Supplementary S7 – Web Server Description

Contents

Caution

This online service is developed on Python, Django, Celery and Redis. Three message queues with different priorities are used to handle three different time-consuming tasks.

For security reasons, no part of the site's functionality will be open sourced. If you encounter any problems or are interested in the development of this online service, please contact me[1].

[1] qizhang18@mails.jlu.edu.cn

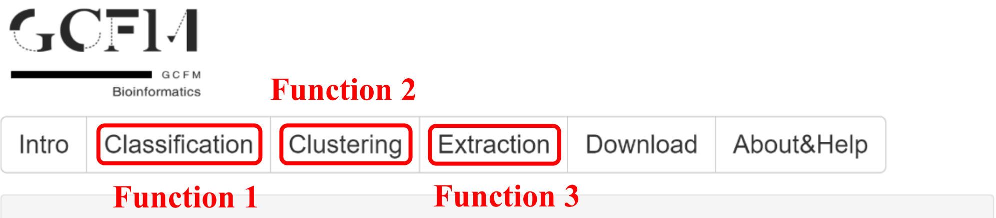
1. Introduction

The web server consists of three functions:

1) Classification function, calculating whether the two ncRNAs are in the same family. When the sequences in the FASTA format of the two RNAs are input by a user, the server obtains similarities between two ncRNAs and evaluate whether they are in the same family.
2) Clustering function, clustering of multiple sequences based on a classification matrix and affinity propagation algorithm. The clustering function implements the derivation of bulk ncRNA family attribution and makes the classification matrix available for download, allowing users to further build phylogenetic trees based on the classification matrix, etc.
3) Batch feature extraction function. For the batch sequence input by users, features can be extracted automatically. For those who just want to use multi-view features, we provide unmatched feature extraction, where the extracted features can be applied to the study of multi-categorization of ncRNA and the study of interactions with ncRNA and protein.

The web server is available at http://bmbl.sdstate.edu/gcfm/. The web server has been tested on several different browsers, including Microsoft Internet Explorer, Mozilla Firefox, Google Chrome and MacOS safari and also on Android OS.

This supplementary file provides a brief instruction to display the usage of the web server of GCFM.



