# **AnnoGene:**
# RESTful web service for annotating genomic features

## User's Manual

Current version: 1.0.0
Last updated: November 13-th 2013

Andrzej Tomski, Marcin Piechota and Ryszard Przewłocki

Department of Molecular Neuropharmacology

Institute of Pharmacology of the Polish Academy of Sciences

Contact author: andrzejtomski@wp.pl

# TABLE OF CONTENTS

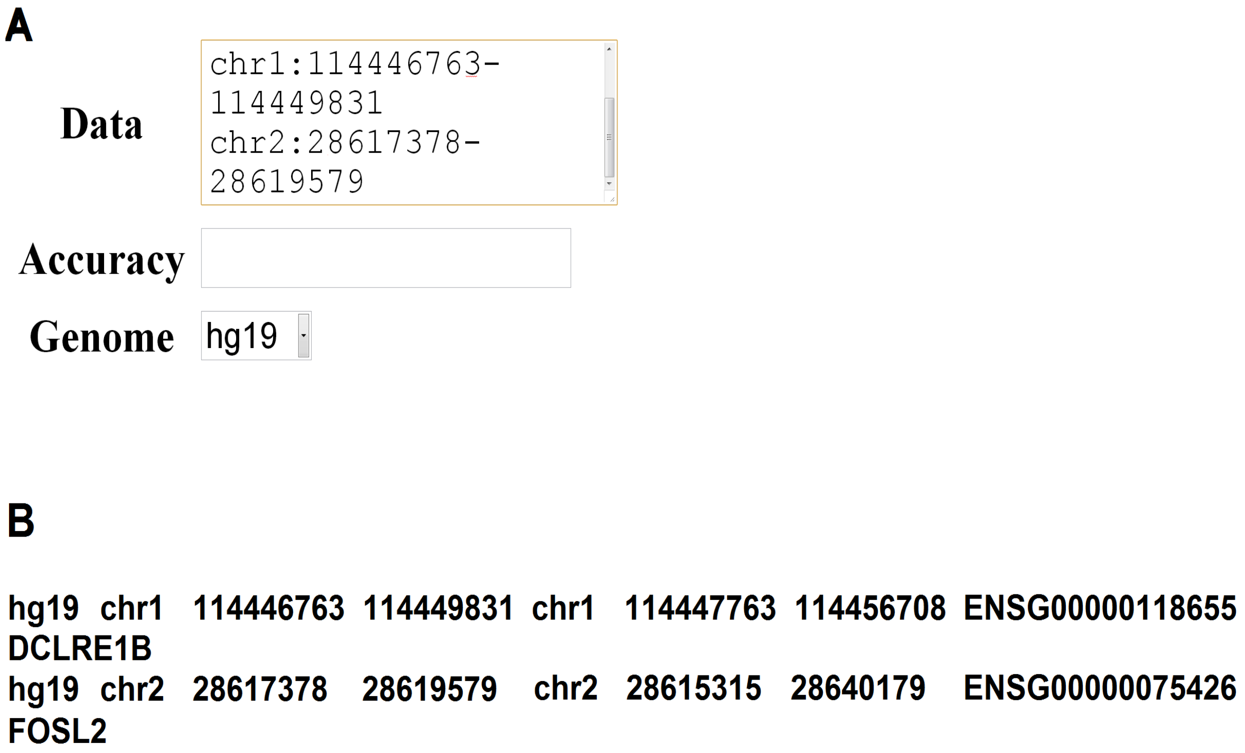
# Overviev

## Background

Modern high-throughput sequencing techniques generate a constantly increasing amount of genomic data. The main problem is quickly identifying the data that may provide information about the nature of various intracellular processes. Typically, thousand of peaks or signals are found across the genome and we want to annotate the nearby genes. **AnnoGene** is a web service implemented in a representational state transfer (REST) style for annotating genomic features. The program searches for the gene nearest to the center of a genomic position. Subsequently, the location and annotations (Ensembl ID and MGI or HGNC symbol) of the gene are shown. AnnoGene is freely available at http://bedanno.cremag.org. Moreover, we provided examples of the REST clients written in the Python, R and Java programming languages. AnnoGene only requires genomic positions from the user. Even when annotating several thousands positions, the output is typically ready in a few seconds.

