

APPENDIX B: INTEGRATION OF GIMME

Basically, GSE data integration has three steps. These steps are demonstrated in Figure B.1. In the first step, GSE code for medulloblastoma is required because it is aimed to investigate this disease in the present study.

GSE62600 dataset (Hooper *et al.*, 2014) including 28 MB samples including 4 different subgroups and normal neural tissue samples is selected and downloaded from Gene Expression Omnibus (GEO) (Edgar, 2002) (See Figure B.2.).

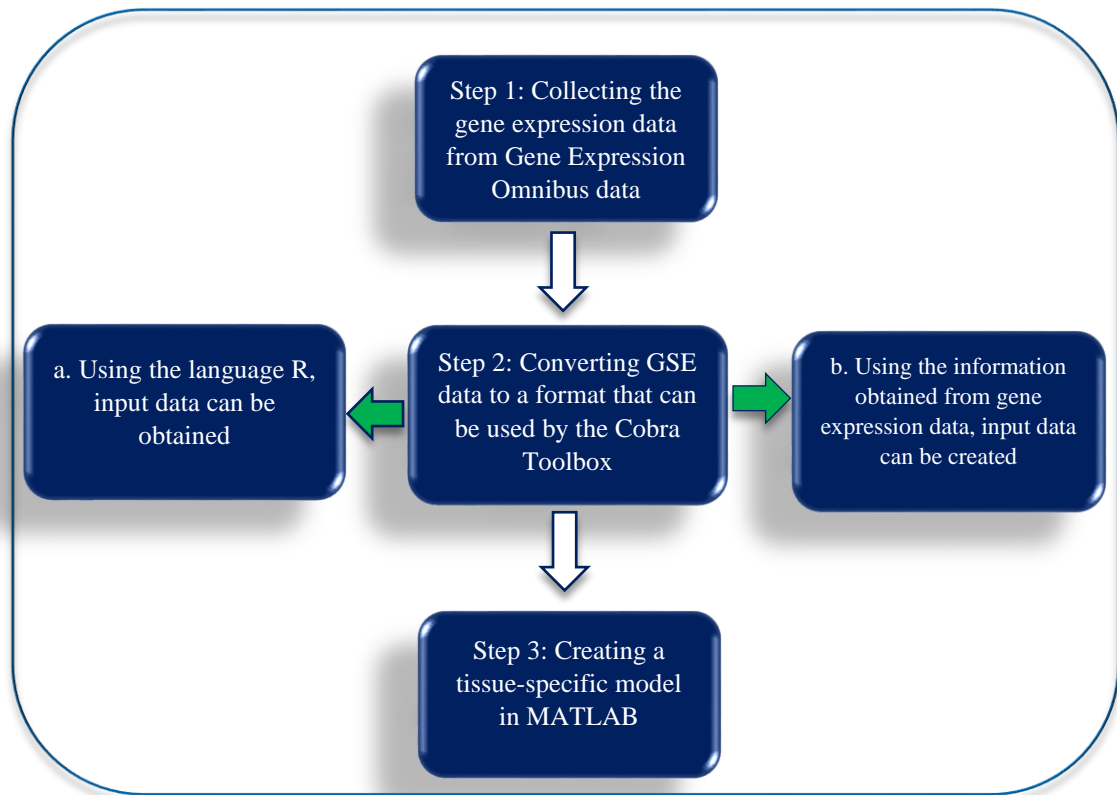


Figure B.1. Flow chart of GSE data Integration.

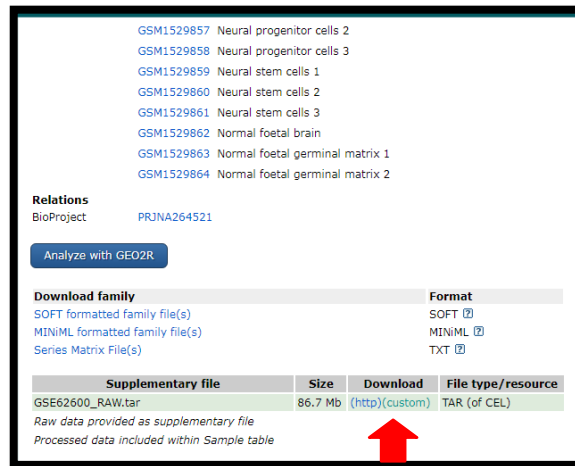


Figure B.2. GEO data downloaded from GEO website.

It is also possible to download GEO Data from GEO website and save it by using following function in MATLAB.

```
a = getgeodata('GSE62600', 'ToFile', 'GSE62600.txt')
```

Then, the data can be called by using geoseries read command. The output shows matrix data of series and Geo data named as 'a' is created in workplace.

```
a= geoseriesread('GSE62600.txt')
```

After downloading GSE62600, the expression values of 28 different MB transcriptome datasets collected from tumor biopsies of MB patients are obtained by using following command.

```
MB_samples=a.Data(:, 1:28)
```

Afterward, MB transcriptome datasets of GSE62600 and their gene expression values are demonstrated as a table on Cobra Toolbox.

Each column represents gene expression values of 28 individuals while first column includes genes.

The expression values can be also obtained directly from the Gene Omnibus website. After clicking GSM sample links shown in Figure B.3, the expression values of each patient can be seen as a table. For example, in Figure B.4., expression values of GSM335185 are demonstrated.

