

BioRuby入門

後藤直久

2005年7月9日

BioRubyとは？

- バイオインフォマティクスに必要な機能や環境をオブジェクト指向スクリプト言語Rubyを用いて統合的に実装したライブラリ
- <http://bioruby.org/>
- バイオインフォマティクス(Bioinformatics)
 - バイオ(bio) : 生物学
 - インフォマティクス(informatics): 情報科学

祝！IPA「未踏ソフト」採択

BioRubyおよびChemRubyは、「Ruby言語による生物化学情報基盤ライブラリの開発」というテーマで、IPA（独立行政法人情報処理推進機構）の2005年度上期未踏ソフトウェア創造事業に採択されました。

<http://www.ipa.go.jp/jinzai/esp/2005mito1/gaiyou/10-26.html>

BioRuby

- 2000/11/21 BioRubyプロジェクト開始
- 2001/06/21 バージョン0.1をリリース
- ... (この間, リリース18回, 学会発表8回など)
- 2004/12/13 バージョン0.62をリリース
- 現在
 - ファイル数: 130以上
 - 行数: 37,000行以上
 - 開発者: 累計 10人以上(うち海外3人以上)

現在・過去の開発者

- Toshiaki Katayama (*)
- Mitsuteru Nakao (*)
- Yoshinori Okuji
- Shuichi Kawashima
- Masumi Itoh
- Naohisa Goto (*)
- Hiroshi Suga
- Alex Gutteridge
- Moses Hohman (*)
- Pjotr Prins (*)
- and some other contributors on the internet.

* 現在、CVSのコミット権を持っている人

Rubyを使う意義

- Rubyはすべてがオブジェクト
 - データ構造を自然に表現
 - 生物学はデータの塊
- スクリプトを書きやすく読みやすい
 - 開発効率が高い
 - 情報科学に詳しくない人にもわかりやすい
- 拡張モジュールを(C言語で)書きやすい
 - パワーが必要な処理は拡張モジュールへ
 - 解析のプラットフォームとしての利用

他言語による先行プロジェクト

- Perl BioPerl
- Java BioJava
- Python Biopython

言語により得意分野が異なるので共存

- Open Bioinformatics Foundation (OBF) を結成
 - 情報交換や開発協力など
- データ入出力形式の標準化 (OBDA)

BioRubyの機能(1)

基本的なデータ構造・アルゴリズム

- 塩基・アミノ酸配列 (Bio::Sequence)
 - 部分配列の切り出し・翻訳など
- 配列上の位置情報 (Bio::Locations)
- アノテーション (Bio::Features)
- マルチプルアライメント (Bio::Alignment)
- 二項関係 (Bio::Relation)
- パスウェイ (Bio::Pathway)
- 文献情報 (Bio::References)
- ...

BioRubyの機能(2)

データベース等のデータフォーマット対応

- FASTA形式 (Bio::FastaFormat)
- GenBank/DDBJ (Bio::GenBank)
- EMBL (Bio::EMBL)
- SwissProt/TrEMBL (Bio::SPTR)
- PIR(NBRF形式) (Bio::NBRF)
- PDB (Bio::PDB)
- PROSITE (Bio::PROSITE)
- KEGG (Bio::KEGG::*)
- TRANSFAC (Bio::TRANSFAC)
- FANTOM (Bio::FANTOM)
- MEDLINE (Bio::MEDLINE)
- Gene Ontology (Bio::GO)
- 他、合計約26種類のデータ形式に対応

BioRubyの機能(3)

解析ソフトウェアの結果処理

- BLAST (Bio::Blast)
- FASTA (Bio::Fasta)
- HMMER (Bio::HMMER)
- CLUSTAL W (Bio::ClustalW)
- MAFFT (Bio::MAFFT)
- sim4 (Bio::Sim4)
- BLAT (Bio::BLAT)
- Spidey (Bio::Spidey)
- GenScan (Bio::GenScan)
- PSORT (Bio::PSORT)
- TarrgetP (Bio::TargetP)
- SOSUI (Bio::SOSUI)
- TMHMM (Bio::TMHMM)
- 他、合計約15種類の解析ソフトウェアに対応

BioRubyの機能(4)

ファイルやネットワーク経由のデータ入出力

- Bio::FlatFile
- Bio::FlatFileIndex
- Bio::Fetch
- Bio::SQL
- Bio::Registry
- Bio::DAS
- Bio::KEGG::API
- Bio::DDBJ::XML
- Bio::PubMed
- ...

