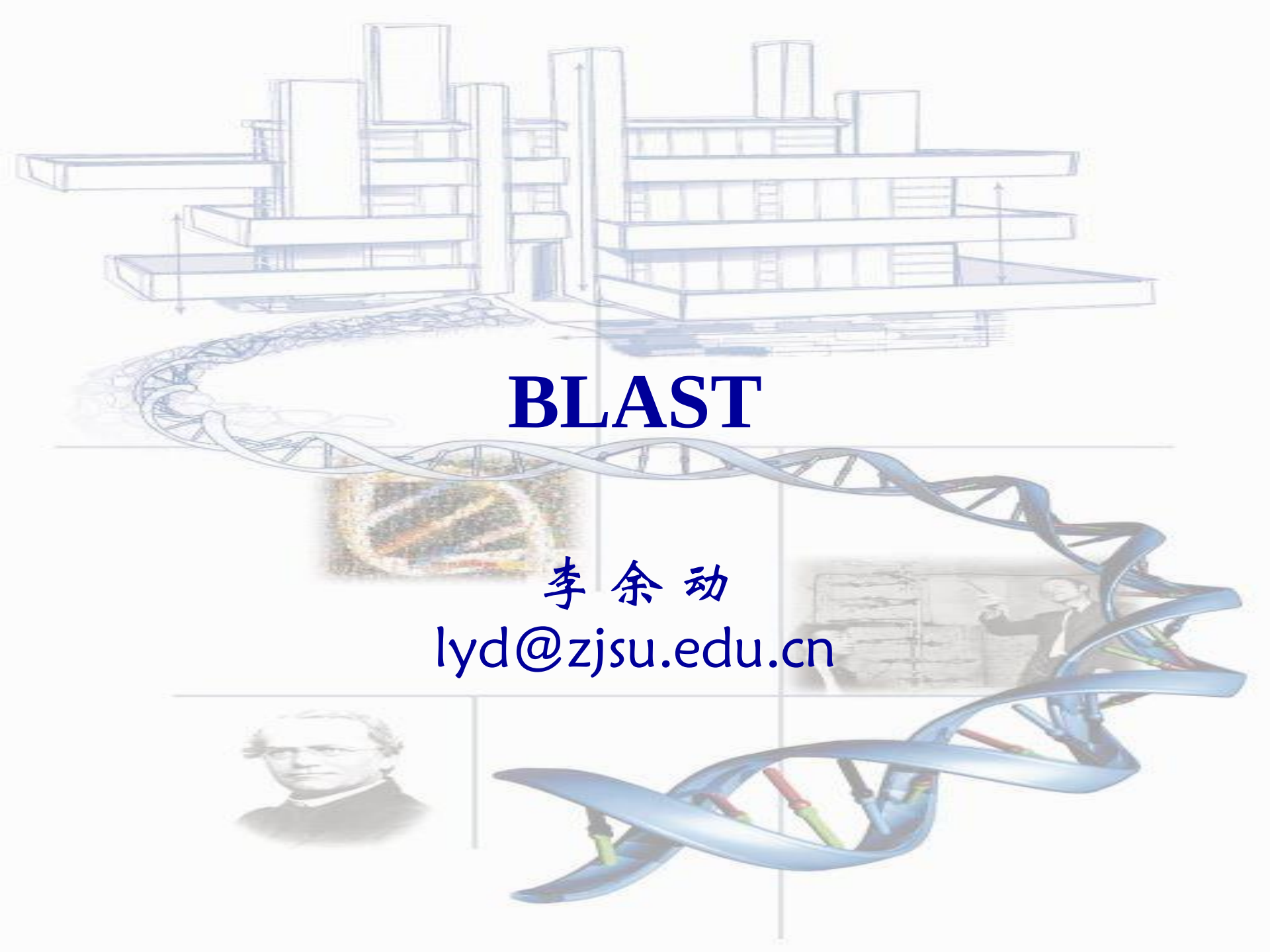


BLAST

李 余 动
lyd@zjsu.edu.cn

Outline

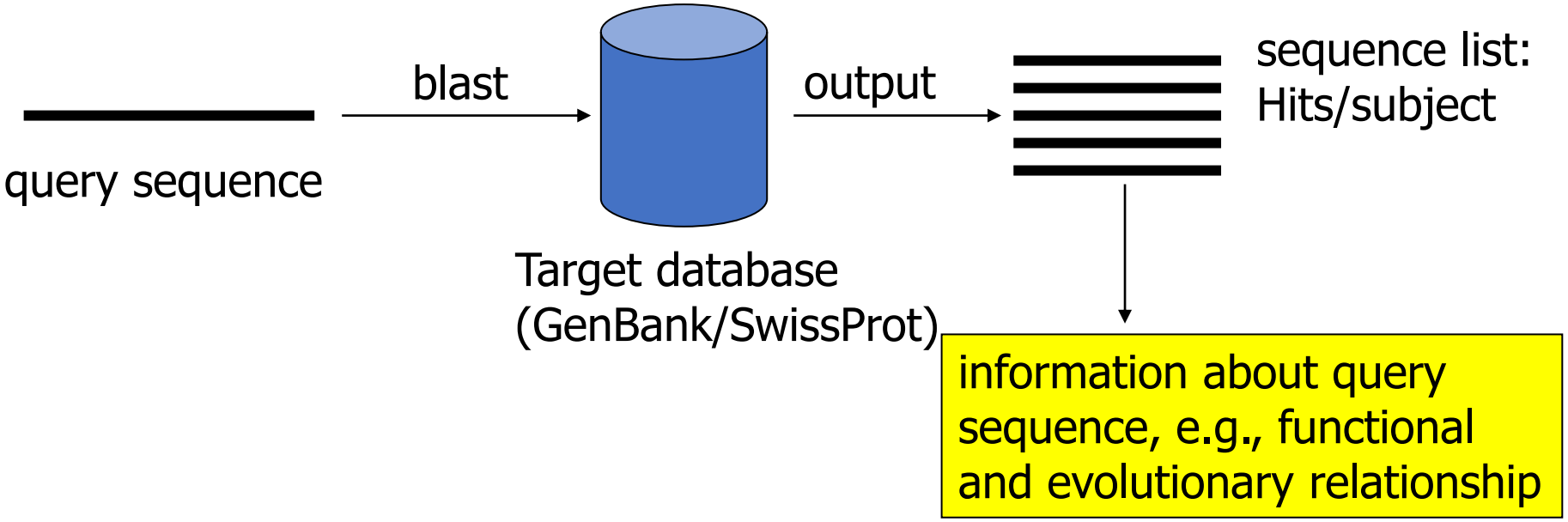
BLAST简介

NCBI网络BLAST使用

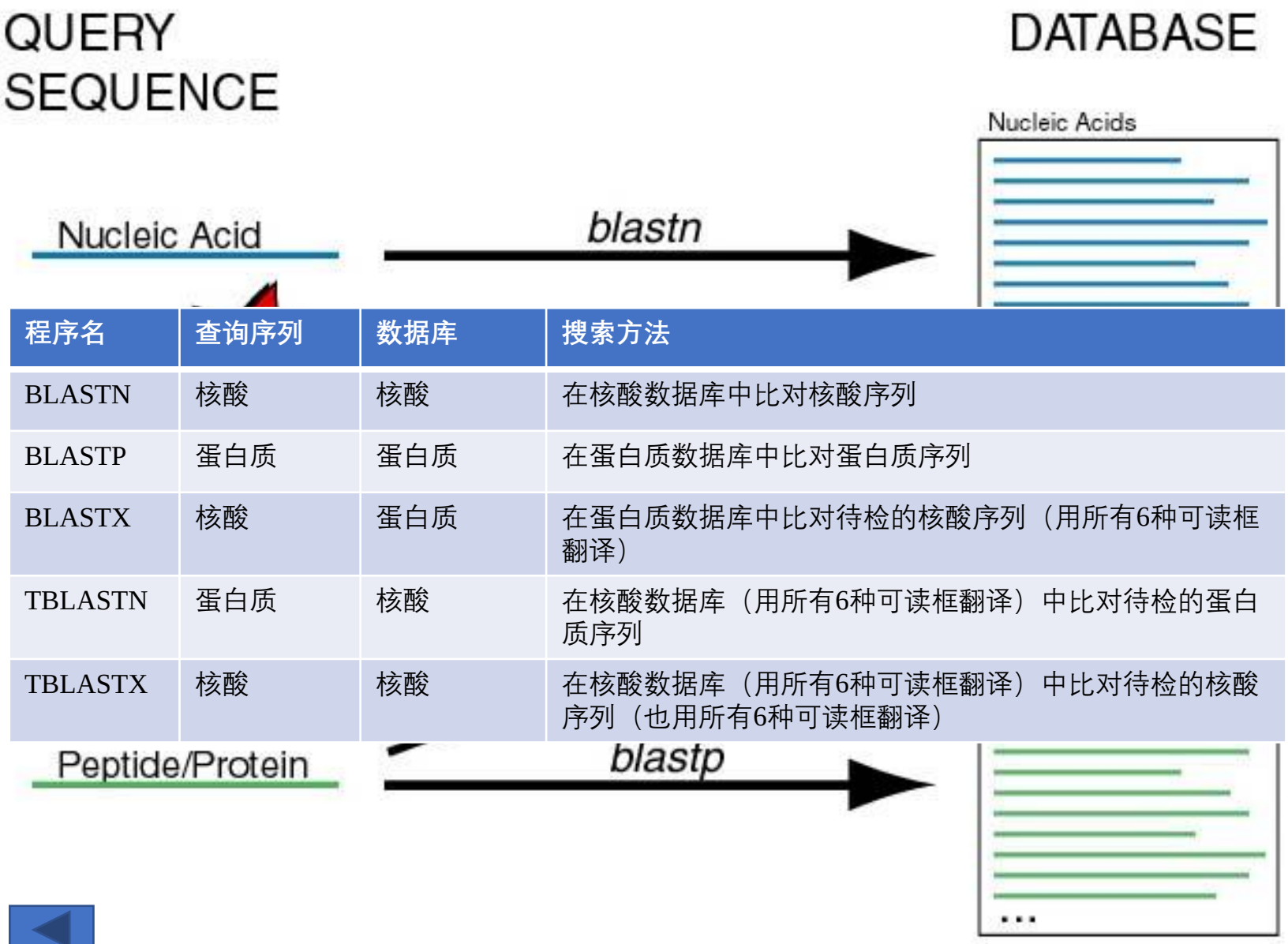
本地BLAST使用

序列数据库搜索

- BLAST： Basic Local Alignment Search Tool (基本局部比对搜索工具)
  - 把查询序列（query sequence）与数据库中的序列进行快速序列比对，找出与查询序列相似的目标序列（subject sequence）。
  - Allows rapid sequence comparison of a query sequence against a database. (牺牲灵敏度，提高计算速度)



主要BLAST程序



程序名	查询序列	数据库	搜索方法
BLASTN	核酸	核酸	在核酸数据库中比对核酸序列
BLASTP	蛋白质	蛋白质	在蛋白质数据库中比对蛋白质序列
BLASTX	核酸	蛋白质	在蛋白质数据库中比对待检的核酸序列（用所有6种可读框翻译）
TBLASTN	蛋白质	核酸	在核酸数据库（用所有6种可读框翻译）中比对待检的蛋白质序列
TBLASTX	核酸	核酸	在核酸数据库（用所有6种可读框翻译）中比对待检的核酸序列（也用所有6种可读框翻译）

BLAST算法论文

J. Mol. Biol. (1990) 215, 403–410

Basic Local Alignment Search Tool

Stephen F. Altschul[1], Warren Gish[1], Webb Miller[2]
Eugene W. Myers[3] and David J. Lipman[1]

[1]*National Center for Biotechnology Information*
National Library of Medicine, National Institutes of Health
Bethesda, MD 20894, U.S.A.