Abstract

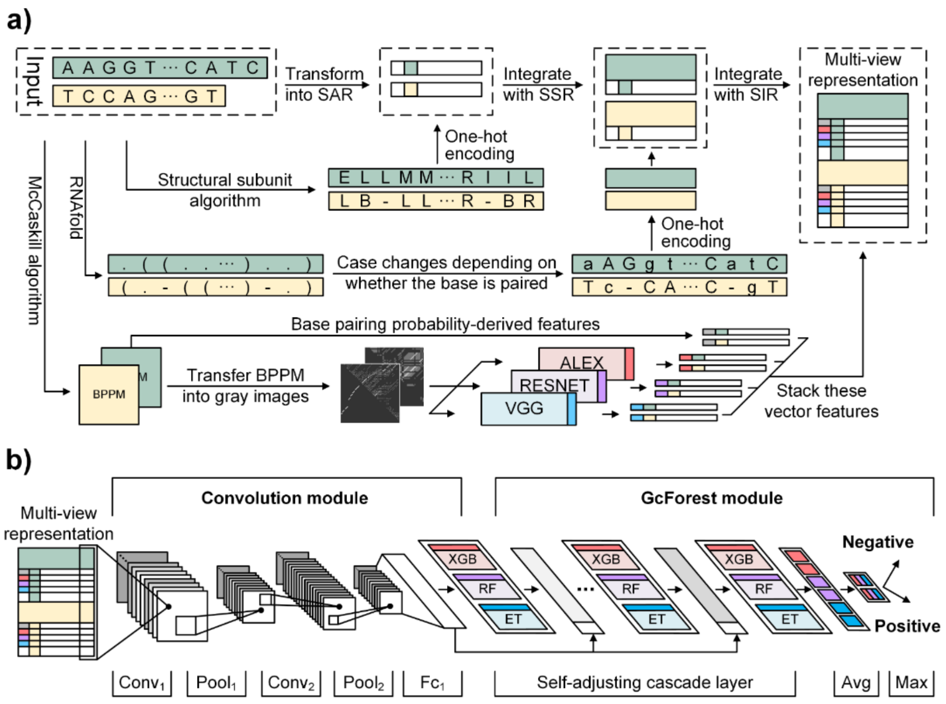
Motivation

Non-coding RNAs (ncRNAs) play crucial roles in multiple biological processes. However, only a few ncRNAs' functions have been well studied. Given the significance of ncRNAs classification for understanding ncRNAs' functions, more and more computational methods have been introduced to improve the classification automatically and accurately.

Result

In this paper, based on a convolutional neural network and a deep forest algorithm, multi-grained cascade forest (GcForest), we propose a novel deep fusion learning framework, GcForest fusion method (GCFM), to classify alignments of ncRNA sequences for accurate clustering of ncRNAs. GCFM integrates a multi-view structure feature representation including sequence-structure alignment encoding, structure image representation, and