M.A.G.E

Meta-analysis of Gene Expression

User Guide

http://www.compgen.org/tools/mage
https://github.com/pbagos/mage

Version 1.0; July 2021

Ioannis A. Tamposis[#], Georgios A. Manios[#], Theodosia Charitou, Konstantina E. Vennou, Panagiota I. Kontou and Pantelis G. Bagos*

(*) Correspondence: Pantelis G. Bagos (pbagos@compgen.org)

Document Revisions

Edition	Date	Author(s)	Description
1.0.0	July 2021	Ioannis Tamposis, Georgios Manios	First Edition

Contents

1. Introduction

MAGE, an acronym for Meta analysis of Gene Expression is a meta-analysis tool developed for data which come from gene expression studies.

The overall aim of this work has been to develop a software tool that would offer a large collection of meta-analysis options, as well as several extensions to evaluate the software applied to various biological problems. The MAGE framework is characterized by:

Speed: It takes a small amount of time to perform the functions which are included

Effectiveness: It gives reliable estimations and results, thanks to the mathematical models which are implemented.

Compatibility: It can be executed either from a Windows or a UNIX operating system. Also, there is no need for the user to be expert of any programming or computer science knowledge to run M.A.G.E.

MAGE is consisted of three basic functions.

- ***GISU:*** Converts probes to gene Identifiers
- ***Meta-analysis:*** Standard meta-analysis, bootstrap meta-analysis and multivariate meta-analysis functions are offered.
- ***Enrichment analysis:*** Given a list of statistically significant genes that come from the meta-analysis, functional enrichment analysis is conducted with the gProfiler toolkit.

1.1 The theory behind MAGE

Meta – analysis: A statistical method, used to combine studies addressing the same question, to extract a safe estimation. It is a reliable method for the evaluation of the effects of the under-study genes in complex phenotypes. Also, the heterogeneity of each study can be calculated too, as well as the total effect size. In M.A.G.E, the meta-analysis function is based to the Random Effects model with the Hedge's g correction for the effect size.

Bootstrap resampling method: The small size of the micro array experiments and the non-normal distribution of the expression values of the genes, are two major problems in these analyses. So, with this method, which is based on the t-test, the dataset is resampled many times (usually 200 or 500, even 1000 times). The final estimations are more accurate than those of the standard meta-analysis, but it is a more time-consuming method.

Multiple Tests: In these analyses, the genes that were found as statistically significant (according to their p -values) are statistically significant indeed? The answer to that question is given thanks to the multiple tests. Those tests take a p-value list as input and return a list of the corrected p-values. The multiple tests are separated in two categories:

a) FWER (Bonferroni, Sidak, Holm, Holland) methods
b) FDR (Hochberg, Simes) methods

The Bonferonni and Sidak methods belong to the one step methods, the Holm and Holland methods belong to the step-up methods and the Hochberg and Simes methods belong to the step-down methods.

Multivariate meta-analysis: Generally, when we refer to meta-analysis, we refer to the univariate meta-analysis, where only one effect is reported by each study. However, there are many cases in practice where a study may report multiple effect sizes. This challenge can be dealt with the multivariate meta-analysis methods.

Enrichment analysis: It is a method to identify classes of genes that are over-represented in a large set of genes and may have an association with disease phenotypes. This method utilizes statistical approaches to identify significantly enriched or depleted groups of genes.

2. Getting started

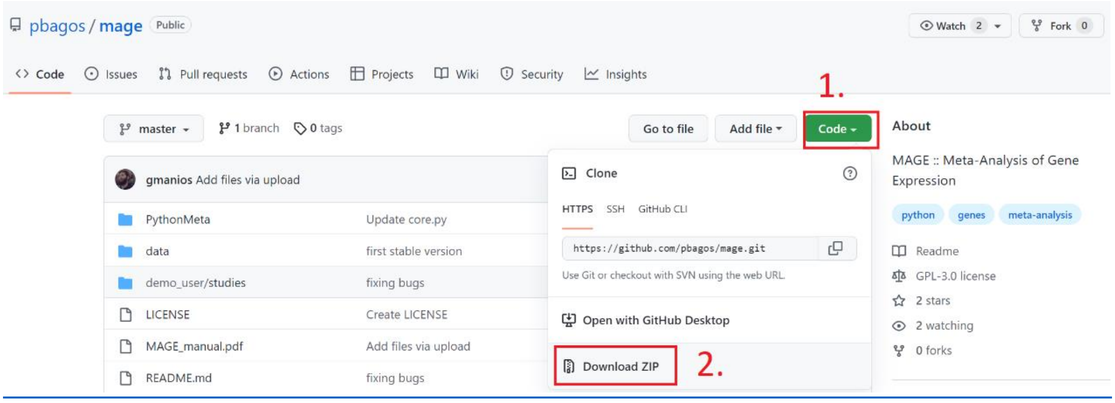
MAGE is a Python package that can be run from the command line. MAGE is written in Python (ver. 3.7.9) and requires the following Python libraries and packages to run:

Requirements

- Pandas~=1.2.4
- Numpy~=1.19.5
- Matplotlib~=3.3.4
- scipy~=1.6.2
- statsmodels~=0.12.2
- PythonMeta~=1.23
- requests~=2.25.1
- statistics~=1.0.3.5
- Seaborn~=0.11.1

Download MAGE from: https://github.com/pbagos/mage

Otherwise, you can run MAGE from its online infrastructure at:
http://www.compgen.org/tools/mage (Mozila Firefox browser is suggested)



pbagos / mage Public

Watch 2 Fork 0

Code Issues Pull requests Actions Projects Wiki Security Insights

1.

master 1 branch 0 tags Go to file Add file Code

gmanios Add files via upload	
PythonMeta	Update core.py
data	first stable version
demo_user/studies	fixing bugs
LICENSE	Create LICENSE
MAGE_manual.pdf	Add files via upload
README.md	fixing bugs

Clone

HTTPS SSH GitHub CLI

https://github.com/pbagos/mage.git

Use Git or checkout with SVN using the web URL.