[2]*Department of Computer Science*
The Pennsylvania State University, University Park, PA 16802, U.S.A.

[3]*Department of Computer Science*
University of Arizona, Tucson, AZ 85721, U.S.A.

(Received 26 February 1990; accepted 15 May 1990)

A new approach to rapid sequence comparison, basic local alignment search tool (BLAST), directly approximates alignments that optimize a measure of local similarity, the maximal segment pair (MSP) score. Recent mathematical results on the stochastic properties of MSP scores allow an analysis of the performance of this method as well as the statistical significance of alignments it generates. The basic algorithm is simple and robust; it can be implemented in a number of ways and applied in a variety of contexts including straightforward DNA and protein sequence database searches, motif searches, gene identification searches, and in the analysis of multiple regions of similarity in long DNA sequences. In addition to its flexibility and tractability to mathematical analysis, BLAST is an order of magnitude faster than existing sequence comparison tools of comparable sensitivity.

1. Introduction

The discovery of sequence homology to a known protein or family of proteins often provides the first clues about the function of a newly sequenced gene. As the DNA and amino acid sequence databases continue to grow in size they become increasingly useful in the analysis of newly sequenced genes and proteins because of the greater chance of finding such homologies. There are a number of software tools for searching sequence databases but all use some measure of similarity between sequences to distinguish biologically significant relationships from chance similarities. Perhaps the best studied measures are those used in conjunction with variations of the dynamic programming algorithm (Needleman & Wunsch, 1970; Sellers, 1974; Sankoff & Kruskal, 1983; Waterman, 1984). These methods assign scores to insertions, deletions and replacements, and compute an alignment of two sequences that corresponds to the least costly set of such mutations. Such an alignment may be thought of as minimizing the evolutionary distance or maximizing the similarity between the two sequences compared. In either case, the cost of this alignment is a measure of similarity; the algorithm guarantees it is optimal, based on the given scores. Because of their computational requirements, dynamic programming algorithms are impractical for searching large databases without the use of a supercomputer (Gotoh & Tagashira, 1986) or other special purpose hardware (Coulson *et al.*, 1987).

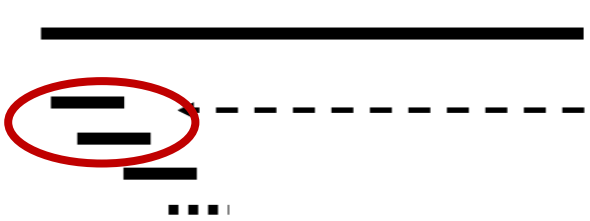
Rapid heuristic algorithms that attempt to approximate the above methods have been developed (Waterman, 1984), allowing large databases to be searched on commonly available computers. In many heuristic methods the measure of similarity is not explicitly defined as a minimal cost set of mutations, but instead is implicit in the algorithm itself. For example, the FASTP program (Lipman & Pearson, 1985; Pearson & Lipman, 1988) first finds locally similar regions between two sequences based on identities but not gaps, and then rescores these regions using a measure of similarity between residues, such as a PAM matrix (Dayhoff *et al.*, 1978) which allows conservative replacements as well as identities to increment the similarity score. Despite their rather indirect approximation of minimal evolution measures, heuristic tools such as FASTP have been quite popular and have identified many distant but biologically significant relationships.

Stephen Altschul, PhD

SF Altschul, W Gish, W Miller, EW Myers, DJ Lipman. Basic local alignment search tool. *Journal of Molecular Biology*, 1990, 215 (3): 403-410. 累计被引用48148次

BLAST算法原理: Seeding-and-Extending

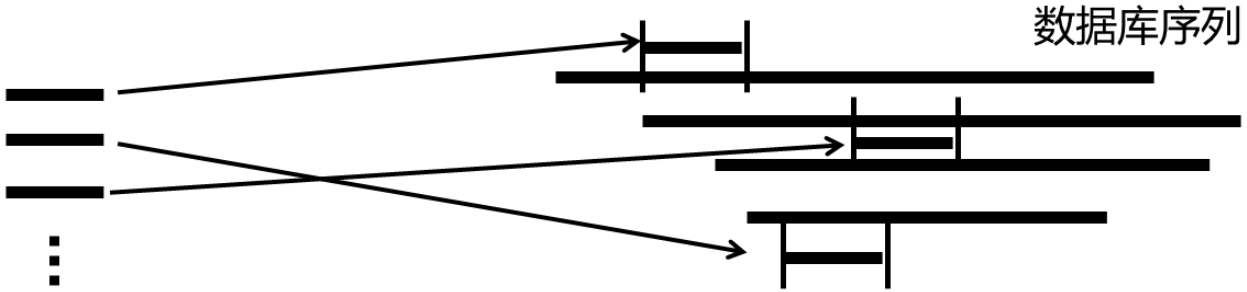
① 查询序列分为不同的words列表



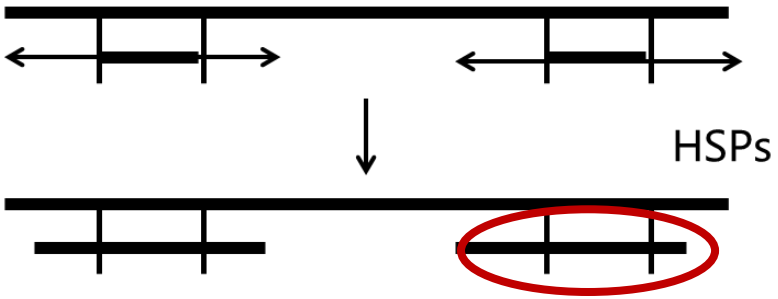
长度为L的查询序列

最多[L-W + 1]个words（通常蛋白质W = 3）

② Words列表与数据库进行比较，确保精确匹配

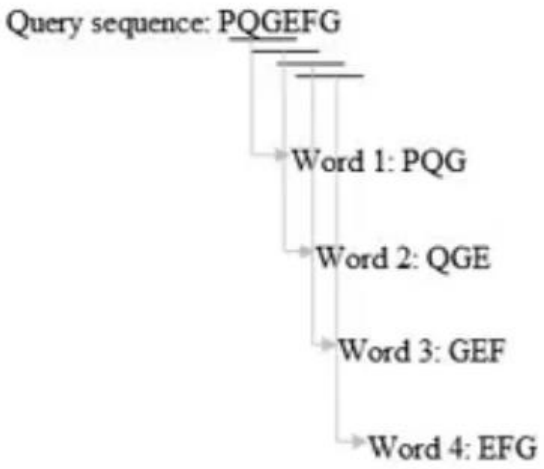


③ 从每个匹配的word的两端进行延伸，直到局部比对得分低于给定的阈值S



STEP 1 :Seeding

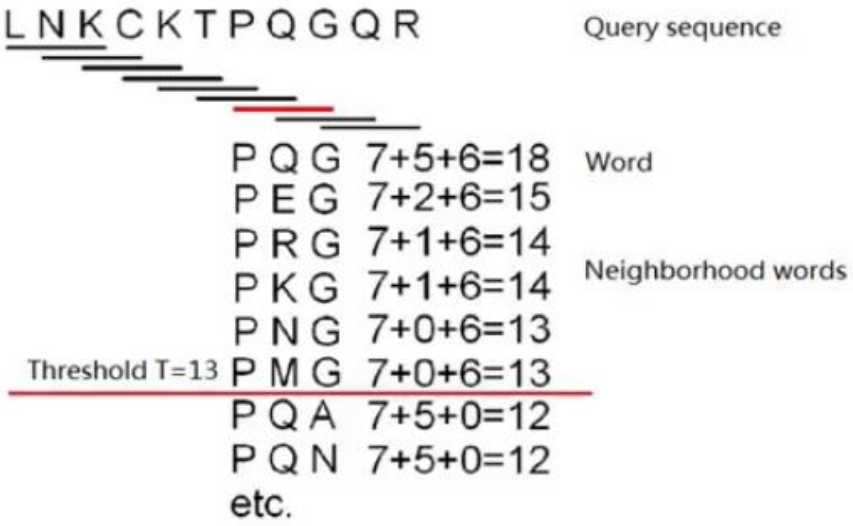
- 将查询序列划分多个固定长度为w的“seed word”：
  - 蛋白质通常以3个氨基酸为一个seed word;
  - DNA通常以11个碱基为一个seed word;
  - word的长度越小，最终的比对准确率就越高，所需要的时间也越长;
  - 长度为N的序列，seed word数量为N-w+1.



- -*W* flag：设定seed word长度的参数

STEP 2 :search Word Hits

- 查询打分矩阵：
  - 蛋白质：BLOSUM或PAM;
  - DNA：match +5分，mismatch -4分.

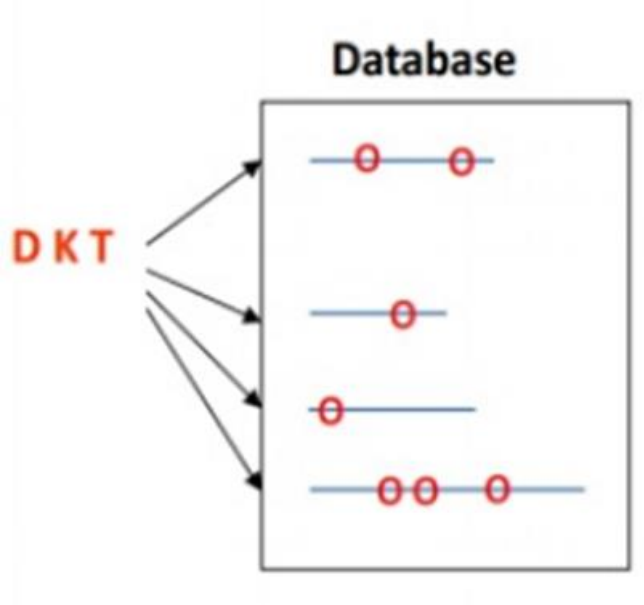


- 不允许出现空位;
- 得分超过设定的阈值**T**的保留，存入words list中作为相似的neighborhood word 保留。

使用者可修改BLAST参数“word size”大小来改变阈值T，如降低T值来找到更多的初始比对序列，提高查询灵敏度。

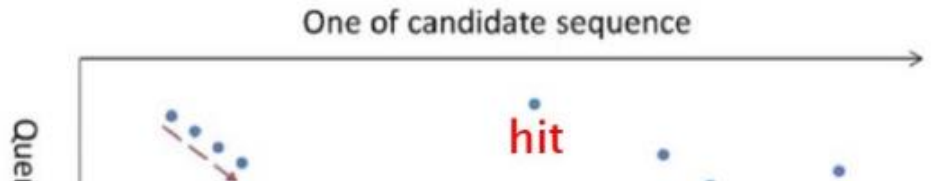
STEP 3 :Scaning

- 在数据库中定位种子找到Hit;
  - 查询words list中的每一个word，在数据库中找到其对应的每一个位置;
  - 数据库中的参考序列已预先做好索引，能快速完成检索.
- 得到每一个seed word在参考比对序列对应的位置;
- 检索方法:
  - 哈希索引Hash Table;
  - 后缀树;
  - Aho-Corasick自动计算法。



STEP 4 :Extending

- 将匹配得到的seed word延伸成更长的片段；
  - 最优的匹配结果和主对角线方向是平行的，因此沿着主对角线方向进行双方向的延伸；
  - 延伸的方法和Smith-Waterman算法基本一致，就是只计算MATCH的得分；
  - 不允许出现空位。



$$F(0,0)=0$$

$$F(i,j)=\max\begin{cases}F(i-1,j-1)+s(x_i,y_j)\\F(i-1,j)+d\\F(i,j-1)+d\\0\end{cases}$$

Query sequence: R P P Q G L F

Database sequence: D P **P E G** V V

Exact match is scanned.

