# Bayesian Analysis of Gene Expression Levels

# (BAGEL)

### Version 4.1, July 2009

Zhang Zhang, Colin Meiklejohn and Jeffrey Townsend

Introduction
Statistical model
Implementation
Command-line Execution
Example analysis
Output
Comparison of error models
Bug Reports
Copyright information
More Information
References

## Introduction

Bayesian Analysis of Gene Expression Levels (BAGEL) is a program that allows statistical inferences to be made regarding differential gene expression between two or more samples measured on spotted (two-channel) microarrays. BAGEL makes these inferences from normalized ratio data, on a gene-by-gene basis. The advantages of BAGEL include ease of use, straightforward interpretation of results, statistical robustness, flexibility in accepting different experimental designs, and that it is free. BAGEL was written by Jeffrey Townsend, who periodically updates and improves the program, and to whom bugs should be reported. BAGEL is available for

Windows, Mac OS9, Mac OSX, and Linux. BAGEL can be downloaded from the Townsend Lab web site, http://www.yale.edu/townsend/software.html.

Please cite:
Townsend, J.P., and D.L. Hartl. 2002. Bayesian analysis of gene expression levels: statistical quantification of relative mRNA level across multiple strains or treatments. Genome Biology 3 (12): research0071.1-0071.16.

## Statistical model

A number of factors can influence the signal intensity of labeled DNA hybridizing to a microarray spot, such as hybridization efficiency or concentration of target sequences in the spot. Any such factors that will be shared by samples hybridizing to the same spot will be eliminated by considering the ratio of the two signal intensities. BAGEL explicitly takes this into account by using ratio measurements, not single-channel signals, coming from two-dye competitive hybridizations. BAGEL makes transitive comparisons across ratios, for example inferring the ratio of sample A to sample C across a set of hybridizations that directly compare sample A to sample B and sample B to sample C. Data that are appropriate for analysis by BAGEL must therefore have the following properties:

1) The data should be collected in such a way that pairs of samples share sources of variation that are non-trivial and that are not of interest to the researcher; and the *relative* magnitude of some metric between the two samples is the measurement of interest. The originally envisioned use for BAGEL, two-channel microarray data, is an obvious example of data with this structure, but in principle BAGEL could be used to analyze any other kind of data that fit these criteria. For this reason, tiled microarray platforms such as Affymetrix do not lend themselves easily to BAGEL analysis.
2) All genotypes, tissues, treatments etc. (hereafter referred to as ��expression nodes��) to be analyzed must be connected to all other nodes through direct or indirect comparisons. For example, given four nodes, a set of microarray experiments that competitively hybridized node 1 vs. node 2, node 2 vs. node 3, and node 3 vs. node 4 would permit the estimation of relative expression levels of genes across all four nodes. On the other hand, a set of experiments that competitively hybridized node 1 vs. node 2 and node 3 vs. node 4 would permit the estimation of gene expression levels between nodes 1 and 2 and between 3 and 4, but no estimates could be made regarding comparisons between 1 and 3. Any experimental design incoporating a reference sample will necessarily fulfill this criterion, as all nodes are connected through their direct comparisons with the reference.
3) There must be a sufficient number of measurements (replicate experiments) to estimate the parameters. In principle, this means as many measurements as nodes, or half as many hybs as nodes (when estimating a single variance parameter, see below). In practice, requirement 2 will usually necessitate more than this many hybs, and of course, the greater the replication, the more precise the estimates of gene expression. Experience suggests that an experimental design providing at least three measurements for each node is a good target number for providing reasonable statistical power.

Formally, the statistical model employed by BAGEL assumes that the measured fluorescence intensity for one channel is a function of (1) the true quantity of the labeled mRNA species; (2) some number of multiplicatively and/or additively confounding factors that are specific to the spot in question but shared by the measured intensity from the other channel; and (3) some number of unbiased, randomly distributed error terms (for example, reverse transcription or labeling efficiency). If the error terms contribute additively, the observed ratios of gene expression between the *i*th and the *j*th samples ($z_{ij}$) should be similar to the ratio of two normal distributions, and can be approximated by the function

