Biopython

Python for Genomic Data Science

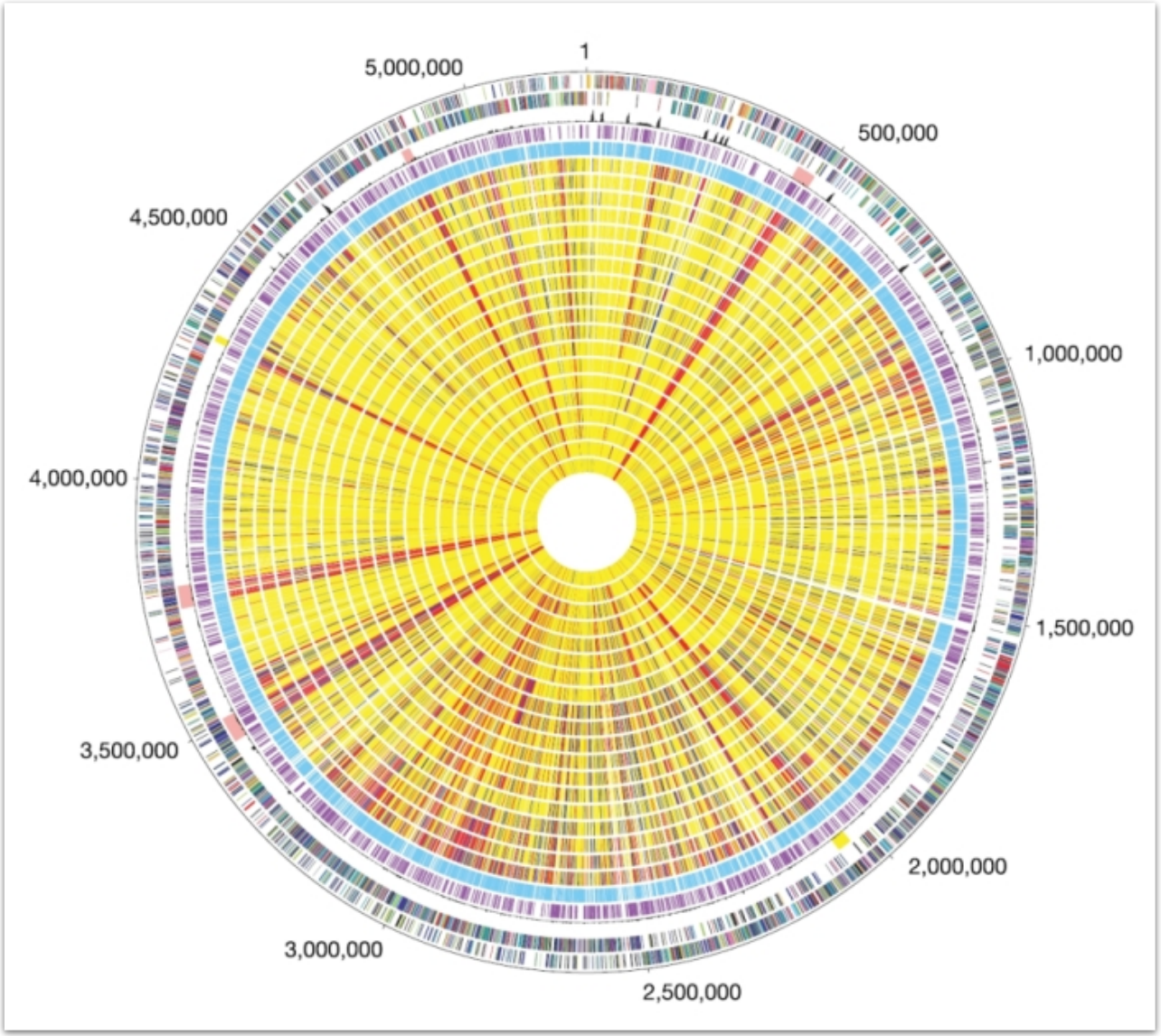


Image: circular representation of the *Bacillus anthracis* genome

The Biopython Project

- http://www.biopython.org : an online resource for modules, scripts, and web links for developers of Python-based software for bioinformatics use and research.

- Biopython includes parsers for various bioinformatics file formats (such as FASTA, Genbank), access to online services like NCBI Entrez or Pubmed databases, interfaces to common bioinformatics programs such as BLAST, Clustalw, and others.

Checking If Biopython Is Installed

```
>>> import Bio
>>> print(Bio.__version__)
1.65
```

If the “import Bio” line fails, Biopython is not installed!