Open with GitHub Desktop

Download ZIP 2.

About

MAGE :: Meta-Analysis of Gene Expression

python genes meta-analysis

Readme

GPL-3.0 license

2 stars

2 watching

0 forks

After downloading the .zip folder of MAGE from GitHub, extract it to a working directory. To compile MAGE, do as it follows below:

Compile:

```
python mage.py  -c conf.txt  -o results/
```

3. Arguments and Options

The M.A.G.E program provides the following command-line arguments:

-c: The configuration (.txt) file which contains the settings selected from the user.

-o: The output file where the user wants to store the results extracted from MAGE

4. Input files

4.1 Configuration file

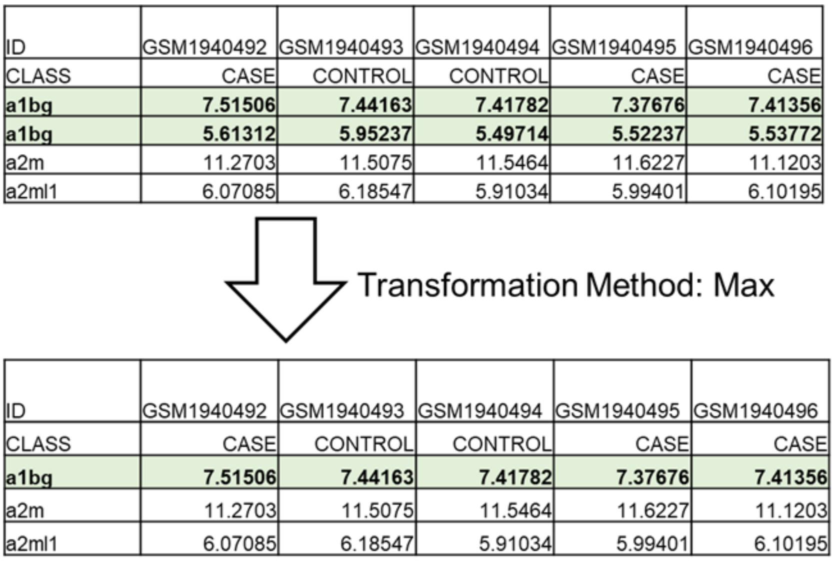
This is the first and most important input file. From this file, the user can give the settings for the execution of MAGE. As it can be seen below, the user can choose the following settings:

```
[SETTINGS]
#General options
#set the folder name contains study files
study_dir = demo_user/studies/

#Study files list separated by comma (,)
study_files = 1.txt, 2.txt,3.txt, 4.txt, 5.txt,6.txt,7.txt,8.txt,9.txt,10.txt

plots = NO # or YES

#Annotation - Define Classes
controls = 0 #or control, or other string identifier
cases = 1 #or case, or other string identifier

#only for Multivariate Analysis
cases2 = 2 # or case2, or other string identifier

#GISU Options
RUN_GISU = NO #or YES
#if YES load data from web, if NO load local data from folder data/
gene_data_online = NO #or YES

transformation_method = mean # or min, max

#Platform name for file. For multiple study files insert a list of platforms separated by comma (,) one per study
platform = GPL96
gene_history_file = data/Gene_History_Reference/gene_history.txt
homo_sapiens_file = data/Homo_Sapiens_reference/homo_sapiens_gene.txt
```

```
platforms_folder = data/Platforms/

#Meta - Analysis Options
bootstrap = NO #or YES
num_of_reps = 100
#Level of Significance of Mulitple Comparisons
significance_level = 0.01
multiple_comparisons = all # or one_step, step_down, step_up, all


#Multivariate Analysis Options
multivariate = NO #or YES
venn_correction = bonferroni #or sidak, holm, holland, hochberg, simes, none

venn_choice = global1_stoufer_weighted # or global1_RE, global1_stoufer, global1_p_fisher,  global1_p_edg1
global1_p_edg2, global2

alpha = 0.05
multiple_tests = one_step # or step_down, step_up, none

#Enrichment Analysis Options
enrichment_analysis = NO #or YES
organism = hsapiens
threshold = 0.05
#Default 'fdr'. Other options are 'bonferroni' and 'g_SCS'
threshold_method = fdr
```

4.2 GISU

MAGE uses an optional component called GISU (Gene ID/Symbol update) to transform the platform's probe identifiers to gene symbols identifiers. This can be helpful when one is comparing datasets arising from different platforms, in which case the probe identifiers must be converted to gene identifiers. Considering that multiple probes may correspond to the same gene in a microarray experiment [2], the multiple entries of the same gene can be combined resolving the “many-to-many” relationship between probe and gene symbols. The software offers three options for this task: the minimum, maximum or arithmetic mean (average) [2,3]. If a particular platform is not included in the list, the user can upload the platform file to proceed to the transformation.



