EBSeq: An R package for differential expression analysis using RNA-seq data

Graphical User Interface Manual

Ning Leng, Haolin Xu and Christina Kendziorski

Table of Contents

1. Installation

The empirical Bayes model in Leng *et al.*, 2013 is implemented in an R package called EBSeq (biostat.wisc.edu/~kendzior/EBSEQ). This manual is a guideline for using the add-on EBSeq interface functions, which will allow a user to run EBSeq without directly using R. Files can be uploaded in an xls, xlsx, or csv format.

The EBSeq interface requires installing the EBSeq package, and also requires the installation of the C packge GTK+ and R package RGtk2. Both packages are available online for free. Detailed installation instructions are specified below.

1.1 Install EBSeq

The EBSeq package is available at biostat.wisc.edu/~kendzior/EBSEQ

- Linux and Mac users please download the EBSeq.tar.gz file
- Windows users please download the EBSeq.zip file.

Download EBSeq package and EBSeq_Interface.zip to a folder. We will refer to this folder later in this manual as *YOUR_PATH*. Unzip the .R files in EBSeq_Interface.zip into *YOUR_PATH*.

To install EBSeq in R:

Start R and type:

```
    install.packages("gplots")

    install.packages("blockmodeling")

    install.packages("YOUR_PATH/EBSeq_1.1.6.tar.gz",
repos=NULL, type="source")
```

1.2 Install Gtk+ and RGtk2 for user interface

Gtk+ downloads are available at http://www.gtk.org/download/index.php

Linux Users:

The GTK+2.X version is suggested for simple installation on linux. GTK+3.X will require higher versions of libraries. More detailed instructions can be found at gtk.org.

1) GTK+:
To start, type in your bash shell:

```
    sudo apt-get install libgtk2.0-dev
    sudo apt-get install glade
```

To check, type

```
    gtk-demo
```

2) RGtk2:
Start up R, type

```
    install.packages("RGtk2")
```

Choose a binary, and installation will start automatically.

Next type:

```
library(RGtk2)
```

Windows Users:

1) GTK+
   a. Download and unzip the all-in-one GTK bundle of any version
   b. Copy the complete bin/ folder of the bundle to *YOUR_PATH*
   c. Open cmd.exe and set path into the bin/ folder in *YOUR_PATH*
   d. Run commends to install

```
pkg-config --cflags gtk+-2.0
```

   e. Check to see if the demo works

```
gtk-demo
```

2) RGtk2
   f. Open R, set the working directory to *YOUR_PATH* (where bin/ is copied to)
   g. Type

```
install.packages("RGtk2")
library(RGtk2)
```

   h. If there are error messages, restart R. Make sure you're set to the right path, and try installing RGtk2 again.

Mac Users:

Start R and type:

```
install.packages("RGtk2")
```

Choose your favorite country/school binary website and R will automatically install Gtk+2.X on your MAC. Then type:

```
library(RGtk2)
```

These steps should be sufficient for RGtk2 to work on a Mac.

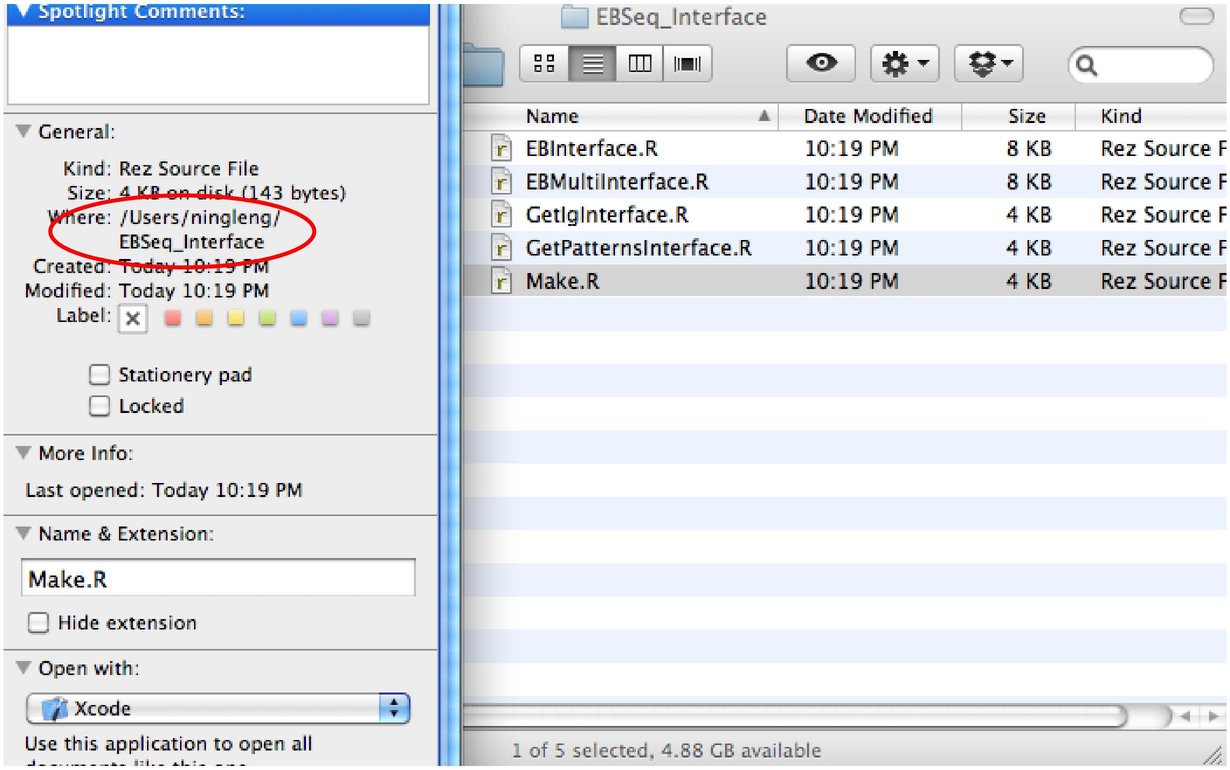
2. Preparation for the analysis

In R, type:

```
setwd("YOUR_PATH")
source("Make.R")
```

Example with screenshot:

As shown on the next page, my *YOUR_PATH* directory is: /Users/ningleng/EBSeq_Interface/



Spotlight Comments:

General:

Kind: Rez Source File
Size: 4 KB on disk (143 bytes)
Where: /Users/ningleng/EBSeq_Interface
Created: Today 10:19 PM
Modified: Today 10:19 PM
Label:

□ Stationery pad
□ Locked

More Info:

Last opened: Today 10:19 PM

Name & Extension:

Make.R

□ Hide extension

Open with:

Xcode

Use this application to open all

EBSeq_Interface

Name	Date Modified	Size	Kind
EBInterface.R	10:19 PM	8 KB	Rez Source F
EBMultiInterface.R	10:19 PM	8 KB	Rez Source F
GetIgInterface.R	10:19 PM	4 KB	Rez Source F
GetPatternsInterface.R	10:19 PM	4 KB	Rez Source F
Make.R	10:19 PM	4 KB	Rez Source F

