LeARN/smallA 1.5

Admin Guide

September, 2009

Erika Sallet, Céline Noirot, Christine Gaspin, Thomas Schiex, Jérôme Gouzy

learn@toulouse.inra.fr

Table of contents

1 Download

Download the most recent tarball, then initialize the LEARN_DIR environment variable to the LeARN directory.

```
% wget http://symbiose.toulouse.inra.fr/LeARN/Download/LeARN-1.5.0.tar.gz
% gzip -cd LeARN-1.5.0.tar.gz | tar xvf -

% cd LeARN

% setenv LEARN_DIR /www/LeARN (csh/tcsh)
or
% export LEARN_DIR=/www/LeARN (sh/bash)
```

2 Install required software

LeARN installation was tested in a 64-bit architecture.
LeARN uses external programs which must be either installed or linked to `$LEARN_DIR/bin/ext/bin`. To permit traceability over years of the annotation process, LeARN links all analyses to its program (name+version). But to ensure over years the consistency of this information, the LeARN admin must have a full control of program versions, it means that at any time he/she must be certain of which release of program he/she is running. To do so we suggest to install (and to keep) all releases of programs inside the `$LEARN_DIR/bin/ext` directory.

2.1 Required software list

- xsltproc: http://xmlsoft.org/XSLT/xsltproc2.html
- ImageMagick: http://www.imagemagick.org/
- wget: http://www.gnu.org/software/wget/
- bioperl >=1.4
- perl >=5.6 and modules http://www.cpan.org/
  - XML::Simple
  - LWP
  - Class::XML
  - Class::Accessor
  - Class::Data::Inheritable
  - XML::XPath
  - XML::Twig
  - XML::TreeBuilder

  - Convert::UU
  - HTML::Entities
  - Bio::SeqIO
  - Bio::Seq
  - BerkeleyDB
- clustalw: http://www.clustal.org/#Download
- UNAFold: http://www.bioinfo.rpi.edu/applications/hybrid/download.php
- ncbi-blast ftp://ftp.ncbi.nlm.nih.gov/blast/executables/release
- multalin: http://lipm-bioinfo.toulouse.inra.fr/download/multalin/multalin.5.4.1/
- tRNAScanSE: ftp://selab.janelia.org/pub/software/tRNAscan-SE/tRNAscan-SE.tar.Z
- infernal: ftp://selab.janelia.org/pub/software/infernal/infernal-0.72.tar.gz
- Vienna RNA package: http://www.tbi.univie.ac.at/~ivo/RNA/
- rfam_scan.pl: http://www.sanger.ac.uk/Software/Rfam/help/scripts/search/rfam_scan.pl
- rfam: ftp://ftp.sanger.ac.uk/pub/databases/Rfam/9.1/Rfam.full.gz
- paraloop: http://lipm-bioinfo.toulouse.inra.fr/download/paraloop/
- MiRfold: http://wwwabi.snv.jussieu.fr/research/publi/small_ncRNA/mirfold_0.2b.tgz

2.2 Software required only for smallA

- Perl modules http://www.cpan.org/
  - Config::General
  - Data::Dumper
  - File::Basename
  - File::Spec::Functions
  - FindBin
  - GD::Polyline
  - GD
  - Getopt::Long
  - List::Util
  - Math::Bezier
  - Math::BigFloat
  - Math::Round
  - Math::VecStat
  - Memoize
  - POSIX

  - Params::Validate
  - Pod::Usage
  - Readonly
  - Set::IntSpan
  - Storable
  - Readonly
- miRanda: http://cbio.mskcc.org/research/sander/data/miRNA2003/miranda_new.html
- circos: http://mkweb.bcgsc.ca/circos/distribution/
- patch : http://ftp.gnu.org/gnu/patch/

Optional:

- cross_match.manyreads: http://www.phrap.org/ (required to clean **454 sequencing data**)
- the Washington University nrdb program (fast). If you don't have access to this program, use the Perl version available in the LEARN binaries directory `$LEARN_DIR/bin/ext/bp_nrdb.pl` instead.

2.3 How install the main software

Below we present how we install the main software inside the LeARN directory (tested with linux debian and ubuntu):

* **Infernal** (***Version 0.72 mandatory version***)

```
% cd $LEARN_DIR/bin/ext
% wget ftp://selab.janelia.org/pub/software/infernal/infernal-0.72.tar.gz
% gzip -cd  infernal-0.72.tar.gz | tar xvf -
% cd $LEARN_DIR/bin/ext/infernal-0.72
% more LICENSE
%./configure --prefix $LEARN_DIR/bin/ext
% make
% make install
% cp squid/sreformat $LEARN_DIR/bin/ext/bin
```

* **tRNAscan-SE**

```
% cd $LEARN_DIR/bin/ext
% wget ftp://selab.janelia.org/pub/software/tRNAscan-SE/tRNAscan-SE.tar.Z
% zcat tRNAscan-SE.tar.Z | tar xvf -
% cd $LEARN_DIR/bin/ext/tRNAscan-SE-1.23
% more GNULICENSE
% setenv HOMEBK $HOME ; export HOMEBK=$HOME
% setenv HOME $LEARN_DIR/bin/ext ; export HOME=$LEARN_DIR/bin/ext
% make
% make install
% setenv HOME $HOMEBK ; export HOME=$HOMEBK
```

* UNAFold /MFOLD_UTIL (require gcc and g++)

See requirements at http://www.bioinfo.rpi.edu/applications/hybrid/.
On debian/ubuntu, you can install the required packages with:

```
% sudo apt-get install libgd2-xpm-dev gnuplot libglut3-dev
```

```
% cd $LEARN_DIR/bin/ext
% wget http://www.bioinfo.rpi.edu/applications/hybrid/download/unafold-3.7.tar.gz
% gzip -cd unafold-3.7.tar.gz | tar xvf -
% cd unafold-3.7
% ./configure --prefix $LEARN_DIR/bin/ext
% make
% make install

% cd $LEARN_DIR/bin/ext
% wget  http://www.bioinfo.rpi.edu/applications/hybrid/download/mfold_util-
4.0.tar.gz
% gzip -cd mfold_util-4.0.tar.gz | tar xvf -
% cd mfold_util-4.0
% ./configure --prefix $LEARN_DIR/bin/ext
% make
% make install
```

* Vienna RNA Package

```
% cd $LEARN_DIR/bin/ext
% wget http://www.tbi.univie.ac.at/~ivo/RNA/ViennaRNA-1.8.3.tar.gz
% gzip -cd ViennaRNA-1.8.3.tar.gz | tar xvf -
% cd $LEARN_DIR/bin/ext/ViennaRNA-1.8.3
% more COPYING
% ./configure --prefix $LEARN_DIR/bin/ext --without-kinfold --without-forester --
without-perl
% make
% make install
```