$$f(z_{ij}\,|\,\mu_i,\sigma_i^2,\mu_j,\sigma_j^2)=\frac{\sigma_i^2\mu_j+\sigma_j^2,\mu_i z_{ij}}{\sqrt{2}\pi(\sigma_i^2+\sigma_j^2 z_{ij}^2)^{3/2}}\,Exp\left(-\frac{(\mu_i-\mu_j z_{ij})^2}{2(\sigma_i^2+\sigma_j^2 z_{ij}^2)}\right)$$

where $\mu_i$ is the expression level and $\sigma^2{}_i$ is the variance of a gene in sample *i*. If the error terms contribute multiplicatively, then the ratio $z_{ij}$ can be approximated by the ratio of two lognormal distributions using the function

$$f(z_{ij}\,|\,\mu_i,\sigma_i^2,\mu_j,\sigma_j^2)=\frac{1}{z_{ij}\sqrt{2\pi(\sigma_i^2+\sigma_j^2)}}\,Exp\left(-\frac{(\ln z_{ij}-(\mu_i-\mu_j))^2}{2(\sigma_i^2+\sigma_j^2)}\right)$$

BAGEL can implement both kinds of error models.

This approach requires the estimation of 2*n* �C1 parameters (*n*-1 expression levels and *n* variances) for each gene. One way to reduce the number of parameters is to assume that, for a given gene, all the nodes have the same error variance. This reduces the number of parameters to *n*, and consequently reduces the number of replicates required for statistical power. Similarly, one can assume that, for a given gene, all samples have a constant relationship with the expression level, *i.e.* that they have a common coefficient of variation ($\nu = \sigma_i/\mu_i$ for all *i*). This approach also requires estimating *n* free parameters.

BAGEL explores the likelihood function derived from either of the ratio formulas above for all nodes using a markov chain monte carlo (MCMC) approach in a Bayesian framework. This method starts with a random vector of parameters and then changes two of the parameters by small, random steps. At each step the likelihood of the data given the model above and the new parameter values is calculated. If the new parameters give a better fit to the data, then the new values are accepted. If the new parameters give a worse fit to the data, then the new values are accepted with a probability proportional to their likelihood. In this way the markov chain searches the parameter space, finding combinations of relative gene expression levels that produce the greatest likelihood, and samples from the chain are used to construct the Bayesian posterior probability of the parameters given the data.

Figures 1-3 show an example of a markov chain from BAGEL analysis of a single gene with four nodes (gene BcDNA:LD09936 from the example dataset discussed below). The relative expression levels for the four nodes are indicated on the y axis in the upper panel (note that they have not yet been normalized such that the node with the lowest expression is set to one) and the step in the chain is shown on the x axis. The lower panel shows the likelihood of the data given the parameters (relative expression levels) at that step in the chain. Figure 1 shows the first 200 steps and Figure 2 shows the first 1000 steps in the chain. Note that the relative expression levels quickly move away from their (random) initial values towards greater or lesser expression levels more consistent with the data, as indicated by the jumps in likelihood. This initial exploration of parameter space, while important for finding the peak of greatest likelihood, should not contribute to the data sampled from the chain, or it will not converge on the posterior probability of the Bayesian formulation of the functions shown above. Put another way, we don��t want BAGEL to make inferences from the data in regions of low likelihood simply as a result of the (random) initial parameter settings. For this reason, BAGEL runs the chain for a ��burn-in�� period before beginning to sample parameter values for the posterior probability. The default length of this burn-in is 20,000 steps in the chain; you can see from Figures 1 and 2 that the parameter values move rather quickly to a region of parameter space and stay there. Figure 3 shows the last 1000 steps sampled from the analysis, following a 20,000 step burn-in. The relative expression levels for all four nodes have stabilized around the most likely values. One thing to note is that Figures 1 and 2 show the parameter values at each step of the chain, whereas Figure 3 shows every 20th step in the chain (so although 1000 steps are shown, the data actually spans 20,000 steps). This is known as the period of sampling from the chain, and is important because it removes correlations between successive steps (the plateaus for a given node in Figures 1 and 2) which otherwise compromise the ability of the markov chain to explore the full range of potential parameter values.

