Installing Bayesian Decomposition
v1.0
Bioinformatics, Fox Chase Cancer Center
7 June 2002

Bayesian Decomposition is provided as a tar file. The file should be downloaded from the Fox Chase Bioinformatics web site and expanded with

```
tar xvf FileName.tar
```

which will create a bd directory with executables and associated files within it.

Citing Use of Bayesian Decomposition
If the results of Bayesian Decomposition analysis are included in a publication or presentation, please cite Moloshok, T. D., *et al.* (2002). "Application of Bayesian Decomposition for analysing microarray data." *Bioinformatics* **18**(4): 566-575.

Running the Program
The provided control file (Test.ctrl) and data file (Test.data) will provide templates for the necessary files to run Bayesian Decomposition. These files also provide a test for successful installation as noted below. The detailed structure of these files is discussed in the associated file, BD_Files.pdf.

Bayesian Decomposition is run from the command line by giving the executable name together with the root form of the control and data files. For the control and data files named Test.ctrl and Test.data, the program is invoked with

```
./genpat Test
```

where it is assumed that the executable is in the working directory. Alternatively, for knowledgeable UNIX users, the executable can be placed in a directory included in the PATH environmental variable. The program will echo to the terminal window the parameters for the run, including the input data size, the number of dimensions posited, and the number of iterations. Then the program will print lines during annealing at each tenth of the way until reaching the run temperature (1/T = 1). A typical line appears as (with interpretation noted below it in red):

```
1/T = 0.10 of 1.00     2194 ( 702)       78503.06
```

present inverse temperature and target	# of atoms in distributions	# of atoms in the patterns	present value of χ^2

Depending on the number of iterations and speed of the computer, this can take considerable time (~10 minutes on a 700 MHz Pentium III running RedHat Linux). Once equilibration has finished, sampling will begin. The progress of sampling is reported every 100 iterations, with a typical line being (with interpretation in red)

```
Sample  1400 of    2000    4289 ( 351)        33665.26
```

present sample and total number to generated | # of atoms in distributions | # of atoms in the patterns | present value of χ^2

Upon completion, the program summarizes the process with a few final lines

```
Random seed was   2147483647
< Chisquared >   =  33545.66
< A Atoms >      =   4273.13
< F Atoms >      =    323.30
log[e]Prob(Data) = -21495.20
```

which give the random seed, the average χ^2 value of the samples, the average number of atoms in the samples, and the evidence, which is not used in this example. It should be noted that this is not the χ^2 of the mean, but rather is an average of the χ^2 of each individual sample. One useful measure that can be made on the mean model returned by the algorithm is to verify that its χ^2 value is lower than the mean χ^2 of the samples. This helps to verify that the sampler has visited a reasonably large space in the posterior distribution.

The success of the installation can be determined by comparing the output files (Test.inf, Test.out, and Test.mov) with the included files (TestOut.inf, TestOut.out, TestOut.mov). The files should be the same within small differences that may result from slight differences in the specific systems that might lead to differences in insignificant digits.

Output Files

The program creates three output files upon completion. The first is the inference file (.inf) that includes the mean model and associated uncertainties. The second is the data file (.dat) that includes the input data, the input uncertainties, and the calculated mock data (the data reconstructed from the model). The third is a movie file (.mov) that contains individual samples (snapshots) taken during the Markov chain process.

The results can be visualized directly from the ASCII output files using a 4GL program such as Matlab or IDL. In addition, a parsing tool that takes these files and creates tab-delimited files is included as a java class file with the documentation files. This can be invoked with

```
java readinf Test.inf
```

after the run is complete. The output of this Java program is a file Test.tab that can be used as input into a standard spreadsheet program for generating graphs of the patterns and distributions. In addition, it is often useful to compare the mock and data matrices to verify that the algorithm has created a model that reconstitutes the data. We are presently

completing a Java tool that will allow these views directly from the output without the use of spreadsheet programs. In addition, the residuals can provide indications of the need for more patterns (if a clear pattern remains in the residuals) or indications of which parts of the data are particularly poorly fit by the model.

Visualization is often most useful for the pattern matrix. For data with expected coherent structure, the individual patterns should make physiological sense (for the included Test.dat file, the patterns are three underlying Gaussian shapes). Visualization is often most useful for time series data or data which comprises multiple groups (e.g. malignant tissue vs. benign tissue).

For gene expression data, the visualization of the distribution matrix is often uninformative. Instead it is useful to rank the genes by their assignment to the individual patterns, noting which genes are linked exclusively to a pattern and which genes are strongly represented in a pattern. Although many genes presently being spotted on microarrays are of unknown function, this ranking of genes can provide an indication of the signaling pathway associated with their expression.

Using Bayesian Decomposition

To use Bayesian Decomposition, you need to convert your data and uncertainties into the proper format. The file BD_Files.pdf contains detailed description of the input files. In addition, the Test.ctrl and Test.data files can be copied and used.

Typically at least 2000 iterations should be done before sampling, however in complex, highly overlapping data sets more may be needed. By performing multiple runs, the adequacy of the equilibration time can be tested (i.e. all runs of BD should recover the same model within the uncertainties). Finally the number of atoms in the A and P atomic domains should be estimated. Generally these estimates only weakly affect the program, and a good estimate is generally to use the number of elements in each matrix respectively, if there is no convolution function as is the case here.

When exploring data sets with Bayesian Decomposition, it will be necessary to provide the number of patterns to the algorithm. Since it is rare that this number is actually known, it is usually a good idea to run Bayesian Decomposition many times using different values for the number of patterns and to explore the results.