* NCBI-BLAST

```
% cd $LEARN_DIR/bin/ext
% wget ftp://ftp.ncbi.nlm.nih.gov/blast/executables/release/2.2.21/blast-2.2.21-
x64-linux.tar.gz
% gzip -cd blast-2.2.21-x64-linux.tar.gz | tar xvf -
% ln -s blast-2.2.21 ncbi-blast
% cp blast-2.2.21/bin/blastall blast-2.2.21/bin/formatdb $LEARN_DIR/bin/ext/bin
```

* CLUSTALW

```
% cd $LEARN_DIR/bin/ext
% wget ftp://ftp.ebi.ac.uk/pub/software/unix/clustalw/clustalw1.83.linux.tar.gz
% gzip -cd clustalw1.83.linux.tar.gz | tar xvf -
% cd  clustalw1.83.linux
% more README_W
% make -f makefile.linux
% cp clustalw $LEARN_DIR/bin/ext/bin
```

* Multalin

```
% cd $LEARN_DIR/bin/ext
% wget http://lipm-
bioinfo.toulouse.inra.fr/download/multalin/multalin.5.4.1/multalin.LINUX.Z
% mkdir multalin
% cd multalin
% zcat ../multalin.LINUX.Z | tar xvf -
% cp ma $LEARN_DIR/bin/ext/bin
```

* JgoughPhylo.pm

```
% cd $LEARN_DIR/lib
% wget http://supfam.mrc-lmb.cam.ac.uk/treedraw/JgoughPhylo.pm
```

* Paraloop

This is a program that manages the distribution of jobs over computing resources. The default configuration is for multiprocessors machines (SMP).

For SUN grid engine users:

```
% cd $LEARN_DIR/bin/ext/paraloop/etc
% cp templates/paraloop.root.cfg_SGE paraloop.root.cfg
```

For PBS users:

```
% cd $LEARN_DIR/bin/ext/paraloop/etc
% cp templates/paraloop.root.cfg_PBS paraloop.root.cfg
% vi paraloop.root.cfg to set PARALOOP_queue to your local configuration
```

The programs `MapWithBlast.pl` and `miRNA_PrecursorPrediction.pl` use Paraloop. Edit their configuration files (respectively `$LEARN_DIR/cfg/Template_MapWithBlast.cfg` and `$LEARN_DIR/cfg/Template_miRNA_PrecursorPrediction.cfg)` to change the number of CPU (--ncpu) to be used.

* MiRfold

```
% cd $LEARN_DIR/bin/ext
% wget http://wwwabi.snv.jussieu.fr/research/publi/small_ncRNA/mirfold_0.2b.tgz
% gzip -cd mirfold_0.2b.tgz | tar xvf -
% cd mirfold_0.2b
% more licence/Licence_CeCILL_V2-en.txt
% patch < ../patch/patch_mirfold_c
% patch < ../patch/patch_mirfold_h
% perl -pi -e 's,/usr/include/rna,\${LEARN_DIR}/bin/ext/ViennaRNA-1.8.3/H,'
Makefile
% perl -pi -e 's,/usr/lib,\${LEARN_DIR}/bin/ext/ViennaRNA-1.8.3/lib,' Makefile
% make
% cp mirfold $LEARN_DIR/bin/ext/bin
```

* miRanda

```
% cd $LEARN_DIR/bin/ext
% wget
```

```
http://cbio.mskcc.org/research/sander/data/miRNA2003/src1.9/binaries/miRanda-1.9-i686-linux-gnu.tar.gz
% gzip -cd miRanda-1.9-i686-linux-gnu.tar.gz | tar xvf -
% cp miRanda-1.9-i686-linux-gnu/bin/miranda $LEARN_DIR/bin/ext/bin
```

*** Circos**

```
% cd $LEARN_DIR/bin/ext
% wget http://mkweb.bcgsc.ca/circos/distribution/circos-0.51.tgz
% gzip -cd circos-0.51.tgz | tar xvf -
% cd  circos-0.51
% wget http://mkweb.bcgsc.ca/circos/distribution/circos-0.51-1.tgz
% gzip -cd circos-0.51-1.tgz | tar xvf -
% cp tools/binlinks/etc/ideogram.conf tools/binlinks/etc/ticks.conf .
% ./install-unix
% ln -s $LEARN_DIR/bin/ext/circos-0.51/bin/circos $LEARN_DIR/bin/ext/bin/circos
```

3 Install LeARN

3.1 Edit the configuration file

Edit the file `$LEARN_DIR/cfg/Template_Learn.cfg` to search all keys containing the strings "_cmd" and "_version". Then you must modify the values of the command line (_cmd) and the values of the program versions (_version) according to your local configuration. At that time you can decide to use either the binaries installed into your LeARN directory (refered as MASK_LEARN_DIR) or elsewhere on your hard drive (specifying a complete path). In both cases you can decide to specify either a directory including the program version as `MASK_LEARN_DIR/bin/ext/infernal-0.72/bin/cmsearch` (better for the traceability in our opinion) or without the version (`MASK_LEARN_DIR/ bin/ext/bin/cmsearch` or `/usr/local/bioinfo/bin/cmsearch`)

3.2 Configure the instance

3.2.1 Check Perl module installation

Run the following command line to check if all mandatory Perl modules are installed.
If you want to install a standard version, run:

```
% $LEARN_DIR/bin/EvalPerlModule.pl
```

Or to install a smallA version (dedicated to the smallRNA), run:

```
% $LEARN_DIR/bin/EvalSmallaPerlModule.pl
```

The program fails if a module is missing. Install all mandatory modules until it displays « OK ».

3.2.2 Create configuration files

Run the following command line to create the configuration files. At this step, you have to decide to install either a standard version to manage all type of non coding RNA, or a version dedicated to the smallRNA.

```
% $LEARN_DIR/bin/LeARN_Install.pl --learn_dir $LEARN_DIR  \
 --learn_url http://localhost/LeARN
Do you want to install the LeARN interface dedicated to the smallRNA [Y/N]? (If
N, the default interface will be installed)
Writing : /www/LeARN/cfg/LearnCgi.cfg
Writing : /www/LeARN/lib/Var.pm
Writing : /www/LeARN/web/learn.xml [smalla.xml]
Writing : /www/LeARN/cfg/RNAPlotEval.cfg
Writing : /www/LeARN/cfg/RNALocalFold.cfg
Writing : /www/LeARN/cfg/miRNA_PhylogeneticProfiling.cfg
Writing : /www/LeARN/cfg/miRNA_PrecursorPrediction.cfg
Writing : /www/LeARN/cfg/smallRNA_CleanSeq.cfg
Writing : /www/LeARN/cfg/smallRNA_ManageExpression.cfg
Writing : /www/LeARN/cfg/MapWithBlast.cfg
Writing : /www/LeARN/cfg/MapWithGlint.cfg
Writing : /www/LeARN/cfg/miRNA_TargetPrediction.cfg
Writing : /www/LeARN/cfg/Learn.cfg
Writing : /www/LeARN/cfg/ForCgi.cfg
```

This installation program tests that programs specified in the configuration file exist on your system and fails if not. When it fails, you must correct the error and run again the program until it completes without any error.