ID	GSM1940492	GSM1940493	GSM1940494	GSM1940495	GSM1940496
CLASS	CASE	CONTROL	CONTROL	CASE	CASE
a1bg	**7.51506**	**7.44163**	**7.41782**	**7.37676**	**7.41356**
a1bg	**5.61312**	**5.95237**	**5.49714**	**5.52237**	**5.53772**
a2m	11.2703	11.5075	11.5464	11.6227	11.1203
a2ml1	6.07085	6.18547	5.91034	5.99401	6.10195

Transformation Method: Max

ID	GSM1940492	GSM1940493	GSM1940494	GSM1940495	GSM1940496
CLASS	CASE	CONTROL	CONTROL	CASE	CASE
a1bg	**7.51506**	**7.44163**	**7.41782**	**7.37676**	**7.41356**
a2m	11.2703	11.5075	11.5464	11.6227	11.1203
a2ml1	6.07085	6.18547	5.91034	5.99401	6.10195

Figure 1. A simple example how GISU transforms probes to gene identifiers. Two probe identifiers correspond to the same gene identifier (highlighted gene identifier). In this case, the max transformation method is applied. The row with the biggest expression values remains in the final dataset and the other row is deleted.

4.3 META-ANALYSIS

Standard meta-analysis

The following array shows the format which the study (.txt) files need to follow to execute the analysis.

gene_symbol	**gsm1940492**	**gsm1940493**	**gsm1940494**	**gsm1940495**	**gsm1940496**	**gsm1940501**
class	1	0	0	1	1	1
a1bg	7.51506	7.44163	7.41782	7.37676	7.41356	7.48818
a1cf	5.61312	5.95237	5.49714	5.52237	5.53772	5.5811
a2m	11.2703	11.5075	11.5464	11.6227	11.1203	11.542
a2ml1	6.07085	6.18547	5.91034	5.99401	6.10195	6.07594
a3galt2	6.37637	6.20717	6.25116	6.15003	6.21651	6.23363

The first column contains the genes that took part in the experiment. At the second row of the first column, there is a class label for each column.

0 (or other string identifier) → Represents the controls of the micro-array experiment (blue columns).

1(or other string identifier) → Represents the cases of the micro-array experiment (green columns).

After the meta-analysis is completed, the results are stored to a .txt file and the plots to the corresponding folder.

Bootstrap meta-analysis

The Bootstrap meta-analysis function takes the same input and returns the same output format with the standard meta-analysis function.

Multivariate analysis

In this function, the input file format is the same, but with an extra class label.

gene_symbol	**gsm76030**	**gsm76031**	**gsm76032**	**gsm76033**	**gsm76034**	**gsm76035**	**gsm76036**
class	2	2	1	1	0	0	2
a1cf	89.2408	72.5536	78.3414	78.2159	53.763	52.4481	66.3112
a2m	27.6737	5.72072	22.4551	26.1235	11.5605	14.6682	11.9491
a4galt	34.7204	51.4307	36.7044	27.6687	24.3128	27.5027	21.6227
a4gnt	25.7449	30.8878	52.2123	16.4948	3.3895	11.1433	13.9207

The first column contains the genes that took part in the experiment. At the second row of the first column, there is a class label for each column.

0 (or other string identifier) → Represents the controls of the micro-array experiment (blue columns)

1 (or other string identifier) → Represents the first class of cases of the micro-array experiment (green columns)

2 (or other string identifier) → Represents the second class of the cases of the micro-array experiment (pink columns)

4.4 ENRICHMENT ANALYSIS

Moreover, the software uses g: Profiler [16] to perform functional enrichment analysis with a given gene list produced as the result from the meta-analysis by using the implemented python module. The software returns multiple files containing gene definitions, a list with statistically significant enriched GO terms, biological pathways, regulatory motifs in DNA, or phenotype ontologies that these genes are highly enriched and provides the user the option to visualize results with a Manhattan or a heatmap plot.

5. Examples

5.1 Meta-analysis

5.1.1 Standard meta-analysis example

1. Collect the studies (see at 4.3 how the gene expressions files should look to execute this function) and put them to the correct directory (demo_user/studies)

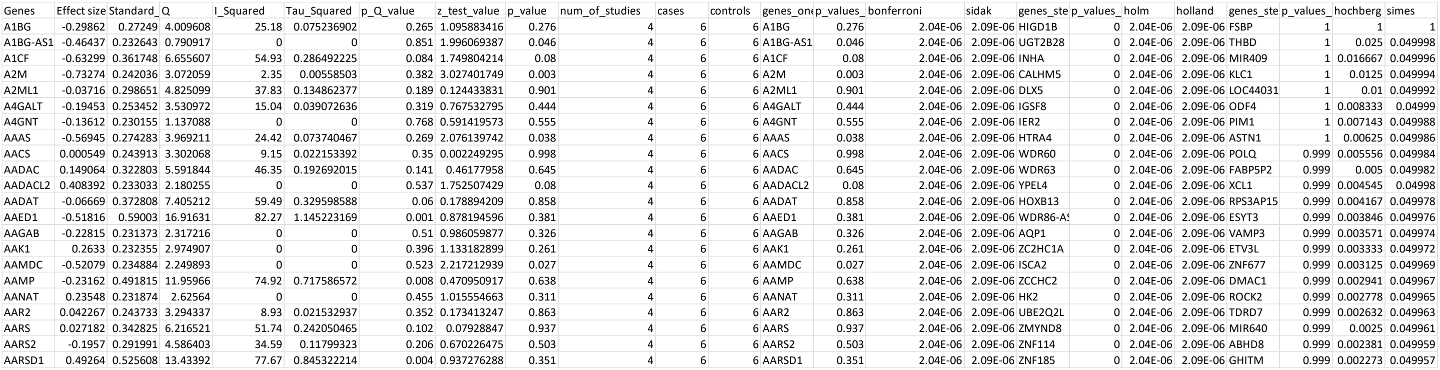
2. Set up the tool parameters in the conf.txt file (see the example below):

```
[SETTINGS]

[General options]
#Set the folder name contains study files
study_dir = demo_user/studies/
#Study files list separated by comma (,)
study_files = study1.txt, study2.txt, study3.txt, study4.txt… study10.txt
# WARNING: The study files must be tab delimited (\t) .txt files
plots = YES

[Meta-Analysis Options]
#Meta - Analysis Options
bootstrap = NO
num_of_reps = 100 # only if bootstrap = YES
#Level of Significance of Mulitple Comparisons
significance_level = 0.01
multiple_comparisons = all
```