1 of 5 selected, 4.88 GB available

So in my case, I typed:

```
setwd("/Users/ningleng/EBSeq_Interface/")
source("Make.R")
```

3. Gene level DE analysis – two conditions

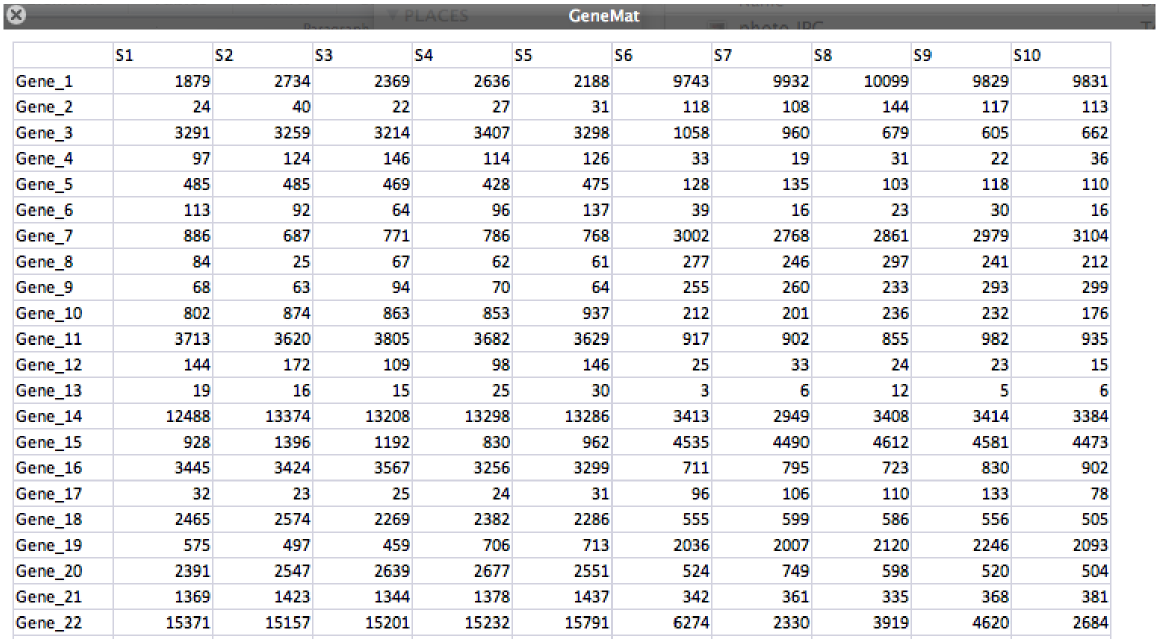
Input requirement:

The input file formats supported by EBSeq are .csv, .xls, or .xlsx.
In your input file, the rows should be the genes and the columns should be the samples.
In other words, your first row stores the sample names and the first column shows your gene names.

Note: This example does not use isoform level expression data.
An example of isoform expression analysis is shown in Section 4.

Example data set in .xls format:

GeneMat.xls (as shown on the next page)



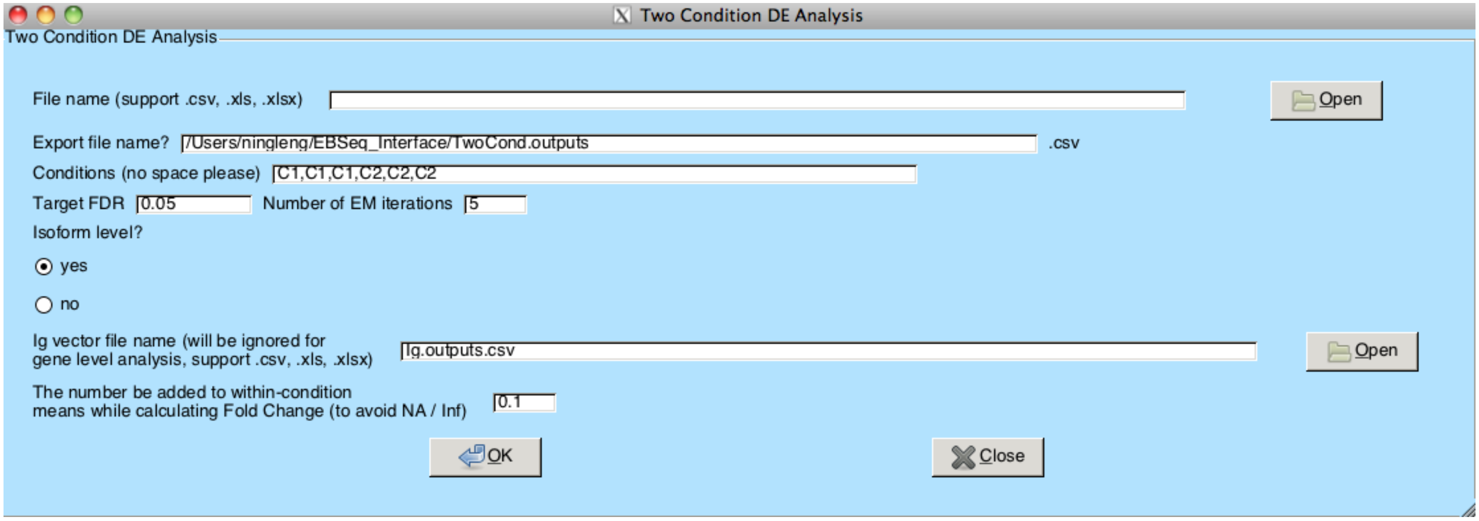
GeneMat

	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
Gene_1	1879	2734	2369	2636	2188	9743	9932	10099	9829	9831
Gene_2	24	40	22	27	31	118	108	144	117	113
Gene_3	3291	3259	3214	3407	3298	1058	960	679	605	662
Gene_4	97	124	146	114	126	33	19	31	22	36
Gene_5	485	485	469	428	475	128	135	103	118	110
Gene_6	113	92	64	96	137	39	16	23	30	16
Gene_7	886	687	771	786	768	3002	2768	2861	2979	3104
Gene_8	84	25	67	62	61	277	246	297	241	212
Gene_9	68	63	94	70	64	255	260	233	293	299
Gene_10	802	874	863	853	937	212	201	236	232	176
Gene_11	3713	3620	3805	3682	3629	917	902	855	982	935
Gene_12	144	172	109	98	146	25	33	24	23	15
Gene_13	19	16	15	25	30	3	6	12	5	6
Gene_14	12488	13374	13208	13298	13286	3413	2949	3408	3414	3384
Gene_15	928	1396	1192	830	962	4535	4490	4612	4581	4473
Gene_16	3445	3424	3567	3256	3299	711	795	723	830	902
Gene_17	32	23	25	24	31	96	106	110	133	78
Gene_18	2465	2574	2269	2382	2286	555	599	586	556	505
Gene_19	575	497	459	706	713	2036	2007	2120	2246	2093
Gene_20	2391	2547	2639	2677	2551	524	749	598	520	504
Gene_21	1369	1423	1344	1378	1437	342	361	335	368	381
Gene_22	15371	15157	15201	15232	15791	6274	2330	3919	4620	2684

In R, type:

```
EBInterface()
```

A window will pop up (shown below):



Two Condition DE Analysis

Two Condition DE Analysis

File name (support .csv, .xls, .xlsx)

Export file name? /Users/ningleng/EBSeq_Interface/TwoCond.outputs .csv

Conditions (no space please) C1,C1,C1,C2,C2,C2

Target FDR 0.05 Number of EM iterations 5

Isoform level?