3.3 Initialize the web server

Add the directives in your web server configuration file. For Apache, add the lines below by replacing /www/LeARN by the content of your $LEARN_DIR.

```
<Directory "/www/LeARN/">
    Order deny,allow
    Deny from all
</Directory>

<Directory "/www/LeARN/cgi-bin/">
    Options +ExecCGI
    Order deny,allow
    Allow from all
</Directory>

<Directory "/www/LeARN/web/">
    Options +FollowSymLinks
    Order deny,allow
    Allow from all
</Directory>

<Directory "/www/LeARN/data/">
    Options +FollowSymLinks
    Order deny,allow
    Allow from all
</Directory>
```

```
ScriptAlias /LeARN/cgi-bin/ /www/LeARN/cgi-bin/
AddHandler cgi-script .cgi
```

Note: the more efficient is to ask your system manager to check/adapt your configuration.

3.4 Add a new software [Optional]

LeARN allows the integration of any detection software providing results in the standard GFF format release 2.

```
AC146854.19     Rfam    domain  84517   84590   .       +       .       Att "ACC=RF00005;ID=tRNA;SCORE=62.95;"
AC146854.19     Rfam    domain  48268   48370   .       +       .       Att "ACC=RF00451;ID=mir-395;SCORE=41.08;"
AC146854.19     trnascan        tRNA    84517   84590   .       +       .       Att "Val;AAC;SCORE=76.95"
```

The other constraint is that your program must have the following usage:

```
% my_wonderful_program « any parameters » --input file --output file.out
```

When your program does not have a valid output format and/or a valid usage the easier way to integrate it into LeARN is to embed the program in a Perl/python script which matches the requirements.

When the requirements are solved, you must register the additional software following the example below:

```
% $LEARN_DIR/bin/LeARN_AddSoftware.pl --name=snoscan \
 --cmd=$LEARN_DIR/bin/get_SnoScan.pl  \
 --param "--cfg $LEARN_DIR/cfg/get_snoscan.cfg" --version=0.01
```

The script will ask for a priority value for the program. This priority is critical to manage redundancy when several detection programs predict the same ncRNA.

Finally, you must run again `$LEARN_DIR/bin/LeARN_Install.pl` in order to take into account the new program.

3.5 Create user accounts [Optional]

```
% $LEARN_DIR/bin/LeARN_AddUser.pl
        --help           : this message
        --login string   : login of the user
        --email string   : email of the user
        --privilege char : number [0|1]
```

Example:

```
% $LEARN_DIR/bin/LeARN_AddUser.pl --login alien --email alien@mars.org
```

Warning: the installation program creates two demo user accounts

```
login: demo1 password: demo1 (with guest privilege)
login: demo2 password: demo2 (with annotator privilege)
```

If you do not want to use these accounts, you must delete the corresponding lines in `$LEARN_DIR/data/auth.priv`

4 Quick Start (standard LeARN version)

Considering that you have successfully installed a standard "LeARN" instance (See sections 2 and 3). Here are the commands that you can run in order to create your own demo web server. The "LeARN" demo server corresponding to the automatic analysis of 4 archea genomes: *Pyrococcus abyssi, Pyrococcus horikoshi, Thermococcus kodakaraensis KOD1, Pyrococcus furiosus.*

1) Fetch the data and copy the fasta files in your $LEARN_DIR directory as defined during the set-up (e.g /www/LeARN/)

```
% cd $LEARN_DIR/data

% wget ftp://ftp.ncbi.nlm.nih.gov/genomes/Bacteria/Pyrococcus_abyssi/NC_001773.fna
% wget ftp://ftp.ncbi.nlm.nih.gov/genomes/Bacteria/Pyrococcus_abyssi/NC_000868.fna
% cat NC_001773.fna NC_000868.fna > Pyrococcus_abyssi.fna

% wget ftp://ftp.ncbi.nlm.nih.gov/genomes/Bacteria/Pyrococcus_horikoshii/NC_000961.fna
% mv NC_000961.fna Pyrococcus_horikoshii.fna

% wget ftp://ftp.ncbi.nlm.nih.gov/genomes/Bacteria/Pyrococcus_furiosus/NC_003413.fna
% mv NC_003413.fna Pyrococcus_furiosus.fna

% wget ftp://ftp.ncbi.nlm.nih.gov/genomes/Bacteria/Thermococcus_kodakaraensis_KOD1/NC_006624.fna
% mv NC_006624.fna Thermococcus_kodakaraensis_KOD1.fna
```

2) Run the pipeline on the 4 genomes (~ 3/4 hours depending your hardware)

```
% cd $LEARN_DIR

% $LEARN_DIR/bin/LeARN.pl --path_new_release $LEARN_DIR/data/relTherm1 --input \
   data/Pyrococcus_horikoshii.fna --species "Pyrococcus horikoshii" --log log/relTherm.1.log

% $LEARN_DIR/bin/LeARN.pl --path_old_release $LEARN_DIR/data/relTherm1 --path_new_release \
  $LEARN_DIR/data/relTherm2 --input data/Pyrococcus_abyssi.fna --species "Pyrococcus abyssi" --log \
   log/relTherm.2.log

% $LEARN_DIR/bin/LeARN.pl --path_old_release $LEARN_DIR/data/relTherm2 --path_new_release  \
   $LEARN_DIR/data/relTherm3 --input data/Pyrococcus_furiosus.fna --species "Pyrococcus furiosus"  \
   --log  log/relTherm.3.log

% $LEARN_DIR/bin/LeARN.pl --path_old_release $LEARN_DIR/data/relTherm3 --path_new_release \
   $LEARN_DIR/data/relTherm4 --input data/Thermococcus_kodakaraensis_KOD1.fna --species \
   "Thermococcus kodakaraensis KOD1" --log log/relTherm.4.log
```