BAGEL infers relative expression levels and statistical significance from the parameter values it samples from the chain. The relative expression level of a node for a given gene is the median value across the samples from the chain, normalized relative to the lowest median expression level. The 95% credible intervals are the values within which 95% of the samples from the chain are bounded, and *P*-values for the ��hypothesis�� that node A > node B are simply the proportions of samples from the chain where node A��s expression level was greater than node B��s.

## Implementation

BAGEL does not perform normalization of raw microarray data (for example, to account for systematic differences in signal intensity between the two fluorophores), and an appropriate normalization method should therefore be implemented prior to BAGEL analysis. Following normalization, the data must be formatted in a way that BAGEL can use. The appropriate format is a tab-delimited text file that contains (normalized) ratio data for all the relevant genes and hybridizations, as well as some header rows and columns (see below).

The details for loading and running BAGEL will differ depending on your platform. See the readme file that comes with the version of BAGEL you have downloaded for specific instructions.

Upon executing BAGEL, the following text will appear on your terminal:

```
B.A.G.E.L.
```

```
Acceptable files for Unix B.A.G.E.L. are tab-delimited text files with three header
rows. The second and third rows must containing unique names for each experimental
expression node and reference expression node, followed by any number of data rows for
each gene of interest:

[Your Notes] [Your Notes] [Label1] [Label2] [Label3]...
[Your Notes] [Channel1]  Exp1   Exp2   Exp3      ...
[Your Notes] [Channel2]  Ref1   Ref2   Ref3  ...
ORF1     CommonName1  Ratio1  Ratio2  Ratio3 ...
...      ...      ...    ...        ...

Please type the exact name of a text file of microarray ratio results to analyze:
```

This text shows you the tab-delimited text format in which BAGEL expects your input file. There are three header rows to a properly formatted BAGEL input file. Square brackets indicate information that may be in your input file for your own reference only. All unbracketed entries must be present. There is no need for the experiments (Label1, Label2, Label3...) to be in any particular order, and there is no inherent difference between experimental (Exp) and reference (Ref) samples. In fact, in any experimental design wisely incorporating dye-swaps, sample names will presumably appear in both row 2 and row 3. It is, however, essential that a sample name be exactly consistent across all columns, or else BAGEL will infer two different samples when there is in fact only one.

At the prompt following the example data format, type the name of the input file, including the directory path. For example, in UNIX, directory pathnames look something like

/Users/jeff/DOCUMENTS/RESEARCH/Software/BAGEL/Datafilename.unx

NOTE: Keep Datafilename and your experimental node names short. If the Datafilename is too long, BAGEL has to truncate it and use a far less intutitive name for your output file.

BAGEL then asks you to verify the number of hybs and the names of the expression nodes (sample•treatments) in your experimental design. Press RETURN to verify, or press ��q�� to quit and correct your input file.

You are then presented with a menu of options:

```
Current MCMC settings:
(E)rror Model: Additive errors, estimating/constraining Coefficient of Variation terms
(C)onstrained Coefficient of Variation: True
(I)nitial values:
Mu[M1-2] := 1.00    Coefficient of Variation[M1-2] := 0.2000
Mu[M2-8] := 1.00    Coefficient of Variation[M2-8] := 0.2000
Mu[M5-7] := 1.00    Coefficient of Variation[M5-7] := 0.2000
Mu[M7-8] := 1.00    Coefficient of Variation[M7-8] := 0.2000
(M)u step size: 0.50
(V)ariance/CV step size: 0.500
(B)urn in, # generations: 20000
(P)eriod of sampling from the Markov chain: 20
(G)enerations to be sampled: 10000
(F)ull output of the chain: False
(T)uning depth maximum: 8
```

(E)rror Model: This option allows you to choose between additive and multiplicative errors, and whether you wish to estimate or constrain the variance or the CV of the gene expression levels, as was described above. All the models BAGEL uses work fairly well, and results are usually quite similar (see below), so unless you have reason to, it��s probably best to use the default settings (additive errors, constraining coefficients of variation).

(C)onstrained Variance/Coefficient of Variation: If TRUE, variances/coefficents of variation for all expression nodes are assumed to be the same. With this option, you must have as many measurements as expression nodes. If FALSE, variances for all expression nodes are separately estimated. In this case, you must have at least twice as many measurements as expression nodes (minus one). Unless your data is very highly replicated, using constrained variance is recommended. With underreplicated data, estimation of variances for each sample is very imprecise and can lead to misleading results. When a design is well-replicated, to the extent it has been tested so far, it seems that estimating each variance independently changes BAGEL estimates of the expression levels very little.