shape alignment encoding of structural subunits, enabling us to capture the potential specificity between ncRNAs. For the classification of pairwise alignment of two ncRNA sequences, the F-value of GCFM improves 6% than an existing alignment-based method. Furthermore, the clustering of ncRNA families is carried out based on the classification matrix generated from GCFM. Results suggest better performance (with 20% Accuracy improved) than existing ncRNA clustering methods (RNAclust, Ensembleclust, and CNNclust). Additionally, we apply GCFM to construct a phylogenetic tree of ncRNA and predict the probability of interactions between RNAs. Most ncRNAs are located correctly in the phylogenetic tree, and the prediction Accuracy of RNA interaction is 90.63%.

Overall



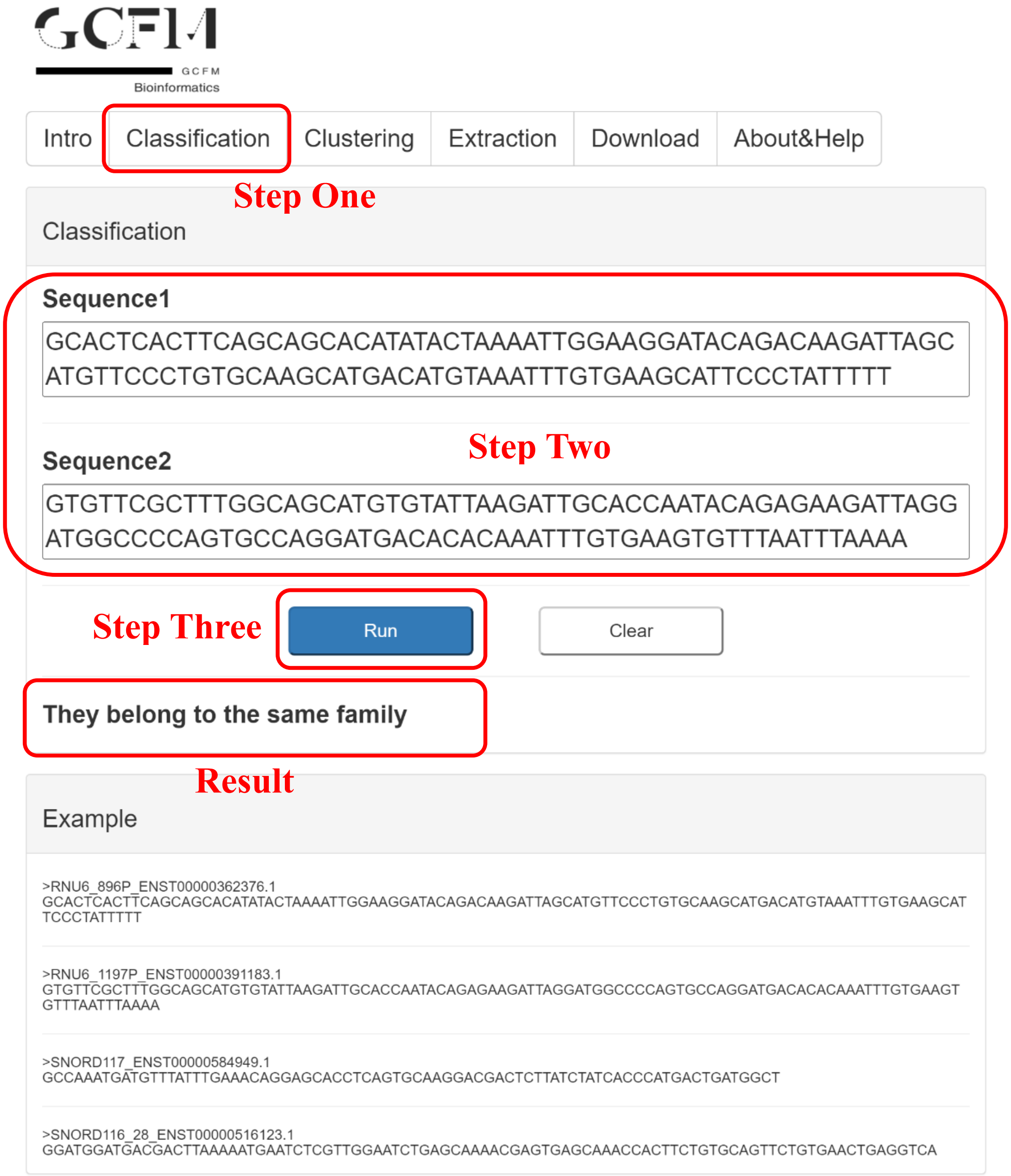
The framework of GCFM.