3) Add a new user with editing privilege

```
% $LEARN_DIR/bin/LeARN_AddUser.pl --login therm --email name@domain.org --privilege 1 --passwd
motdepasse
```

4) Select the latest release as the default web release

```
% $LEARN_DIR/bin/LeARN_SetWebRelease.pl --root $LEARN_DIR --release_dir data/relTherm4
```

5) Access the web server (the URL depends on your local installation)

At that stage you should be able to browse and query the database

6) Edit the database (in a private workspace)

 * first click on Home/Connexion to enter the login (therm) and the password

(motdepasse)

 * then go to Home/Status to load a copy of the database in you private workspace

 * after that step, you can select either the public database for browsing or select the private one to edit ncRNA and/or families

5 smallA Quick Start: manage smallRNA projects

Considering that you have successfully installed an instance dedicated to the smallRNA (See sections 2 and 3). Here are the commands that you can run in order to manage smallRNA libraries and to install your own demo web server. The "smallA" demo corresponding to the analysis of 2 smallRNA libraries of *Medicago truncatula (Mt)* sequenced with 454 sequencing technology: first one contains smallRNA expressed in roots of the plant, second in root nodules of the plant.
WARNING: the QuickStart uses a selection of Mt genome and smallRNA data.

First, initialize the SMALLA_DIR environment variable.

```
% setenv SMALLA_DIR $LEARN_DIR/bin/int/smallRNA (csh/tcsh)
or
% export SMALLA_DIR=$LEARN_DIR/bin/int/smallRNA (sh/bash)
```

1) Get the data (two libraries of smallRNA, the chromosomes 0 and 1 of *Mt,* the 454 adaptors, tRNA and rRNA of *Mt*, mRNA of *Mt)*

```
% cd $LEARN_DIR/data
% wget http://symbiose.toulouse.inra.fr/LeARN/Download/raw_nodule_smallrna.fna
% wget http://symbiose.toulouse.inra.fr/LeARN/Download/raw_root_smallrna.fna
% wget http://symbiose.toulouse.inra.fr/LeARN/Download/MtrV2Chr0-1
% wget http://symbiose.toulouse.inra.fr/LeARN/Download/adaptors.fna
% wget http://symbiose.toulouse.inra.fr/LeARN/Download/mt_rna.fna
% wget http://symbiose.toulouse.inra.fr/LeARN/Download/MtrV2mrna
```

2) Cleaning the sequences

Cleaning 454 data requires to install cross_match software.
If you don't want to install cross_match
- Bypass this step: download the cleaned files and go directly to the step 3.

```
% wget http://symbiose.toulouse.inra.fr/LeARN/Download/nodule_smallrna.fna
% wget http://symbiose.toulouse.inra.fr/LeARN/Download/root_smallrna.fna
```

Else
- Install cross_match.manyreads (See section 2.2)
- Edit the configuration file `$LEARN_DIR/cfg/smallRNA_CleanSeq.cfg`:

```
% vi $LEARN_DIR/cfg/smallRNA_CleanSeq.cfg
```

```
# Multifasta file of the adaptator sequences masked by cross_match or Blast
adaptator_lib=/www/LeARN/data/adaptors.fna
...
# Library of the sequences to remove (rRNA, tRNA, ...)
rna_db=/www/LeARN/data/mt_rna.fna
...
# Crossmatch command
cm_cmd=/www/LeARN/bin/ext/bin/cross_match.manyreads
```

- Launch the cleaning of the two libraries:

```
% $SMALLA_DIR/smallRNA_CleanSeq.pl --input $LEARN_DIR/data/raw_root_smallrna.fna \
  --output $LEARN_DIR/data/root_smallrna --cfg $LEARN_DIR/cfg/smallRNA_CleanSeq.cfg
% $SMALLA_DIR/smallRNA_CleanSeq.pl --input $LEARN_DIR/data/raw_nodule_smallrna.fna \
  --output $LEARN_DIR/data/nodule_smallrna --cfg $LEARN_DIR/cfg/smallRNA_CleanSeq.cfg
```

3) Compute the expression of smallRNA

Edit the configuration file `$LEARN_DIR/cfg/smallRNA_ManageExpression.cfg` to fill the path of the smallRNA files and the corresponding library names:

```
% vi $LEARN_DIR/cfg/smallRNA_ManageExpression.cfg
```

```
# List of multifasta files. One file contains a set of miRNA produced in the same condition.
mir_files=/www/LeARN/data/root_smallrna.fna;/www/LeARN/data/nodule_smallrna.fna
# String containing a list of conditions. miRNA of the first file
# of the string mir_files was produced in the first condition.
conditions=R;N
```

Delete redundancy (If you can, use WU nrdb program. Otherwise use the Perl version `$LEARN_DIR/bin/ext/bp_nrdb.pl` as in the example), then compute the expression information.

```
% cd $LEARN_DIR/data
% $LEARN_DIR/bin/ext/bp_nrdb.pl -o smallrna.nr -d # root_smallrna.fna nodule_smallrna.fna
% $SMALLA_DIR/smallRNA_ManageExpression.pl --nrdb_file $LEARN_DIR/data/smallrna.nr \
  --output_fasta_file $LEARN_DIR/data/smallrna.fna --cfg $LEARN_DIR/cfg/smallRNA_ManageExpression.cfg
% more $LEARN_DIR/data/smallrna.fna
```

4) Run the complete pipeline (~ 3/4 hours depending your hardware):

```
% $LEARN_DIR/bin/smallA_BuildCmd.pl --input_fasta $LEARN_DIR/data/MtrV2Chr0-1 \
  --smallrna_db $LEARN_DIR/data/smallrna.fna --path_new_release $LEARN_DIR/data/RELEASE_N_R \
  --species 'Medicago truncatula'
% more $LEARN_DIR/data/RELEASE_N_R/smalla.cmd
% sh $LEARN_DIR/data/RELEASE_N_R/smalla.cmd
```

5) Search targets of the miRNA of precursors

```
% cd $LEARN_DIR/data
% $SMALLA_DIR/miRNA_TargetPrediction.pl --path_release $LEARN_DIR/data/RELEASE_N_R  \
   --lib $LEARN_DIR/data/MtrV2mrna --class all
```

6) Select the latest release as the default web release

```
% $LEARN_DIR/bin/LeARN_SetWebRelease.pl --root $LEARN_DIR --release_dir data/RELEASE_N_R
```

7) Access the web server (the URL depends on your local installation)

At that stage you should be able to navigate in the web site and to visualize precursor classification, phylogenetic profiles, targets of miRNA and genomic view.