Installing Biopython

- available from http://biopython.org/wiki/Download
- runs on many platforms: Windows, Mac, and on the various flavors of Linux and Unix
- supports both Python 2 and Python 3

A Biopython Usage Simple Example

Problem. Find out from what species an unknown DNA sequence came from.

`myseq.fa`

```
>sequence_unknown
CATGCTACGGTGCTAAAAGCATTACGCCCTATAGTGATTTTCGAGACATACTGTGTTT
TTAAATATAGTATTGCC
```

Running BLAST over the Internet

```
>>> from Bio.Blast import NCBIWWW
>>> fasta_string = open("myseq.fa").read()
>>> result_handle = NCBIWWW.qblast("blastn",
"nt", fasta_string)
```

which database to search against

program to use

To find out more information:

```
>>> help(NCBIWWW.qblast)
```

The qblast() Function

Help on function qblast in module Bio.Blast.NCBIWWW:

qblast(program, database, sequence, auto_format=None, composition_based_statistics=None, db_genetic_code=None, endpoints=None, entrez_query='(none)', expect=10.0, filter=None, gapcosts=None, genetic_code=None, hitlist_size=50, i_thresh=None, layout=None, lcase_mask=None, matrix_name=None, nucl_penalty=None, nucl_reward=None, other_advanced=None, perc_ident=None, phi_pattern=None, query_file=None, query_believe_defline=None, query_from=None, query_to=None, searchsp_eff=None, service=None, threshold=None, ungapped_alignment=None, word_size=None, alignments=500, alignment_view=None, descriptions=500, entrez_links_new_window=None, expect_low=None, expect_high=None, format_entrez_query=None, format_object=None, format_type='XML', ncbi_gi=None, results_file=None, show_overview=None, megablast=None)

Do a BLAST search using the QBLAST server at NCBI.

Supports all parameters of the qblast API for Put and Get.

Some useful parameters:

- program blastn, blastp, blastx, tblastn, or tblastx (lower case)
- database Which database to search against (e.g. "nr").
- sequence The sequence to search.
- ncbi_gi TRUE/FALSE whether to give 'gi' identifier.
- descriptions Number of descriptions to show. Def 500.
- alignments Number of alignments to show. Def 500.
- expect An expect value cutoff. Def 10.0.
- matrix_name Specify an alt. matrix (PAM30, PAM70, BLOSUM80, BLOSUM45).
- filter "none" turns off filtering. Default no filtering
- format_type "HTML", "Text", "ASN.1", or "XML". Def. "XML".
- entrez_query Entrez query to limit Blast search
- hitlist_size Number of hits to return. Default 50
- megablast TRUE/FALSE whether to use MEga BLAST algorithm (blastn only)
- service plain, psi, phi, rpsblast, megablast (lower case)

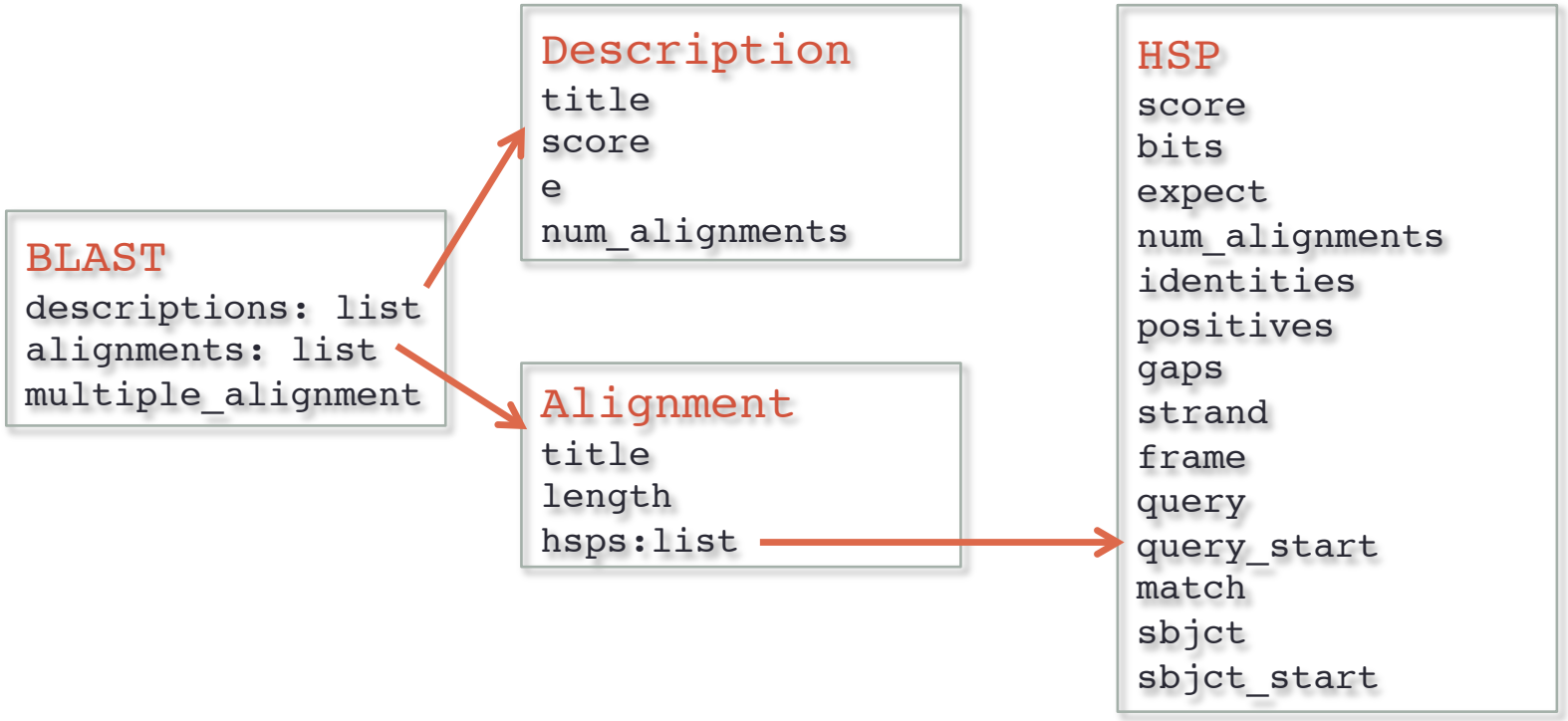
This function does no checking of the validity of the parameters
and passes the values to the server as is. More help is available at:
http://www.ncbi.nlm.nih.gov/BLAST/Doc/urlapi.html