## Getting started: simple example

Let's suppose we have some BED data of ChiP-seq peaks. We want to find and annotate the genes nearest to those peaks. The analysis can be easily done with the help of AnnoGene. The example of annotating hg19 data is presented below:



Data
```
chr1:114446763-
114449831
chr2:28617378-
28619579
```

Accuracy

Genome hg19

```
hg19  chr1   114446763  114449831  chr1   114447763  114456708  ENSG00000118655
DCLRE1B
hg19  chr2   28617378   28619579   chr2   28615315   28640179   ENSG00000075426
FOSL2
```

‘

**A** shows the required fields and **B** is the output.

## Program details

- Program name: AnnoGene

- Availibility: http://bedanno.cremag.org

- Technology: RESTful web service

- Description: searches for the gene nearest to the center of a genomic position with the desired accuracy

- Accession: any web browser or one of the REST clients

- Programming language: Python 2.7

- Source code: freely available at http://github.com/andrewtom/AnnoGene

- Install commands for necessary libraries and frameworks (see References):

  ▷ **sudo easy_install web.py**: web.py v0.37 or greater is a web framework needed to run a copy of AnnoGene on a PC,

  ▷ **available by default with pip**: urllib and urllib2 are the libraries being used by the Python REST client,

  ▷ **install.packages('RCurl')**: RCurl $v1.95-4.1$ or greater is a command to type in the R console. RCurl is a package to handle HTTP requests through the R REST client.

  ▷ running AnnoGene from the command line: **python AnnoGene.py A** and the only parameter **A** is the associated port number.

# The interface

## Options

The web interface requires three simple steps. First, the user enters the data into the **'Data'** form. Then, the user specifies an accuracy for the analysis in the **'Accuracy'** text box (the accuracy indicates the distance from the center of the region the operation is performed). Finally, the user chooses one of the available genomes from **'Genome'** drop-down list . If the accuracy is limited to a very small number of base nucleotides, there may be no genes sufficiently close to the entered position. However, when no accuracy is specified, the nearest gene will be found. The acceptable data format depends on the field:

**'Data'** form is a place to enter the data. Genomic positions can be entered in two different formats:

- divided by whitespace (i.e., tab delimited): chrName Start End
- the format used in most of genomic databases: chrName:Start-End.

For example, **chr1 100000 10000000** is acceptable, but **chr 1 100000 10000000** not.

**'Accuracy'** can be any non-negative integer. When no accuracy is specified, the nearest gene will be found.

**'Genome'** is a drop-down list. The three most popular and most frequently used genome assemblies are available: mm9 and mm10 for mouse and hg19 for human. New assemblies will be added in the future.

## Output

AnnoGene finishes by presenting all of the lines together with their annotations. This is a brief description of the output by AnnoGene.

Each line is a collection of a few fields separated by tabs. These fields are:

- 1. **Genome assembly**
- 2.-4. **Peak location**
- 5.-7. **The nearest gene location**
- 8. **Ensembl ID**
- 9. **HGNC(MGI) symbol**

## Client programs

AnnoGene can also be accessed through one of the provided clients (Java, Python and R). Below we present the client code listings.