(I)nitial values: The starting Mu and Sigma Squared parameter values for the Markov Chain. In some applications of the MCMC method, it is very important to try many different initial starting values to ensure that the chain does not get stuck in one region of the state space. This is not much of an issue with the BAGEL models. With moderately decent microarray data, it does not get stuck in local peaks.

(M)u step size: The step size is a very important parameter in the MCMC implementation, which BAGEL automatically tunes for you as long as (T)uning Depth maximum, below, is greater than one. Note that BAGEL uses information from genes previously analyzed in your dataset to help it guess the right step size.

(V)ariance/CV step size: See (M) above.

(B)urn in, # generations: The default burn-in (20000 iterations) is rather excessive for most data sets. However, it is nice to feel confident that stationerity in the chain has been reached. Decreasing this parameter will make BAGEL run slightly faster. It is up to you to ensure that BAGEL is reaching stationerity. One way to do this is to run multiple chains and check that they are converging on the same results.

(P)eriod of sampling from the Markov chain: This is how many iterations are performed until the current state is sampled from the chain to construct posterior distributions. When the period is greater than one, this is referred to as ��thinning the chain��. It subdues correlations that are present between subsequent states of the chain. Decreasing this period substantially decreases computation time, but compromises the independence of the samples and thereby the adequate mixing of the chain.

(G)enerations to be sampled: Ten thousand generations yields accuracy to about three digits. Increasing the number of generations (iterations) increases the number of digits of accuracy. The product P * G largely determines the amount of time necessary for a BAGEL run for a gene.

(F)ull output of the chain: FALSE. Keep it that way, unless your input file has only one or two genes and you really like BAGEL to talk about everything it��s doing. If you run with Full Output, BAGEL saves the posterior distributions for Mu for every sample. On a genome-wide data set, it could rapidly fill your hard drive.

(T)uning depth maximum: How hard (in MCMC runs) you wish BAGEL to try to find an optimal step size for a gene. In the newest versions of BAGEL (>3.0), an optimal step size is discovered in a few tries, and almost always in six or seven, so there is little point in changing this from the default value (8 tuning iterations). A gene for which no optimal step size is found is marked FALSE under "Acceptable?" in the output file, but this just about never happens.

When you are done changing settings, or if (more likely) you have not changed them at all, press RETURN, and BAGEL will begin to work on your data set. BAGEL currently takes a long time to run, say, a minute per gene or more. Thus, it is frequently convenient to set it going on a computer you won��t need for the night, and leave it alone.

## Command-line Execution

BAGEL also provides a command-line option to run the program without manual intervention, which is suitable for submitting jobs to the queues for processing. The command line is parsed as follows:

```
USAGE: BAGEL [options] InputDataFile
```

Where:
  InputDataFile = Expression data

Options:

| | |
|---|---|
| -e ErrorModel | valid options are 'AM', 'AC', 'EM', or 'EC' |
| -c | estimate individual variances. If this flag is not given, the default (single variance) is used |
| -i Mu1,Mu2,... | initial mu estimates for each expression node (comma delimited, no spaces) |
| -I Var1,Var2... | initial sigma-squared value (if the -c flag is not given) or a list of initial sigma-squared values for each each expression node (if the -c flag is given). |
| -m MuStepSize | step size for varying estimates |
| -v VCVStepSize | step size for varying variances |
| -b BurnIn | # of generations of burn-in prior to sampling |
| -p SamplingPeriod | period of sampling from the Markov chain |

| | |
|---|---|
| -g ChainSampleSize | # of generations to be sampled |
| -f FullOutput | if this flag is given, the full output of the MCMC chain will be written (c.f. the warning in the BAGEL tutorial) |
| -t MaxTuningDepth | how hard BAGEL should try to tune the step size |
| -q | Dry run: parse input file and command line, but abort before starting the BAGEL run. |
| | |
| (the following parameters are not covered in the previous BAGEL tutorial) | |
| -r RandSeed | Seed for random number generator. If not given, the random number generator will be seeded from the system clock (so that multiple BAGEL runs will perform independent samplings of the parameter space). |
| -o ResultsFile | Filename for the BAGEL results file. If not given, a default name will be generated based on the name of the input file (c.f. the BAGEL tutorial). |