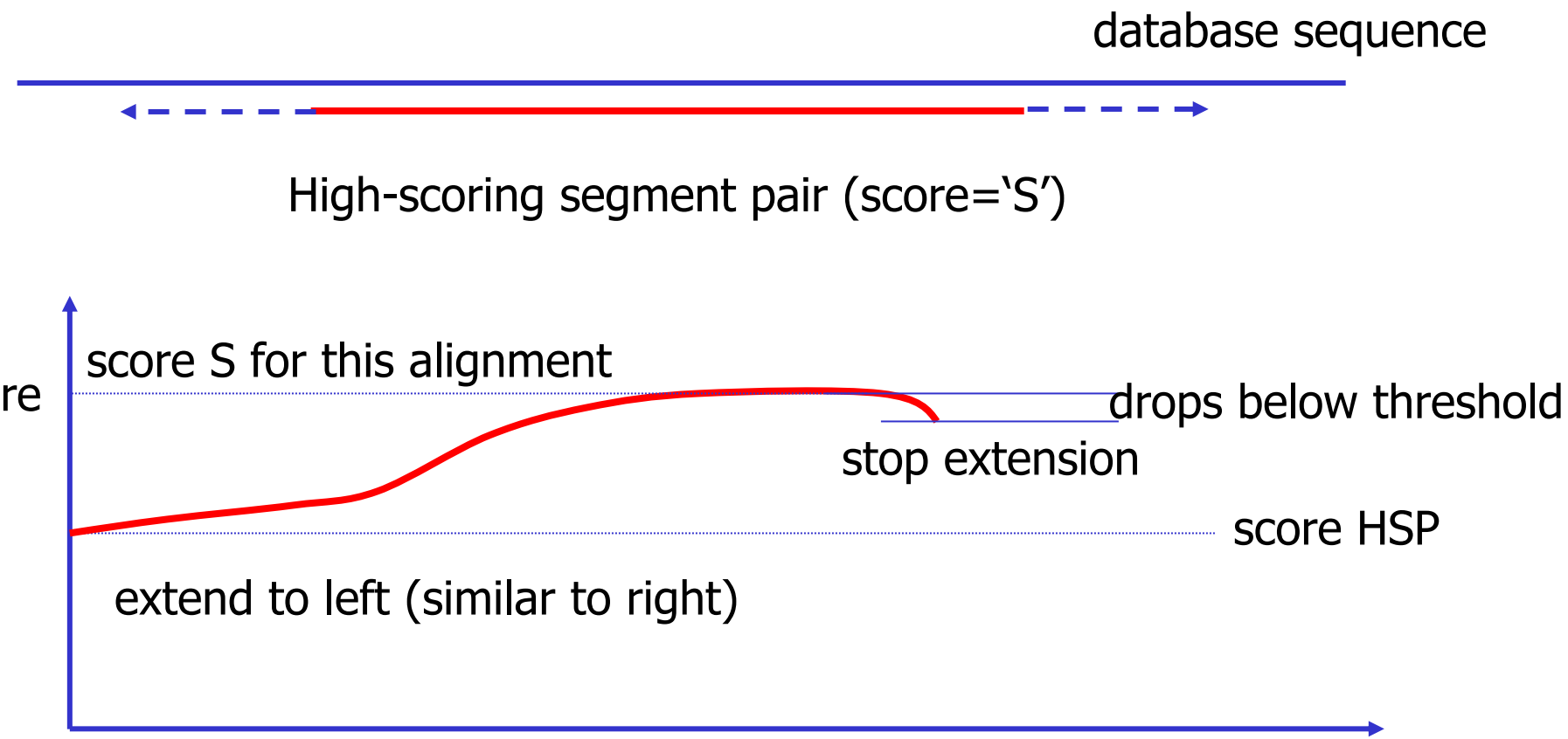
Score: -2 **7 7 2 6 1** -1

HSP

Optimal accumulated score = 7+7+2+6+1 = 23

Join words on same diagonal

- 计算延伸后的得分，当得分小于指定的阈值**S**，停止延伸；
  - 最后的比对结果叫做高分片段对（high-scoring segment pair, HSP）
- 计算出每个seed Word的HSP得分，排序，选取TOP作为候选；



STEP 5 :Significance Evaluation

- 量化比对是统计学显著的匹配，还是随机发生的事件?

- 评价标准:

  - raw scores (原始分数): 不涉及打分系统参数，几乎无意义;
  - bit scores (比特分数): 对raw scores与其参数（如数据库大小、序列长度与组成等）进行归一化处理;

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

λ - 打分系统修正参数
K - 残基比例修正参数
S - HSP分数

STEP 5 :Significance Evaluation

✦E value: 衡量在随机情况下，数据库存在的比当前匹配分数更好的比对的数目；

$$E = mn2^{-S'} = Kmne^{-\lambda S}$$

n - 查询序列残基数

m - 数据库中总残基数

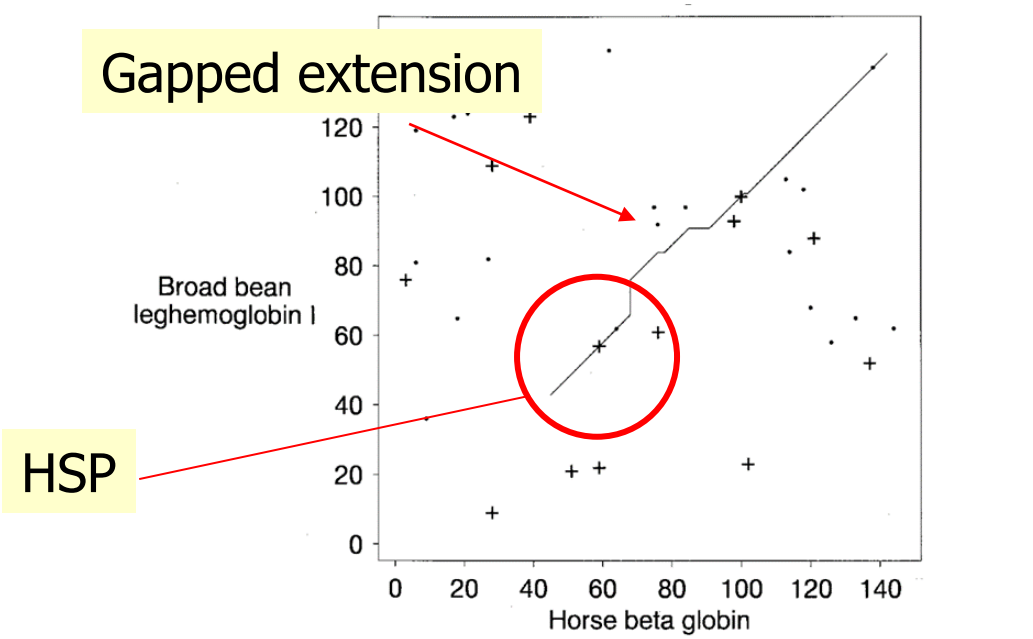
K, λ – Karlin-Altschul统计量

- E >1,序列比对结果不可靠；例如，E=10就意味着会有10个随机的匹配获得与当前比对相等或者更高的分数。
- E<0.05，结果统计学上有意义；
- E<10^{-5}，比对序列与查询序列高度一致。

- 通过设定E值过滤掉不合理的HSP比对序列，得到最终比对的结果。

BLAST：发展与改进

- 早期的BLAST版本：无空位罚分；
- 新版本(BLAST+)：增加Gap Penalties (Existence: 11, Extension: 1)等




```
Leghemoglobin   43 FSFLKDSAGVVDSPKLGAHAEKVFGMVRDSAVQLRATGEVV--LDGKDGS------  90
                   F  L +   V+ +PK+ AH +KV          L + GE V  LD   G+
Beta globin     45 FGDLSNPGAVMGNPKVKAHGKKV----------LHSFGEGVHHLDNLKGTFAALSE  90

Leghemoglobin   91 IHIQKGVLDP-HFVVVKEALLKTIKEASGDKWSEELSAAWEVAYDGLATAI 140
                   +H  K  +DP +F ++   L+  +    G  ++ EL A+++    G+A A+
Beta globin     91 LHCDKLHVDPENFRLLGNVLVVVLARHFGKDFTPELQASYQKVVAGVANAL 141
```

BLAST：发展与改进

- PSI-BLAST: Position-Specific Iterated BLAST(位点特异性迭代BLAST)
  - 位点特异性迭代BLAST每次用位点特异权重矩阵（Position-Specific Scoring Matrix, PSSM）搜索数据库后，再利用搜索结果重新构建PSSM，并再次用新的PSSM搜索数据库，如此反复(iteration)，直到没有新的结果产生为止。

- PHI-BLAST: Pattern-Hit Initiated BLAST(模式识别BLAST)
  - 模式识别BLAST能找到与输入序列相似的并符合某种特定模式(pattern)的序列。例如，N-糖基化位点基序(N-glycosylation site motif)总是符合以下特定模式： 以Asn开始，然后是除了Pro之外的其它氨基酸，再紧跟Ser或Thr，再跟除了Pro之外的其它氨基酸,用正则表达式表示N-糖基化位点基序：N{P}[ST]{P}