a) The whole flow chart of the multi-view structure feature representations. When two ncRNA sequences to be predicted are input, three feature representations (SSR, RNA SIR, and SAR), will be extracted, respectively. BPPM represents the base-pairing probabilities matrix.

b) The overall architecture of the model. According to the obtained multi-view structure feature representations, the multi-view features are extracted through the convolution module. The final classification results are obtained through the GcForest module with cascading.

2. Classification function

Predict if the two sequences belong to the same ncRNA family. First, click on the "**Classification**" button to go to the classification function page. Next, enter the sequence in the two text boxes in step two. Click on the "**Run**" button. The results will be displayed in the "**Result**" area.



Step One

Classification

Sequence1

```
GCACTCACTTCAGCAGCACATATACTAAAATTGGAAGGATACAGACAAGATTAGC
ATGTTCCCTGTGCAAGCATGACATGTAAATTTGTGAAGCATTCCCTATTTTT
```

Sequence2

Step Two

```
GTGTTCGCTTTGGCAGCATGTGTATTAAGATTGCACCAATACAGAGAAGATTAGG
ATGGCCCCAGTGCCAGGATGACACACAAATTTGTGAAGTGTTTAATTTAAAA
```

Step Three

They belong to the same family

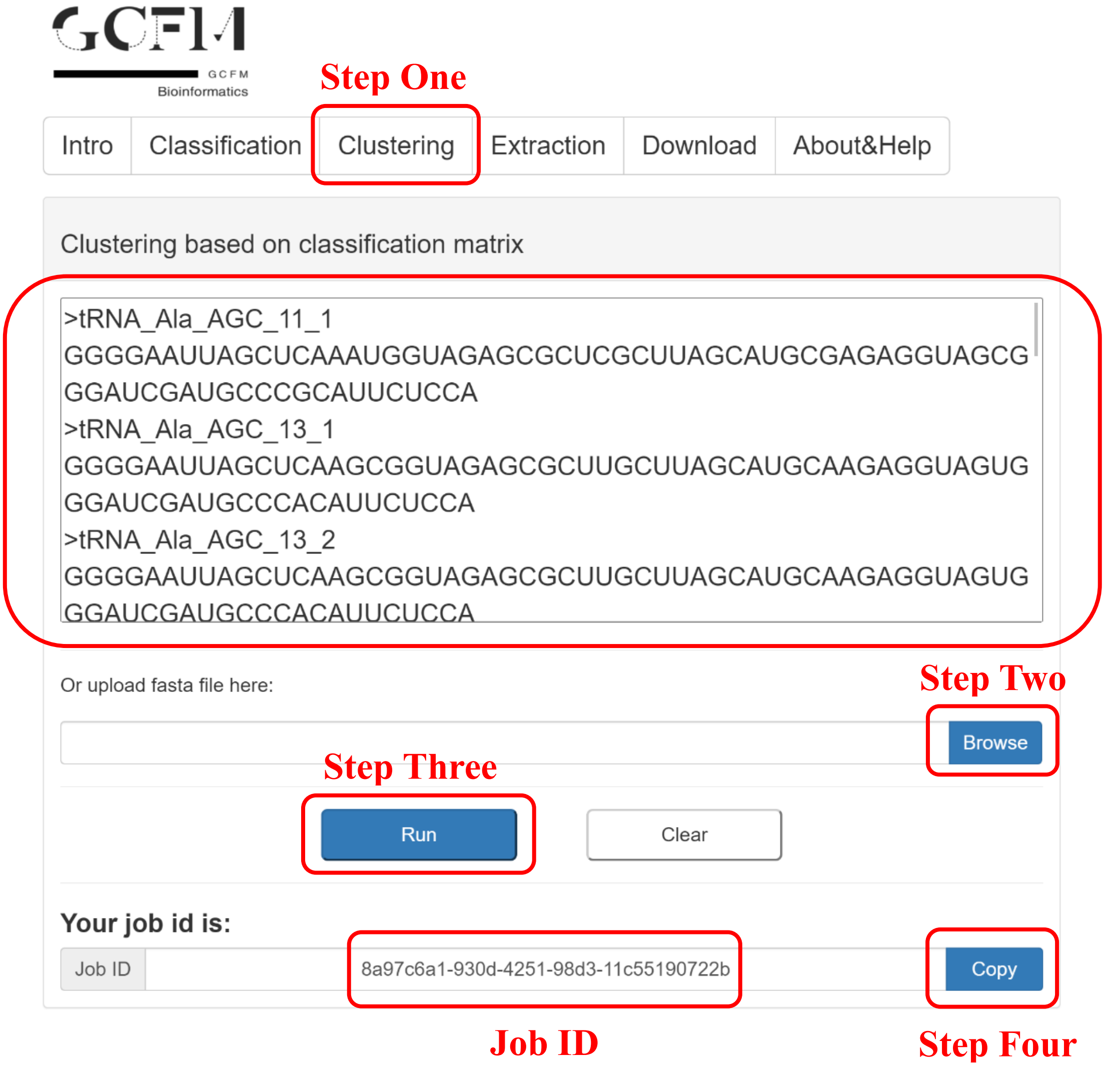
Result

Example

```
>RNU6_896P_ENST00000362376.1
GCACTCACTTCAGCAGCACATATACTAAAATTGGAAGGATACAGACAAGATTAGCATGTTCCCTGTGCAAGCATGACATGTAAATTTGTGAAGCAT
TCCCTATTTTT

>RNU6_1197P_ENST00000391183.1
GTGTTCGCTTTGGCAGCATGTGTATTAAGATTGCACCAATACAGAGAAGATTAGGATGGCCCCAGTGCCAGGATGACACACAAATTTGTGAAGT
GTTTAATTTAAAA

>SNORD117_ENST00000584949.1
GCCAAATGATGTTTATTTGAAACAGGAGCACCTCAGTGCAAGGACGACTCTTATCTATCACCCATGACTGATGGCT

>SNORD116_28_ENST00000516123.1
GGATGGATGACGACTTAAAAATGAATCTCGTTGGAATCTGAGCAAAACGAGTGAGCAAACCACTTCTGTGCAGTTCTGTGAACTGAGGTCA
```

3. Clustering function

First, click on the “**Clustering**” button to go to the clustering function page. Next enter multiple sequences in the text box in step two or click the “**Browse**” button to upload a file containing the sequences in fasta format. Then click on the “**Run**” button to create the task and get the Job ID, which is displayed in the “**Job ID**” area. Click “**Copy**” to copy the Job ID, check the status and get the results in the Download page.



