bcall-68

Jérôme Boulanger

November 22, 2011

1 The big picture

Bcall tries to perform a base calling by estimating the parameters of a mixture of Gaussian. The element of the mixture are related to the classes 00,01,10 and 11. The mixture parameters are estimated using some additional constraints. The maximum a posteriori is found using the well-know Expectation-Maximisation algorithm[1, 2]. Beforehand, a normalization of the data is performed. The normalization can be done image-wise or/and globally. Moreover, several normalization techniques can be used. These one corresponds mainly to several methods for the estimation of the center and the covariance of the data. The package includes 3 different program with one having a graphical user interface.

2 Using the 3 programs

2.1 bcall

`bcall` performs the analysis of several datasets organized in subfolders. For each dataset a 2 by 2 analysis is done and then a global merge of the result is obtainted providing the final results. Each subfolder should have a pair of log file with a consistent number of objects and images.
In the command line window type (matlab or octave):
`>> [experiment, options]=bcall('Directory',options)`
`bcall` is a command line function for matlab that take two parameters. The first one is the path to the `'Directory'` containing SNP or CG sub-folders. The second one is an `options` structure. Type `help bcall` in the commande window to get more help. The function returns the experiment data and the options used.

2.2 gbcall

This is the graphical interface of `bcall`.
In the command line window type (matlab only):
`>> gbcall`
gbcall allows to graphically set the options of bcall and then call bcall.

Table 1: Fields of the options structure for `bcall`. A boolean is true/false or 1/0

name	values	description
`options.debug`	boolean	enable debug
`options.cg`	string	pattern to match subdir names
`options.normalization_type`	$1, 2, 3, 4$	type of normalization. 1:none, 2:std, 3:robust std+median, 4:robust covariance+mean shift
`options.beta`	float [0,1]	strenght of the constraints (the lower the stronger)
`options.max_cluster_size`	float	target size for the clusters
`options.imagewise_normalization`	boolean	if `true` the normalization include a normalization image by image. This options is incompatible with pretrain_bcall and dysswitch_bcall
`options.remove_bad_image`	boolean	if `true` remove images which have too different nomalization parameters than the others
`options.satellites`	boolean	tell if `true` : use additionnal clusters
`options.image`	list of integer	keep the specified images in the list. if equal -1 then all images are used.
`options.pfiltercat`	float	minimal percentage of data dumped (not working for dyeswitch/pretrain_bcall).
`options.iterations`	integer	number of iterations.

- **Data path:** Is used to locate the root of the folder where the data are stored (eg: Z:analysis).
- **Directory:** Indicate the folder containing the subfolders (SNPs or CGs).
- **Pattern:** The pattern allow to select the sub-folders in the folder "directory". A star means "anything".
- **Nucleotid:** Give a sequence of letter to indicate the nucleotids corresponding to the data. The sub-folder order is given by the operating system and the order of letters should correspond this order. For each sub-folder corresponds two nucleotids since each sub-folder contains two channels.
- **Probability:** Set a threshod on the probability to belong to a class.
- **Iterations:** The EM algoritm is iterative. This set the number of iteration.
- **Size cluster:** Is a constraint on the size of the clusters, when set to 0, no constraint is applied. The effect of this parameter has been changed a lot since the previous version. Default value is now 0 (no constraints).
- **Filter output:** Threshold to remove classes with less points than a certain percent of the all data set.
- **Image:** Select a specific image to perform the analysis.
- **Satellites:** Enable the use of additional cluster to represent the 00 class.
- **Debug:** Enable the debug mode.
- **Final plotting:** Enable the plotting of the global graph (time consumming).
- **Close figures:** Close figure before starting the analysis.
- **Imagewise normalization:** Normalize each image using a robust variance (MAD) estimation and the median. This is dangerous when data have less than 50% of points in class 00.
- **Remove bad images:** Remove image whose points are statistically different from the average (based on mean and variance).
- **Normalization:** Select a normalization (options.normalization_type):
  1. none: no normalisation.
  2. standard : use the mean and standard deviation given by `mean` and `std` from matlab. Usually performs poorly due to the outliers ([3]) corresponding to the class 10,01,11.
  3. MAD/Med : use the median absolute deviation [4] and the median (robust up to 50% of outliers)

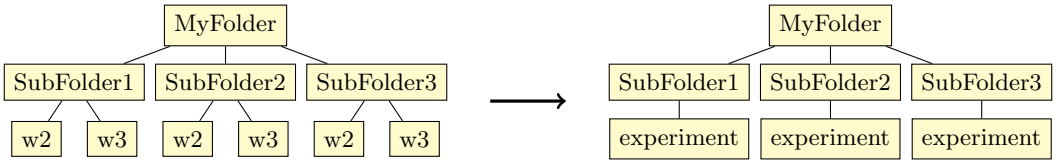

Figure 1: Two possible organizations of files in `bcall` and `pretrain_bcall`. The folders can contains other files. The first case is before any processing, while the second correspond to the state of the folder after it has been loaded once.