**(I) Python code:**

```
#loading the libraries
import urllib
import sys

url=”http://bedanno.cremag.org”
params=urllib.urlencode({”Data”:sys.argv[1],”Accuracy”:sys.argv[2],”Genome”:sys.argv[3]})
if sys.argv[3] in [”mm9”,”mm10”,”hg19”]:

#retrieve a URL containing parameters
   response=urllib.urlopen(url,params).read()
   if ”negative” in response:
        print ”Accuracy must be positive!”
   elif ”Incorrect” in response:
        print ”Wrong match: line 1”
   else:
        print response
else:
   print ”Genome not available. Try again!”
```

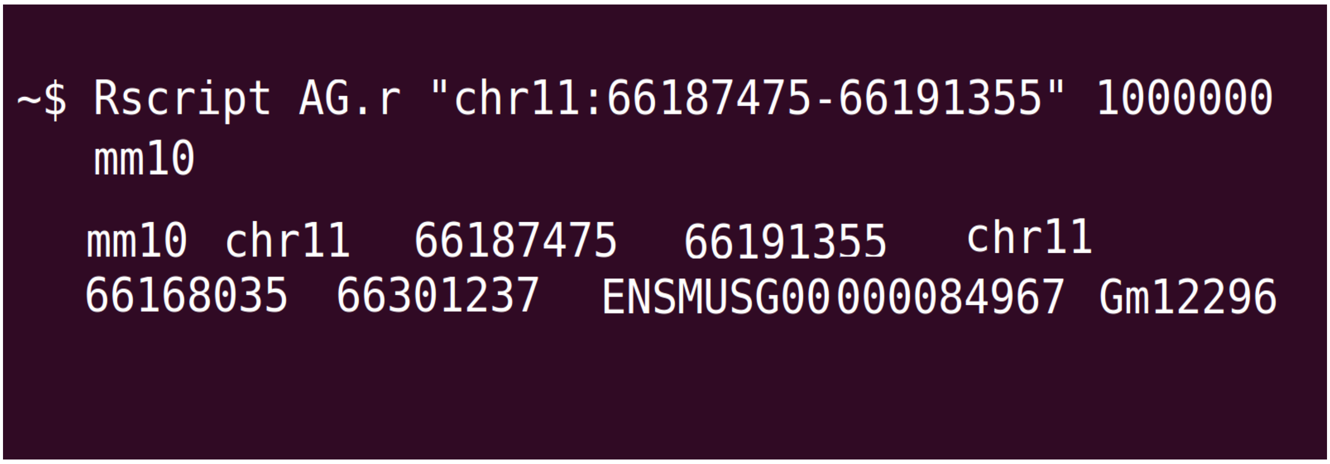
**(II) R code:**

```
args ← commandArgs(TRUE)

#RCurl loading
library(”RCurl”)

#fetch a URL and submit forms
url ← ”http://bedanno.cremag.org”
if (args[2] !=” ”){
   if (as.integer(args[2])<0) {
     cat(”Accuracy must be positive!\n”);quit(save=”no”)
  }
}
tryCatch({
   result ← postForm(url,Data=args[1],Accuracy=args[2],Genome=args[3])
},
error=function(e) {
   cat(”Genome not available!\n”);quit(save=”no)
})
tryCatch({
   cat(rawToChar(result))
},
error=function(e) {
   cat(”Wrong match: line 1 \n”)
})
```

**(III) Java code:**

```
String bedString = ””;
for(BedItem item : bedItems)
          bedString += item.toString() + ”\n”;
bedString = bedString.trim();
String parameters = ”Data=” + bedString +
                              ”&Accuracy=” + ”0” +
                              ”&Genome=” + genome;
URL url = new URL(”http://bedanno.cremag.org”);
URLConnection conn = url.openConnection();
conn.setDoOutput(true);
OutputStreamWriter writer = new OutputStreamWriter(conn.getOutputStream());
//write parameters
writer.write(parameters);
writer.flush();
//get the response
BufferedReader reader = new BufferedReader(new InputStreamReader(conn.getInputStream()))
String line;
while ((line = reader.readLine()) != null) {
         System.out.println(line);
}
writer.close();
reader.close();
```