Clustering based on classification matrix

```
>tRNA_Ala_AGC_11_1
GGGGAAUUAGCUCAAAUGGUAGAGCGCUCGCUUAGCAUGCGAGAGGUAGCG
GGAUCGAUGCCCGCAUUCUCCA
>tRNA_Ala_AGC_13_1
GGGGAAUUAGCUCAAGCGGUAGAGCGCUUGCUUAGCAUGCAAGAGGUAGUG
GGAUCGAUGCCCACAUUCUCCA
>tRNA_Ala_AGC_13_2
GGGGAAUUAGCUCAAGCGGUAGAGCGCUUGCUUAGCAUGCAAGAGGUAGUG
GGAUCGAUGCCCACAUUCUCCA
```

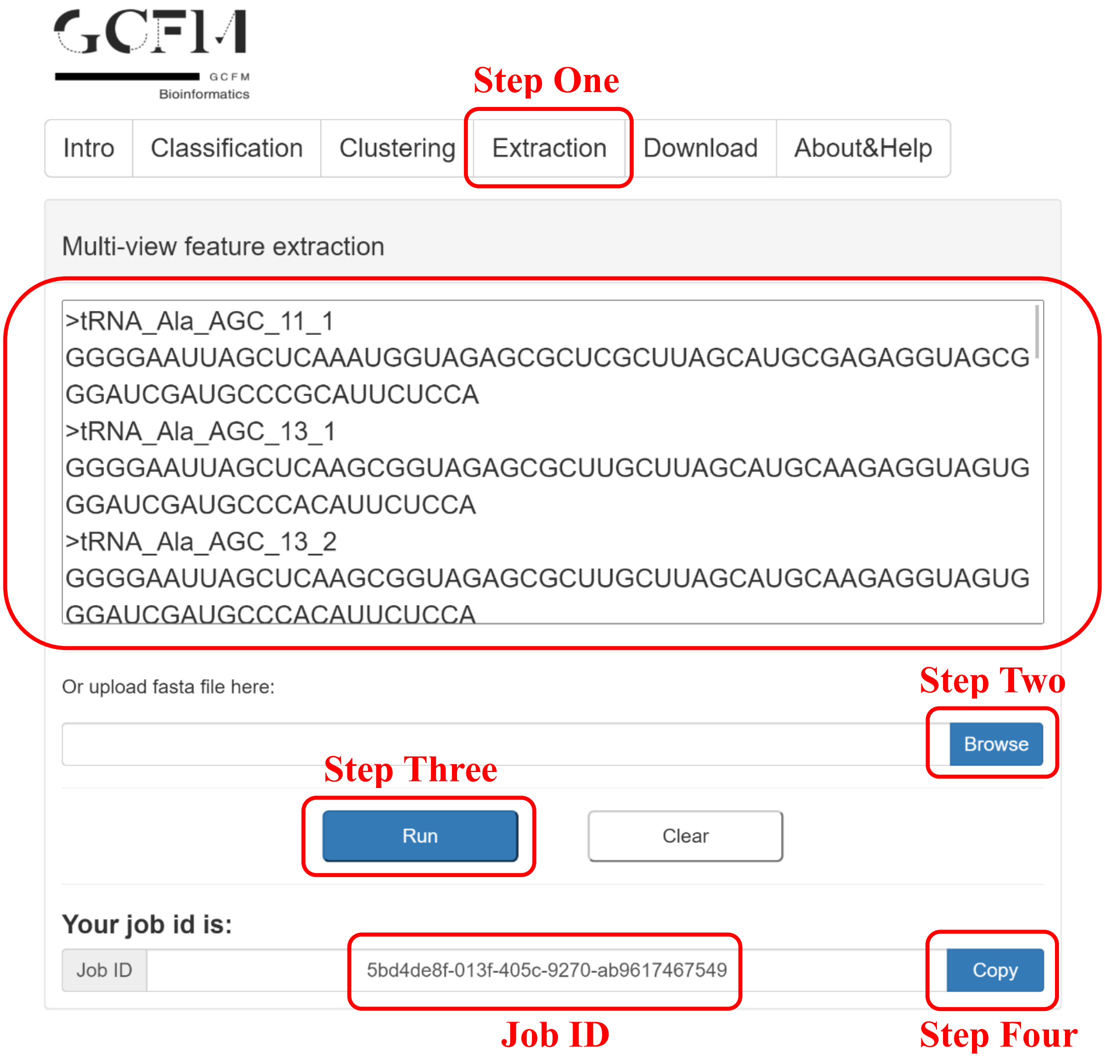
Or upload fasta file here:

Your job id is:

Job ID: 8a97c6a1-930d-4251-98d3-11c55190722b

4. Batch feature extraction function

First, click on the “**Extraction**” button to go to the extraction function page. Next enter multiple sequences in the text box in step two or click the “**Browse**” button to upload a file containing the sequences in fasta format. Then click on the “**Run**” button to create the task and get the Job ID, which is displayed in the “**Job ID**” area. Click “**Copy**” to copy the Job ID, check the status and get the results in the Download page.



Step One

Multi-view feature extraction

```
>tRNA_Ala_AGC_11_1
GGGGAAUUAGCUCAAAUGGUAGAGCGCUCGCUUAGCAUGCGAGAGGUAGCG
GGAUCGAUGCCCGCAUUCUCCA
>tRNA_Ala_AGC_13_1
GGGGAAUUAGCUCAAGCGGUAGAGCGCUUGCUUAGCAUGCAAGAGGUAGUG
GGAUCGAUGCCCACAUUCUCCA
>tRNA_Ala_AGC_13_2
GGGGAAUUAGCUCAAGCGGUAGAGCGCUUGCUUAGCAUGCAAGAGGUAGUG
GGAUCGAUGCCCACAUUCUCCA
```

Or upload fasta file here:

Step Two

Step Three

Your job id is:

Job ID 5bd4de8f-013f-405c-9270-ab9617467549

Job ID

Step Four

5. Download

The download page provides four main functions.

1) Query the status of the job created by the clustering and feature extraction functions.

Operation:

Firstly, click on the “**Download**” button to go to the Download function page. Next, enter the Job ID number in the Job ID box in Step two. Then click on the “**Query**” button to view the status in the “Job Status” area. If the task status is successful, you can download the relevant data by clicking on the blue link.

2) Clustering and feature extraction results can be downloaded when the task is completed.

3) View clustering results online.