4. MDL/MS : use MDL criterion to estimate the covariance matrix and a mean shift to locate the lower left mode. This approach is more robust and can be used in cases where the 00 cluster is less than 50%.
5. same as previous but use the covariance matrix to whiten the data.

- **Save:** Save parameters
- **Reset:** Reset to default (saved) parameters.
- **Ok:** Launch the base calling.

2.3 pretrain_bcall

It is a *command line program* (there is no graphical user interface for this program) which perform the analysis of several files in distinct directories which are name following the pattern `options.cg` using the *same* parameters estimated globally or on the first dataset. This program allow to analyse a set of several folders containing each a pair of file w2/w3 (or w4/w5 etc...). Accordingly, the files should be organized as in Figure 1. The experiments do not necesseraly have the same number of beads/objects/images.
The command line window type (matlab or octave) usage is:
`>> [experiment,options]=pretrain_bcall('Directory',options)`
You can also use it as :
`>> pretrain_bcall('Directory',options)`
The arguments of the program are

- **'Directory'** is a string indicating the path to the folder where are the data, please do not forget that a string is constructed using quotes, eg `'Z:\Analysis\...'`.
- **options** is a structure containing options (see Table 2.1). In addition, set `options.global=true` to enable the analysis of the datasets globally (merging all the data, normalizing and estimating the parameters of the mixture) or using only the data set from the 1st file. By default `options.global=false`.

For example, in the console window:

```
directory='Z:\Analysis\MyFolder\'; % containing SubFolder1
    SubFolder2 SubFolder3
options.cg='/SubFolder*'; % this is compulsory unless it is  'SNP
    *'
options.max_cluster_size=1.5; % Define the cluster size
options.beta=0.1; % set the strengh of the constraint (small is
    strong)
options.satellites=true; % will use satellites
options.remove_bad_images=true; % will remove bad images
options.global=false; % will use only the 1st image to calibrate
    the analysis
pretrain_bcall(directory,options); % finally lauch the analysis
```

2.4 dyeswitch_bcall

Command line which performs the analysis of a dye switch experiment. The set of experiments share the same number of images/objects. The calibration is learned by default on the merged datas (globally : `options.global=true`).
The command line window type (matlab or octave) usage is:
`>> [experiment,options]=dyeswitch_bcall('Directory',options);`
You can also use it as :
`>> dyeswitch_bcall('Directory',options);` The arguments of the program are

- **'Directory'** is a string indicating the path to the folder where are the data, please do not forget that a string is constructed using quotes, eg `'Z:\Analysis\...'`.
- **options** is a structure containing options (see Table 2.1).

For example, in the console window:

```
directory='Z:\Analysis\MyFolder\'; % containing SubFolder1
    SubFolder2 SubFolder3
options.cg='/SubFolder*'; % this is compulsory unless it is  'SNP
    *'
options.max_cluster_size=1.5; % Define the cluster size
options.beta=0.1; % set the strengh of the constraint (small is
    strong)
options.satellites=true; % will use satellites
options.remove_bad_images=true; % will remove bad images
options.global=false; % will use only the 1st image to calibrate
    the analysis
dyeswitch_bcall(directory,options); % finally lauch the analysis
```

3 Two steps analysis

The two functions `calibration` and `apply` allows to perform the analysis in two steps. First estimating the parameters and then applying the classifiacation.

```
% Step 1
directory1='Z:\Analysis\MyFolder1\'; % containing the calibration
    datasets
options.cg='/SubFolder*'; % this is compulsory unless it is  'SNP
    *'
options.max_cluster_size=1.5; % Define the cluster size
options.beta=0.1; % set the strengh of the constraint (small is
    strong)
options.satellites=true; % will use satellites
options.remove_bad_images=true; % will remove bad images
options.global=true; % will use only the 1st image to calibrate
    the analysis
options=calibration(directory1,options);
save('mycalibration.mat','options'); % eventually save the
    calibation

% Step 2
options = load('mycalibration.mat'); % and restore it later
directory1='Z:\Analysis\MyFolder2\'; % containing the folders to
    analyze
apply(directory1,options); % apply the calissifiaction and save
    the results
```

References

[1] Dempster, A.P.; Laird, N.M.; Rubin, D.B. (1977). "Maximum Likelihood from Incomplete Data via the EM Algorithm". Journal of the Royal Statistical Society. Series B (Methodological) 39 (1): 138.

[2] http://en.wikipedia.org/wiki/Expectation-maximization_algorithm

[3] http://en.wikipedia.org/wiki/Outlier

[4] http://en.wikipedia.org/wiki/Median_absolute_deviation