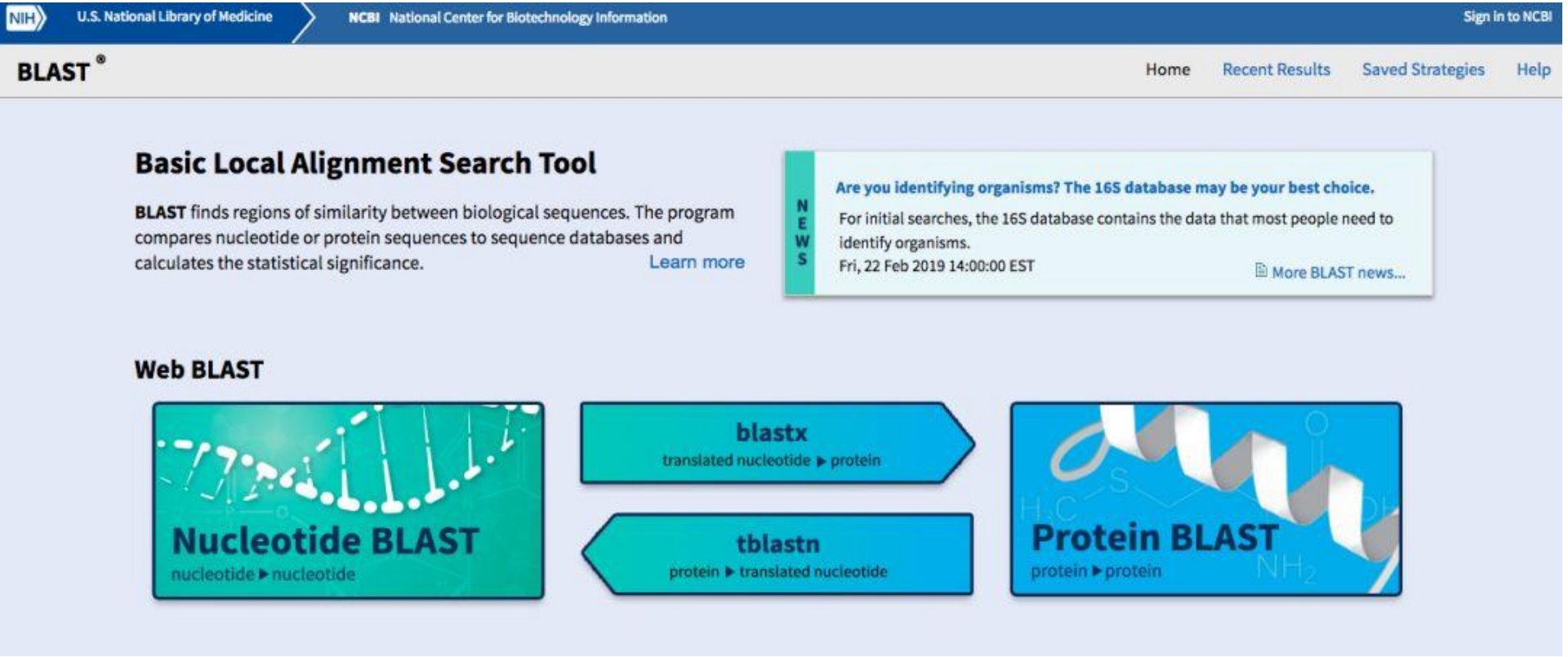
Outline

BLAST简介

NCBI网络BLAST使用

本地BLAST使用

NCBI在线BLAST主页



http://blast.ncbi.nlm.nih.gov/

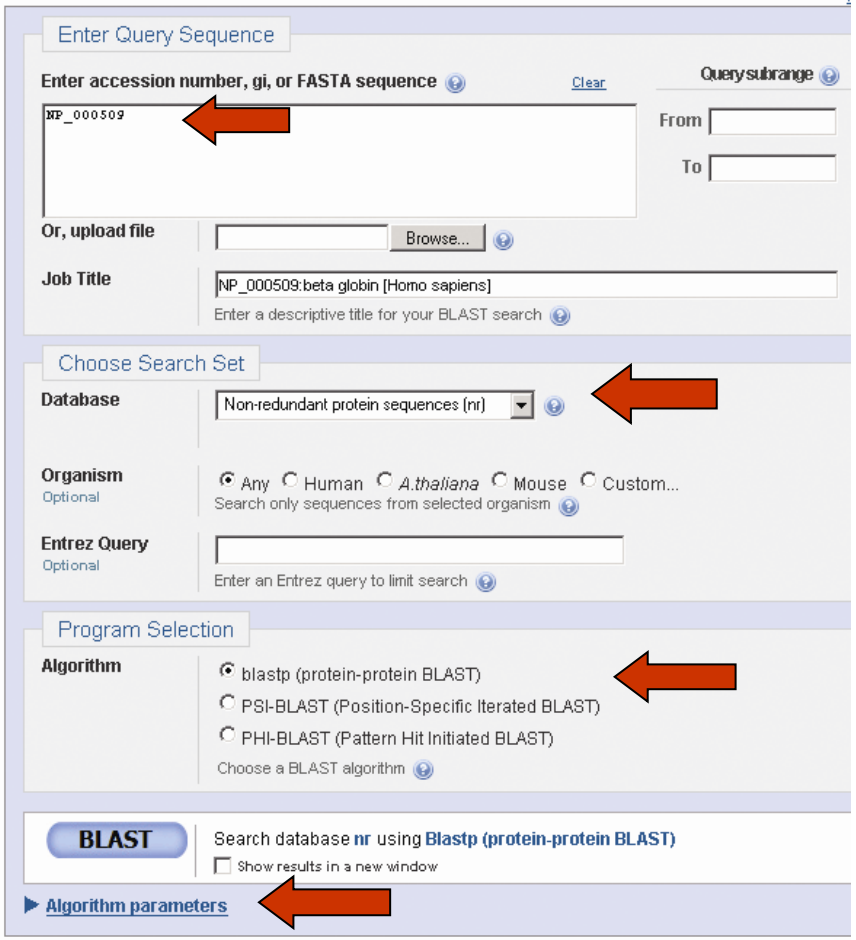
Four Steps to a BLAST search

(1) Choose the sequence (query)

(2) Choose the database to search

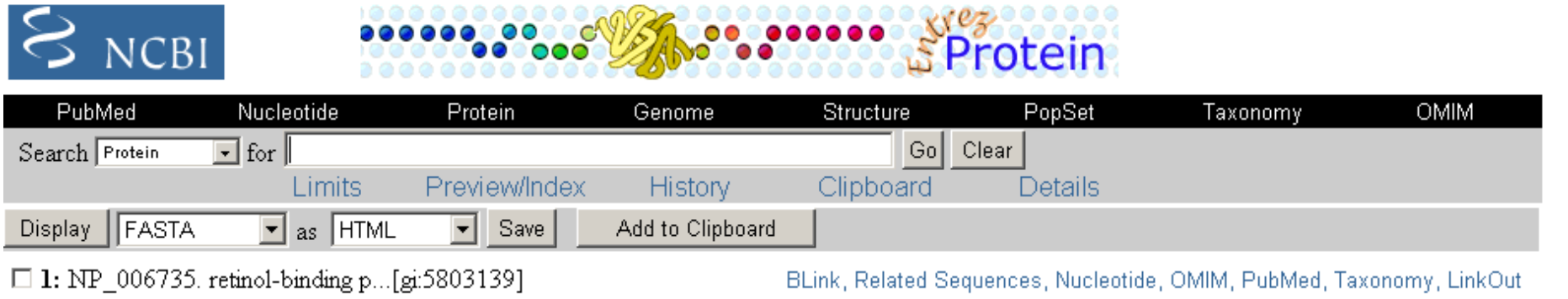
(3) Select the BLAST program

(4) Choose optional parameters



Step 1: Choose your sequence

- BLAST搜索第一步是选定要查询的DNA或蛋白质序列
  - Sequence can be input in FASTA format or as accession number (e.g. NP_006735)

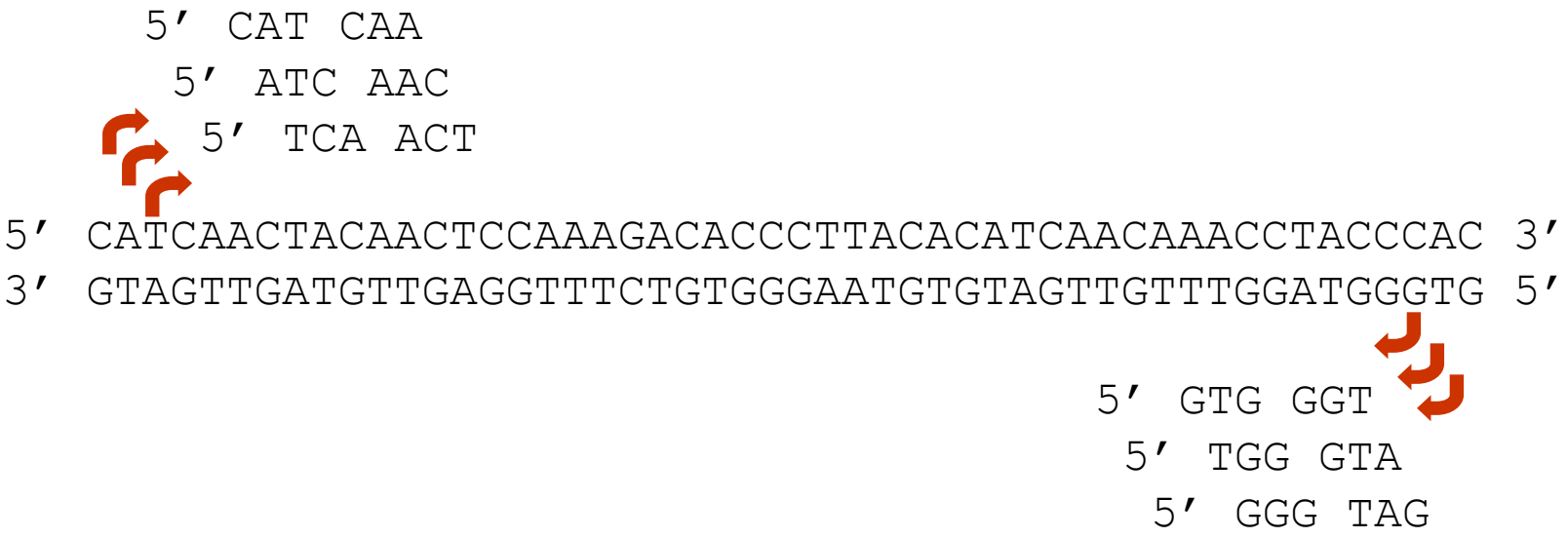



```
>gi|5803139|ref|NP_006735.1| retinol-binding protein 4, plasma precursor; retinol-binding protein 4, interstitial [Hom
MKWVWALLLLAAWAAAERDCRVSSFRVKENFDKARFSGTWYAMAKKDPEGLFLQDNIVAEFSVDETGQMS
ATAKGRVRLLNNWDVCADMVGTFTDTEDPAKFKMKYWGVASFLQKGNDDHWIVDTDYDTYAVQYSCRLLN
LDGTCADSYSFVFSRDPNGLPPEAQKIVRQRQEELCLARQYRLIVHNGYCDGRSERNLL
```

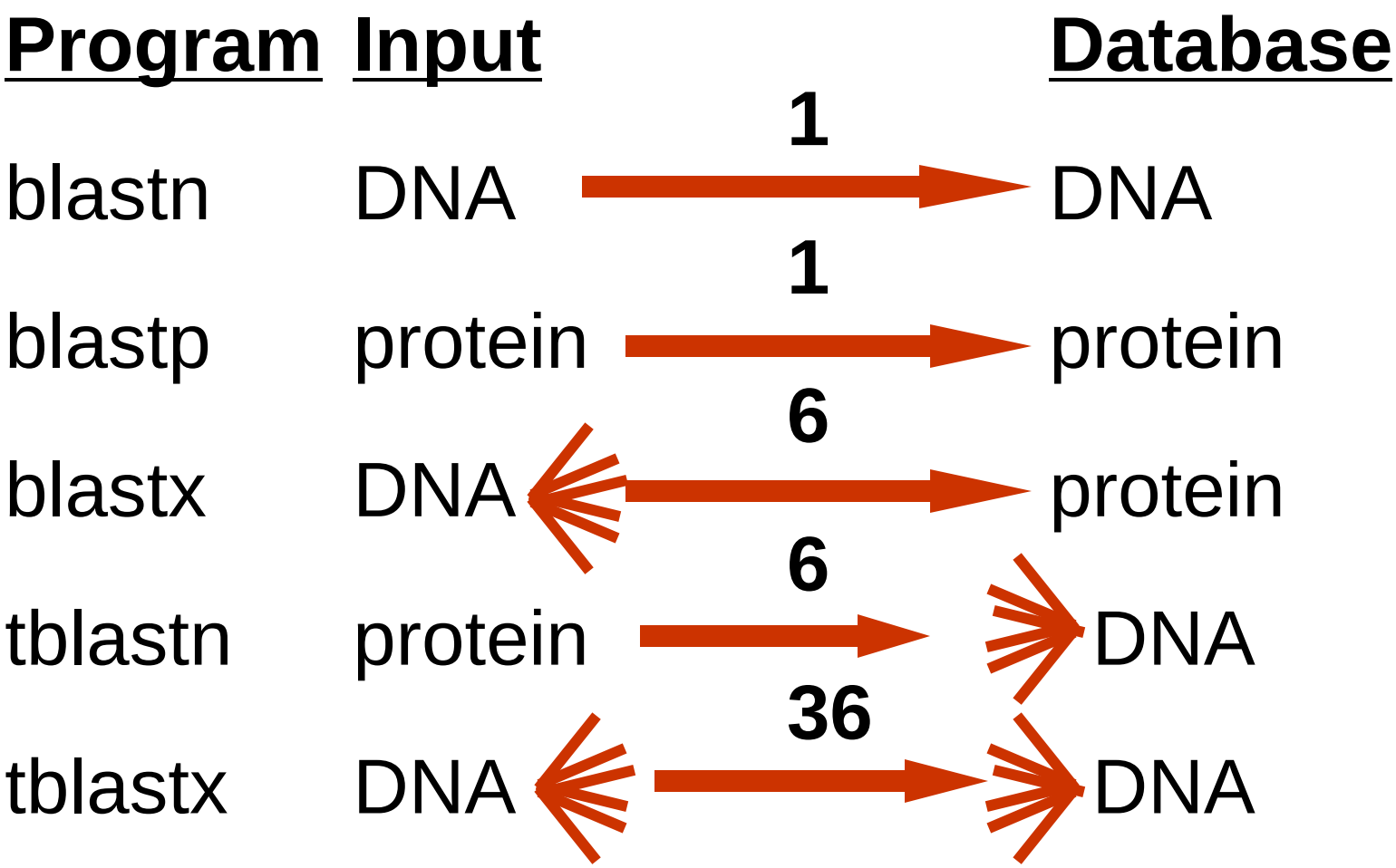
Step 2: Choose the BLAST program

- blastn (**n**ucleotide BLAST)
- blastp (**p**rotein BLAST)
- blastx (translated BLAST)
- tblastn (**t**ranslated BLAST)
- tblastx (**t**ranslated BLAST)