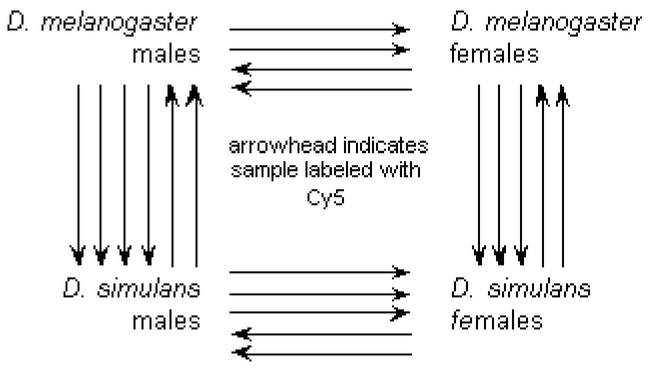
The options may be given in any order. If an option is repeated, the last value of that option will be used. If an option is omitted, the default value (as documented in the BAGEL tutorial) will be used.

The command-line BAGEL was provided by the courtesy of Dr. Mark Voorhies.

## Example analysis

The microarray data used here is taken from Ranz et. al. 2003. These experiments analyzed mRNA levels for ~5000 genes in adult males and females of *Drosophila melanogaster* and a closely related species, *Drosophila simulans*. This dataset therefore has four nodes, and the experimental design and level of replication is shown in Figure 4. For simplicity of illustration, four genes from the full dataset were selected.

**Figure4**



**1) Execute BAGEL**

If you double-click on the application, a window comes up with the following text in it:

```
B.A.G.E.L.

Acceptable files for B.A.G.E.L. are tab-delimited text files
with three header rows. The second and third rows must containing unique names
for each experimental expression node and reference expression node,
followed by any number of data rows for each gene of interest:

[Your Notes]  [Your Notes]  [Label1]      [Label2]      [Label3]      ...
[Your Notes]  [Channel1]           Exp1          Exp2          Exp3          ...
[Your Notes]  [Channel2]           Ref1          Ref2          Ref3          ...
ORF1                 CommonName1          Ratio1 Ratio2 Ratio3 ...
...           ...                  ...           ...           ...
```

This text is meant to give you the tab-delimited text format by which BAGEL expects your input file to be formatted. There are three header rows to a properly formatted BAGEL input file. Square brackets indicate information that may be in your input file for your own reference only. All unbracketed entries must be present. There is no need for the experiments (Label1, Label2, Label3...) to be in any particular order, and there is no inherent difference between experimental (Exp) and reference (Ref) samples. In fact, in any experimental design wisely incorporating dye-swaps, sample names will presumably appear in both row 2 and row 3. It is, however, essential that a sample name be exactly consistent across all columns, or else BAGEL will infer two different samples when there is in fact only one.

### 2) Input file name and verify file format

```
Verifying Input File Dros_spp
Dros_spp    DYE   1A   1B   17B   27A   6A   6B   18A   18B   28B   19A   21A   21B
26A   26B   28A   20B   22A   22B   29A   29B   24B

dataset

Initializing ExpressionNodeNameList...
Assigning names to expression nodes.....................

Number of Hybs: 21

Press RETURN to verify or q to quit:
{RETURN}

gene name

Assigning more names to expression nodes.....................

Please verify that 4 expression nodes are desired,
that all desired nodes are are listed below, and
that each of the following are unique expression nodes.
CS
CSf
Sim
Simf

Press RETURN to verify or q to quit:
{RETURN}
```