6 Create and update LeARN release

6.1 *Create the first release*

You have the possibility to only run the analyses and store the result in a repository, or launch the complete analyse and build the LeARN database.

6.1.1 [Optional] Pre-computation

On large datasets it is required to parallelize the computation of detection software. This program builds a file which contains the list of “atomic” command lines. These commands can be easily run on a cluster.

```
Usage:./bin/LeARN_BuildScanCli.pl
  --help                          this message
  --cfg  filename                 full path of the configuration file
  --path_old_release dirname      path of old release
  --path_new_release dirname      path of new release (to create)
  --input_fasta filename          multifasta input file of one species
  --species quoted_string         name of the organism, eg: 'Casimir vulgaris'
  --log filename                  log file
```

6.1.2 Run the clustering algorithm

This can reuse the precomputed analyses if you define the same repository as for the pre-computation step.

```
$LEARN_DIR/bin/LeARN.pl
[Mandatory]
  --input_fasta filename  filename      multifasta input file of one species
[Optional]
  --help                                this message
  --cfg                   filename      [LEARN_DIR/cfg/Learn.cfg] full path of
                                        the configuration file
  --path_old_release      dirname       path of old release [default set in cfg]
  --path_new_release      dirname       path of new release (to be create)
                                        [default set in cfg]
  --smallrna_db           filename      smallrna multifasta file (Required to
                                        install a smallA instance)
  --overlaps              filename      file contain overlap data [seq1 start1
                                        end1 strand1 seq2 start2 end2 strand2]
  --species               quoted_string name of the organism ,
                                        eg: 'Medicago truncatula'
  --upgraded_sequences    filename      if some version of BAC are upgraded
                                        specify the file wich contain list of
                                        [old_bac new_bac]
  --log                   filename      [LEARN_DIR/log/learn.log] log file
  --no_family_building                  If specified, dont search rna families
  --no_post_process_merging             If specified, dont execute the post
                                        processing merge of families
  --no_rna_redundancy_removing          If specified, don't remove the redundant
                                        or overlapping rna
```

Example:

```
% export RFAM_DIR=$LEARN_DIR/data/release_template/db/Rfam/ # or any other Rfam
directory
```

```
% $LEARN_DIR/bin/LeARN.pl \
                       --path_new_release $LEARN_DIR/data/v1 \
                       --species "Casimir vulgaris" \
                       --input_fasta Cv.multifasta
```

6.2 Update a release

For updating the database, either with a dataset of the same species or using sequences of another species you must run the program by providing both a directory for the new release and the directory of the previous release.

```
% export RFAM_DIR=$LEARN_DIR/data/release_template/db/Rfam/ # or any other Rfam
directory
% $LEARN_DIR/bin/LeARN.pl  \
                      --path_old_release $LEARN_DIR/data/v1  \
                      --path_new_release $LEARN_DIR/data/v2  \
                      --species "Casimir singularis" \
                      --input_fasta Cs.multifasta
```

6.3 Select/Set the default web release

```
$LEARN_DIR/bin/LeARN_SetWebRelease.pl
[Mandatory]
        --root            directory   Path of learn directory
        --release_dir     directory   Relative to learn root directory. e.g.
                                      data/rel_template
        --new                         Do not erase current cfg . Initialize a web
                                      site for testing a new release
[Optional]
          --help                      this message
```

Example:

```
% $LEARN_DIR/bin/LeARN_SetWebRelease.pl --root $LEARN_DIR --release_dir data/v1
```

The parameter --new builds a temporary web site release but doesn't modify the default web release. This feature is useful for testing a new release without modifying the default.

6.4 Export the data of a RNAML directory to GFF3

```
$LEARN_DIR/bin/LeARN_Rnaml2Gff3.pl
[Mandatory]
          --rnaml_dir   dirname   : directory of the RNAML files
          --output_dir  dirname   : output directory
[Optional]
          --help                  : This message
```

Example:

```
% $LEARN_DIR/bin/LeARN_Rnaml2Gff3.pl \
          --rnaml_dir $LEARN_DIR/data/relTherm1/db/learn/rna/xml \
          --output_dir $LEARN_DIR/data/example/gffoutput
```

7 smallA: managing libraries of smallRNA

We developed a set of programs dedicated to analyse data generated by projects aiming at studying the expression of smallRNA on the basis of deep sequencing. Most of these programs are available in the directory $SMALLA_DIR ($LEARN_DIR/bin/int/smallRNA).

7.1 smallA_BuildCmd.pl: generate smallA computing command lines

Create a command line file to compute all the smallA process: map the sequence, predict and classify precursor and generate a smallA instance. Note that some steps can be parallelized (step 2, 3, 4):
Steps:
1 - generate the smallA arborescence
2 - map the smallrna to the genomic sequences (parallelizable)
3 - search mirna precursor (parallelizable)
4 - annotate the precursor (parallelizable)
5 - integrate data in smallA repository
6 - classify the precursor
7 - compute phylogenetic profile according to miRBase
8 - generate a circos circular visualization.

```
$LEARN_DIR/bin/smallA_BuildCmd.pl
[Mandatory]
  --input_fasta      filename      Multifasta input file of one species
                                   (ABSOLUTE PATH)
  --smallrna_db      filename      Multifasta file of the mature sequences and
                                  their Expression in the header. (ABSOLUTE PATH)
[Optional]
  --help                           This message
  --path_new_release filename      Path of new release (to be create)
                                   [default set in cfg]
  --species          quoted_string Name of the organism , eg: 'Medicago tr'
  --cfg              filename      [LEARN_DIR/cfg/smalla.cfg] Full path of the
                                   configuration file
```

Example:

```
$LEARN_DIR/bin/smallA_BuildCmd.pl --input_fasta $LEARN_DIR/data/Chr1 \
  --smallrna_db $LEARN_DIR/data/smallrna.fna \
  --path_new_release $LEARN_DIR/data/RELEASE_N_R --species 'Medicago truncatula'
```

7.2 smallRNA_CleanSeq.pl: clean small RNA reads

Clean smallRNA libraries, specially generated by 454 or solexa sequencing. Remove adaptor sequences (using *cross_match* for 454 data or *Blast* for solexa data*)*, then remove the sequences which are not smallRNA using *Blast* (for instance rRNA or tRNA degradation products) and filter the sequences according to their length. Save the cleaned smallRNA in a multifasta file.

```
$SMALLA_DIR/smallRNA_CleanSeq.pl
[Mandatory]
         --input     filename   Multifasta file containing the reads to clean.
         --output    filename   Root of the names of the output files.
         --method    crossmatch (adapt for 454)|blast (adapt for solexa)
                    [crossmatch] select the method to remove the adaptors
          --cfg       filename   Configuration file path.
[Optional]
         --help                 This message
         --no_clean             Don't remove the temporary files
```

Example:

```
% $SMALLA_DIR/smallRNA_CleanSeq.pl \
         --input  $LEARN_DIR/data/example/raw_lib1.fasta \
         --output $LEARN_DIR/data/example/lib1.fasta \
         --cfg    $LEARN_DIR/cfg/smallRNA_CleanSeq.cfg
```

Prerequisite: to clean 454 sequences, it's required to download and install *cross_match.manyreads* (http://www.phrap.org/) to mask the adaptors.