● yes

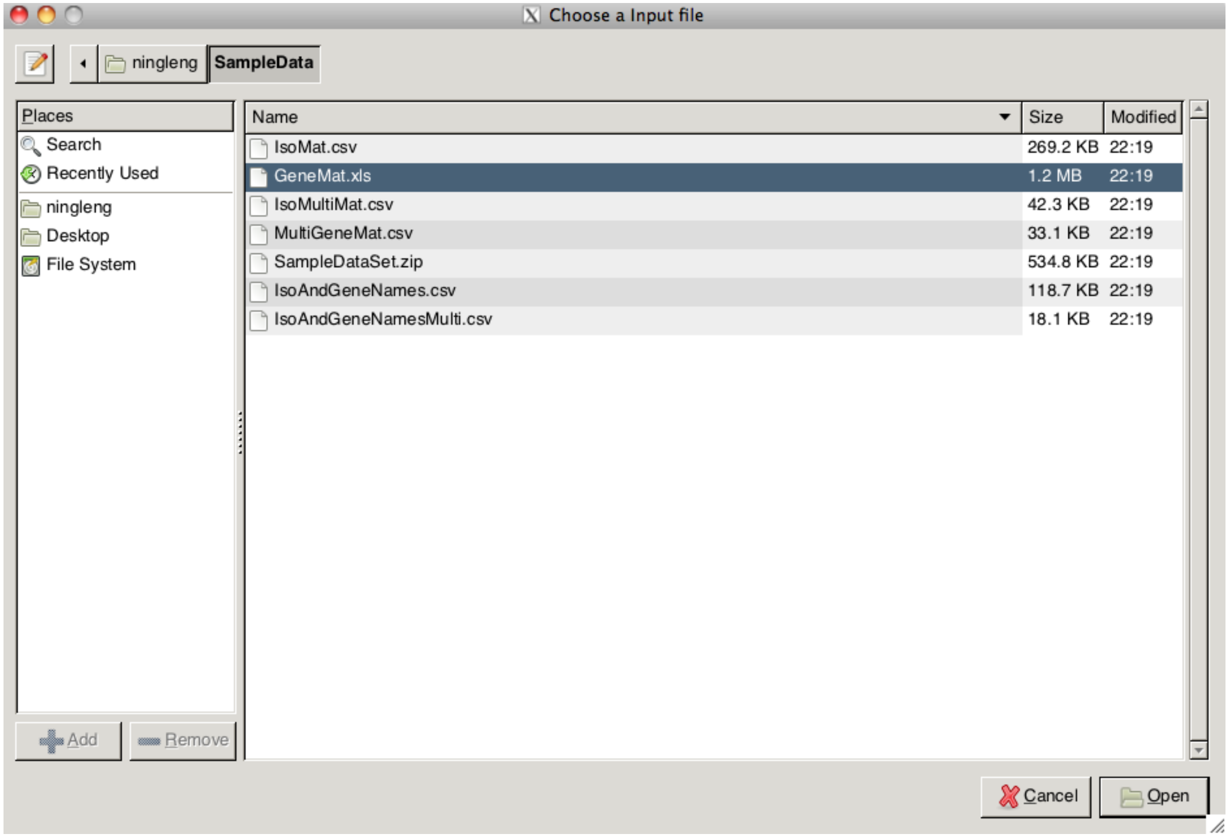
○ no

Ig vector file name (will be ignored for gene level analysis, support .csv, .xls, .xlsx) Ig.outputs.csv

The number be added to within-condition means while calculating Fold Change (to avoid NA / Inf) 0.1

To select the input file, click the upper right “Open” button. A window will pop up and ask for an input file (shown on the next page).

Select GeneMat.xls and click open.

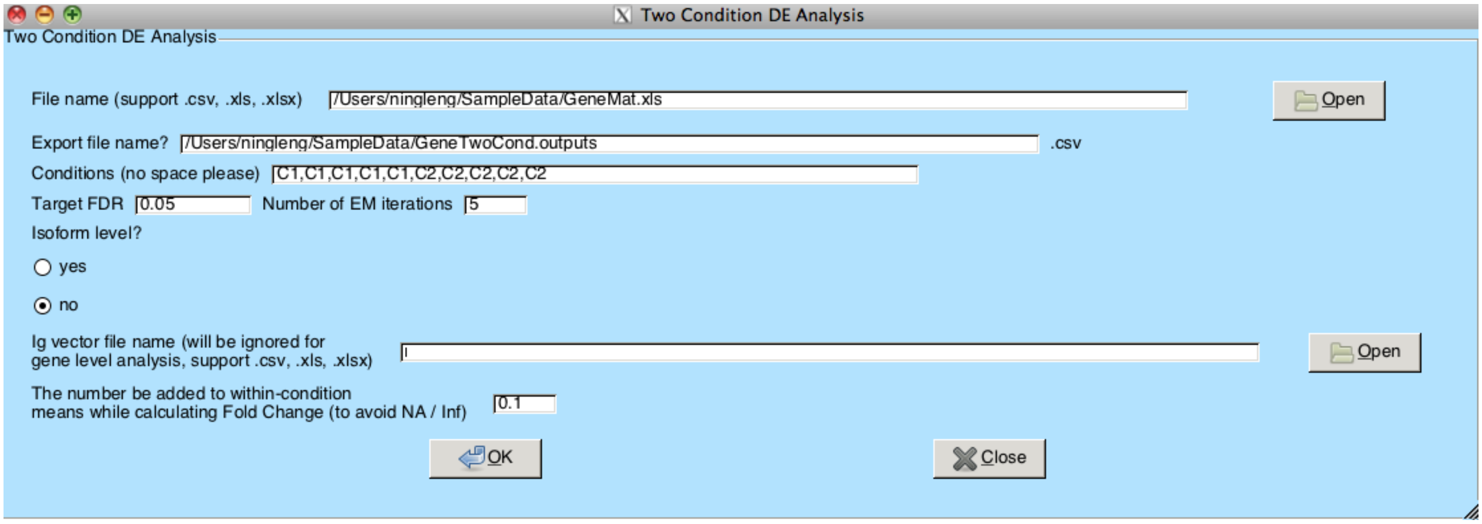


Name	Size	Modified
IsoMat.csv	269.2 KB	22:19
GeneMat.xls	1.2 MB	22:19
IsoMultiMat.csv	42.3 KB	22:19
MultiGeneMat.csv	33.1 KB	22:19
SampleDataSet.zip	534.8 KB	22:19
IsoAndGeneNames.csv	118.7 KB	22:19
IsoAndGeneNamesMulti.csv	18.1 KB	22:19

Next, a user can customize:

i. The input file
ii. The export file (output) name
iii. Conditions of the samples
iv. Target false discovery rate (FDR); the default is 0.05
v. Number of EM iterations; the default of 5 is a good start, but more may be required
vi. Whether gene or isoform level analysis is of interest
vii. The name for the *Ig* vector file
viii. A number *d* to be added to the condition means to avoid invalid entries (NA or ∞) while calculating FC. The default is $d = 0.1$. The formula to calculate FC is $\frac{\bar{X}_{C1}+d}{\bar{X}_{C2}+d}$.