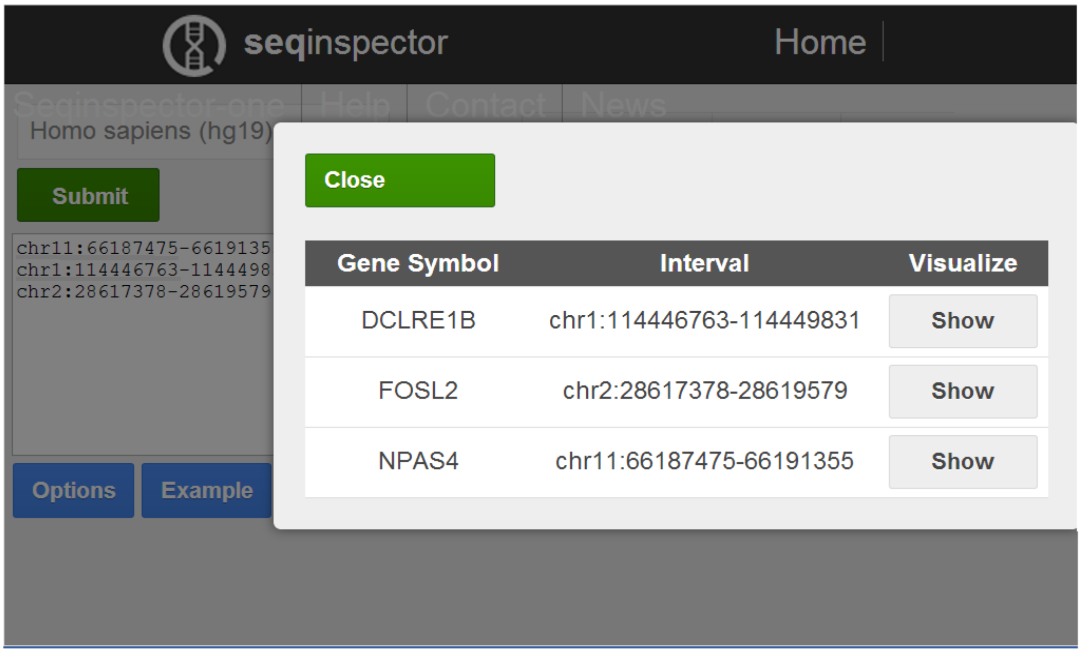
**R client example**



```
~$ Rscript AG.r "chr11:66187475-66191355" 1000000
   mm10
   mm10  chr11   66187475   66191355   chr11
   66168035  66301237   ENSMUSG00000084967 Gm12296
```

# Integration with Seqinspector

AnnoGene has been integrated with Seqinspector (http://seqinspector.cremag.org), a web tool for finding putative regulators of genes and studying protein-protein interactions. A typical application involves obtaining peaks from a ChIP-seq experiment, then calculating the average coverage for all tracks, and performing a two-sample t-test with a comparison to a reference. Low p-value levels that are validated by the Bonferroni correction lead to the identification of the associated transcription factors.



```
chr11:66187475-6619135
chr1:114446763-1144498
chr2:28617378-28619579
```

| Gene Symbol | Interval | Visualize |
|---|---|---|
| DCLRE1B | chr1:114446763-114449831 | Show |
| FOSL2 | chr2:28617378-28619579 | Show |
| NPAS4 | chr11:66187475-66191355 | Show |

## License

AnnoGene is freely available to academic and non-academic users at the http://bedanno.cremag.org.

# References

- **Python(v2.7): http://www.python.org**
- **Apache: http://www.apache.org**
- **ENSEMBL: http://www.ensembl.org**
- **HUGO Gene Nomenclature Committee: http://www.genenames.org**
- **Mouse Genome Informatics: http://www.informatics.jax.org**
- **web.py(v0.37): http://webpy.org**
- **urllib2: http://docs.python.org/2/howto/urllib2.**
- **RCurl(v1.95-4.1): http://www.omegahat.org/RCurl.**