### 3) Choose parameter settings

```
Assigning experimental node names to hyb list.....................
Assigning reference node names to hyb list.....................
File Dros_spp header rows verified.

Current MCMC settings:
(E)rror Model: Additive errors, estimating/constraining Coefficient of Variation terms
(C)onstrained Coefficient of Variation: True
(I)nitial values:
Mu[CS] := 1.00 Coefficient of Variation[CS] := 0.2000
Mu[CSf] := 1.00 Coefficient of Variation[CSf] := 0.2000
Mu[Sim] := 1.00 Coefficient of Variation[Sim] := 0.2000
Mu[Simf] := 1.00    Coefficient of Variation[Simf] := 0.2000
(M)u step size: 0.50
(S)igma / mu step size: 0.500
(B)urn in, # generations: 20000
(P)eriod of sampling from the Markov chain: 20
(G)enerations to be sampled: 10000
(F)ull output of the chain: False
(T)uning depth maximum: 8

Enter a letter to change a parameter, q to quit, or RETURN to go on:
{RETURN}
```

### 4) BAGEL is off and running

```
PGRP-SC1b(GH07464)
Constructing Comparison Matrix:
       CS   CSf   Sim  Simf
   CS  1.29  1.48 0.67  0.53 0.15 0.63 0.35

   CSf  1.05 1.38            0.35 0.40 0.46

   Sim  7.27 1.88            1.29 0.55 1.14

  Simf      3.33 2.19  1.06 2.04

V/CV A.R.: 0.14...
```

Here BAGEL takes all of the available data for the first gene (PGRP-SC1b) and starts to run the markov chain. The line at the bottom (V/CV A.R. . . ) shows BAGEL��s attempts to tune the jump size in the chain to produce an appropriate acceptance rate (between 0.15 and 0.5). Once the chain has run successfully, BAGEL shows you the results:

```
Mu Acceptance ratio: 0.40        True
Coefficient of Variation Acceptance ratio: 0.40     True
Mu SS: 0.500  V/CV SS: 0.1498
PGRP-SC1b(GH07464),   CS: 0.32    1.00  0.34
PGRP-SC1b(GH07464),   CSf: 0.35    1.17  0.35
PGRP-SC1b(GH07464),   Sim: 0.39    2.37  0.38
PGRP-SC1b(GH07464),   Simf: 0.40   2.85  0.38
Density of Best Likelihood: 6.496307351635393e-06
Logmean MuStepSize: 0.50
Logmean VCVStepSize: 0.15
Genes Examined: 1
```

The *D. simulans* female expression level for PGRP-SC1b, for instance, is estimatdd to be 2.85, with a 95% credible interval between 2.45 and 3.23.  All expression levels are estimated in comparison to the sample with the lowest expression level (here, male *D. melanogaster*).

BAGEL then moves on to the next gene . . .

```
BcDNA:LD09936(LD09936)
Constructing Comparison Matrix:
       CS   CSf   Sim  Simf
   CS  1.01  8.9120.68  0.42 0.47 0.42 0.51

   CSf  0.08 0.07            2.08 1.55 1.39

   Sim  1.94 1.35           36.0649.9537.74

  Simf      0.67 0.51  0.02 0.01


Mu Acceptance ratio: 0.25        True
Coefficient of Variation Acceptance ratio: 0.15     True
Mu SS: 0.500  V/CV SS: 0.1498
BcDNA:LD09936(LD09936),     CS: 1.06    11.35  0.96
BcDNA:LD09936(LD09936),     CSf: 0.94    1.78   1.24
BcDNA:LD09936(LD09936),     Sim: 1.53   22.10   1.16
BcDNA:LD09936(LD09936),     Simf: 0.80   1.00  1.01
Density of Best Likelihood: 4.584386846285838e-11
Logmean MuStepSize: 0.50
Logmean VCVStepSize: 0.15
Genes Examined: 2


CG12200(LD30246)
Constructing Comparison Matrix:
       CS   CSf   Sim  Simf
   CS      0.11 0.15  0.87 0.99 1.04

   CSf  5.76 4.34            9.42 7.04 5.92

   Sim  0.85 1.12            1.31 2.06

  Simf      0.12 0.19  0.55 0.80

V/CV A.R.: 0.07...
Mu Acceptance ratio: 0.20        True
Coefficient of Variation Acceptance ratio: 0.32     True
Mu SS: 0.500  V/CV SS: 0.0299
CG12200(LD30246),   CS: 0.20    1.28  0.21
CG12200(LD30246),   CSf: 0.48    7.13  0.45
CG12200(LD30246),   Sim: 0.28    1.43  0.33
CG12200(LD30246),   Simf: 0.19   1.00  0.19
Density of Best Likelihood: 4.516871022806865e-01
Logmean MuStepSize: 0.50
Logmean VCVStepSize: 0.09
Genes Examined: 3


qtc(SD06355)
Constructing Comparison Matrix:
       CS   CSf   Sim  Simf
   CS  0.93  3.64 2.06  0.90 1.13 0.82 0.96

   CSf  0.30 0.39            0.85 0.88 0.97

   Sim  0.89 1.09            1.83 2.61
  Simf      1.05 0.85  0.52 0.28


Mu Acceptance ratio: 0.22        True
Coefficient of Variation Acceptance ratio: 0.18     True
Mu SS: 0.500  V/CV SS: 0.0875
qtc(SD06355),  CS: 0.19    2.59  0.18
qtc(SD06355),  CSf: 0.18    1.00  0.18
qtc(SD06355),  Sim: 0.19    2.65  0.19
qtc(SD06355),  Simf: 0.18   1.11  0.17
Density of Best Likelihood: 1.481589125998954e+03
Logmean MuStepSize: 0.50
Logmean VCVStepSize: 0.09
Genes Examined: 4


Press RETURN when ready.
```