Required parameters.

Default output format is XML.

The BLAST Record

```
>>> from Bio.Blast import NCBIXML
>>> blast_record = NCBIXML.read(result_handle)
```



Parsing BLAST Output

```
>>> len(blast_record.alignments)
50

>>> E_VALUE_THRESH = 0.01
>>> for alignment in blast_record.alignments:
...     for hsp in alignment.hsps:
...             if hsp.expect < E_VALUE_THRESH:
...                     print('****Alignment****')
...                     print('sequence:', alignment.title)
...                     print('length:', alignment.length)
...                     print('e value:', hsp.expect)
...                     print(hsp.query)
...                     print(hsp.match)
...                     print(hsp.sbjct)
...
```

Best Match For DNA Sequence

```
****Alignment****
sequence: gi|733962926|gb|KP271020.1| Zaire ebolavirus isolate Ebola virus/
H.sapiens-wt/COD/2014/Lomela-Lokolia19, complete genome
length: 18861
e value: 2.89428e-29
CATGCTACGGTGCTAAAAGCATTACGCCCTATAGTGATTTTCGAGACATACTGTGTTTTTAAATATAGTATTGCC
||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
CATGCTACGGTGCTAAAAGCATTACGCCCTATAGTGATTTTCGAGACATACTGTGTTTTTAAATATAGTATTGCC
****Alignment****
sequence: gi|733962903|gb|KP271019.1| Zaire ebolavirus isolate Ebola virus/
H.sapiens-wt/COD/2014/Lomela-Lokolia17, partial genome
length: 18760
e value: 2.89428e-29
CATGCTACGGTGCTAAAAGCATTACGCCCTATAGTGATTTTCGAGACATACTGTGTTTTTAAATATAGTATTGCC
||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
CATGCTACGGTGCTAAAAGCATTACGCCCTATAGTGATTTTCGAGACATACTGTGTTTTTAAATATAGTATTGCC
****Alignment****
sequence: gi|733962878|gb|KP271018.1| Zaire ebolavirus isolate Ebola virus/
H.sapiens-wt/COD/2014/Lomela-Lokolia16, complete genome
length: 18941
e value: 2.89428e-29
CATGCTACGGTGCTAAAAGCATTACGCCCTATAGTGATTTTCGAGACATACTGTGTTTTTAAATATAGTATTGCC
||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
CATGCTACGGTGCTAAAAGCATTACGCCCTATAGTGATTTTCGAGACATACTGTGTTTTTAAATATAGTATTGCC
```

...

More Help with Biopython

- **Biopython Tutorial and Cookbook:** *http://biopython.org/DIST/docs/tutorial/Tutorial.html*

- **Byopython FAQ:** *http://biopython.org/DIST/docs/tutorial/Tutorial.html#htoc5*