On the next page, there is a screenshot of my example for GeneMat.xls. Note: I chose “no” for isoform levels, so no *Ig* input is required. The default directory to save the output is *YOUR_PATH.*



File name (support .csv, .xls, .xlsx) /Users/ningleng/SampleData/GeneMat.xls

Export file name? /Users/ningleng/SampleData/GeneTwoCond.outputs .csv

Conditions (no space please) C1,C1,C1,C1,C1,C2,C2,C2,C2,C2

Target FDR 0.05 Number of EM iterations 5

Isoform level?

○ yes

● no

Ig vector file name (will be ignored for gene level analysis, support .csv, .xls, .xlsx)

The number be added to within-condition means while calculating Fold Change (to avoid NA / Inf) 0.1

Explaining the Outputs

Four files will be generated for my GeneMat.xls example:

(1) GeneTwoCond.outputs.csv:
Columns are posterior probability of being DE, Fold Change (*d* is added to both the numerator and denominator), posterior Fold Change, and library size adjusted gene expressions. Rows are the genes in the same order as the input file.

(2) GeneTwoCond.outputs.SortedByPPDE.csv:
Columns are the same as in (1). Genes are sorted decreasingly by PPDE.

(3) GeneTwoCond.outputs.SortedByPPDE.FilteredByFDR.csv:
Columns are the same as in (1) and (2). Only genes with PPDE >= 1 - Target_FDR are listed.

(4) GeneTwoCond.outputs.rda:
The R data file containing all statistical objects in the run.

4. Isoform level DE analysis – two conditions

4.1 Get *Ig* vector

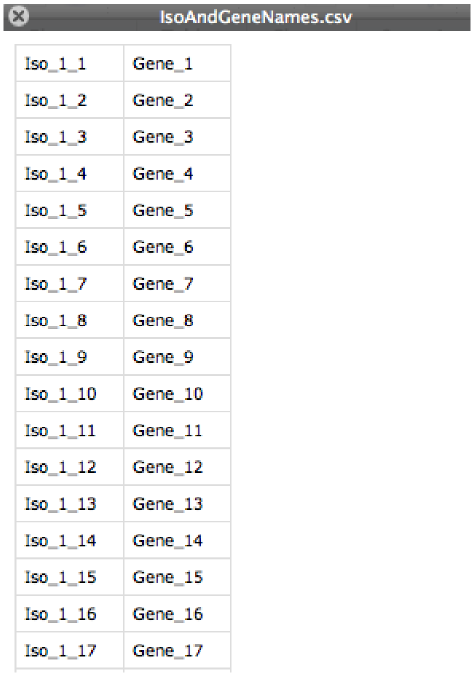
For isoform level analysis, an *Ig* vector is required (see Leng *et al*., 2013, or the EBSeq vignette for details on *Ig*). If you have the *Ig* vector file generated from RSEM, please ignore this subsection.

Input requirement:

Again, csv, xls, or xlsx files are accepted. The first column specifies the isoform names and the second column specifies the corresponding gene names.

Example data set:

IsoAndGeneNames.csv



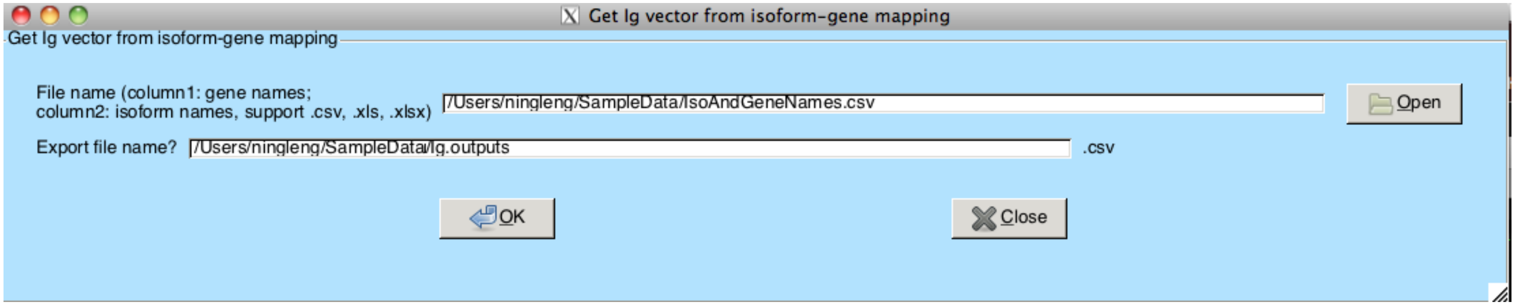
IsoAndGeneNames.csv

Iso_1_1	Gene_1
Iso_1_2	Gene_2
Iso_1_3	Gene_3
Iso_1_4	Gene_4
Iso_1_5	Gene_5
Iso_1_6	Gene_6
Iso_1_7	Gene_7
Iso_1_8	Gene_8
Iso_1_9	Gene_9
Iso_1_10	Gene_10
Iso_1_11	Gene_11
Iso_1_12	Gene_12
Iso_1_13	Gene_13
Iso_1_14	Gene_14
Iso_1_15	Gene_15
Iso_1_16	Gene_16
Iso_1_17	Gene_17

In R, type:

```
GetIgInterface()
```

A user interface will pop up (as shown below):



Get Ig vector from isoform-gene mapping

Get Ig vector from isoform-gene mapping

File name (column1: gene names;
column2: isoform names, support .csv, .xls, .xlsx) /Users/ningleng/SampleData/IsoAndGeneNames.csv

Export file name? /Users/ningleng/SampleData/Ig.outputs .csv

Outputs

A .csv file containing the *Ig* vector isoform level DE analysis will be created in *YOUR_PATH*.

4.2 DE analysis

Input requirement:

The *Ig* vector file from Section 4.1 or RSEM rsem-generate-ngvector function (http://deweylab.biostat.wisc.edu/rsem/rsem-generate-ngvector.html).