Before calling the program, modify the configuration file `$LEARN_DIR/cfg/smallRNA_CleanSeq.cfg`. Fill in:

- 'adaptator_lib' the library of adaptors,
- 'rna_db' the library of RNA (tRNA, rRNA, ...) to remove,
- [Optional] 'cm_cmd', the path of the cross_match.manyread program to clean 454 sequences.

7.3 smallRNA_ManageExpression.pl: compute the expression of smallRNA sequences in libraries

The program studies a set of unredundant smallRNA sequences included in a output nrdb (non redundant database) file. For each smallRNA, it computes the number of time the sequence is expressed in different libraries. The program generated a multifasta file whose headers contains the expression of the sequence. The fasta header `'>ID1 LIB1(10) LIB2(5)'` means that the smallRNA ID1 is expressed 10 times in the library LIB1 and 5 times in the library LIB2.

```
$SMALLA_DIR/smallRNA_ManageExpression.pl
[Mandatory]
     --nrdb_file             filename  Path of the nrdb file
     --output_fasta_file     filename  Path of the output fasta file
     --cfg                   filename  Path of the cfg file

[Optional]
     --output_detailed_file  filename  Path of the output detailed file
     --help                            this message
```

If 'output_detailed_file' parameter is specified, the program searches for each sequence the most similar sequence in a miRNA database (Default, miRBase database) using a mapping program and generates a more informative tabulated file.

Before calling the program, modify the configuration file `$LEARN_DIR/cfg/smallRNA_ManageExpression.cfg` to fill in:

- 'mir_files', list of multifasta files. One file represents a smallRNA library and contains a set of smallRNA produced in the same condition. The nrdb_file has to be produced from these files. File paths are separated by ';'.
- 'conditions', list of expression conditions of the libraries, separated by ';'. First library of 'mir_files' is expressed in the first condition of this list, etc.
- *if you want a detailed output,* 'map_prog_cmd', the mapping program and 'mir_database' the miRNA database .

Prerequisite:

- Delete redundancy of the smallRNA libraries using a nrdb program **with the option -d #**. (See section 2.2 for nrdb installation)

Example:

Part of the cfg file:

```
# List of multifasta files. One file contains a set of miRNA produced in the same
condition.
mir_files=/www/LeARN/data/lib1.fasta;/www/LeARN/data/lib2.fasta;
# String containing a list of conditions. miRNA of the first file
# of the string mir_files was produced in the first condition.
conditions=LIB1;LIB2
```

```
% cd $LEARN_DIR/data
% $LEARN_DIR/bin/ext/bp_nrdb.pl -o smallrna.nr -d # lib1.fasta lib2.fasta
% $SMALLA_DIR/smallRNA_ManageExpression.pl \
         --nrdb_file             $LEARN_DIR/data/smallrna.nr \
         --output_fasta_file     $LEARN_DIR/data/smallrna.fna \
         --cfg                   $LEARN_DIR/cfg/smallRNA_ManageExpression.cfg
```

7.4 smallRNA_GenomicMapping.pl: map smallRNA to the genomic sequence

Map the smallRNA to the genomic sequence 'input', keep the hits which match with a mismatch number lower than 'max_mismatch_nb' and which match less than 'max_loci_nb_per_chr' times. Save the m8 result in the LeARN repository.

```
$SMALLA_DIR/smallRNA_GenomicMapping.pl
Mapping smallrna against genomic sequence, keep the perfect matchs and save it in
the learn repository.

[Mandatory]
    --input                 filename  Fasta sequence
    --smallrna_db           filename  Multifasta file of smallrna
[Optional]
    --max_mismatch_nb       integer   Maximum number of mismatch (0 by default)
    --max_loci_nb_per_chr   integer   Maximum number of hits per chromosome (10 by
default)
    --cfg                   filename  Configuration file path
                                     (LEARN_DIR/cfg/Learn.cfg by default)
```

Example:

```
% $SMALLA_DIR/smallRNA_GenomicMapping.pl \
  --input       $LEARN_DIR/data/Chr \
  --smallrna_db $LEARN_DIR/data/smallrna.fna
```

7.5 miRNA_PrecursorPrediction.pl: predict miRNA precursors knowing mature sequence

Search miRNA precursor candidates knowing mature miRNA, using MiRfold (http://wwwabi.snv.jussieu.fr/research/publi/small_ncRNA/) and applying some structure duplex filters. (filtering about structure energy, precursor length, miR:miR* duplex structure)

```
$SMALLA_DIR/miRNA_PrecursorPrediction.pl
[Mandatory]
   --input         filename   Sequence file (Absolute path)

[Optional]
   --output        filename   [in LeARN repository] Output file  (Absolute path)
   --mirfold_file  filename   Precomputed raw MiRfold output file (Absolute path)
   --mapping_file  filename   Precomputed mapping output file in m8 format
                              (Absolute path)
   --db_column     integer    Number of the column containing the miRNA in
                              mapping_file
   --no_clean                 Use it to don't clean temporary files
   --help                     This message
   --cfg           filename   [LEARN_DIR/cfg/miRNA_PrecursorPrediction.cfg] Full
                              path of the configuration file
```

Before calling the program, modify the configuration file `$LEARN_DIR/cfg/miRNA_PrecursorPrediction.cfg` to fill in 'smallrna_db', the path of the miRNA database (By default, miRBase http://microrna.sanger.ac.uk/)

Required:
if the mapping_file is not specified, the LeARN repository has to contain the result file of the mapping.
Optional:
To thread MiRfold, edit `$LEARN_DIR/cfg/miRNA_PrecursorPrediction.cfg` to fill in 'paraloop_cmd': change the number of CPU to use (`--ncpu`). Initialize the PARALOOP environment variable:

```
% setenv PARALOOP $LEARN_DIR/bin/ext/paraloop (csh/tcsh)
or
% export PARALOOP=$LEARN_DIR/bin/ext/paraloop (sh/bash)
```