3. Compile MAGE with the following command at the terminal

```
python mage.py  -c conf.txt  -o results/
```

4. See the results and the plots to the output folder (results/). See an example of the results below:



Genes	Effect size	Standard_	Q	I_Squared	Tau_Squared	p_Q_value	z_test_value	p_value	num_of_studies	cases	controls	genes_one	p_values_	bonferroni	sidak	genes_ste	p_values_	holm	holland	genes_ste	p_values_	hochberg	simes
A1BG	-0.29862	0.27249	4.009608	25.18	0.075236902	0.265	1.095883416	0.276	4	6	6	A1BG	0.276	2.04E-06	2.09E-06	HIGD1B	0	2.04E-06	2.09E-06	FSBP	1	1	1
A1BG-AS1	-0.46437	0.232643	0.790917	0	0	0.851	1.996069387	0.046	4	6	6	A1BG-AS1	0.046	2.04E-06	2.09E-06	UGT2B28	0	2.04E-06	2.09E-06	THBD	1	0.025	0.049998
A1CF	-0.63299	0.361748	6.655607	54.93	0.286492225	0.084	1.749804214	0.08	4	6	6	A1CF	0.08	2.04E-06	2.09E-06	INHA	0	2.04E-06	2.09E-06	MIR409	1	0.016667	0.049996
A2M	-0.73274	0.242036	3.072059	2.35	0.00558503	0.382	3.027401749	0.003	4	6	6	A2M	0.003	2.04E-06	2.09E-06	CALHM5	0	2.04E-06	2.09E-06	KLC1	1	0.0125	0.049994
A2ML1	-0.03716	0.298651	4.825099	37.83	0.134862377	0.189	0.124433831	0.901	4	6	6	A2ML1	0.901	2.04E-06	2.09E-06	DLX5	0	2.04E-06	2.09E-06	LOC44031	1	0.01	0.049992
A4GALT	-0.19453	0.253452	3.530972	15.04	0.039072636	0.319	0.767532795	0.444	4	6	6	A4GALT	0.444	2.04E-06	2.09E-06	IGSF8	0	2.04E-06	2.09E-06	ODF4	1	0.008333	0.04999
A4GNT	-0.13612	0.230155	1.137088	0	0	0.768	0.591419573	0.555	4	6	6	A4GNT	0.555	2.04E-06	2.09E-06	IER2	0	2.04E-06	2.09E-06	PIM1	1	0.007143	0.049988
AAAS	-0.56945	0.274283	3.969211	24.42	0.073740467	0.269	2.076139742	0.038	4	6	6	AAAS	0.038	2.04E-06	2.09E-06	HTRA4	0	2.04E-06	2.09E-06	ASTN1	1	0.00625	0.049986
AACS	0.000549	0.243913	3.302068	9.15	0.022153392	0.35	0.002249295	0.998	4	6	6	AACS	0.998	2.04E-06	2.09E-06	WDR60	0	2.04E-06	2.09E-06	POLQ	0.999	0.005556	0.049984
AADAC	0.149064	0.322803	5.591844	46.35	0.192692015	0.141	0.46177958	0.645	4	6	6	AADAC	0.645	2.04E-06	2.09E-06	WDR63	0	2.04E-06	2.09E-06	FABP5P2	0.999	0.005	0.049982
AADACL2	0.408392	0.233033	2.180255	0	0	0.537	1.752507429	0.08	4	6	6	AADACL2	0.08	2.04E-06	2.09E-06	YPEL4	0	2.04E-06	2.09E-06	XCL1	0.999	0.004545	0.04998
AADAT	-0.06669	0.372808	7.405212	59.49	0.329598588	0.06	0.178894209	0.858	4	6	6	AADAT	0.858	2.04E-06	2.09E-06	HOXB13	0	2.04E-06	2.09E-06	RPS3AP15	0.999	0.004167	0.049978
AAED1	-0.51816	0.59003	16.91631	82.27	1.145223169	0.001	0.878194596	0.381	4	6	6	AAED1	0.381	2.04E-06	2.09E-06	WDR86-AS	0	2.04E-06	2.09E-06	ESYT3	0.999	0.003846	0.049976
AAGAB	-0.22815	0.231373	2.317216	0	0	0.51	0.986059877	0.326	4	6	6	AAGAB	0.326	2.04E-06	2.09E-06	AQP1	0	2.04E-06	2.09E-06	VAMP3	0.999	0.003571	0.049974
AAK1	0.2633	0.232355	2.974907	0	0	0.396	1.133182899	0.261	4	6	6	AAK1	0.261	2.04E-06	2.09E-06	ZC2HC1A	0	2.04E-06	2.09E-06	ETV3L	0.999	0.003333	0.049972
AAMDC	-0.52079	0.234884	2.249893	0	0	0.523	2.217212939	0.027	4	6	6	AAMDC	0.027	2.04E-06	2.09E-06	ISCA2	0	2.04E-06	2.09E-06	ZNF677	0.999	0.003125	0.049969
AAMP	-0.23162	0.491815	11.95966	74.92	0.717586572	0.008	0.470950917	0.638	4	6	6	AAMP	0.638	2.04E-06	2.09E-06	ZCCHC2	0	2.04E-06	2.09E-06	DMAC1	0.999	0.002941	0.049967
AANAT	0.23548	0.231874	2.62564	0	0	0.455	1.015554663	0.311	4	6	6	AANAT	0.311	2.04E-06	2.09E-06	HK2	0	2.04E-06	2.09E-06	ROCK2	0.999	0.002778	0.049965
AAR2	0.042267	0.243733	3.294337	8.93	0.021532937	0.352	0.173413247	0.863	4	6	6	AAR2	0.863	2.04E-06	2.09E-06	UBE2Q2L	0	2.04E-06	2.09E-06	TDRD7	0.999	0.002632	0.049963
AARS	0.027182	0.342825	6.216521	51.74	0.242050465	0.102	0.07928847	0.937	4	6	6	AARS	0.937	2.04E-06	2.09E-06	ZMYND8	0	2.04E-06	2.09E-06	MIR640	0.999	0.0025	0.049961
AARS2	-0.1957	0.291991	4.586403	34.59	0.11799323	0.206	0.670226475	0.503	4	6	6	AARS2	0.503	2.04E-06	2.09E-06	ZNF114	0	2.04E-06	2.09E-06	ABHD8	0.999	0.002381	0.049959
AARSD1	0.49264	0.525608	13.43392	77.67	0.845322214	0.004	0.937276288	0.351	4	6	6	AARSD1	0.351	2.04E-06	2.09E-06	ZNF185	0	2.04E-06	2.09E-06	GHITM	0.999	0.002273	0.049957