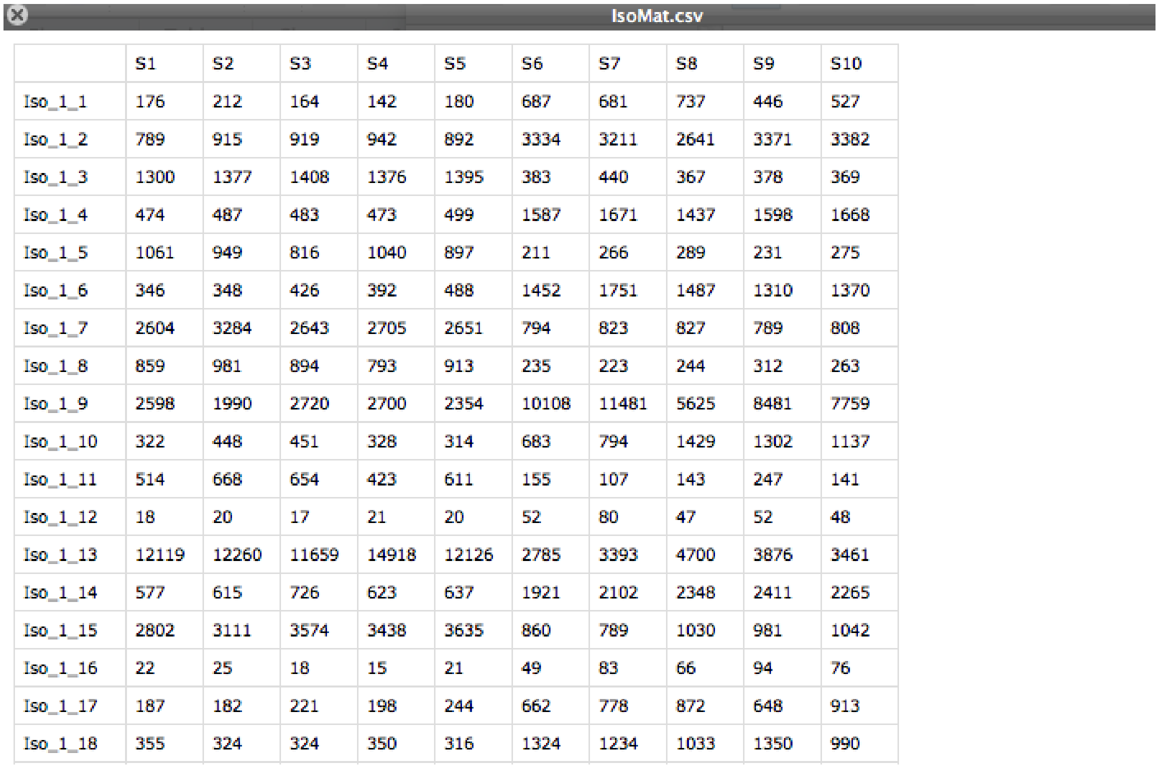
The data input could be .csv, .xls, or .xlsx files.

Rows are isoforms and columns are samples.

- The first row shows the sample names
- The first column shows the isoform names

Example data set

IsoMat.csv

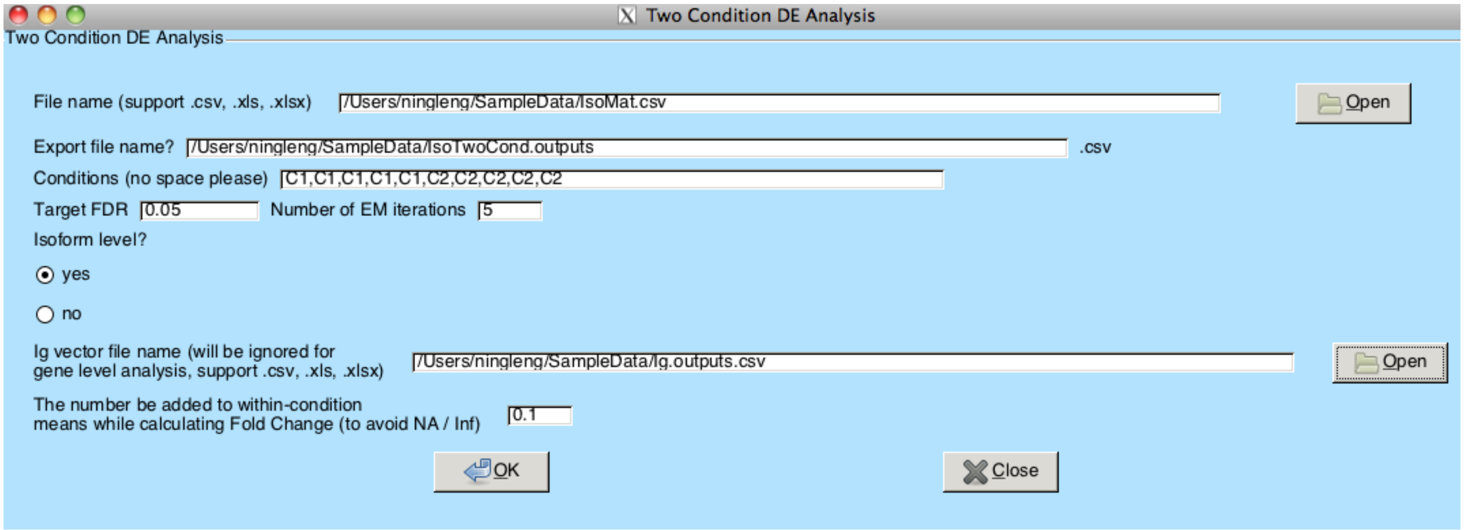


	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
Iso_1_1	176	212	164	142	180	687	681	737	446	527
Iso_1_2	789	915	919	942	892	3334	3211	2641	3371	3382
Iso_1_3	1300	1377	1408	1376	1395	383	440	367	378	369
Iso_1_4	474	487	483	473	499	1587	1671	1437	1598	1668
Iso_1_5	1061	949	816	1040	897	211	266	289	231	275
Iso_1_6	346	348	426	392	488	1452	1751	1487	1310	1370
Iso_1_7	2604	3284	2643	2705	2651	794	823	827	789	808
Iso_1_8	859	981	894	793	913	235	223	244	312	263
Iso_1_9	2598	1990	2720	2700	2354	10108	11481	5625	8481	7759
Iso_1_10	322	448	451	328	314	683	794	1429	1302	1137
Iso_1_11	514	668	654	423	611	155	107	143	247	141
Iso_1_12	18	20	17	21	20	52	80	47	52	48
Iso_1_13	12119	12260	11659	14918	12126	2785	3393	4700	3876	3461
Iso_1_14	577	615	726	623	637	1921	2102	2348	2411	2265
Iso_1_15	2802	3111	3574	3438	3635	860	789	1030	981	1042
Iso_1_16	22	25	18	15	21	49	83	66	94	76
Iso_1_17	187	182	221	198	244	662	778	872	648	913
Iso_1_18	355	324	324	350	316	1324	1234	1033	1350	990

In R, type:

```
EBInterface()
```

A user interface will pop up (as shown on the next page).



Two Condition DE Analysis

File name (support .csv, .xls, .xlsx) /Users/ningleng/SampleData/IsoMat.csv

Export file name? /Users/ningleng/SampleData/IsoTwoCond.outputs .csv

Conditions (no space please) C1,C1,C1,C1,C1,C2,C2,C2,C2,C2

Target FDR 0.05 Number of EM iterations 5

Isoform level?