Example:
Part of the cfg file:

```
# Paths of the miRNA databases, separated by a ";"
smallrna_db=/www/LeARN/data/smallrna.fna
```

```
% $SMALLA_DIR/miRNA_PrecursorPrediction.pl \
         --input $LEARN_DIR/data/genome.fa \
         --output $LEARN_DIR/data/precursors.gff
```

7.6 miRNA_PrecursorAnnotation.pl: annotate miRNA precursors

Select a subset of precursors whose matures have a specific length, and annotate these precursors with smallRNA sequences which map to the precursor sequence. Save the annotated precursors in a GFF file (in the LeARN repository or in the output file if the parameter 'output' is specified).

```
$SMALLA_DIR/miRNA_PrecursorAnnotation.pl
[Mandatory]
  --mirfold_outfile filename  Path of a GFF file, the output of
                              miRNA_PrecursorPrediction.pl: precursors of miRNA
                              to select and annotate
  --smallrna_db     filename  Multifasta file of the miRNA and their expression
                              in the header.
  --input           filename  Sequence file [Absolute path]. REQUIRED ONLY TO
                              INTEGRATE RESULTS IN LEARN REPOSITORY.
[Optional]
  --cfg             filename  [LEARN_DIR/cfg/miRNA_PrecursorAnnotation.cfg] Full
                              path of the configuration file
  --output          filename  To force the output file. (default is in the LeARN
                              Repository)
  --mapping_file    filename  To force the use of this mapping m8 output (can be
                              gz). Mapping of smallRNA on the genomic sequence
                              with no mismatch
  --db_column        integer  In m8 file, number of the column of the smallRNA
                              [1 or 2]
  --help                      This message
  --verbose                   Print to the standard output some extra information
                              (the ID of the matures, ...)
```

Example:

```
% $SMALLA_DIR/miRNA_PrecursorAnnotation.pl \
    --smallrna_db $LEARN_DIR/data/smallrna.fna \
    --mirfold_outfile $LEARN_DIR/data/precursors.gff \
    --input $LEARN_DIR/data/Chr.fna
```

7.7 Create a smallA release

Create a smallA release which integrate the predicted and annotated miRNA precursors.

```
$LEARN_DIR/bin/smallA.pl
[Mandatory]
  --input_fasta         filename        multifasta input file of one species
  --smallrna_db         filename        smallrna multifasta file
[Optional]
  --help                                this message
  --cfg                 filename        [LEARN_DIR/cfg/Learn.cfg] full path
                                             of the configuration file
  --path_old_release    dirname         path of old release [default set in
                                             cfg]
  --path_new_release    dirname         path of new release (to be create)
                                             [default set in cfg]
  --overlaps            filename        file contain overlap data
                                             [seq1 start1 end1 strand1 seq2
                                             start2 end2 strand2]
  --species             quoted_string   name of the organism , eg: 'Medicago tr'
  --upgraded_sequences  filename       If some version of BAC are upgraded
                                        specify the file which contain list of
                                        [old_bac new_bac]
  --log                 filename        [LEARN_DIR/log/learn.log] log file
  --no_rna_redundancy_removing          If specified, don't remove the redundant
                                        or overlapping rna
```

Example:

```
$LEARN_DIR/bin/smallA.pl --input $LEARN_DIR/data/Chr.fna --species 'my species' \
 --path_new_release $LEARN_DIR/data/MY_RELEASE \
 --smallrna_db $LEARN_DIR/data/smallrna.fna  --no_rna_redundancy_removing
```

7.8 miRNA_PrecursorClassification.pl: classify miRNA precursors

Classify the precursors according to the miR/miR* duplex. For each class, generate an excel file (.xls), a fasta file of the precursors (.fna) formatted for miRBase submission and a fasta file of the seed sequences (.fna.mir).

```
$SMALLA_DIR/miRNA_PrecursorClassification.pl
There are two ways to classify:
1) Specify a learn release path to automatically classify the precursors of the
release and save results in the release directory.
[Mandatory]
   --path_release   dirname   Directory path of the release

2) Specify input and output files:
[Mandatory]
    --rnaml_dir    dirname   Directory of the RNAML files
    --smallrna_db  filename  Multifasta file of the mature sequences and their
                             expression in the header.
    --outprefix    filename  Prefix of the output files


[Optional]
  --cfg            filename  [LEARN_DIR/cfg/miRNA_PrecursorClassification.cfg]
                             Configuration file
  --help                     This message
  --verbose                  Verbose mode
```

Example:

```
% $SMALLA_DIR/miRNA_PrecursorClassification.pl   \
         --path_release $LEARN_DIR/data/MY_RELEASE
```

7.9 miRNA_TargetPrediction.pl: predict miRNA targets

Call miRanda program to predict targets of miRNA. Filter the results according to the length of the alignment and to the pairs between the miRNA and its target (Jones-Rhoades and Bartel).

```
$SMALLA_DIR/miRNA_TargetPrediction.pl
There are two ways to use miRNA_TargetPrediction.pl:
1) Specify a learn release path to automatically save the targets in the release
directory.
[Mandatory]
    --path_release   dirname   Directory path of the release
    --lib            filename  Multifasta file of the sequences in which search
                               targets
[Optional]
    --class          string    [mir:mirstar]
                               List of precursor classes (separated by ;).
                               Search the targets of the mature miRNAs of the
                               precursors of these classes. Accepted values:
                               all (all the classes), mir:mirstar,
                               mir:mirstar:inv, mir:mirstar:other,
                               mirmult, mir:other, mirsngl, misc:expr, undef

2) Specify input and output files:
[Mandatory]
    --input           filename  Multifasta file of miRNA
    --lib             filename  Multifasta file of the sequences in which search
                                targets
    --raw_output      filename  Path of the raw output file
    --tab_output      filename  Path of the tabulated output file


[Optional for 1 and 2]
    --miranda_file   filename  miRanda result file (.gz or .lzma allowed): in
                               this case, miRanda is not run (the 'input' and
                               'lib' parameter are not required)
    --cfg            filename  [LEARN_DIR/cfg/miRNA_TargetPrediction.cfg] Path of
                               the cfg file
    --help                    this message
```

If the 'path_release' is defined, read the smallRNA sequences in the release and save the result files in the release directory.
If the 'class' parameter is defined, search targets only for the mature miRNA of the precursors which belong to these classes.