DNA potentially encodes six proteins



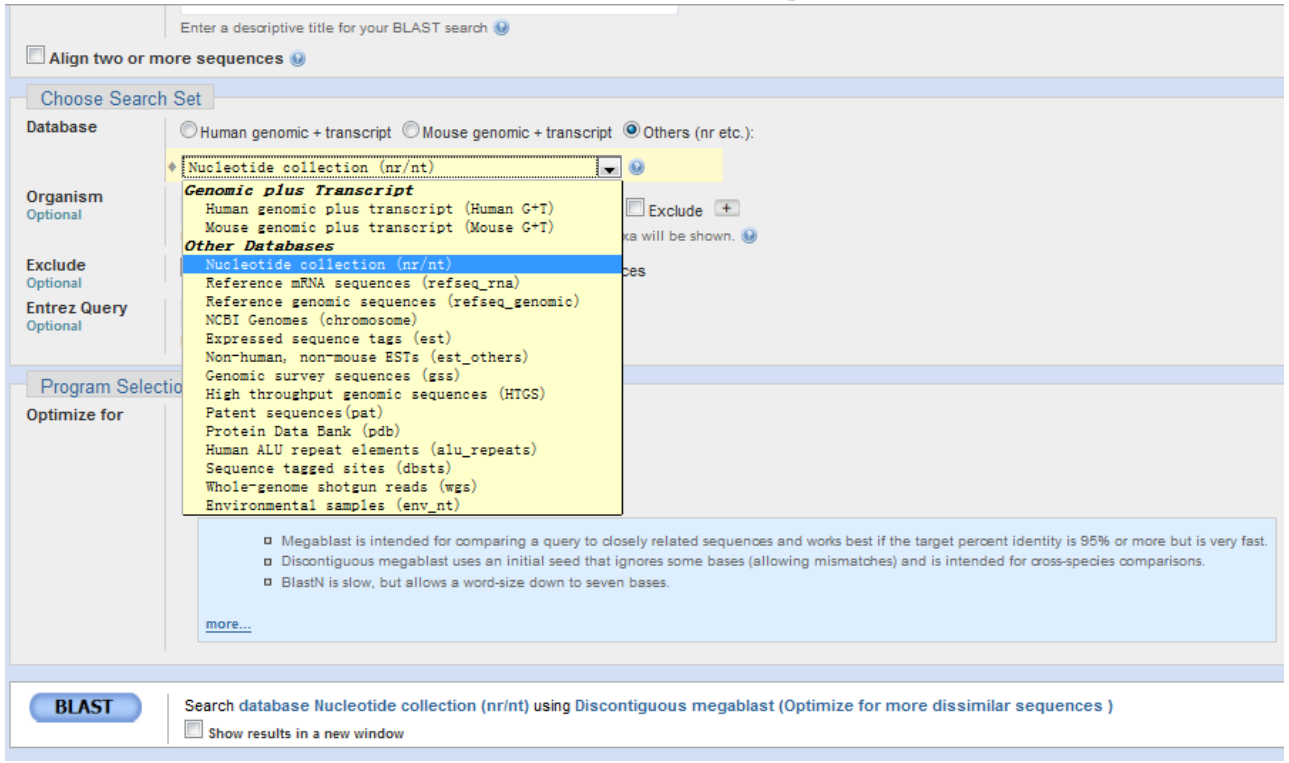
Choose the BLAST program



Step 3: choose the database

- nr/nt = non-redundant (most general database)
- refseq_rna = Reference RNA sequences
- refseq_genomic = NCBI参考序列中的基因组序列
- pdb = Protein Data Bank （已经蛋白三维结构）
- htgs = high throughput genomic sequence
- ……

选择一个数据库



Step 4: Choose optional parameters

You can...

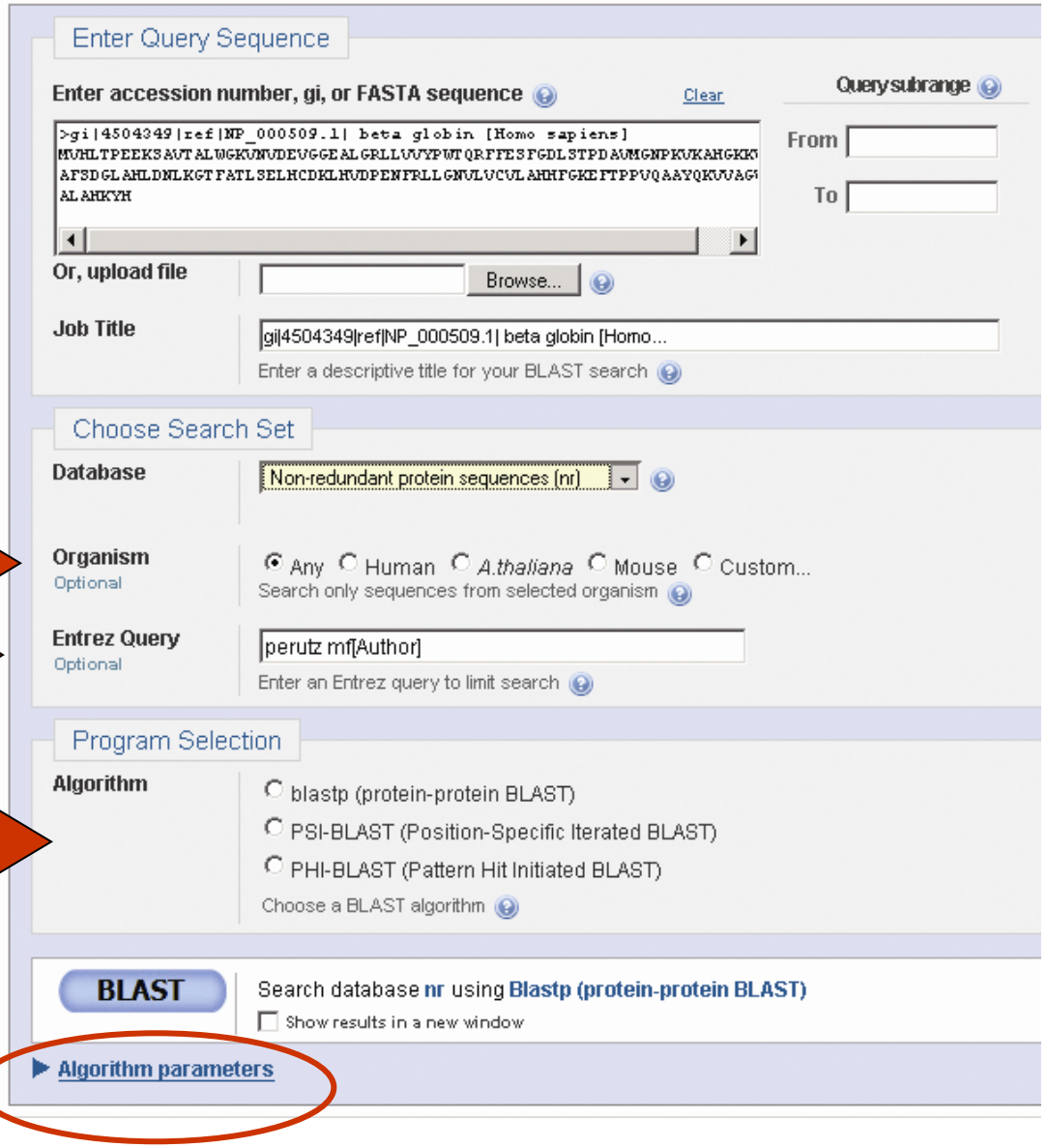
- change max target sequences
- turn filtering on/off
- change the substitution matrix
- change the expect (e) value
- change the word size
- change the output format to display

Select optional search parameters

organism

Entrez

algorithm



optional blastn search parameters

过滤(Filtering)

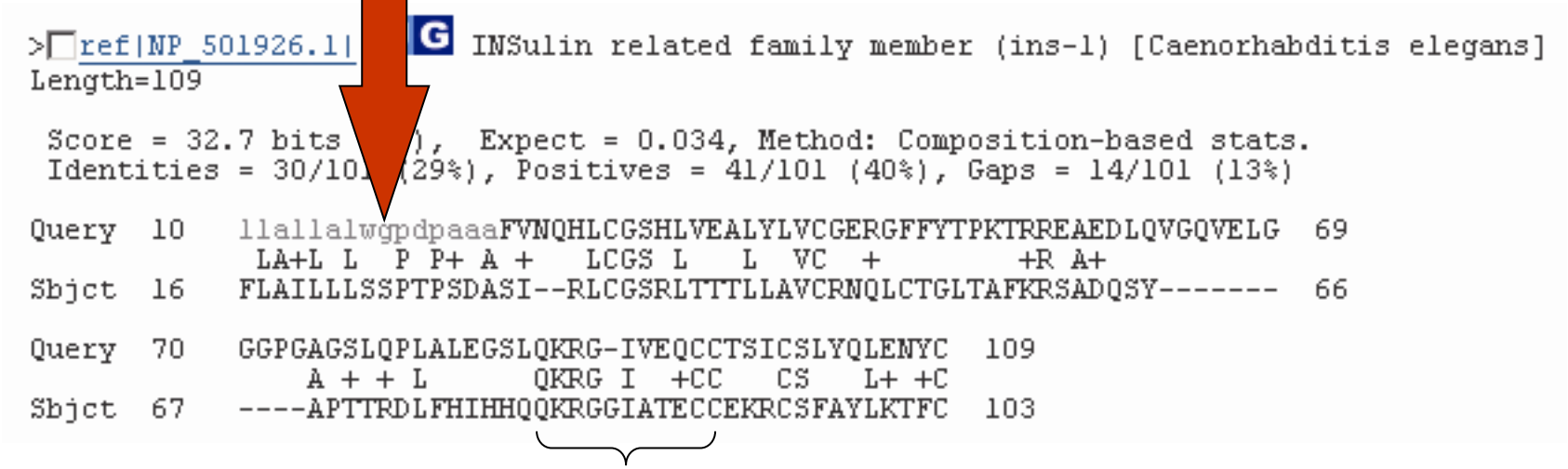
- 过滤掉低复杂度区域（"Low-complexity region"）：
  - 很少碱基或氨基酸的大量重复：
    - CACACACACACACACA…
    - KLKLKLKLKLLKLKL…
- 防止大量具有统计学显著意义，却不具备生物学意义的序列干扰比对：
  - 低复杂度区域得分很高，影响比对；
- 用符号代替这些序列，并在搜索时忽略：
  - DNA用N，蛋白质用X.