Figure 2. A screenshot from the meta-analysis results.

Explanation of the results

Genes: The names of the genes.

Effect Size (Hedge’s g): The overall effect size with the Hedge’s g correction.

Standard Error: The overall standard error of each gene.

I – Squared, Tau – Squared and Q: The heterogeneity tests.

p_Q value: p-value of the Q test.

z_test _value: The value of the z-test.

p-value: The p_ value of the gene.

num_of_studies: A counter that depicts in how many studies was each gene

cases: A counter that show us in how many samples each gene appears as a case control: A counter that show us in how many samples each gene appears as a control

bonferroni, sidak, holm, holland,hochberg and simes: The corrected p-values with the multiple comparison tests.

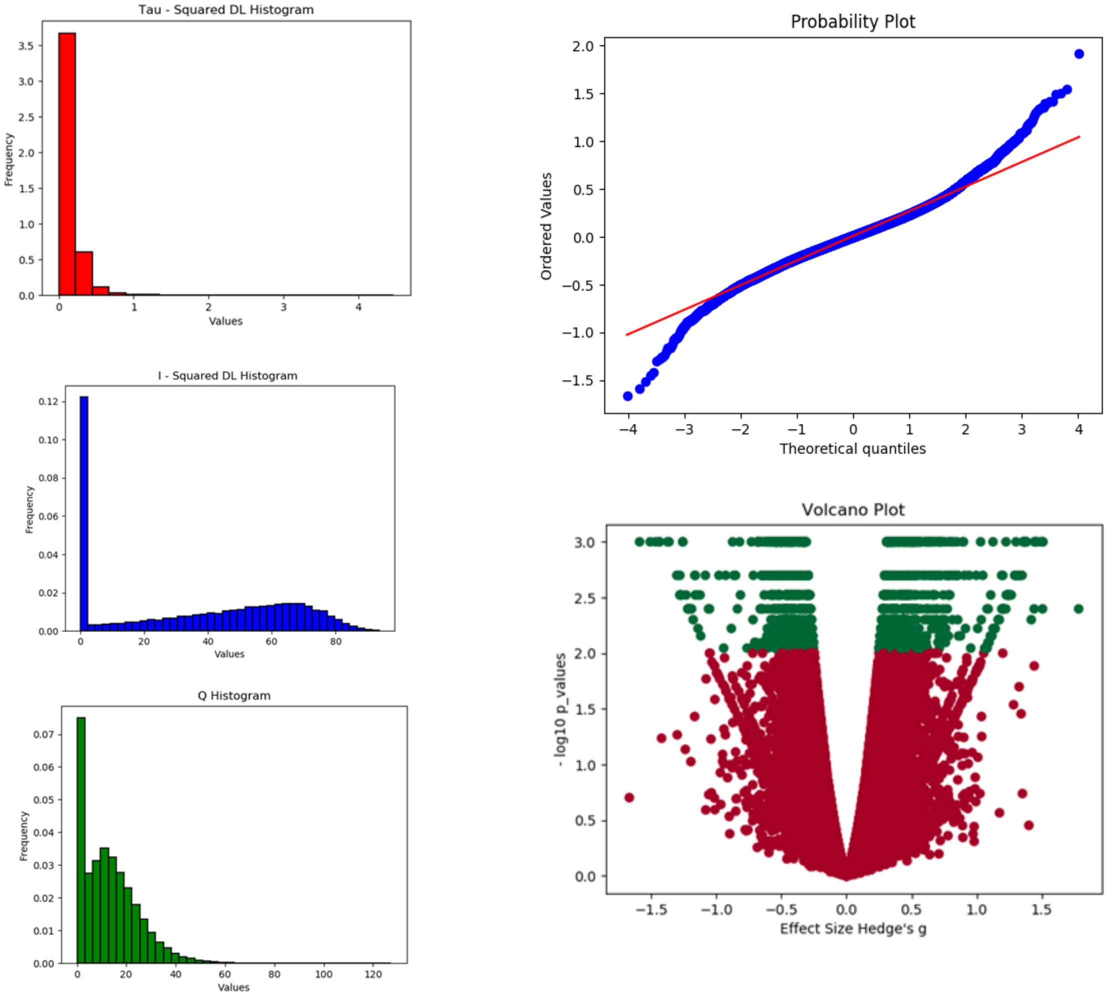


Figure 3. The plots of a standard meta-analysis of MAGE

QQ plot of probability plot: A scatter plot which portrays a theoretical distribution of the corrected effect sizes (x-axis) against the real distribution of the corrected effect sizes of the genes (y-axis).

Histograms (for *Q*, Tau^2, I^2): Histograms for the measures of heterogeneity.

Volcano plot: A scatter plot which portrays the corrected effect sizes (x-axis) against the negative decimal logarithm ($-\log_{10}$) of the p-values (y-axis). The genes which were found as statistically significant, are colored with green and the genes which were characterized as not statistically significant are colored with red.

5.1.2 Bootstrap meta-analysis example

1. Collect the studies (see at 4.3 how the studies should look to execute the multivariate meta-analysis function) and put them to the correct directory.

2. Set up the tool parameters in the conf.txt file (see the example below):

```
[SETTINGS]

[General options]
#Set the folder name contains study files
study_dir = demo_user/studies/
#Study files list separated by comma (,)
study_files = study1.txt, study2.txt, study3.txt, study4.txt… study10.txt
# WARNING: The study files must be tab delimited (\t) .txt files
plots = YES

[Meta-Analysis Options]
bootstrap = YES
num_of_reps = 200
#Level of Significance of Multiple Comparisons
significance_level = 0.05
multiple_comparisons = one_step
```

3. Compile MAGE with the following command at the terminal

```
python mage.py  -c conf.txt  -o results/
```

