BASTA – Bayesian statistical tissue profiling using DNA copy number amplifications (Manual updated 26.2.2009)

BASTA – Bayesian statistical tissue profiling using DNA copy number amplifications (Manual updated 26.2.2009) 1
Introduction 1
Installation 1
Running the clustering analysis 2
Viewing the results 3
Input files 4

Introduction

BASTA is a software tool for unsupervised classification of tissue samples based on copy number amplifications. BASTA can be used to learn optimal partition (i.e. clustering) for the samples and recognize the aplification patterns that are characteristic for each of the clusters in the partition. The details of BASTA are given in:

Marttinen, Pekka, Myllykangas, Samuel and Corander, Jukka (2009) Bayesian clustering and feature selection for tissue samples using DNA copy number amplifications. BMC Bioinformatics, accepted for publication.

BASTA can be used freely and comes with no warranties. If you use BASTA, please cite the above-specified paper. Questions and feedback can be sent to

pekka.marttinen@helsinki.fi

or to

jukka.corander@abo.fi

Installation

BASTA was created with Matlab and is currently available as a compiled version for Windows XP and Mac OS X users.

To run BASTA you need to install Matlab Runtime Component (version 7.6, the same as for BAPS 5.x), which is free and can be downloaded with BAPS for Windows XP from http://web.abo.fi/fak/mnf//mate/jc/software/baps_xp.html, and for Mac OS X from http://web.abo.fi/fak/mnf//mate/jc/software/installation_mac_ppc.html(PowerPC version) or http://web.abo.fi/fak/mnf//mate/jc/software/installation_mac_intel.html(Intel Mac version)

In the Windows XP, after the installation of the runtime component, unzip the BASTA package in any folder, and the program is ready for use. In the Mac OS X environment,

unzip the BRAT package in any folder, and check BAPS installation instructions (web links shown above) to run it (see in particular point 6.).

The runtime component can be installed anywhere in the system, EXCEPT under the Matlab path, if Matlab is installed on your computer.

Running the clustering analysis

Once you run Basta.exe, two windows open: one for output and the other with one button "Perform clustering of profiles". Clicking the button with mouse starts the analysis. You will first be asked whether you wish to use 'Binary data' or 'Pre-processed data'. (This is done because you may wish to run to search algorithm more than once, and pre-processing of a large data set may take some time. It suffices to do the pre-processing just once.) If you select 'Binary data' the specified data will be pre-processed and you will be asked if you wish to save the preprocessed data. By selecting 'Pre-processed data' this intermediate step will be skipped.

Once the data is pre-processed, you need to initialize the search algorithm. First, you need to specify maximum number of clusters allowed by the search algorithm. This number is required for implementational purposes and it is recommended that the user specifies the value to be large enough not to constrain the analyses (For example, if the search algorithm ends up with a solution in which the number of clusters is close to the specified upper bound, we recommend that the analysis would be repeated with a higher upper bound). On the other hand, specifying unnecessarily high upper bound may slow down the analysis. After specifying the upper bound, the user needs to initialize the state of the algorithm, by specifying an initial partition. There are two ways of doing this: 1) user may input a manually created partition, or 2) user may specify some reasonable value for the number of clusters, and the software automatically creates an initial partition with the specified number of clusters using a simple clustering algorithm (complete linkage clustering).

The search algorithm starts immediately after the initialization. After the search has converged (i.e. reached a state from which no improvement to the current model is available) the user must specify a file in which the results will be written (such a file will have .mat extension). This result file is needed when viewing the results. The result file is of no use as such, and the user is required to use some of the options described in the next Section to investigate the results (one can investigate the contents of the file by loading it to Matlab, though). The number of clusters in the found optimal partition as well as its Log(ml*prior) value are printed on the output window. The log posterior values, i.e. Log(ml*prior), can be used to compare the goodness of different solutions if it happens that the search algorithm ends up in different solutions on different runs. In general, it is a good idea to repeat the search at least a couple of times with different initial numbers of clusters (even with the same number more than once), to see if better solutions can be reached.

Viewing the results

Draw Partition: By selecting "Draw Partition" from Results menu, you can produce a graphical representation (see Figure 1) of the obtained partition (i.e. clustering). In this representation, the amplification profiles belonging to the same cluster are drawn next to each other. First, you need to specify the result file from the clustering analysis. In addition, two optional files: "chromosome lengths file" and "sample names file" (see Section Input files for details) can be specified. If the chromosome lengths file is selected, the corresponding boundaries between different chromosomes will be shown in the figure by vertical black lines. You can also select which clusters you wish to be drawn in the figure. The clusters are numbered such that the largest cluster has label "1", the second largest label "2", and so on. The clusters are separated in the figure by horizontal black lines. Details of any cluster in the figure can be obtained by clicking a particular cluster by mouse. Then, information (the samles contained in the cluster, and the cluster-specific amplifications) about the cluster will be printed in the output window.