Platforms (1)	GPL96 [HG-U133A] Affymetrix Human Genome U133A Array
Samples (28)	GSM1529837 Medulloblastoma 2
Less...	GSM1529838 Medulloblastoma 6
	GSM1529839 Medulloblastoma 7
	GSM1529840 Medulloblastoma 8
	GSM1529841 Medulloblastoma 10
	GSM1529842 Medulloblastoma 11
	GSM1529843 Medulloblastoma 14
	GSM1529844 Medulloblastoma 15
	GSM1529845 Medulloblastoma 17
	GSM1529846 Medulloblastoma 18
	GSM1529847 Medulloblastoma 19
	GSM1529848 Medulloblastoma 21
	GSM1529849 Medulloblastoma 26
	GSM1529850 Medulloblastoma 28
	GSM1529851 Medulloblastoma 29
	GSM1529852 Medulloblastoma 30
	GSM1529853 Medulloblastoma 31
	GSM1529854 Medulloblastoma 32A
	GSM1529855 Medulloblastoma 37
	GSM1529856 Neural progenitor cells 1
	GSM1529857 Neural progenitor cells 2
	GSM1529858 Neural progenitor cells 3

Figure B.3. Required Samples.

In the second step, to integrate the GSE data by using GIMME (Becker and Palsson, 2008) (or other algorithms such as iMAT- developed by Shlomi), it should be converted to a format that can be used by Cobra Toolbox.

GIMME algorithm takes out the reactions which correspond to gene expression values below a specified threshold. Absence/Presence (AP.txt) data which includes highly and lowly expressed genes to be used as an input to GIMME can be obtained by using the language R. It can be also created in Excel by using gene expression values.

Data table header descriptions

ID_REF
VALUE Expression data are reported as unlogged RMA values

Data table

ID_REF	VALUE
1007_s_at	2983.760276
1053_at	112.4005042
117_at	82.14598043
121_at	356.8399314
1255_g_at	23.29270402
1294_at	128.5769945
1316_at	64.37545624
1320_at	41.36458412
1405_i_at	26.75933521
1431_at	17.23322308
1438_at	365.2221829
1487_at	167.3912879
1494_f_at	92.73307833
1598_g_at	782.722688
160020_at	377.4008162
1729_at	205.5688981
1773_at	150.5854475
177_at	36.00551561
170_*	462.7707106

Figure B.4. Expression values of GSE62600 of sample GSM1529853.

The average of gene expression values of each gene in different patients is calculated as 337. The threshold is specified as 30% of the mean of transcriptome data of MB dataset (See section 3.1.4.1). Genes with an expression value lower than the threshold were identified as Absent (A) and genes with an expression higher than the threshold were identified as present (P). An example table for 3 patients is shown as in Table B.1.

Table B.1. Example of Absence/Presence (AP.text) data table.

Genes	P1	P2	P3	Mean	P/A
1007_s_at	2984	580	1306	1623	P
1053_at	112	106	113	110	P
117_at	82	73	82	79	A
121_at	357	367	289	337	P

Afterwards, another table involving Gene ID's, in the same format as in the MB model, corresponding to ones found in GSE data should be created. The table involving all gene ID formats in GPL96 can be downloaded from GEO website. An example table for four genes are given in Table B.2.

Table B.2. Example of Gene ID (Entrez ID's) data.

ID	ENTREZ GENE ID
1007_s_at	780 /// 100616237
1053_at	5982
117_at	3310
121_at	7849

Once (AP.txt) data and EntrezIDs (EID.txt) are ready (Figure 1.8.), MB specific model can be created.

```
EID=importdata('eid.txt');
AP1=importdata('ap.txt');
```

After importing texts, AP1 is named as Data1. It is converted to character array and transposed for For loop. Then, a new matrix named D in the dimension of Data1 is generated.

```
Data1=cellstr(AP1);
Data1=char(Data1);
Data1=Data1.';
D=zeros(1,length(Data1));
```

A and P in Data1 are converted to 0 and 1 by using For loop.

for

```
i=[1] Data=Data1((i),:);
```

```
A=strfind(Data,'A');
```

```
P=strfind(Data,'P');
```

```
i=i/2+.5;
```

```
D(i,A)=0;
```

```
D(i,P)=1;
```

end

After D matrix is transposed again, both D and EID are assigned in a struct named ExpressionData as Data and Locus respectively.

```
D=D.';
```

```
ExpressionData.Data=D;
```

```
ExpressionData.Locus=EID;
```

Then following code is used to generate MB-specific model.

```
[tissue,Rxns]=createTissueSpecificRec(model,expressionData,proceedExp,orphan,  
exRxnRemove, solver,options,funcModel)
```

Since data is already processed, proceedExp should be 1. Orphan parameter should be 1 to leave orphan reactions in MB model. exRxnRemove was not used as it was not desired to remove the exchange reactions. Solver should be GIMME to create MB-specific model with GIMME algorithm. Objective coefficient should be entered in options as shown in the following code. funcModel generates a model including only reactions that can carry a flux. In order to skip this step 0 should be entered. Once the constraints and objective function are set, the following codes are entered. GIMME is run in default settings.

```
options = [find(model2.c) 0.90]  
changeCobraSolver ('gurobi', 'all');  
[MODEL_gimme,Rxns]= createTissueSpecificRec(model2, ExpressionData,1,1,[], 'GIMME',  
options,0);
```