Our starting point: search human insulin against worm RefSeq proteins by blastp using default parameters

```
>ref|NP_501926.1| UG INSulin related family member (ins-1) [Caenorhabditis elegans]
Length=109

 Score = 32.7 bits (73),  Expect = 0.034, Method: Composition-based stats.
 Identities = 30/101 (29%), Positives = 41/101 (40%), Gaps = 14/101 (13%)

Query  10   LLALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGERGFFYTPKTRREAEDLQVGQVELG  69
            LA+L L   P P+ A +   LCGS L    L  VC  +         +R A+
Sbjct  16   FLAILLLSSPTPSDASI--RLCGSRLTTTLLAVCRNQLCTGLTAFKRSADQSY-------  66

Query  70   GGPGAGSLQPLALEGSLQKRG-IVEQCCTSICSLYQLENYC  109
                A + + L      QKRG I  +CC   CS   L+ +C
Sbjct  67   ----APTTRDLFHIHHQQKRGGIATECCEKRCSFAYLKTFC  103
```

(a) Query: human insulin NP_000198
Program: blastp
Database: *C. elegans* RefSeq
Default settings:
Unfiltered (“composition-based statistics”)

Filtering
(the filtered sequence is the query in lowercase and grayed out)



```
>ref|NP_501926.1|  INSulin related family member (ins-1) [Caenorhabditis elegans]
Length=109

 Score = 32.7 bits (73),  Expect = 0.034, Method: Composition-based stats.
 Identities = 30/101 (29%), Positives = 41/101 (40%), Gaps = 14/101 (13%)

Query  10   llallalwgpdpaaaFVNQHLCGSHLVEALYLVCGERGFFYTPKTRREAEDLQVGQVELG  69
            LA+L L   P P+ A +   LCGS L   L  VC  +        +R A+
Sbjct  16   FLAILLLSSPTPSDASI--RLCGSRLTTTLLAVCRNQLCTGLTAFKRSADQSY-------  66

Query  70   GGPGAGSLQPLALEGSLQKRG-IVEQCCTSICSLYQLENYC  109
                A + + L      QKRG I  +CC   CS   L+ +C
Sbjct  67   ----APTTRDLFHIHHQQKRGGIATECCEKRCSFAYLKTFC  103
```

(b) Query: human insulin NP_000198
Program: blastp
Database: *C. elegans* RefSeq
Option: Filter low complexity regions

Note that the bit score, Expect value, and percent identity all change with the “no compositional adjustment” option

```
>ref|NP_501926.1| UG INSulin related family member (ins-1) [Caenorhabditis elegans]
Length=109

 Score = 34.7 bits (78),  Expect = 0.009
 Identities = 30/100 (30%), Positives = 41/100 (41%), Gaps = 14/100 (14%)

Query  11   LALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGERGFFYTPKTRREAEDLQVGQVELGG  70
            LA+L L  P P+ A +   LCGS L   L  VC  +        +R A+
Sbjct  17   LAILLLSSPTPSDASIR--LCGSRLTTTLLAVCRNQLCTGLTAFKRSADQSY--------  66

Query  71   GPGAGSLQPLALEGSLQKRG-IVEQCCTSICSLYQLENYC  109
               A + + L      QKRG I  +CC   CS   L+ +C
Sbjct  67   ---APTTRDLFHIHHQQKRGGIATECCEKRCSFAYLKTFC  103
```

(c) Query: human insulin NP_000198
Program: blastp
Database: *C. elegans* RefSeq
Option: No compositional adjustment