分子生物学入門

- 基本は「細胞」
 - 脂質でできた膜(細胞膜)で仕切られている
 - 細胞質基質, 細胞内小器官, 核
- 細胞を構成する分子
 - タンパク質
 - 核酸(DNA, RNA)
 - 糖質
 - 脂質
 - ...

タンパク質とアミノ酸

■ タンパク質

- 数個～たくさんのアミノ酸が結合した1個の分子
- タンパク質を構成するアミノ酸は20種類のみ(例外あり)
 - 細菌からヒトまで全生物に共通
- 直線状に連結
- 方向がある(N末端→C末端)
- 折りたたみ・立体構造(3次元構造)
- 情報学的には文字列(String)として扱える

DNA

- DNA (デオキシリボ核酸)
- ヌクレオチドが連結した分子
 - ヌクレオチド: リン酸+糖(デオキシリボース)+塩基
 - 塩基は下記の4種類
 - A (アデニン, adenin)
 - G (グアニン, guanin)
 - C (シトシン, cytosine)
 - T (チミン, tymine)
 - 直線的に連結, 方向がある(5'→3')

DNAの二重らせん

- AとT, GとCが水素結合
- 二本鎖DNA
- 相補鎖
 - 5'-AAGTCGT-3' の相補鎖は 5'-ACGACTT-3'
 - 3'-TTCAGCA-5'
 - Ruby的には `str.tr('ACGT', 'TGCA').reverse`
- 半保存的複製

RNA

- RNA (リボ核酸)
- DNAと似ているが少し異なる
 - ヌクレオチド: リン酸+糖(リボース)+塩基
 - DNAとは糖が違う
 - 塩基4種類
 - T(チミン)のかわりにU(ウラシル)になっているところがDNAと違う
 - A (アデニン, adenin)
 - G (グアニン, guanin)
 - C (シトシン, cytosine)
 - U (ウラシル, uracil)

遺伝情報の流れ

- DNA: 遺伝情報を蓄積
- 転写: DNA → RNA
 - メッセンジャーRNA (mRNA)
- 翻訳: RNA → タンパク質
 - 3塩基(コドン) → 1アミノ酸
- 基本的には片方向の情報の流れ
 - セントラルドグマ
 - 例外: RNA → DNA: 逆転写
 - ウイルスなどで行われる

コドン表(遺伝暗号表)

- DNA(RNA)3塩基→1アミノ酸
- ほとんどすべての生物で同じ(例外あり)

UUU: F	UCU: S	UAU: Y	UGU: C
UUC: F	UCC: S	UAC: Y	UGC: C
UUA: L	UCA: S	UAA: *	UGA: *
UUG: L	UCG: S	UAG: *	UGG: W
CUU: L	CCU: P	CAU: H	CGU: R
CUC: L	CCC: P	CAC: H	CGC: R
CUA: L	CCA: P	CAA: Q	CGA: R
CUG: L	CCG: P	CAG: Q	CGG: R
AUU: I	ACU: T	AAU: N	AGU: S
AUC: I	ACC: T	AAC: N	AGC: S
AUA: I	ACA: T	AAA: K	AGA: R
AUG: M	ACG: T	AAG: K	AGG: R
GUU: V	GCU: A	GAU: D	GGU: G
GUC: V	GCC: A	GAC: D	GGC: G
GUA: V	GCA: A	GAA: E	GGA: G
GUG: V	GCG: A	GAG: E	GGG: G