● yes

○ no

Ig vector file name (will be ignored for gene level analysis, support .csv, .xls, .xlsx) /Users/ningleng/SampleData/Ig.outputs.csv

The number be added to within-condition means while calculating Fold Change (to avoid NA / Inf) 0.1

As in the gene level analysis, a user can customize:

i. The input file
ii. The export file (output) name
iii. Conditions of the samples
iv. Target false discovery rate (FDR); the default is 0.05
v. Number of EM iterations; the default of 5 is a good start, but more may be required
vi. Whether gene or isoform level analysis is of interest
vii. The name for the *Ig* vector file.
viii. A number *d* to be added to the condition means to avoid invalid entries (NA or ∞) while calculating FC. The default is *d* = 0.1. The formula to calculate FC is $\frac{\bar{X}^{C_1}+d}{\bar{X}^{C_2}+d}$.

Explaining the Outputs

Four files will be generated:

(1) IsoTwoCond.outputs.csv:
Columns are posterior probability of being DE, Fold Change (*d* is added to both the numerator and denominator), posterior Fold Change, and library size adjusted isoform expressions. Rows are the isoforms in the same order as the input file.

(2) IsoTwoCond.outputs.SortedByPPDE.csv:
Columns are the same as in (1). Isoforms are sorted decreasingly by PPDE.

(3) IsoTwoCond.outputs.SortedByPPDE.FilteredByFDR.csv:
Columns are the same as in (1) and (2). Only isoforms with PPDE >= 1 - Target_FDR are listed.

(4) IsoTwoCond.outputs.rda:
The R data file containing all statistical objects in the run.

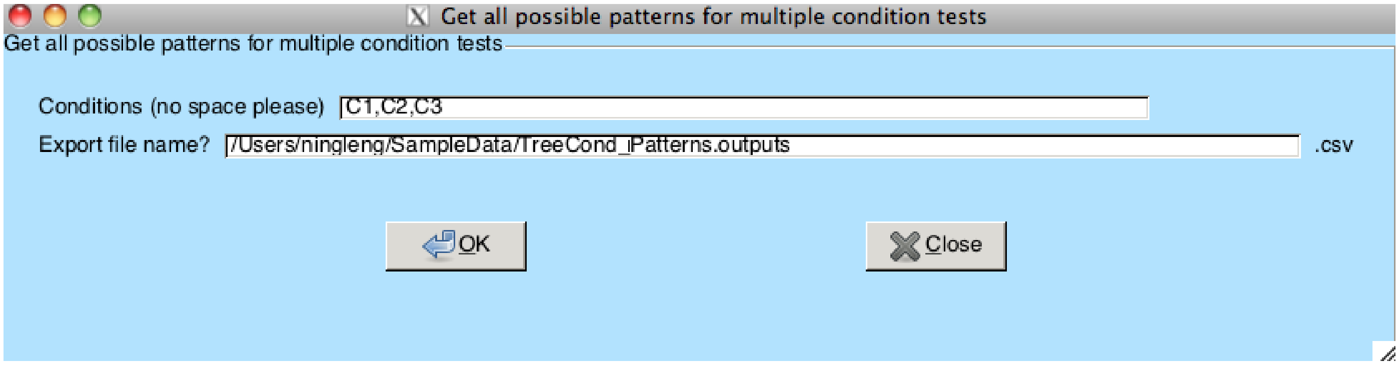
5. Gene level DE analysis – multiple conditions

5.1 Get all possible patterns

In R, type:

```
GetPatternsInterface()
```

A user interface will pop up (as shown below), fill in the condition names to be tested:

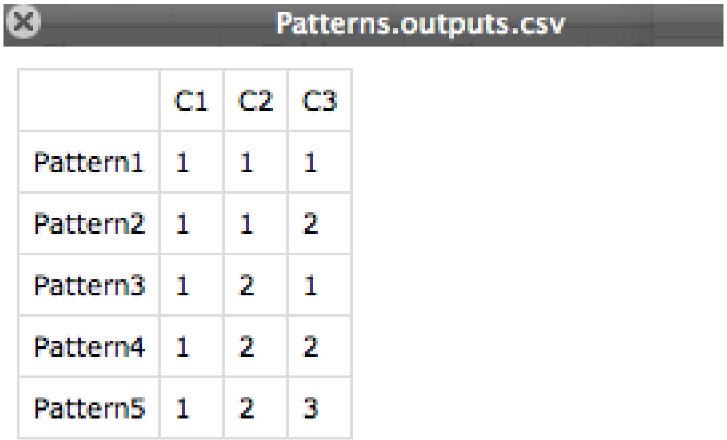

Get all possible patterns for multiple condition tests
Get all possible patterns for multiple condition tests
OK
Close

Conditions (no space please) C1,C2,C3

Export file name? /Users/ningleng/SampleData/TreeCond_Patterns.outputs .csv

Outputs:

A .csv file containing all possible patterns for multiple condition testing will be generated. For example:


Patterns.outputs.csv

	C1	C2	C3
Pattern1	1	1	1
Pattern2	1	1	2
Pattern3	1	2	1
Pattern4	1	2	2
Pattern5	1	2	3

The first pattern is C1 = C2 = C3
The second pattern is C1 = C2 ≠ C3

A user can delete any pattern that is not of interest directly from this .csv file before continuing to further analysis (e.g. try delete Pattern2 from this sample csv prior to the analysis in Section 5.2). When the number of conditions is greater than 4, it’s recommended to use a subset of the patterns (fewer than 10 patterns).

5.2 DE Analysis

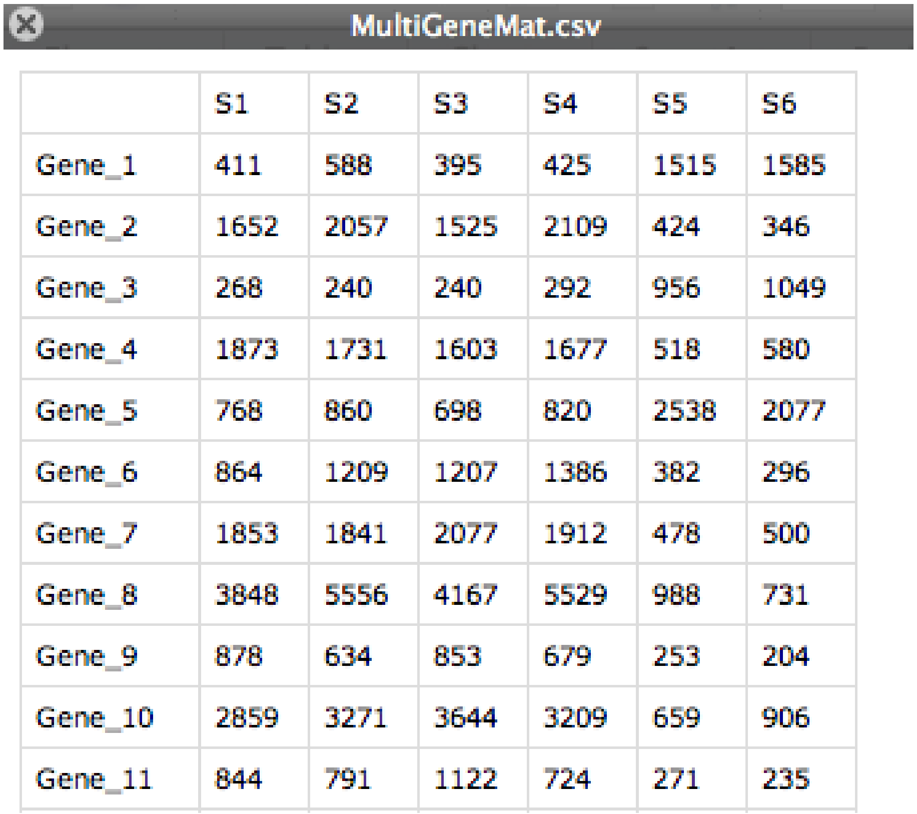
Input requirement:

The file contains patterns of interest.
Again, the input could be .csv, .xls, or .xlsx files.
Rows are isoforms and columns are samples.