4. See the results and the plots in the output folder (results/). It returns the same output as the standard meta-analysis function.

5.1.3 Multivariate meta-analysis example

1. Collect the studies (see at 4.4 how the studies should look in order to execute this function) and put them to the correct directory

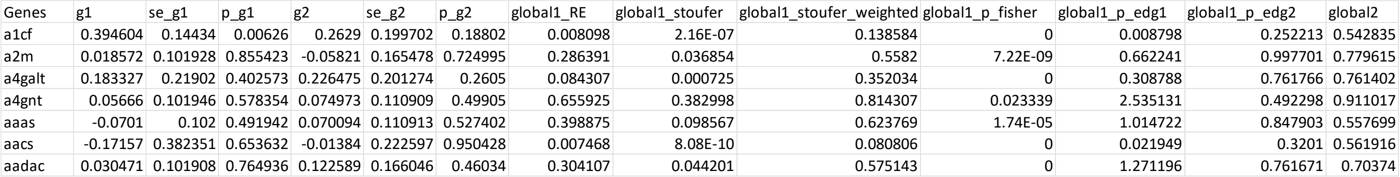
2. The configuration file must be set up like this (In the other M.A.G.E functions, type NO):

```
[SETTINGS]

[General options]
#Set the folder name contains study files
study_dir = demo_user/studies/
#Study files list separated by comma (,)
study_files = study1.txt, study2.txt, study3.txt, study4.txt… study10.txt
# WARNING: The study files must be tab delimited (\t) .txt files
plots = YES

[Multivariate meta-analysis options]
multivariate = NO (or YES)
venn_correction = none #(available choices: bonferroni, sidak, holm, holland, hochberg,
simes, none)
venn_choice = global1_stoufer_weighted #(available choices = 'global1_RE',
global1_stoufer, global1_stoufer_weighted, global1_p_fisher, global1_p_edg1,
global1_p_edg2, global2)
alpha = 0.05
multiple_tests = none (available choices: one_step, step_down, step_up, none)
```

3. Compile M.A.G.E with the following command at the terminal

```
python mage.py  -c conf.txt  -o results/
```

4. See the results and the plots to the output folder (results/). See an example below:



Genes	g1	se_g1	p_g1	g2	se_g2	p_g2	global1_RE	global1_stoufer	global1_stoufer_weighted	global1_p_fisher	global1_p_edg1	global1_p_edg2	global2
a1cf	0.394604	0.14434	0.00626	0.2629	0.199702	0.18802	0.008098	2.16E-07	0.138584	0	0.008798	0.252213	0.542835
a2m	0.018572	0.101928	0.855423	-0.05821	0.165478	0.724995	0.286391	0.036854	0.5582	7.22E-09	0.662241	0.997701	0.779615
a4galt	0.183327	0.21902	0.402573	0.226475	0.201274	0.2605	0.084307	0.000725	0.352034	0	0.308788	0.761766	0.761402
a4gnt	0.05666	0.101946	0.578354	0.074973	0.110909	0.49905	0.655925	0.382998	0.814307	0.023339	2.535131	0.492298	0.911017
aaas	-0.0701	0.102	0.491942	0.070094	0.110913	0.527402	0.398875	0.098567	0.623769	1.74E-05	1.014722	0.847903	0.557699
aacs	-0.17157	0.382351	0.653632	-0.01384	0.222597	0.950428	0.007468	8.08E-10	0.080806	0	0.021949	0.3201	0.561916
aadac	0.030471	0.101908	0.764936	0.122589	0.166046	0.46034	0.304107	0.044201	0.575143	0	1.271196	0.761671	0.70374

Figure 4. The results of the multiple outcomes meta-analysis.