いい加減な用語集

■ ゲノム

- 生物の遺伝情報全体
- 複数(または1本)の染色体から構成される

■ 染色体

- 1本の2本鎖DNA

■ 遺伝子

- 概念的なもの
- 1個のタンパク質になる塩基配列
- または、その配列が存在するゲノム上の領域

生物の分類

- 分子レベルで見ると3つの「ドメイン」に分類
 - 細菌 (Bacteria)
 - 例: 大腸菌、乳酸菌
 - 古細菌 (Archaea)
 - 例: メタン菌
 - 細菌と古細菌をあわせて原核生物と言う
 - 真核生物 (Eukaryota, Eukaryotes)
 - 酵母やカビからヒトまで
 - 植物も動物も真核生物という点では同じ
 - 単細胞の生物も多細胞の生物もいる

バイオインフォマティクス

- Bioinformatics
- 日本語訳は「生物情報学」
- 生物に関する情報を、情報科学や生物学の手法を組み合わせて解析し理解する学問
- 現在はゲノムや遺伝子やタンパク質の各種情報解析がメイン

国際塩基配列データベース

- アメリカ: GenBank

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

- ヨーロッパ: EMBL

<http://www.ebi.ac.uk/embl/>

- 日本: DDBJ

<http://www.ddbj.nig.ac.jp/>

- データや情報は相互に交換している

データの例 (GenBank)

- 1エントリ1配列
- 重複しない「アクセッション番号」が割り当てられている

```
LOCUS          HUMADH1CB              1400 bp    mRNA     linear   PRI 08-JUN-1995
DEFINITION     Homo sapiens class I alcohol dehydrogenase (ADH1) alpha subunit
               mRNA, complete cds.
ACCESSION      M12271
VERSION        M12271.1  GI:178091
KEYWORDS       ADH1 gene; alcohol dehydrogenase; alcohol dehydrogenase I;
               dehydrogenase.
SOURCE         Homo sapiens (human)
  ORGANISM     Homo sapiens
               Eukaryota; Metazoa; Chordata; Craniata; Vertebrata; Euteleostomi;
               Mammalia; Eutheria; Euarchontoglires; Primates; Catarrhini;
               Hominidae; Homo.
REFERENCE      1 (bases 1 to 1400)
  AUTHORS      Ikuta,T., Szeto,S. and Yoshida,A.
  TITLE        Three human alcohol dehydrogenase subunits: cDNA structure and
               molecular and evolutionary divergence
  JOURNAL      Proc. Natl. Acad. Sci. U.S.A. 83 (3), 634-638 (1986)
  PUBMED       2935875
COMMENT       Original source text: Homo sapiens (clone: pUCADH-alpha-15L) liver
               cDNA to mRNA.
               A draft entry and printed copy of the sequence in [1] were kindly
               provided by A.Yoshida, 30-MAY-1986.
               The other human class I ADH1 alpha subunit sequence is found under
               accession M11307.1
```

```

FEATURES                                 Location/Qualifiers
    source                                 1..1400
                                           /organism="Homo sapiens"
                                           /mol_type="mRNA"
                                           /db_xref="taxon:9606"
                                           /map="4q21-q23"
                                           /clone="pUCADH-alpha-15L"
                                           /tissue_type="liver"
    gene                                    1..1400
                                           /gene="ADH1"
    mRNA                                   <1..1400
                                           /gene="ADH1"
                                           /note="G00-119-650"
    CDS                                     16..1143
                                           /gene="ADH1"
                                           /EC_number="1.1.1.1"
                                           /note="alpha subunit"
                                           /codon_start=1
                                           /product="alcohol dehydrogenase 1"
                                           /protein_id="AAA68131.1"
                                           /db_xref="GI:178092"
                                           /db_xref="GDB:G00-119-650"
                                           /translation="MSTAGKVIKCKAAVLWELKKPFSIEEVEVAPPKAHEVRIKMVAV
GICGTDDHVVSGTMVTPLPVILGHEAAGIVESVGEGVTTVKPGDKVIPLAIPQCGKCR
ICKNPESNYCLKNDVSNPQGTLDGTSRFTCRRKP IHHFLGISTFSQYTVVDENAVAK
IDAASPLEKVCLIGCGFSTGYGSAVNVAKVTPGSTCAVFGLGGVGLSAIMGCKAAGAA
RIIAVDINKDKFAKAKELGATECINPQDYKKPIQEVLKEMTDGGVDFSFEVIGRLDTM
MASLLCCHEACGTSVIVGVPPDSQNL SMNPMLLLTGRTWKGAILGGFKSKECVPKLV A
DFMAKKFSLDALITHVLPFEKINEGFDLLHSGKSIRTI LMF"