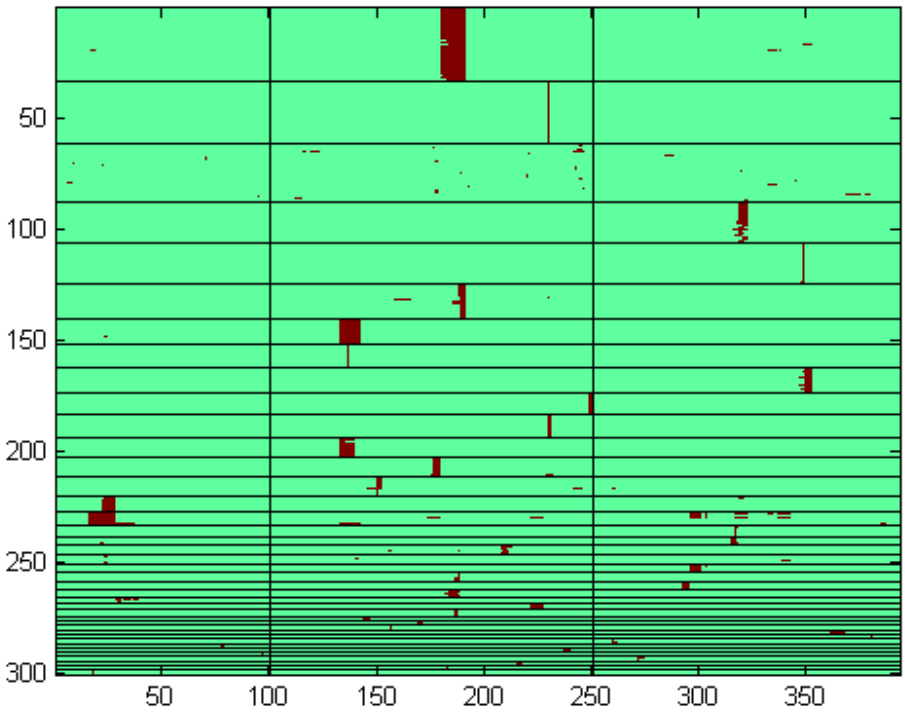

Figure 1: Output obtained by selecting "Draw Partition". In the figure the amplification profiles (horizontal colored lines) belonging to the same cluster are drawn next to each other, and the boundaries of clusters are specified by horizontal black lines. The profiles are drawn such that 'ones' (the amplified sub-bands) are colored brown, and 'zeros' light green. Thus, for example, the first cluster (on top) contains about forty aplification profiles. The boundaries of the chromosomes are shown by vertical black lines. Thus, there are three chromosomes in the figure, the lengths of which are approximately 100, 150 and 150 sub-bands.

Write Cross-Reference Table: By selecting "Write Cross-Reference Table" from the results menu, one can generate and print a cross-reference table between the obtained result clustering and some other reference clustering (see Section Input files for details). The table will be printed as tab-delimited text (.txt) in a file specified by the user. As such, it can be read for example in Excel.

Write Results in Text: Writes in text the obtained partition and cluster-specific amplifications in a file (.txt) specified by the user. Writes also for each sample the changes that would occur in the log posterior of the optimal model, if the sample were moved to any other cluster.

Draw sub-band significance profile: Draws a plot of calculated (log) Posterior Ratio values for the sub-band positions. These values specify the significance of a specific position for the clustering. The higher the value, the more information a particular sub-band position carries about the underlying clustering, see Marttinen et al. (2008) for details about these posterior ratios.

Input files

In this Section we describe various input files required by the program. The most important of the files are "Binary data file" and "Result file".

Binary data file: This is the main data file, which you need to have before you can do anything else! This file is a text file (.txt) containing the amplification profiles as binary data, such that rows correspond to samples, and columns correspond to the different sub-band positions. (Example: example_data.txt. You can try this with upper bound for the number of clusters equal to 70, for example, and initial number of clusters equal to the default value 68, for example. When we analyzed this data set with upper bounds 50, 60, 70 and 80 using default initial numbers of clusters, the best outcome was a partition with 47 clusters, and Log(ml*prior) value equal to –2905.7588. Notice that a higher, i.e. less negative, value is better. A single analysis should take a couple of minutes to complete.)

Pre-processed data: After the binary data is pre-processed, you can save the pre-processed data (.mat). The saved data can be used in subsequent runs of the search algorithm, without need to redo the pre-processing.

Result file: After the search algorithm is completed, the user must specify the file in which the results are saved. This file is in Matlab data file format (.mat), and it is needed when viewing the results using some options available from the "Results" menu.

Chrosome lengths file: This is a text file (.txt) with as many rows as there are concatenated chromosomes in a data set. Each row specifies the length of a chromosome. The total length of chromosomes must be equal to the number of columns in the data file. (Notice that, in a data file, different chromosomes are concatenated.) For example, in the example data file (example_data.txt) there are 393 columns (corresponding to

chromosome sub-bands). Thus, the chromosome length file (example_chr_lengths.txt) for these data might contain for example three rows:
100
150
143
specifying that there are three chromosomes with lengths 100, 150 and 143 sub-bands in the data. This file is is needed (optionally) when drawing image of the result partition.

Sample names file: This is a text file (.txt) with as many rows as there are rows in the data file. Each row in the file specifies the name of the corresponding sample. As an example, see ‘example_names.txt’, which specifies the names for the samples in the example data file. These names are simply p1, p2, ... , p300, i.e. the name of a sample profile is ‘p’ with the index of the sample (in the example data there are amplification profiles from a total of 300 samples). This file is needed when drawing image of the result partition. Expecially, when such an image is clicked, the contents of the cluster which was clicked will be displayed on the output window using the given names for the samples. If no name file is specified, the program will use default names, which simply are the indices (1, 2, etc.) of the samples in the data file.

Reference clustering: This file is required when “Write Cross-Reference Table” is selected. It is a text file (.txt) with as many rows as there are rows in the data file. On each row, there is a name of the reference cluster to which the corresponding sample belongs. See for example ‘example_ref_clustering.txt’.