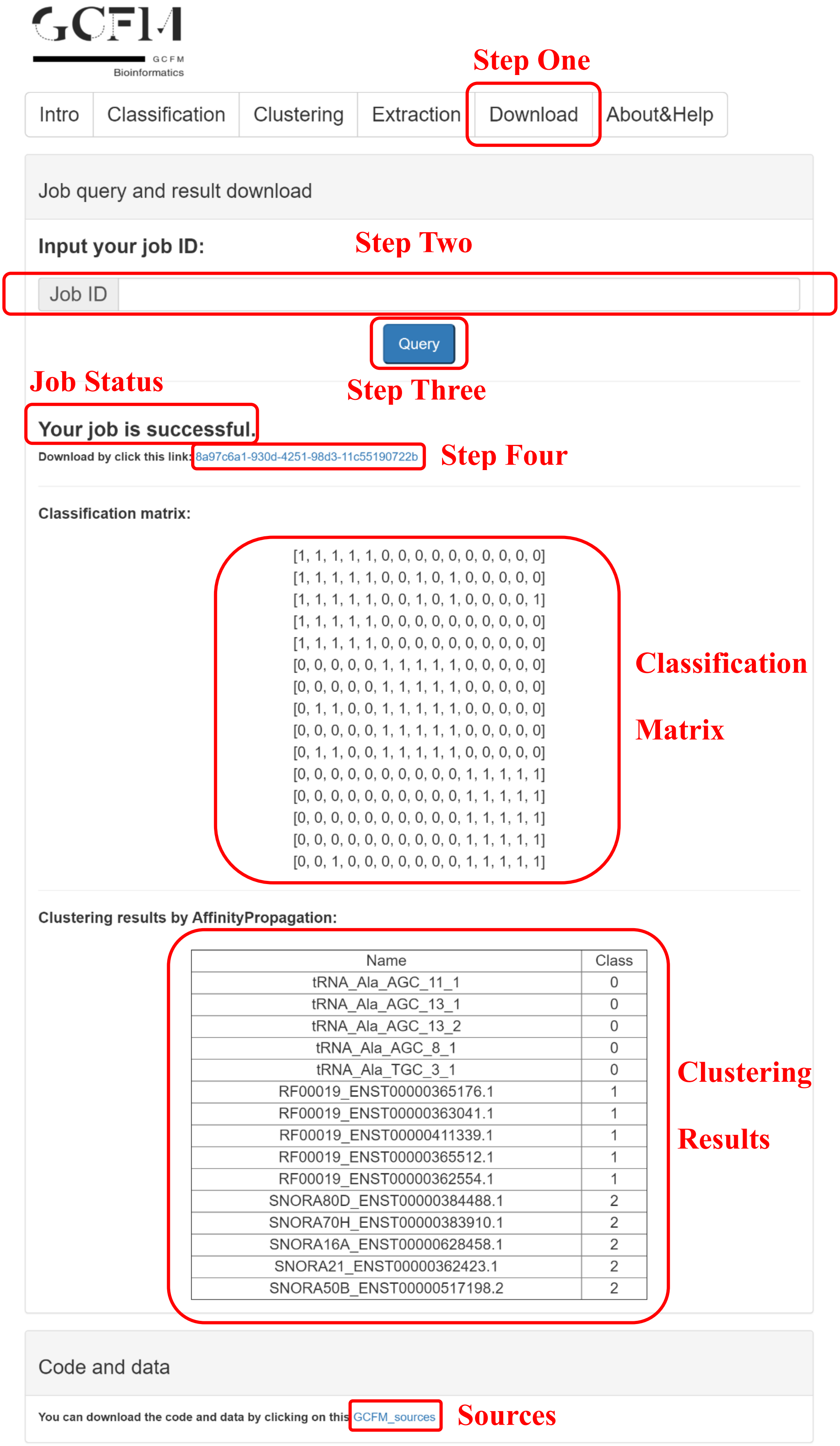
Note:

When the query is for a clustering job, the corresponding content is displayed in the “Classification Matrix” and “Clustering Results” areas.

4) Download source code and related data.

Note:

The download link can be found at the bottom of the Download page.



Job query and result download

Input your job ID:

Your job is successful.

Download by click this link: 8a97c6a1-930d-4251-98d3-11c55190722b

Classification matrix:

```
[1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
[1, 1, 1, 1, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0]
[1, 1, 1, 1, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1]
[1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
[1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0]
[0, 1, 1, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0]
[0, 1, 1, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1]
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1]
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1]
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1]
[0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1]
```

Clustering results by AffinityPropagation:

Name	Class
tRNA_Ala_AGC_11_1	0
tRNA_Ala_AGC_13_1	0
tRNA_Ala_AGC_13_2	0
tRNA_Ala_AGC_8_1	0
tRNA_Ala_TGC_3_1	0
RF00019_ENST00000365176.1	1
RF00019_ENST00000363041.1	1
RF00019_ENST00000411339.1	1
RF00019_ENST00000365512.1	1
RF00019_ENST00000362554.1	1
SNORA80D_ENST00000384488.1	2
SNORA70H_ENST00000383910.1	2
SNORA16A_ENST00000628458.1	2
SNORA21_ENST00000362423.1	2
SNORA50B_ENST00000517198.2	2

Code and data

You can download the code and data by clicking on this GCFM_sources