```


ORIGIN 52 bp upstream of PvuII site; chromosome 4q21.

```
1 gaagacagaa tcaacatgag cacagcagga aaagtaatca aatgcaaagc agctgtgcta
61 tgggagttaa agaaaccctt ttccattgag gaggtggagg ttgcacctcc taaggcccat
121 gaagttcgta ttaagatggt ggctgtagga atctgtggca cagatgacca cgtggtagt
181 ggtaccatgg tgacccact tctgtgatt ttaggccatg aggcagccgg catcgtggag
241 agtgttggag aaggggtgac tacagtcaaa ccaggtgata aagtcatccc actcgtctatt
301 cctcagtgtg gaaaatgcag aatttgtaa aaccggaga gcaactactg cttgaaaaac
361 gatgtaagca atcctcaggg gaccctgcag gatggcacca gcaggttcac ctgcaggagg
421 aagcccatcc accacttctt tggcatcagc accttctcac agtacacagt ggtggatgaa
481 aatgcagtag ccaaaattga tgcagcctcg cctctagaga aagtctgtct cattggctgt
541 ggattttcaa ctggttatgg gtctgcagtc aatgttgcca aggtcacccc aggctctacc
601 tgtgctgtgt ttggcctggg aggggtcggc ctatctgcta ttatgggctg taaagcagct
661 ggggcagcca gaatcattgc ggtggacatc aacaaggaca aatttgcaa ggccaaagag
721 ttgggggcca ctgaatgcat caaccctcaa gactacaaga aacccatcca ggagggtgcta
781 aaggaaatga ctgatggagg tgtggatttt tcatttgaag tcatcggtcg gcttgacacc
841 atgatggctt ccctgttatg ttgtcatgag gcatgtggca caagtgtcat cgtaggggta
901 cctcctgatt cccaaaacct ctcaatgaac cctatgctgc tactgactgg acgtacctgg
961 aaggagacta ttcttggtgg ctttaaagt aaagaatgtg tcccaaaact tgtggctgat
1021 tttatggcta agaagttttc attggatgca ttaataacc atgttttacc ttttgaaaa
1081 ataaatgaag gatttgacct gcttcactct gggaaaagta tccgtaccat tctgatgttt
1141 tgagacaata cagatgtttt cccttgtggc agtcttcagc ctctctacc ctacatgatc
1201 tggagcaaca gctgggaaat atcattaatt ctgctcatca cagattttat caataaatta
1261 catttggggg ctttccaaag aatggaaat tgatgtaaaa ttatttttca agcaaatgtt
1321 taaaatccaa atgagaacta aataaagtgt tgaacatcag ctggggaatt gaagccaata
1381 aaccttctt ctttaaccatt
```