Compositional adjustment:组成校正，使用针对序列组成的统计方法，对每个数据库序列产生一个稍微区别的打分系统

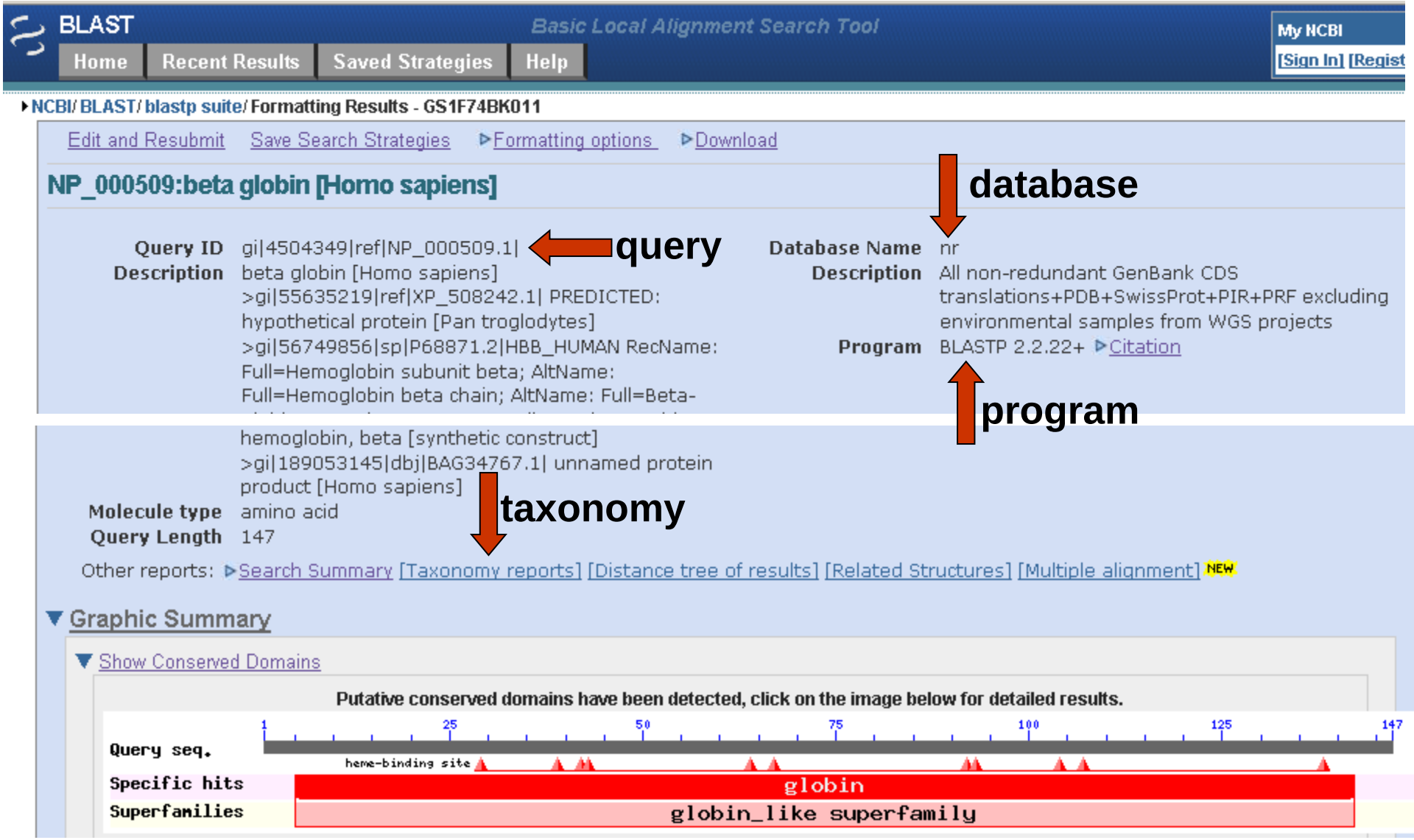
Output format

Descriptions

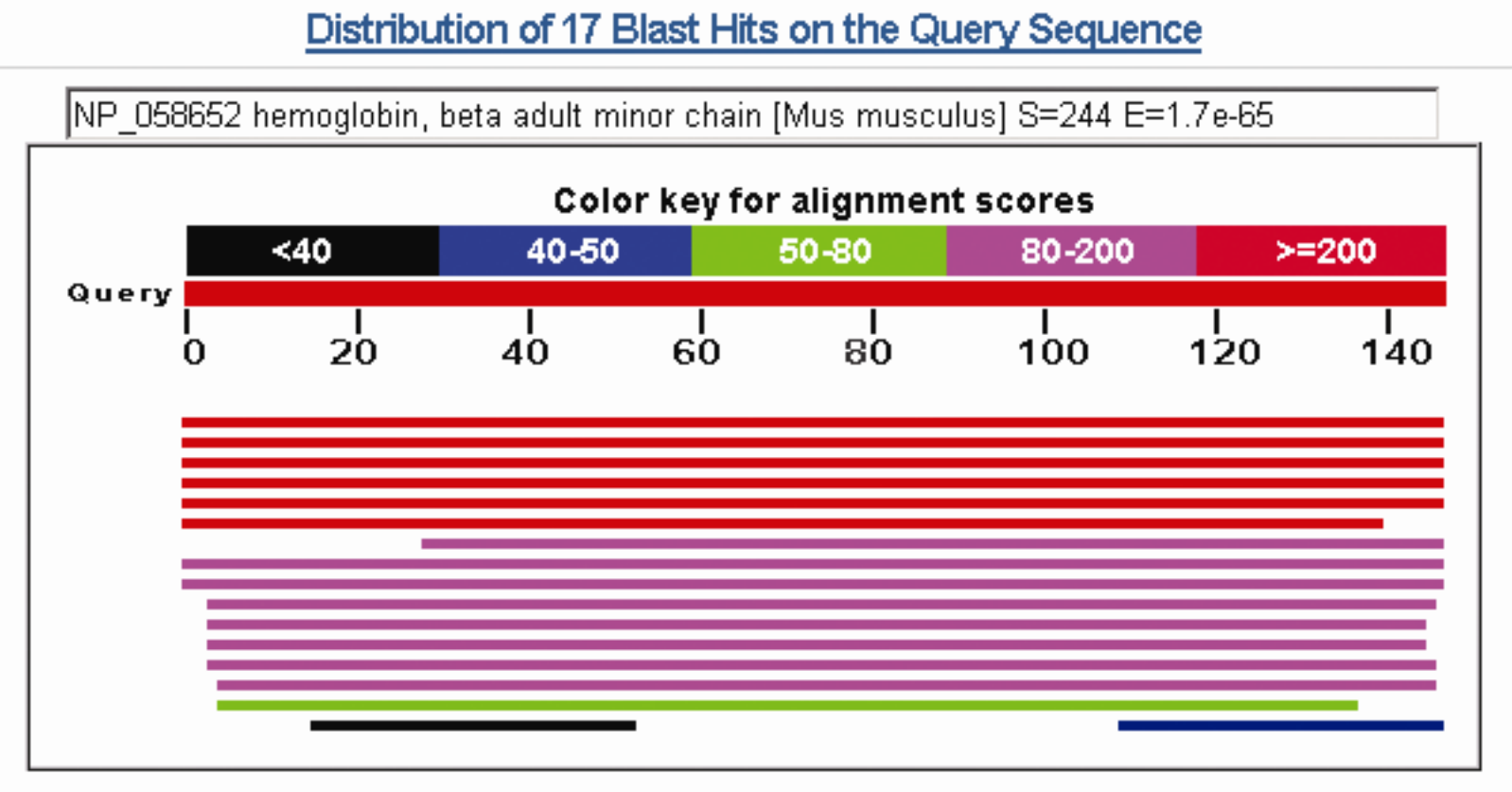
Graphical Summary

Alignment view

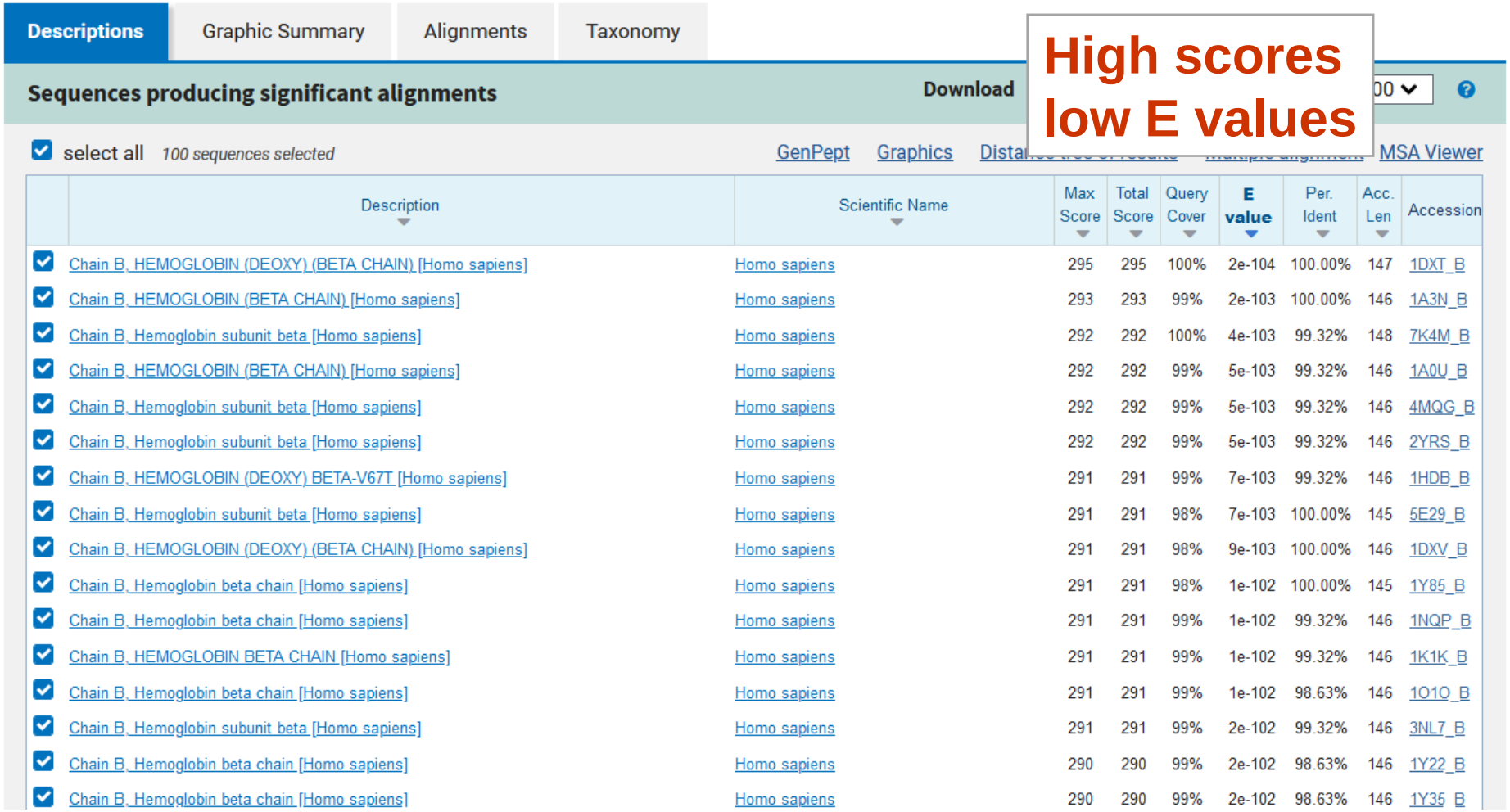
BLAST search output: top portion



BLAST search output: graphical output



BLAST searcah output: Descriptions



Descriptions | Graphic Summary | Alignments | Taxonomy

Sequences producing significant alignments Download

High scores low E values

select all *100 sequences selected* GenPept Graphics Distance tree of results Multiple alignment MSA Viewer

Description	Scientific Name	Max Score	Total Score	Query Cover	E value	Per. Ident	Acc. Len	Accession
Chain B, HEMOGLOBIN (DEOXY) (BETA CHAIN) [Homo sapiens]	Homo sapiens	295	295	100%	2e-104	100.00%	147	1DXT_B
Chain B, HEMOGLOBIN (BETA CHAIN) [Homo sapiens]	Homo sapiens	293	293	99%	2e-103	100.00%	146	1A3N_B
Chain B, Hemoglobin subunit beta [Homo sapiens]	Homo sapiens	292	292	100%	4e-103	99.32%	148	7K4M_B
Chain B, HEMOGLOBIN (BETA CHAIN) [Homo sapiens]	Homo sapiens	292	292	99%	5e-103	99.32%	146	1A0U_B
Chain B, Hemoglobin subunit beta [Homo sapiens]	Homo sapiens	292	292	99%	5e-103	99.32%	146	4MQG_B
Chain B, Hemoglobin subunit beta [Homo sapiens]	Homo sapiens	292	292	99%	5e-103	99.32%	146	2YRS_B
Chain B, HEMOGLOBIN (DEOXY) BETA-V67T [Homo sapiens]	Homo sapiens	291	291	99%	7e-103	99.32%	146	1HDB_B
Chain B, Hemoglobin subunit beta [Homo sapiens]	Homo sapiens	291	291	98%	7e-103	100.00%	145	5E29_B
Chain B, HEMOGLOBIN (DEOXY) (BETA CHAIN) [Homo sapiens]	Homo sapiens	291	291	98%	9e-103	100.00%	146	1DXV_B
Chain B, Hemoglobin beta chain [Homo sapiens]	Homo sapiens	291	291	98%	1e-102	100.00%	145	1Y85_B
Chain B, Hemoglobin beta chain [Homo sapiens]	Homo sapiens	291	291	99%	1e-102	99.32%	146	1NQP_B
Chain B, HEMOGLOBIN BETA CHAIN [Homo sapiens]	Homo sapiens	291	291	99%	1e-102	99.32%	146	1K1K_B
Chain B, Hemoglobin beta chain [Homo sapiens]	Homo sapiens	291	291	99%	1e-102	98.63%	146	1O1O_B
Chain B, Hemoglobin subunit beta [Homo sapiens]	Homo sapiens	291	291	99%	2e-102	99.32%	146	3NL7_B
Chain B, Hemoglobin beta chain [Homo sapiens]	Homo sapiens	290	290	99%	2e-102	98.63%	146	1Y22_B
Chain B, Hemoglobin beta chain [Homo sapiens]	Homo sapiens	290	290	99%	2e-102	98.63%	146	1Y35_B

Cut-off: .05? 10^{-10}?

BLAST的得分与统计显著性

- S (Score)：比对得分
- E (Expect)：比对随机找出的序列的期望数目
- P (Probability)： 比对随机找出的一条或多条序列，其比对得分大于等于S的可能性
- E与P值比较低说明了该结果与查询序列具有进化上的关系，而并非由于随机因素得到该结果。
  - 当E值接近零时，一个比对随机发生的可能性也会接近零。
  - 人类基因组分析中，当E值低于0.001(10^{-3})时，搜索结果通常被认为有统计上的显著性。

BLAST search output: alignment output

Outline

BLAST简介

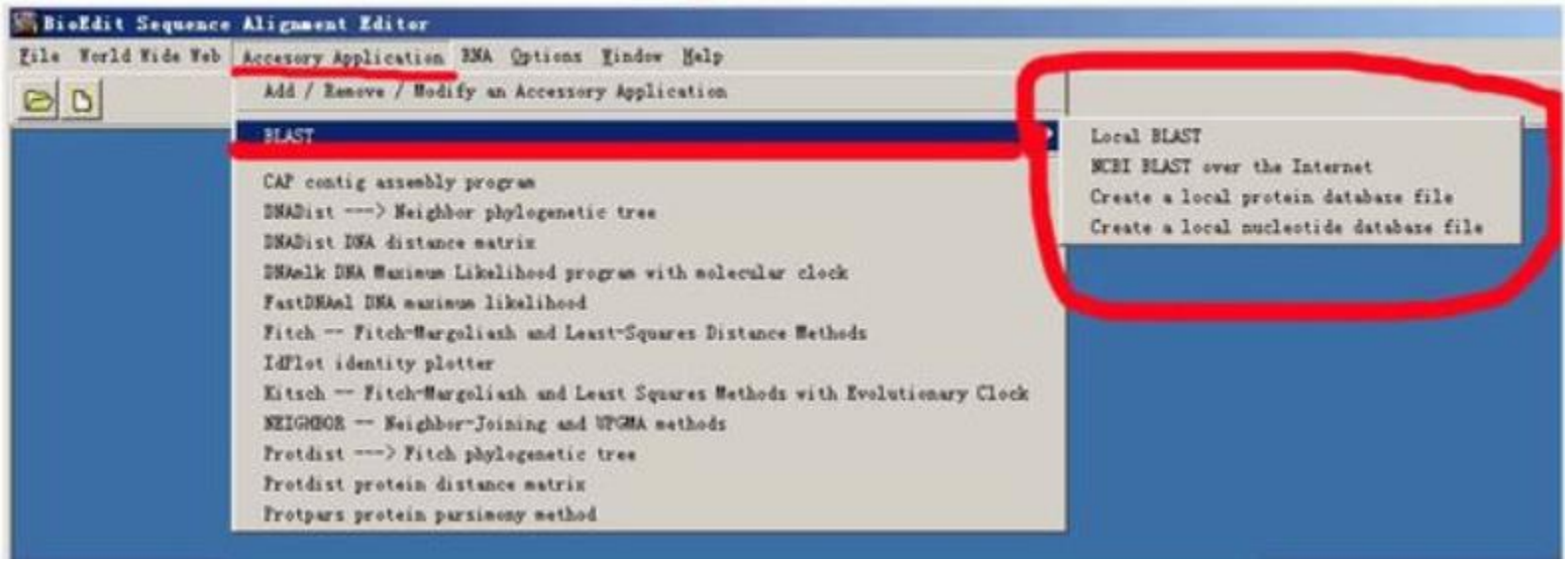
NCBI网络BLAST使用

本地BLAST使用

BioEdit本地BLAST

BioEdit提供了本地BLAST的功能：

菜单Accesory Application -> BLAST



准备序列数据

序列数据文件(fasta格式)

- **查询序列(Query):16S_rRNA.fasta**
- **序列数据库(Subject)：16SMicrobial.tar.gz**

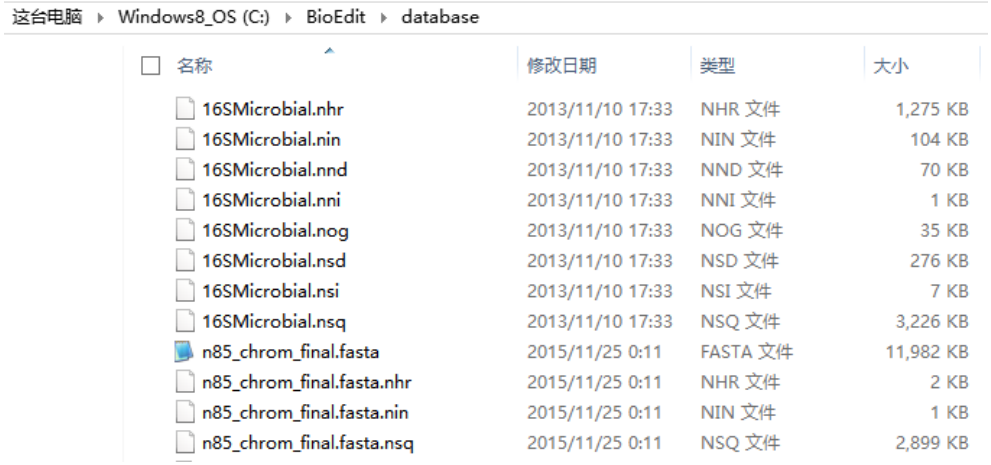
- 查询序列: 1条测序得到的细菌16S rDNA序列
- 序列数据库： 此数据库为NCBI网站已经构建好的细菌16S 数据库，使用压缩软件(如WINRAR)解压后会看到8个文件。