And it��s done.

## Output

### 1) BAGEL results

BAGEL results output is a tab-delimited text file with estimates for each expression node, additions for 95% upper- bounds, and subtractions for 95% lower-bounds. These are formatted such that creation of an EXCEL column or bar graph should be very easy. Other columns let you know of the Mu and Variance/CV step acceptance rate as well as an ��Acceptable?�� column which discloses whether BAGEL found acceptable acceptance rates (between 0.15 and 0.5) for both parameters.

The results output from the example datafile looks like this:

```
gene name
Unique ID           Common Name         CS      CSf     Sim     Simf    ...
PGRP-SC1b           GH07464             1       1.17    2.37    2.85    ...
BcDNA:LD09936       LD09936             11.35   1.78    22.10   1       ...
CG12200             LD30246             1.28    7.13    1.43    1       ...
qtc                 SD06355             2.59    1       2.65    1.11    ...
...


(-)97.5%[CS]        (-)97.5%[CSf]       (-)97.5%[Sim]       (-)97.5%[Simf]
0.32146             0.34871             0.38685             0.39713       ...
1.055               0.94196             1.52557             0.79782       ...
0.20477             0.48235             0.28134             0.18666       ...
0.18859             0.17951             0.18848             0.17595       ...
...
```

### 2) BAGEL *P*-values

*P*-values for whether expression level is greater in one sample than another are saved as separate files according to expression node names. You should be aware that these *P*-values are not "corrected" in any way for multiple tests; you should have the appropriate scientific skepticism and look carefully for corroborating biologically consistent evidence. Furthermore, the *P*-value is only as precise as the number of samples taken from the chain. If 10,000 samples were taken, then a *P*-value of zero really means that in none of the samples taken from the chain did the comparison hold, and so the *P*-value is really *P* < 0.0001. The *P*-value text files are on the same drive in the same folder as your input file. The output filename will be the same as the original Datafilename, but will have the characters ��.pvalue�� appended.

The *P*-value from the example datafile looks like this:

```
Unique ID            Common Name   P(CS>CSf)     P(CS>Sim)           ...
PGRP-SC1b            GH07464       0.2543        0.0001              ...
BcDNA:LD09936        LD09936       1             0                   ...
CG12200              LD30246       0             0.1591              ...
qtc                  SD06355       1             0.3336              ...
...
```

If BAGEL is halted mid-run, a number of files with ".BAM" suffixes may be found in the BAGEL folder. These files temporarily store the sampled Mu values and may be summarily deleted.

## Comparison of error models

Figure 5 shows the results of BAGEL analysis of the four genes under different error models. Each error model was run twice in order to assess variation between independent runs within a model. With the level of replication shown in Figure 4, changing between additive and multiplicative error models, and constraining variances or CVs makes very little difference to the results. For three of the genes, choosing unconstrained variances also has little effect on the results, aside from increasing the width of the 95% credible intervals. However, for BcDNA:LD09936, choosing an unconstrained variance model does have a significant impact on the relative expression levels inferred by the markov chain. This result, given the well replicated nature of these experiments, underscores the need to be cautious when choosing an unconstrained variance/CV model.