//

- 基本的にはテキスト形式
- 配列だけでなく付加情報も付いてくる

Fasta形式

- 配列データだけを扱う場合のシンプルな形式
- >から始まる行に配列のIDや説明など
- その直後に配列データ(配列データ中の改行は無視)

```
>M12271 human ADH1 alpha subunit mRNA
gaagacagaatcaacatgagcacagcaggaaaagtaatcaaagcagcagctgtgctatgggaggttaa
agaaacccttttccattgaggaggtggaggttgcacctcctaaggcccatgaagttcgtattaagatgg
ggctgttaggaatctgtggcacagatgaccacgtgggttagtggtaccatgggtgacccacttcctgtgatt
ttaggccatgaggcagccggcatcgtggagagtggtggagaaggggtgactacagtcaaaccaggtgata
aagtcaccccactcgctattcctcagtggtgaaaatgcagaatttgtaaaaacccggagagcaactactg
cttgaaaaacgatgtaagcaatcctcaggggaccctgcaggatggcaccagcaggttcacctgcaggagg
aagcccatccaccacttccttggcatcagcaccttctcacagtacacagtggtggatgaaaatgcagtag
ccaaaattgatgcagcctcgctctagagaaagtctgtctcattggctgtggattttcaactgggttatgg
gtctgcagtcaatggttgccaagggtcaccacaggctctacctgtgctgtgtttggcctgggaggggtcggc
ctatctgctattatgggctgtaagcagctggggcagccagaatcattgcggtggacatcaacaaggaca
aatttgcaaaggccaaagagttgggggccactgaatgcatcaaccctcaagactacaagaaacccatcca
ggaggtgctaaaggaaatgactgatggaggtgtggatttttcatttgaagtcacggctcggcttgacacc
atgatggcttcctgttatggtgtcatgaggcatgtggcacaagtgatcctaggggtacctcctgatt
ccaaaacctctcaatgaaccctatgctgctactgactggacgtacctggaagggagctattccttgggtg
ctttaaagtaagaatgtgtccaaaacttgtggctgattttatggctaagaagttttcattggatgca
ttaataaccatgttttacctttgaaaaataaatgaaggatttgacctgcttcactctgggaaaagta
tccgtaccattctgatgttttgagacaatacagatgtttcccttgtggcagcttcagcctcctctacc
ctacatgatctggagcaacagctgggaaatatcattaattctgctcatcacagattttatcaataaatta
catttgggggctttccaaagaaatggaaattgatgtaaaattattttcaagcaaatgtttaaatacca
atgagaactaaataaagtgttgaacatcagctggggaattgaagccaataaaccttccttctaaccatt
```

タンパク質データベース

■ UniProt

- <http://www.uniprot.org/>
- タンパク質配列データベース
- SwissProt, TrEMBL, PIR が統合してできた
- 実験データに基づいた高品質なデータ

■ PDB

- <http://www.rcsb.org/>
(日本ミラー: <http://www.pdbj.org/>)
- 立体構造データベース

データベース

- nr : non-redundant (冗長性のないという意味)
 - (塩基配列の場合は nt と称することも多い)
 - 古今東西のあらゆる配列を格納したデータベース
 - ただし、一部は含まない
 - NCBI, EMBL, DDBJ, GenomeNet などがそれぞれ独自作成
- データ量は年々増加
 - <http://www.ncbi.nlm.nih.gov/Genbank/genbankstats.html>
 - 2GB, 4GB越えも珍しくない
 - 32ビットの壁
 - 1ファイルで2GB,4GBを越えることもある
 - 例: <ftp://ftp.ncbi.nih.gov/blast/db/FASTA/>

ホモロジーサーチ

■ ホモロジーサーチ

- ある配列に「似た」配列をデータベースから検索すること

■ BLAST

- Basic Local Alignment Search Tool
- バイオインフォマティクスでもっともよく使われているソフトのひとつ
- <http://www.ncbi.nlm.nih.gov/BLAST/>

分子進化の中立説

- 1968年 木村資生(きむらもとお)が提唱
- 分子レベルの進化は、生物の生存に有利でも不利でもない中立な突然変異が集団に広まる(固定することにより起こる)
 - ある個体に偶然に起こった突然変異は
 - 有害で致死なら集団全体に広まらない
 - 不利でも有利でもない(中立)なら、偶然による
 - 有利だからといって必ずしも集団全体に広まるとは限らない
 - いずれにせよ、ほとんどの突然変異は集団全体に広まらず消えてしまう
- 配列の機能的に重要な部分ほど変わりにくい
- 機能的にあまり重要でない部分は変わりやすい

BioRubyのインストール方法

- Rubyのみで書かれているので簡単
 - % tar zxvf bioruby-0.6.2.tar.gz
 - % cd bioruby-0.6.2
 - % ruby install.rb config
 - % ruby install.rb setup
 - % sudo ruby install.rb install
- または、RubyGems を利用
 - % gem install bioruby
 - ただし、対応したばかりなのでテストは不十分

まず、使ってみる

```
#!/usr/bin/env ruby
require 'bio'

# require 'rubygems'          # RubyGems使用の場合
# require_gem 'bioruby'      # RubyGems使用の場合

#塩基配列を変数に格納
dna = Bio::Sequence::NA.new('ATGAGCACAGCAGGAAAAGTAATC')

# タンパク質に翻訳した結果を表示
print dna.translate, "\n"

# 相補鎖を表示
print dna.complement, "\n"
```