- The first row stores the sample names
- The first column stores the isoform names

Example data set:

MultiGeneMat.csv



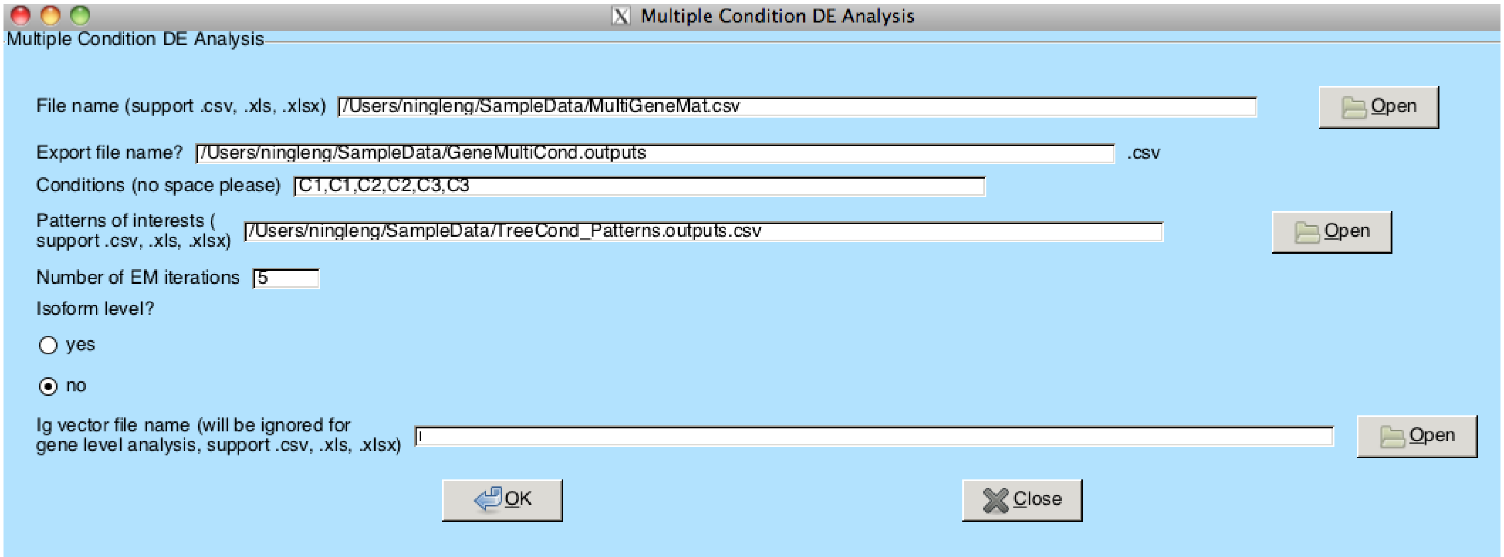
MultiGeneMat.csv

	S1	S2	S3	S4	S5	S6
Gene_1	411	588	395	425	1515	1585
Gene_2	1652	2057	1525	2109	424	346
Gene_3	268	240	240	292	956	1049
Gene_4	1873	1731	1603	1677	518	580
Gene_5	768	860	698	820	2538	2077
Gene_6	864	1209	1207	1386	382	296
Gene_7	1853	1841	2077	1912	478	500
Gene_8	3848	5556	4167	5529	988	731
Gene_9	878	634	853	679	253	204
Gene_10	2859	3271	3644	3209	659	906
Gene_11	844	791	1122	724	271	235

In R, type:

```
EBMultiInterface()
```

A user interface will pop up (shown on the next page).


Multiple Condition DE Analysis
Multiple Condition DE Analysis
Open
Open
Open
OK
Close

File name (support .csv, .xls, .xlsx) /Users/ningleng/SampleData/MultiGeneMat.csv

Export file name? /Users/ningleng/SampleData/GeneMultiCond.outputs .csv

Conditions (no space please) C1,C1,C2,C2,C3,C3

Patterns of interests (support .csv, .xls, .xlsx) /Users/ningleng/SampleData/TreeCond_Patterns.outputs.csv

Number of EM iterations 5

Isoform level?

○ yes

● no

Ig vector file name (will be ignored for gene level analysis, support .csv, .xls, .xlsx)

Again, user can customize:

i. Input file name
ii. The export file (output) name
iii. Conditions of the samples
iv. Patterns of interest. The output file from Section 5.1 can be used. When the number of conditions is greater than 4, using a subset of the patterns is recommended (fewer than 10 patterns).
v. Number of EM iterations; the default of 5 is a good start, but more may be required
vi. Whether gene or isoform level analysis is of interest
vii. The name for the *Ig* vector file.

Explaining the Outputs

Three files will be generated:

(1) GeneMultiCond.outputs.PP.Pattrns.csv:
Columns are posterior probability of being each pattern. Rows are the genes with the same order as input.

(2) GeneMultiCond.outputs.MAP.csv:
Column 1 shows the pattern with the highest posterior probability for each gene. The other columns are the library size adjusted gene expressions. Rows are the genes with the same order as input.

(3) GeneMultiCond.outputs.rda:
The R data file containing all statistical objects in this run.

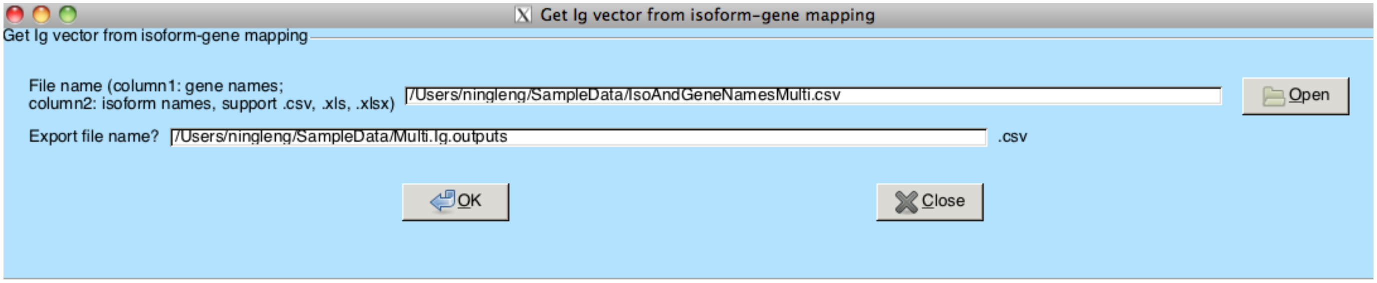
6. Isoform level DE analysis – multiple conditions

6.1 Get *Ig* vector

In R, type:

```
GetIgInterface()
```

A window will pop up (shown below); analysis proceeds as in Section 4.1.