名称	修改日期	类型	大小
16SMicrobial.nhr	2013/11/10 17:33	NHR 文件	1,275 KB
16SMicrobial.nin	2013/11/10 17:33	NIN 文件	104 KB
16SMicrobial.nnd	2013/11/10 17:33	NND 文件	70 KB
16SMicrobial.nni	2013/11/10 17:33	NNI 文件	1 KB
16SMicrobial.nog	2013/11/10 17:33	NOG 文件	35 KB
16SMicrobial.nsd	2013/11/10 17:33	NSD 文件	276 KB
16SMicrobial.nsi	2013/11/10 17:33	NSI 文件	7 KB
16SMicrobial.nsq	2013/11/10 17:33	NSQ 文件	3,226 KB

（ftp://ftp.ncbi.nlm.nih.gov/blast/db/FASTA/）

构建本地数据库

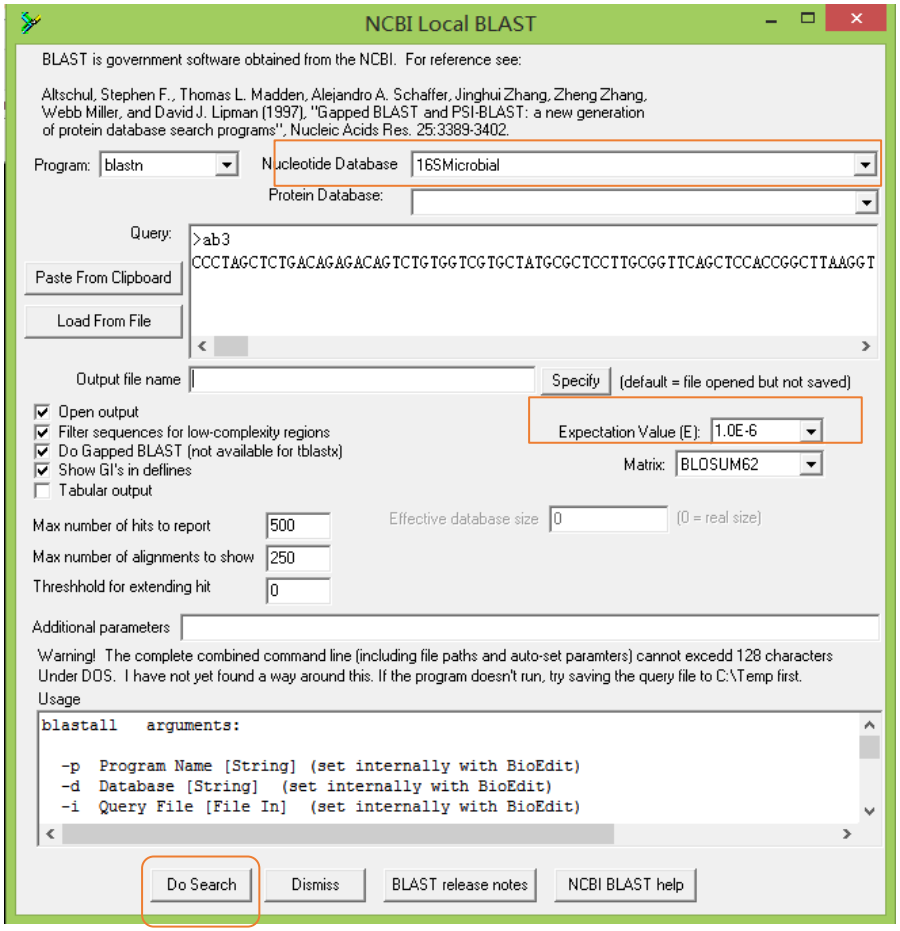
- 把序列数据做成本地数据库
  - 菜单create a local nucleotide database file(DNA数据库)
  - 菜单create a local protein database file(蛋白质数据库)
- 建库的原始数据需要做成fasta格式
- 建成的数据库存放在BIOEDIT文件夹(c:\bioedit\database)。



注：本练习直接用NCBI网站提供的已经格式化的16S序列数据库，不需要从头建库，将解压后16SMicrobial目录下的8个文件复制到BioEdit安装目录下的database文件夹(e.g., c:\bioedit\database)即可。

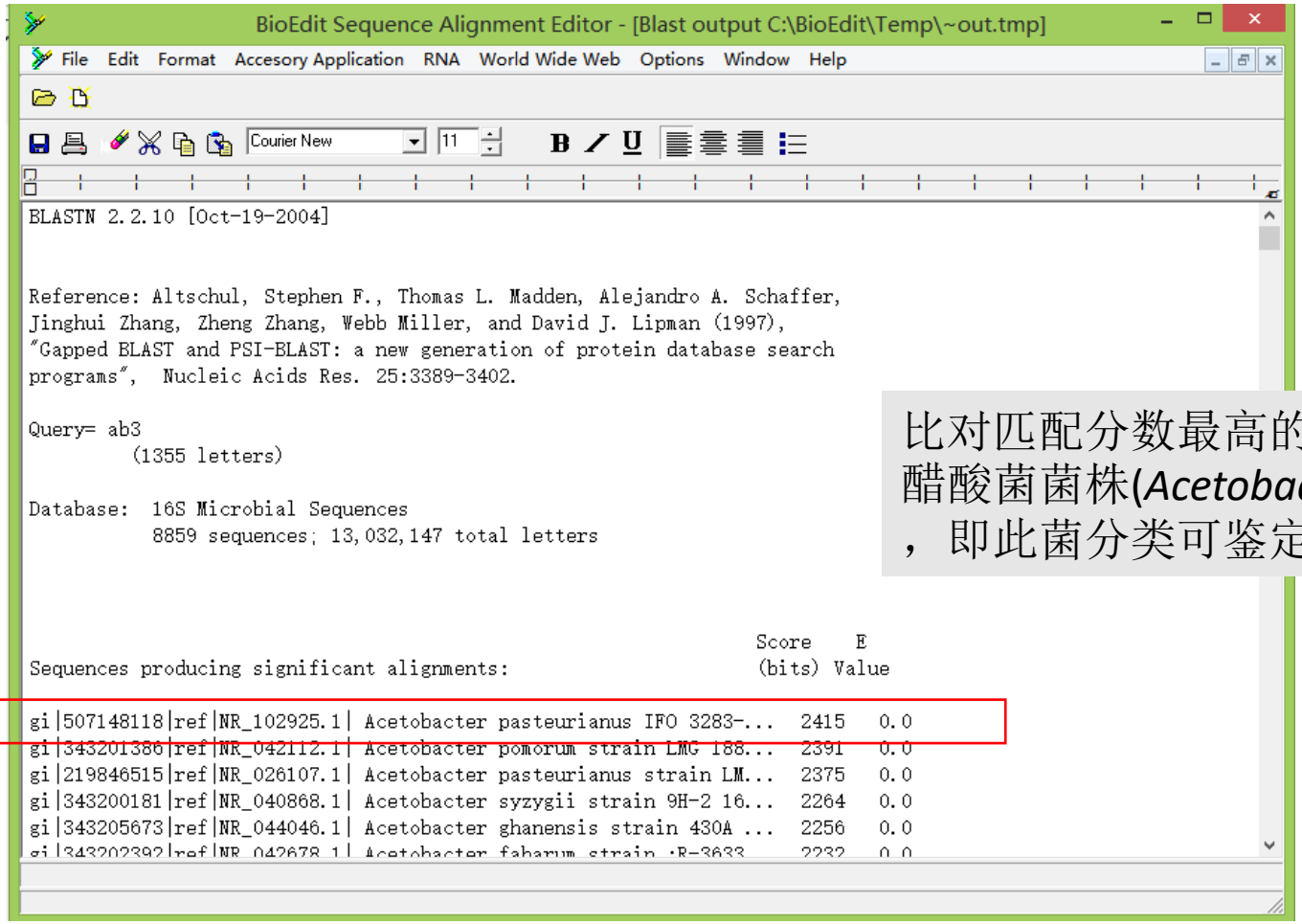
序列间的相似性检索

- 运行BLAST：菜单Accessory application-BLAST-local BLAST
- 用待查询序列作为query, 数据库选择刚刚建好的那个，设置其它blastn参数，通常e-value越小越好。



查看BLAST结果

- BLAST比对结果找到数据库中与查询序列相似的序列。如下图所示，结果会有相应数据库序列的序列号，可查看对应序列。



比对匹配分数最高的序列是源自巴氏醋酸菌菌株(*Acetobacter pasteurianus*)，即此菌分类可鉴定为巴氏醋酸菌。

Linux本地BLAST

1. Ubuntu安装BLAST:

```
$sudo apt-get install ncbi-blast+
```

最新版的本地BLAST+软件可以从NCBI网站下载安装

2.准备数据

直接从NCBI下载BLAST数据库:

ftp://ftp.ncbi.nlm.nih.gov/blast/db/FASTA/

也可以自己创建fasta格式的数据库

Linux本地BLAST

3.格式化数据库

```
$makeblastdb -in db.fasta -dbtype nucl -out dbname -parse_seqids
```

参数说明:

-in：待格式化序列的输入文件（fasta格式）

-dbtype：数据库序列类型，prot(蛋白质)或nucl(核酸)

-out：数据库名

-parse_seqids：参数可选，表示从输入fasta格式中解析序列标识符(SeqIds)

Linux本地BLAST

4.运行BLAST比对程序

这里以核酸序列比对核酸数据库（blastn）为例：

```
$blastn –task blastn -query seq.fasta -out seq.blast -db dbname -outfmt 6 -evalue 1e-5 -num_threads 4
```

参数说明:

-task： 共五个程序选择'blastn' 'blastn-short' 'dcmegablast' 'megablast' 'rmblastn' ，默认megablast。

-query： 输入文件的路径及文件名

-out：输出文件的路径及文件名

-db：格式化后得到的数据库路径及数据库名

-outfmt：输出文件格式，共有18种格式， 0是默认比对格式，6是tabular格式。

-evalue：设置输出结果的e-value值

-num_threads：使用的线程数

其它BLAST程序用法与blastn类似，如蛋白序列比对蛋白数据库（blastp）以及核酸序列比对蛋白数据库（blastx）等。

Linux本地BLAST

5.查看BLAST比对结果

```
$more gyrB_blast.txt
gb|CP000422.1|:4284-6230    LX-4_contig1        99.28       1947        14          0           1
          1947       148455    150401    0.0         3518
```

结果中从左到右，每一列的意义分别是：

- Query id：查询序列标识符，如“gb|CP000422.1|:4284-6230”
- Subject id：数据库中比对的目标序列标识符，如“LX-4_contig1”
- % identity：查询序列与目标序列比对的一致性(%)，如“99.28”
- alignment length：查询序列与目标序列比对上的片段长度，如“1947”
- mismatches：查询序列与目标序列比对错误的计数，如“14”
- gap openings：空位数，如“0”
- q. start：查询序列比对起始位点，如“1”
- q. end：查询序列比对终止位点，如“1947”
- s. start：目标序列比对起始位点，如“148455”
- s. end：目标序列比对终止位点，如“150401”
- e-value：E值，如“0.0”
- bit score：序列匹配得分，如“3518”

作业

- 某实验室从土壤中分离到一株细菌，并通过16S rRNA基因测序获得一条序列(16S_rRNA.fasta)，进行BLAST分析鉴定细菌的分类，并使用不同的BLAST参数比较结果的差别。
- 可使用下面任一种方法：
  - 使用NCBI在线BLAST工具
  - 使用BioEdit软件的BLAST程序
  - 在Linux命令行中的BLAST程序 ($sudo apt install ncbi-blast+)

注：16S rRNA基因测序序列与16SMicrobial数据库可从课程网站下载