<u>Prerequisite</u>: *miRanda* (http://www.microrna.org/microrna/getDownloads.do) has to be installed. (See 2.3 section)

Example to search targets of the miRNA of the precursors of the classes mir:mirstar and mirmult:

```
% $SMALLA_DIR/miRNA_TargetPrediction.pl \
   --path_release $LEARN_DIR/data/MY_RELEASE  \
   --lib          $LEARN_DIR/data/mrna.fna    \
   --class        "mir:mirstar;mirmult"
```

7.10 miRNA_PhylogeneticProfiling.pl: generate phylogenetic profiles

Compare the miRNA sequences of a RNAML directory to a miRNA database (default, miRBase). For each miRNA, compute the number of similar sequences in the database, sorted by species. Generate the phylogenetic profiles and the similarities between the candidate miRNA and the miRNA of the database.

```
$SMALLA_DIR/miRNA_PhylogeneticProfiling.pl
[Mandatory]
    --input_dir    dirname     Directory of the RNAML files of the miRNA
    --output       filename    Path of the output phylogenetic profile file
OR
    --path_release dirname     Directory of a learn release

[Optional]
    --cfg          filename    [LEARN_DIR/cfg/miRNA_PhylogeneticProfiling.cfg]
                               Path of the configuration file
    --no_clean                 If specified, don't delete temporary files
    --help                     This message
```

If 'path_release' is defined, parse the RNAML directory of the release and save the output files in the release directory.
Before calling the program, modify `$LEARN_DIR/cfg/miRNA_PhylogeneticProfiling.cfg` to fill in:

- 'mir_db', the path of the miRNA database (default, miRBase)
- 'sorted_species', the sorted list of species codes. The LeARN miRNA will be compared only with the miRNA of these species.
  The species code of a miRNA of the database is the prefix of its name before the character '-'. For instance, the species code of ath-miR857 is ath (for *A. Thaliana*).

<u>Prerequisite</u>:

- 'mir_db' file has to be indexed with formatdb. (`$LEARN_DIR/bin/ext/bin/formatdb -i db.fa -p F`)
- The RNAML files describing a miRNA have to have a 'sequence' element whose 'analysis-ids' attribute contains the string 'mir'. (For instance 'mirfold-1.0') Required to distinguish the miRNA from the other non coding RNA.

Example:

```
% $SMALLA_DIR/miRNA_PhylogeneticProfiling.pl \
         --path_release $LEARN_DIR/data/MY_RELEASE
```

7.11 smallRNA_Circos.pl: visualize smallRNA on the genomic sequence

Generate a circular visualization of the mapping of the smallRNA to the genome using Circos program (mkweb.bcgsc.ca/circos).

```
$SMALLA_DIR/smallRNA_Circos.pl
There are two ways to generate circos file:
1) Specify a learn release path to automatically load data from the release and
generate circos image in the release directory
[Mandatory]
   --path_release dirname    Directory of a learn release


2) Specify input and output files:
[Mandatory]
   --sequences    string     List of files separated by ';': fasta files of the
                             genomic sequences
   --m8_files     string     List of files separated by ';': m8 files of the
                             genomic sequences
   --output       filename   Circos output image [Absolute path]

[Optional for 1 and 2]
   --cfg          filename   [LEARN_DIR/cfg/smallRNA_Circos.cfg] Full path of
                             the configuration file
   --no_clean                Use this option to keep the directory containing
                             temporary files
```

If 'path_release' is specified, generate an image which integrates all the genomic sequences loaded in the release. The image is saved in the release directory.

Example:

```
% $SMALLA_DIR/smallRNA_Circos.pl \
  --path_release $LEARN_DIR/data/MY_RELEASE
```

7.12 LeARN_ManageExpression.pl: compute expression information on smallRNA libraries

Compute the expression of the libraries (total number of expressed sequences, number of expressed sequences with length=20, number of non redundant expressed sequences ...) Results are saved in the XML file 'lib_out'.

```
$LEARN_DIR/bin/LeARN_ManageExpression.pl
[Mandatory]
    --lib           filename     XML file describing a tree of libraries.
    --expressed_seq filename     Multifasta File of miRNA whose headers get their
                                 expression. Example: output_fasta_file of
                                 smallRNA_ManageExpression.pl
    --lib_out       filename     Path of the output XML file describing the tree
                                 of libraries and their expression.
    --cfg           filename     Configuration file path.
[Optional]
    --rna_dir       path         Directory containing RNAML files.
    --rna_dir_out   path         Path of the output directory of XML files.
                                 A XML file contains the expression of a miRNA.
    --help                       This message
```

If the parameters 'rna_dir' and 'rna_dir_out' are defined, extract the expression of the miRNAs included in the LeARN RNAML repository and save the results in XML files in the rna_dir_out directory.

Example:
Example of 'lib' file, containing the tree of libraries:

```
<?xml version="1.0"?>
<collection>
         <class key="1">
                  <definition>Class of two libraries libraries</definition>
                  <library key="1" id="Lib1" tag=" LIB1(\d+)">
                           <definition>Library 1</definition>
                  </library>
                  <library key="2" id="Lib2" tag=" LIB2(\d+)">
                           <definition>Library 2</definition>
                  </library>
         </class>
         <library key="2" id="OTHER" tag=" OTH(\d+)">
                  <definition>An other library</definition>
         </library>
</collection>
```

```
% $LEARN_DIR/bin/LeARN_ManageExpression.pl \
         --lib             $LEARN_DIR/data/example/lib.xml \
         --expressed_seq   $LEARN_DIR/data/example/lib1_2.fasta.nr.expr \
         --lib_out         $LEARN_DIR/data/example/lib_out.xml \
         --cfg             $LEARN_DIR/cfg/Learn_ManageExpression.cfg
```

7.13 miRNA_MappingSynthesis.pl: generate an overview of the genomic mapping

Extract from mapping result files (m8 files) the number of time the smallRNA are mapped to the different genomic sequences. Write results to the standard output.

```
$SMALLA_DIR/miRNA_MappingSynthesis.pl
[Mandatory]
    --genome_mapping_files  filename list  List of m8 files containing the
                            mapping of smallrna against genomic sequences
                            (separator is ';')
    --mapping_genomes       string         List of genome code
    --smallrna_db          filename        Multifasta file of the miRNA and their
                                           expression in the header.
[Optional]
    --mirna_mapping_file   filename        M8 file: mapping of the mirna against
                                           a mirna database (ex: miRBase)
    --help                       This message
```

'mirna_mapping_file' is an optional parameter: a m8 file containing the mapping of the smallRNA against a smallRNA database (miRBase for instance). If specify, compute the similarities between the smallRNA and those of the database.

Prerequisite:
In the configuration file `$LEARN_DIR/cfg/miRNA_MappingSynthesis.cfg`, following parameters has to be filled:
 - 'genome_mapping_files', the list of the genomic mapping files,
 - 'mapping_genomes', the list of the genome names.