Bio::Sequenceクラス

- 塩基配列やアミノ酸配列を格納するクラス
- Bio::Sequence 汎用
- Bio::Sequence::NA 塩基配列
 - 塩基配列独自の処理を追加
- Bio::Sequence::AA アミノ酸配列
 - タンパク質独自の処理を追加
- Stringクラスを継承している

標準クラスを継承する際の注意点

```
class Foo < String; end
a = Foo.new('aaa')
b = a + 'bbb'
p b.class # ==> String #先祖返りしてしまう
```

```
# 必要なメソッドは上書きする必要がある
class Foo < String
  def +(s)
    self.class.new(super)
  end
end
a = Foo.new('aaa')
b = a + 'bbb'
p b.class # ==> Foo
```

- Ruby 1.6.6より前ではバグがあるので注意
- 詳細は[ruby-list:31866] から始まるスレッド参照

Bio::Sequence::NA 主なメソッド一覧

- `to_fasta(label, width)`
FASTAフォーマットに変換。`width`は省略時無限大。
- `subseq(from, to)`
部分配列を得る
- `splicing(position)`
スプライシングを行う。"1..100"や"`complement(join(1..10,20..30))`"
のような形式で指定
- `composition`
組成をハッシュとして返す
- `complement`
相補鎖を返す。
- `translate(frame = 1, table = 1)`
タンパク質への翻訳を行う。`frame`, `table`は省略可能。
Bio::Sequence::AAクラスのインスタンスを作成

Bio::Sequence::AA 主なメソッド一覧

- `to_fasta(label, width)`
FASTAフォーマットに変換。widthは省略時無限大。
- `subseq(from, to)`
部分配列を得る
- `composition`
組成をハッシュとして返す
- `codes`
3文字表記を返す
- `molecular_weight`
分子量を返す

ばらばらなデータ形式

■ 生物学関連のデータベースは719個存在

- Galperin, M.Y. (2005) The Molecular Biology Database Collection: 2005 update. *Nucleic Acids Research*, 33: D5-D24.

http://nar.oxfordjournals.org/cgi/content/full/33/suppl_1/D5

■ データベース毎にデータの形式は異なると考えたほうがよい＝それぞれパーサが必要

■ 各種解析ソフトの出力についても同様

- 解析ソフトは捕捉できるだけでも129～448種類以上

<http://bioinformatics.org/software/>

<http://sourceforge.net/> のBioinformaticsカテゴリ

データベース等のデータフォーマット対応

- FASTA形式 (Bio::FastaFormat)
- GenBank/DDBJ (Bio::GenBank)
- EMBL (Bio::EMBL)
- SwissProt/TrEMBL (Bio::SPTR)
- PIR(NBRF形式) (Bio::NBRF)
- PDB (Bio::PDB)
- PROSITE (Bio::PROSITE)
- KEGG (Bio::KEGG::*)
- TRANSFAC (Bio::TRANSFAC)
- FANTOM (Bio::FANTOM)
- MEDLINE (Bio::MEDLINE)
- Gene Ontology (Bio::GO)

など、合計約26種類のデータ形式に対応

解析ソフトウェアの出力のパーサ

- BLAST (Bio::Blast)
- FASTA (Bio::Fasta)
- HMMER (Bio::HMMER)
- CLUSTAL W (Bio::ClustalW)
- MAFFT (Bio::MAFFT)
- sim4 (Bio::Sim4)
- BLAT (Bio::BLAT)
- Spidey (Bio;;Spidey)
- GenScan (Bio::GenScan)
- PSORT (Bio::PSORT)
- TarrgetP (Bio::TargetP)
- SOSUI (Bio::SOSUI)
- TMHMM (Bio::TMHMM)

など、合計約15種類の解析ソフトウェアに対応

Bio::FlatFileでの自動判別

- データ形式をいちいち指定するのは面倒
- BioRubyでは自動判別に対応
 - Bio::FlatFileクラス (lib/bio/io/flatfile.rb)
 - 内部では単純に順番に正規表現で引っ掛けてるだけ