Explanation of the results

Genes: The names of the genes.

g1: Overall effect size for the first effect.

se_g1: Overall standard error for the first effect.

p_g1: p-value of the first effect.

g2: Overall effect size for the second effect.

se_g2 : Overall standard error for the second effect.

p_g2: p-value of the second effect.

global1_RE: The p-value which comes from the Random Effects model of the multivariate meta-analysis.

global1_stoufer: Stouffer’s method result for global1.

global1_stouffer_weighted: Stouffer’s weighted method result for global1.

global1_p_fisher: Fisher method result of the overall effect size

global1_p_edg1: Edgington’s first method result for global1.

global1_p_edg2: Edgington’s second method result for global1.

global2: The p-value that occurs from the multivariate meta-analysis which takes as input the difference of g1 and g2 as effect size and the standard error of the difference of g1 and g.

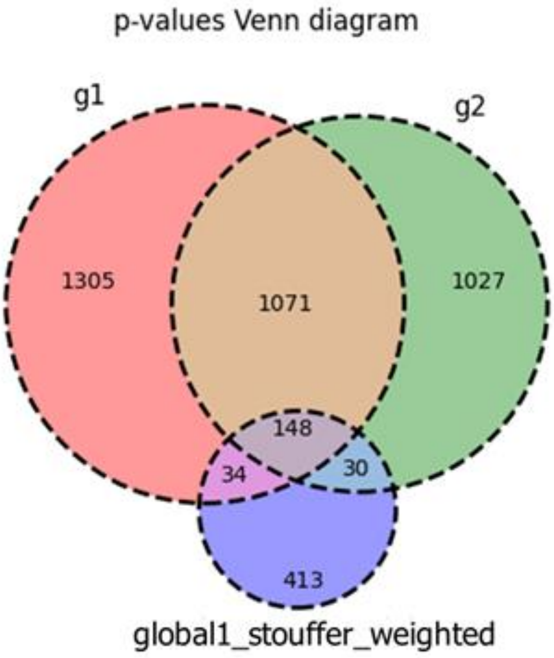


Figure 5. The Venn diagram for the p-values of the multivariate meta-analysis.

Venn diagram: This diagram shows how many p values were found as statistically significant with each effect. It is consisted of three p-value sets.

g1**:** p-values from the first effect

g2**:** p-values from the second effect

The third set of p-values that would be portrayed, depends on the venn_choice option of the configuration file. In this case, the global1_stouffer_weighted was chosen to be the third set of p-values.

global1_stouffer_weighted: p -values from the Stouffer's weighted method

5.2 Enrichment analysis examples

The enrichment analysis is available for the standard meta-analysis only. That means that the meta-analysis function needs to be executed first. So, if the user choses to perform functional enrichment analysis, the corresponding option should be toggled to 'YES' in the configuration file. The functional enrichment analysis of the input gene list is performed using the g:Profiler toolkit in python using the g:GOSt core tool that detects statistically significantly enriched biological processes, molecular functions, cellular components, biological pathways, regulatory motifs and protein complexes.

The enrichment results in g:GOSt are highlighted in a Manhattan plot (Figure 7). The enrichment results are presented in a Manhattan Plot with all significant terms identified per source and it is accompanied by a more extensive readable output format with detailed information about every term with gene list and p-values. Each functional enriched term is derived from the most common data sources which are regularly updated such as Gene Ontology, KEGG, Reactome, WikiPathways, miRTarBase, TRANSFAC, Human Protein Atlas, CORUM, and the Human Phenotype Ontology. Furthermore, a heatmap visualization illustrates results for genes participating in significant enrichment terms (Figure 8).

Table 1. Functional enrichment analysis results output.

source	**term_name**	**term_id**	**adjusted_p_value**	**negative_log10_of_adjusted_p_value**
GO:BP	response to organic substance	GO:0010033	5.54E-07	6.256117189
GO:BP	cellular response to chemical stimulus	GO:0070887	1.25257E-06	5.902196824
GO:BP	response to chemical	GO:0042221	4.66537E-06	5.331113705
GO:BP	cellular response to organic substance	GO:0071310	1.01484E-05	4.993600908
GO:BP	multicellular organism development	GO:0007275	0.002198326	2.657907808
GO:CC	vesicle	GO:0031982	3.24914E-05	4.488232185
GO:CC	cytoplasm	GO:0005737	5.69367E-05	4.244607716

