transmembrane conductance reg	6118	6118	72%	0.0	90%	U G
NM_001082716.1	Oryctolagus cuniculus cystic fibrosis transmembrane cond	6015	6015	72%	0.0	90%	U G
NM_001007143.1	Canis familiaris cystic fibrosis transmembrane conductance	5873	5873	72%	0.0	89%	U G
U40227.1	Oryctolagus cuniculus CFTR chloride channel mRNA, compl	5238	5962	70%	0.0	92%	U G
AY916796.1	Trichosurus vulpecula cystic fibrosis transmembrane cond	4560	4560	72%	0.0	82%	U G
NM_001368413.1	PREDICTED: Monodelphis domestica similar to cystic fibros	4498	4498	71%	0.0	82%	U G
NM_021050.2	Mus musculus cystic fibrosis transmembrane conductance	4264	4264	72%	0.0	81%	U G
M60493.1	Mouse cystic fibrosis transmembrane conductance regulat	4264	4264	72%	0.0	81%	U G
M69298.1	Mouse cystic fibrosis transmembrane conductance regulat	4233	4233	72%	0.0	81%	U G
XM_001059206.1	PREDICTED: Rattus norvegicus cystic fibrosis transmembr	4183	4183	72%	0.0	80%	U G
XM_001062374.1	PREDICTED: Rattus norvegicus cystic fibrosis transmembr	4183	4183	72%	0.0	80%	U G
NM_031506.1	Rattus norvegicus cystic fibrosis transmembrane conduct	4183	4183	72%	0.0	80%	U G
AY299332.1	Synthetic construct CFTR/EGFP fusion protein mRNA, comp	2695	2695	24%	0.0	99%	U G
X65256.1	X.laevis CFTR mRNA	2634	2634	72%	0.0	73%	U G
U60209.1	Xenopus laevis cystic fibrosis transmembrane conductance	2598	2598	72%	0.0	73%	U G
AK036261.1	Mus musculus adult male cecum cDNA, RIKEN full-length e	1642	1642	25%	0.0	83%	U G
XM_001249650.1	PREDICTED: Bos taurus similar to cystic fibrosis transmem	1559	1559	17%	0.0	92%	U G
M89906.1	Rattus norvegicus cystic fibrosis transmembrane conduct	1326	1326	25%	0.0	79%	U G
DQ388145.1	Homo sapiens isolate cfr13838_B cystic fibrosis transmem	1319	8178	72%	0.0	100%	U G
DQ388144.1	Homo sapiens isolate cfr13838_A cystic fibrosis transmem	1319	7208	63%	0.0	100%	U G
DQ388143.1	Homo sapiens isolate cfr11589_B cystic fibrosis transmem	1319	8173	72%	0.0	100%	U G
DQ388142.1	Homo sapiens isolate cfr11589_A cystic fibrosis transmem	1319	8178	72%	0.0	100%	U G
DQ388141.1	Homo sapiens isolate cfr11521_B cystic fibrosis transmem	1319	8178	72%	0.0	100%	U G
DQ388140.1	Homo sapiens isolate cfr11521_A cystic fibrosis transmem	1319	8173	72%	0.0	100%	U G
DQ388139.1	Homo sapiens isolate cfr11376_B cystic fibrosis transmem	1319	8173	72%	0.0	100%	U G
DQ388138.1	Homo sapiens isolate cfr11376_A cystic fibrosis transmem	1319	8167	72%	0.0	100%	U G
DQ388137.1	Homo sapiens isolate cfr11373_B cystic fibrosis transmem	1319	7719	68%	0.0	100%	U G
DQ388136.1	Homo sapiens isolate cfr11373_A cystic fibrosis transmem	1319	7113	63%	0.0	100%	U G
DQ388135.1	Homo sapiens isolate cfr11321_B cystic fibrosis transmem	1319	8167	72%	0.0	100%	U G