例: 入力ファイルの配列データを表示

```
#!/usr/bin/env ruby
require 'bio' #require_gem 'bioruby'
ARGV.each do |filename|
  ff = Bio::FlatFile.auto(filename)
  ff.each do |x|
    print x.seq, "\n"
  end
end
end
```


パーサ高速化のための遅延評価

(情報科学的に厳密に遅延評価と言えるのかどうかは謎)

- まず、データ全体をほとんど手を加えずインスタンス変数に蓄える
- メソッドが呼ばれたときに初めて、そのメソッドで要求されているデータだけ取り出す
 - ついでに他のデータも容易に取り出せるときはそうする
- 取り出したデータもインスタンス変数に保存
 - 次回以降そのメソッドが呼ばれたときはその変数の値を返す
- メモリは食うがトータルでは速いことが多い
 - データの一部分しか使わないことのほうが多いため

BLAST結果の例

BLASTN 2.2.6 [Apr-09-2003]

Reference: Altschul, Stephen F., Thomas L. Madden, Alejandro A. Schaffer, Jinghui Zhang, Zheng Zhang, Webb Miller, and David J. Lipman (1997), "Gapped BLAST and PSI-BLAST: a new generation of protein database search programs", Nucleic Acids Res. 25:3389-3402.

Query= ri|0610005A07|R000001A15|1277 contigs=2 ver=1 seqid=2
(1277 letters)

Database: fantom2.00.seq
60,770 sequences; 119,956,725 total letters

Searching.....done

バージョン

Reference

Queryの情報

データベースの情報

Sequences producing significant

```
ri|0610005A07|R000001A15|1277 c
ri|0610039M06|R000004L05|1061 c
ri|4930431E11|PX00030N13|1181 c
ri|1110004G14|R000015H01|1462 c
ri|1700124M20|ZX00096C11|926 co
ri|2900019E12|ZX00083B15|841 co
ri|0610033N11|R000004G20|840 co
ri|9430011C20|PX00107J21|1874 c
ri|B830049N13|PX00073P19|1106 c
```

HSP

High-Scoring Segment Pair の略。

BLASTによる相同性検索結果の最小単位

```
>ri|0610005A07|R000001A15|1277 contigs=2 ver=1 seqid=2
Length = 1277
```

Score = 2531 bits (1277), Expect = 0.0
Identities = 1277/1277 (100%)
Strand = Plus / Plus

```
Query: 1   gggcagctctctgaacagccaaggctagattgacactgagcctgtccggttcagacctcg 60
          |||
Sbjct: 1   gggcagctctctgaacagccaaggctagattgacactgagcctgtccggttcagacctcg 60
```

HSP

Hit

~~~~~(中略)~~~~~

```
>ri|1110004G14|R000015H01|1462 contigs=2 ver=1 seqid=1271
Length = 1462
```

Score = 297 bits (150), Expect = 3e-79  
Identities = 207/226 (91%)

~~~~~(中略)~~~~~

>ri|1110004G14|R000015H01|1462 contigs=2 ver=1 seqid=1271
Length = 1462

Score = 297 bits (150), Expect = 3e-79
Identities = 207/226 (91%)
Strand = Plus / Plus

Query: 113 attcgctgttctcctggaatacacagactcaagctatgaggagaagagatacaccatgggt 172

||||| ||| ||| ||||||||| ||||||||| ||||||||| |||||||||

Sbjct: 29 attcggtctctcctagaatacacaggctcaagctatgaagagaagagatacaccatggga 88

Query: 173 gatgctcctgactatgaccaaagccagtggctgaatgagaaattcaagctgggctggac 232

|| ||||||||| ||||||||| ||||||||| ||||||| ||||| |||||||||

Sbjct: 89 gacgctcctgactatgaccgaagccagtggctgagtgagaagttaaattgggctggac 148

Query: 233 tttcctaacctgcctacttgatcgatgggtcacacaagatcacgcagagcaatgccatc 292

||||||| ||| ||||||||| ||||||||| ||||||||| ||||||||| |||||||||

Sbjct: 149 tttcccaattgccttacttgattgatgggtcacacaagatcacgcagagcaatgccatc 208

Query: 293 ctgcgctaccttggccgcaagcacaacctgtgtggggagacagagg 338

||||||| ||| ||||||||| ||||||||| ||||||||| |||||||||

Sbjct: 209 ctgcgctacattgcccgcaagcacaacctgtgtggggagacagagg 254

HSP

Score = 93.7 bits (47), Expect = 1e-17
Identities = 110/131 (83%)
Strand = Plus / Plus