Get Ig vector from isoform-gene mapping

Get Ig vector from isoform-gene mapping

File name (column1: gene names; column2: isoform names, support .csv, .xls, .xlsx) /Users/ningleng/SampleData/IsoAndGeneNamesMulti.csv

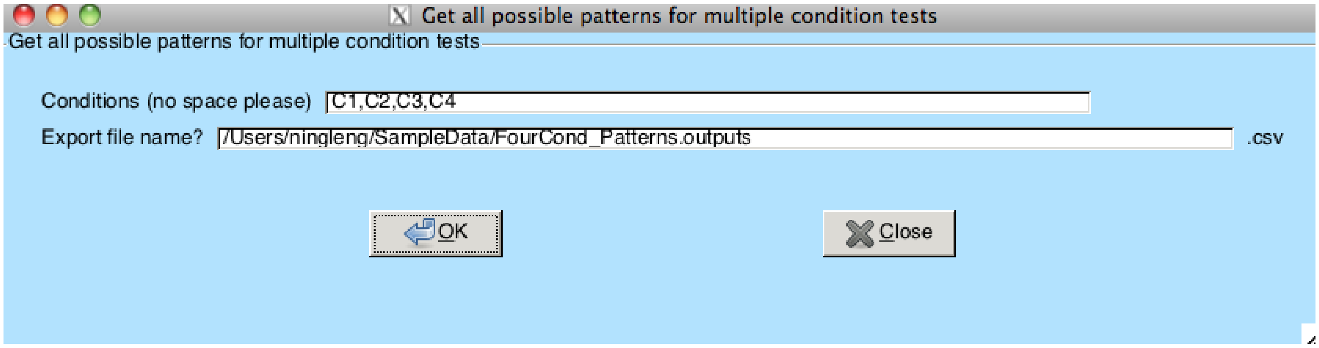
Export file name? /Users/ningleng/SampleData/Multi.Ig.outputs .csv

6.2 Get all possible patterns

In R, type:

```
GetPatternsInterface()
```

A window will pop up (shown below); analysis proceeds as in Section 5.1.



Get all possible patterns for multiple condition tests

Get all possible patterns for multiple condition tests

Conditions (no space please) C1,C2,C3,C4

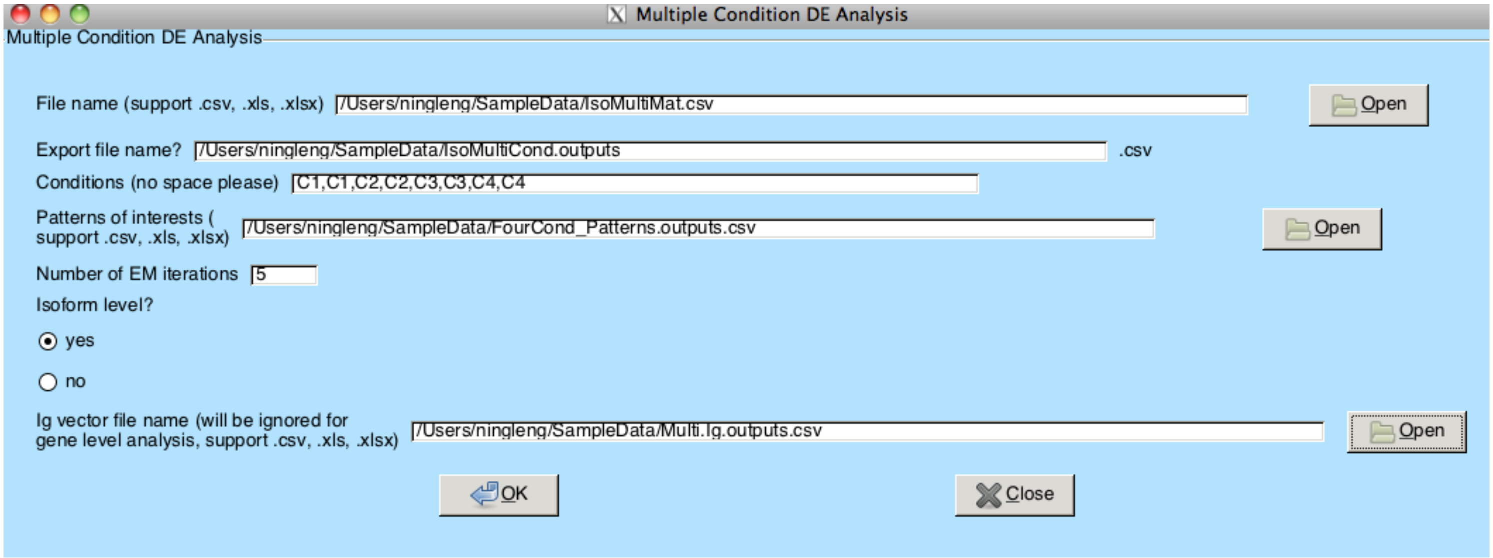
Export file name? /Users/ningleng/SampleData/FourCond_Patterns.outputs .csv

6.3 DE Analysis

In R, type:

```
EBMultiInterface()
```

Again, a window will pop up (shown on the next page); analysis proceeds as in Section 5.2.



Multiple Condition DE Analysis

File name (support .csv, .xls, .xlsx) /Users/ningleng/SampleData/IsoMultiMat.csv

Export file name? /Users/ningleng/SampleData/IsoMultiCond.outputs .csv

Conditions (no space please) C1,C1,C2,C2,C3,C3,C4,C4

Patterns of interests (support .csv, .xls, .xlsx) /Users/ningleng/SampleData/FourCond_Patterns.outputs.csv

Number of EM iterations 5

Isoform level?

● yes

○ no

Ig vector file name (will be ignored for gene level analysis, support .csv, .xls, .xlsx) /Users/ningleng/SampleData/Multi.Ig.outputs.csv

7. Problem shooting

More details of the EBSeq implementation can be found at http://www.biostat.wisc.edu/~kendzior/EBSEQ/EBSeq_Vignette.pdf.

If you have additional questions not addressed in this manual regarding the EBSeq interface, please see the Q&A section on the EBSeq website biostat.wisc.edu/~kendzior/EBSEQ, or contact us at nleng@wisc.edu.

Reference:

Leng, N., J.A. Dawson, J.A. Thomson, V. Ruotti, A.I. Rissman, B.M.G. Smits, J.D. Haag, M.N. Gould, R.M. Stewart, and C. Kendziorski. (2013). EBSeq: An empirical Bayes hierarchical model for inference in RNA-seq experiments, *Bioinformatics*, [e-pub ahead of print 21 February 2013] [Download].

Li, B., V. Ruotti, R.M. Stewart, J.A. Thomson, and C. Dewey. (2010). RNA-Seq gene expression estimation with read mapping uncertainty. *Bioinformatics 26*(4): 493-500.