

# Bioinformatics Recipes with BioRuby

Mitsuteru Nakao<sup>1</sup>  
n@bioruby.org

Naohisa Goto<sup>2</sup>  
ng@bioruby.org

Toshiaki Katayama<sup>3</sup>  
k@bioruby.org

- <sup>1</sup> Computational Biology Research Center (CBRC), National Institute of Advanced Industrial Science and Technology (AIST), Aomi Frontier Building 17F, 2-43 Aomi, Koto-ku, Tokyo 135-0064, Japan
- <sup>2</sup> Genome Information Research Center, Osaka University, Yamadaoka 3-1, Suita, Osaka 565-0871, Japan
- <sup>3</sup> Human Genome Center, Institute of Medical Science, University of Tokyo, 4-6-1 Shirokanedai, Minato-ku, Tokyo 108-8639, Japan

**Keywords:** open source, bioinformatics library, Ruby language, object oriented

## 1 Introduction

The BioRuby project [1] is developing an open source class library for bioinformatics written in the object oriented scripting language Ruby [2]. The BioRuby library provides various methods for manipulating biological sequences, accessing biological databases, parsing database entries, executing biological analysis applications and parsing their results. We have abstracted a number of biological objects into classes having intuitive methods, so that BioRuby can be easily learned, as well as effectively reduce the amount of code that users need to write. During the past year, support for protein structures (PDB), support for the KEGG API v3.0, and unit tests for several classes have been added.

In this software demonstration, we will present a series of recipes for typical “dry”-tasks often requested by biologists. We are also going to provide a hands-on seminar that includes a 10 minute guide for getting started and a demonstration showing the steps of developing a complex pipeline for real world bioinformatics tasks.

## 2 Mini HOWTO

### 2.1 Biological sequence manipulation

BioRuby provides classes for DNA sequences and protein sequences. They allow us to manipulate the sequences easily.

```
require 'bio'
str = 'tcgaaactg'
na = Bio::Sequence::NA.new(str)
na.length          #=> 9
na.complement      #=> "cagtttcga"
na.subseq(2, 4)    #=> "cga"
na.composition     #=> {"a"=>3, "c"=>2, "g"=>2, "t"=>2}
na.gc_percent      #=> 44.4
window_width = 3
step_size = 3
```

```

na.window_search(window_width, step_size) do |window|
  #=> "cag", "ttt", "cga"
end
aa = na.translate #=> "SKL"
aa.molecular_weight #=> 346.42
aa.composition #=> {"K"=>1, "L"=>1, "S"=>1}

```

## 2.2 Working with sequence annotations from biological databases

BioRuby can treat over 20 major database formats including GenBank, SwissProt and KEGG as well as clients for bioinformatics applications including BLAST, Fasta and PSORT. The following example shows how to obtain a SwissProt entry by the BioFetch method which is a part of the OBDA (Open Bio\* Database Access) [3], and then to predict the subcellular localization by PSORT WWW Server via the Internet.

```

require 'bio'
entry = Bio::Fetch.new.fetch("sp", "UBCI_HUMNA")
sp = Bio::SwissProt.new(entry)
sp.entry_id #=> "UBCI_HUMAN"
sp.definition #=> "Ubiquitin-like protein SUMO-1 conjugating enzyme ..."
sp.seq.length #=> 158
# Feature table (FT Lines)
sp.ft.keys #=> ["HELIX", "TURN", "CONFLICT", "ACT_SITE", "STRAND"]
# Comments (CC Lines)
sp.cc.keys #=> ["CATALYTIC ACTIVITY", "TISSUE SPECIFICITY", "FUNCTION", ...]
sp.cc["TISSUE SPECIFICITY"] #=> ["In mouse, expressed in spleen, kidney, ..."]
# Predicting protein subcellular localization by PSORT WWW Server
psort2_serv = Bio::PSORT::PSORT2.imsut
report = psort2_serv.query(sp.seq.to_fasta(sp.entry_id))
report.entry_id #=> "UBCI_HUMAN"
report.pred #=> "nuc"

```

## 3 Conclusion

We have been providing well defined classes which correspond to biological objects, focusing on maximizing usability and efficiency for biologists so that the users of BioRuby library can easily develop their own analysis pipeline. We are trying to enhance the reliability of our library by adding unit tests for each classes. We welcome novel users to start using BioRuby. Users can use the BioRuby mailing list to exchange information concerning the library with us.

## References

- [1] Goto, N., Nakao, M., Kawashima, S., Katayama, T., and Kanehisa, M., BioRuby: Open-source bioinformatics library, *Genome Informatics*, 14:629–630, 2003.
- [2] <http://www.ruby-lang.org/>
- [3] <http://obda.open-bio.org/>