Query: 583 gtgcctggatgcgttcccaaacctgaaggacttcatagcgcgtttgagggcctgaagaa 642

||||||| || ||||||||| ||||||||| || || ||||||||| |||||||||

Sbjct: 499 gtgcctggacgccttcccaaacctgaaggactttgtggcccgcgtttgaggtactgaagag 558

Query: 643 gatctccgactacatgaagaccagtcgcttccctcccaagaccatgttcacaaagatggc 702

|||||| | ||||||||| ||||||||| || |||| | |||||| ||||

Sbjct: 559 gatctctgcttacatgaagaccagccgcttccctccgaacaccctatatacaaagtggtg 618

Query: 703 aacttggggca 713

|||||||

Sbjct: 619 cacttggggca 629

Hit

HSP

Score = 56.0 bits (28), Expect = 2e-06
Identities = 106/132 (80%)
Strand = Plus / Plus

BLASTパーサの比較

- BioRuby
- BioPerl
- Zerg
 - 高速なBLASTパーサとして最近発表された
 - C言語で実装されたライブラリ(`lex`使用)
 - Perlからも使用可能
 - Paquola, A.C.M., *et al.* (2003) Zerg: a very fast BLAST parser library, *Bioinformatics*, 19, 1035-1036.

機能比較

| | BioRuby
(0.5.3) | BioPerl
(1.2.1) | Zerg
(1.0.3) |
|--------------|--------------------|--------------------|--------------------|
| 言語 | Ruby | Perl | C
(Perlからも使用可能) |
| NCBI BLAST対応 | ○ | ○ | ○* |
| HSPのアライメント取得 | ○ | ○ | × |
| PSI-BLAST対応 | ○ | ○ | × |
| WU-BLAST対応 | ○* | ○ | × |

* 一部の統計情報には未対応

実行速度比較

- ベンチマークプログラムを10回動作させたときの平均所要時間と処理速度およびBioPerlを1としたときの速度比を求めた。
 - テストデータ
 - BLASTN実行結果 104,921,408バイト 8014エン트리
 - マシンのスペック
 - PentiumIII 1GHz, メモリ1GB, HDD 27GB
 - OS: Linux 2.4.18

実行速度比較

| | 所要時間(s) | S.D. | 速度(MB/s) | 速度比 |
|-------------------------------|---------|-------|----------|------|
| BioRuby
(Ruby1.8.0) | 35.325 | 0.032 | 2.83 | 21.3 |
| BioRuby
(Ruby1.6.7) | 49.724 | 0.048 | 2.01 | 15.1 |
| BioPerl
(Perl5.6.1) | 751.067 | 2.915 | 0.133 | 1 |
| Zerg-C | 2.437 | 0.002 | 41.1 | 308 |
| Zerg-Perl | 2.605 | 0.002 | 38.4 | 288 |
| Zerg-Perl2 | 36.687 | 0.051 | 2.73 | 20.5 |

考察

- 機能は BioPerl \doteq BioRuby $>$ Zerg
- 速度は Zerg $>$ BioRuby $>$ BioPerl
- BioRubyはBioPerlと同等の機能を持ちながら
20倍以上高速
- ZergはBioRubyよりさらに15倍以上高速だが
 - 機能が少ない
 - コンパイルやインストールが必要

今後の課題

- ドキュメントやサンプルの整備
- UnitTest
- 対応データベース・ソフトウェアの拡大
- リファクタリング
- 解析機能の充実
- BioRubyを使用したソフトウェアの開発
- ...

<http://bioruby.org/>