## Bug Reports

Please report bugs or suggestions for changes to Jeffrey Townsend (Jeffrey.Townsend@yale.edu).

## Copyright information

1620, (510) 643-7201, for commercial licensing opportunities. Created by *Jeffrey Townsend*, *Department of Plant and Microbial Biology*, University of California, Berkeley.

IN NO EVENT SHALL REGENTS BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, INCLUDING LOST PROFITS, ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN IF REGENTS HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

REGENTS SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE AND ACCOMPANYING DOCUMENTATION, IF ANY, PROVIDED HEREUNDER IS PROVIDED "AS IS". REGENTS HAS NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

## More Information

This tutorial was compiled by Zhang Zhang. Its contents were constructed from the original tutorial by Colin Meiklejohn, the original publication describing BAGEL (Townsend and Hartl 2002), a subsequent report elaborating the error models (Townsend 2003), Mark Voorhies�� command-line help file, and Jeffrey Townsend��sreadme file that accompanies the BAGEL download. Questions regarding the use of BAGEL and this tutorial can be addressed to Jeffrey Townsend (Jeffrey.Townsend@yale.edu).

## References

**1) Original publication describing BAGEL:**

Townsend JP, Hartl DL, 2002. Bayesian analysis of gene expression levels: statistical quantification of relative mRNA level across multiple strains or treatments. *Genome Biol.* **3(12):** RESEARCH0071.

**2) Extension of the original additive model to include multiplicative errors and constrained CVs, as well as power tests of the models in BAGEL are described in:**

Townsend JP, 2004. Resolution of large and small differences in gene expression using models for the Bayesian analysis of gene expression levels and spotted DNA microarrays. *BMC Bioinformatics* **5**: 54.

**3) Research reports that have used BAGEL and might provide ideas about post-BAGEL analyses of microarray data include:**

Aubin-Horth N, Letcher BH, Hofmann HA, 2005. Interaction of rearing environment and reproductive tatic on gene expression profiles in Atlantic salmon. *Journal of Heredity* 96(2): 1-18.

Lemos B, Meiklejohn CD, Hartl DL, 2004. Regulatory evolution across the protein interaction network. *Nature Genetics* 36(10): 1059-1060.

Ranz JM, Namgyal K, Gibson G, Hartl DL, 2004. Anomalies in the expression profile of interspecific hybrids of *Drosophila melanogaster* and *Drosophila simulans*. Genome Research 14: 373-379.

Grozinger CM, Sharabash NM, Whitfield CW, Robinson, GE, 2003. Pheromone-mediated gene expression in the honey bee brain. *PNAS* 100 (suppl 2):14519-14525.

Whitfield CW, Cziko A, Robinson, GE, 2003. Gene expression profiles in the brain predict behavior in individual bees. *Science* 302:296-299.

Silverman N, Zhou R, Erlich RL, Hunter M, Bernstein E, Schneider D, Maniatis T, 2003. Immune activation of NF-kappaB and JNK requires Drosophila TAK1. *J. Biol. Chem.* 278(49):48928-48934.

Ranz JM, Castillo-Davis CI, Meiklejohn CD, Hartl DL, 2003. Sex-dependent gene expression and evolution of the Drosophila transcriptome. *Science* 300(5626):1742-5.

Meiklejohn CD, Parsch J, Ranz JM, Hartl DL, 2003. Rapid evolution of male-biased gene expression in Drosophila. *PNAS* 100(17):9894-9899.

Townsend JP, Cavalieri D, Hartl DL, 2003. Population genetic variation in genome-wide gene expression. *Mol. Biol. Evol.* 20(6):955-63.

**4) Introductory reviews on the design of Bayesian multifactorial microarray experiments:**

Townsend, J.P., 2003. Multifactorial experimental design and the transitivity of ratios with spotted DNA microarrays. *BMC Genomics* 4: 41.

Townsend, J.P., and J.W. Taylor, 2005. Designing experiments using spotted microarrays to detect gene regulation differences within and among species. In ��Methods in molecular evolution: producing the biochemical data��, Zimmer, E., ed., Academic Press: *Methods in Enzymology* 224.