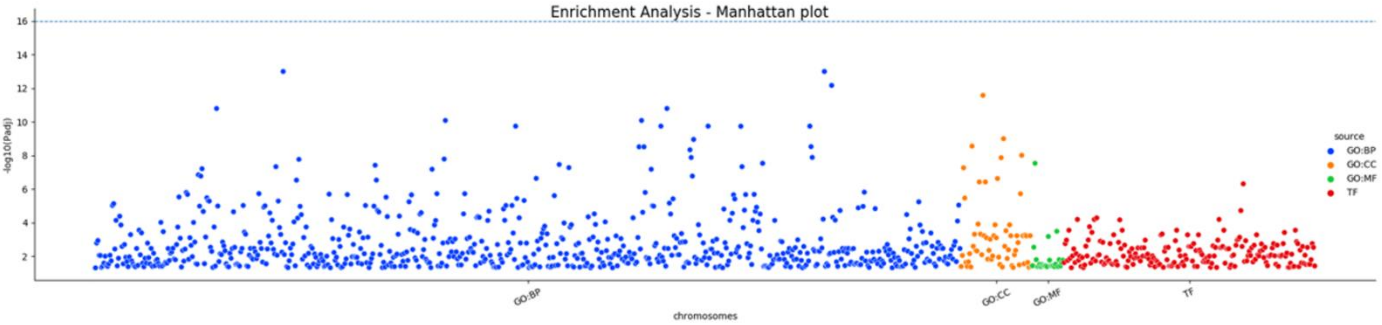


Figure 7. Manhattan plot. This plot maps genes to known functional information sources and detects statistically significant enriched terms using the well-proven cumulative hypergeometric test. The x-axis shows the functional terms, and the y-axis shows the corresponding enrichment p-values in negative log10 scale. Each circle represents the significant functional term across all the analysed term categories. The circles are colour-grouped by data source.

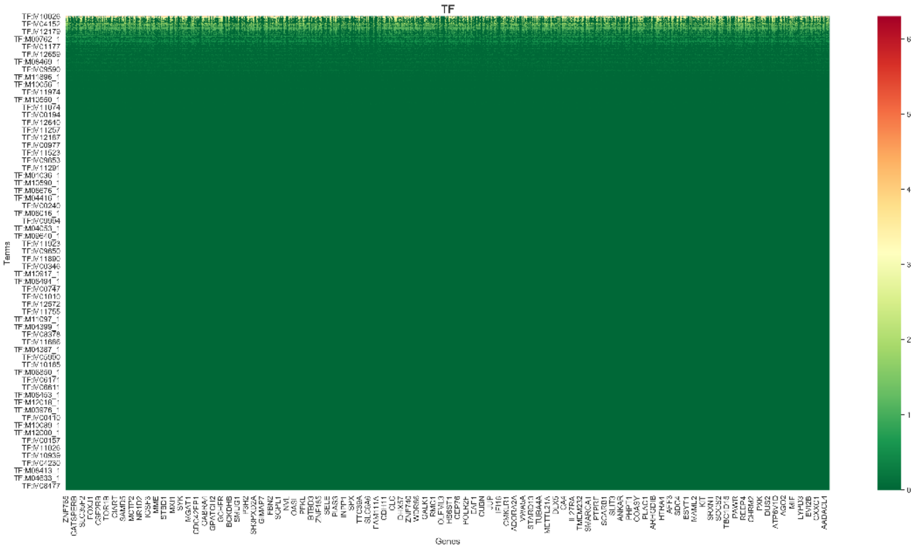


Figure 8. A heatmap plot from TF.

6.References

1. Forero, D. A.; Lopez-Leon, S.; González-Giraldo, Y.; Bagos, P. G. Ten Simple Rules for Carrying out and Writing Meta-Analyses. PLoS Comput. Biol. 2019, 15 (5), e1006922.
2. Dai, M.; Wang, P.; Boyd, A. D.; Kostov, G.; Athey, B.; Jones, E. G.; Bunney, W. E.; Myers, R. M.; Speed, T. P.; Akil, H. Evolving Gene/Transcript Definitions Significantly Alter the Interpretation of GeneChip Data. Nucleic Acids Res. 2005, 33 (20), e175–e175.
3. Ramasamy, A.; Mondry, A.; Holmes, C. C.; Altman, D. G. Key Issues in Conducting a Meta-Analysis of Gene Expression Microarray Datasets. PLoS Med. 2008, 5 (9), e184.
4. DerSimonian, R.; Laird, N. Meta-Analysis in Clinical Trials. Control. Clin. Trials 1986, 7 (3), 177–188.
5. Hedges, L. V. Distribution Theory for Glass's Estimator of Effect Size and Related Estimators. J. Educ. Stat. 1981, 6 (2), 107–128.
6. Choi, J. K.; Yu, U.; Kim, S.; Yoo, O. J. Combining Multiple Microarray Studies and Modeling Interstudy Variation. Bioinformatics 2003, 19 (suppl_1), i84–i90.
7. Doerge, R. Combining Affymetrix Microarray Results. BMC Bioinformatics 2005.
8. Diciccio, T. J.; Romano, J. P. A Review of Bootstrap Confidence Intervals. J. R. Stat. Soc. Ser. B Methodol. 1988, 50 (3), 338–354.
9. Meuwissen, T. H.; Goddard, M. E. Bootstrapping of Gene-Expression Data Improves and Controls the False Discovery Rate of Differentially Expressed Genes. Genet. Sel. Evol. 2004, 36 (2), 191–205.
10. Vennou, K. E.; Piovani, D.; Kontou, P. I.; Bonovas, S.; Bagos, P. G. Methods for Multiple Outcome Meta-Analysis of Gene-Expression Data. MethodsX 2020, 7, 100834.
11. Ludbrook, J. Multiple Comparison Procedures Updated. Clin. Exp. Pharmacol. Physiol. 1998, 25 (12), 1032–1037.
12. Holm, S. A Simple Sequentially Rejective Multiple Test Procedure. Scand. J. Stat. 1979, 65–70.
13. Holland, B. S.; Copenhaver, M. D. An Improved Sequentially Rejective Bonferroni Test Procedure. Biometrics 1987, 417–423.
14. Benjamini, Y. Discovering the False Discovery Rate. J. R. Stat. Soc. Ser. B Stat. Methodol. 2010, 72 (4), 405–416.
15. Simes, R. J. An Improved Bonferroni Procedure for Multiple Tests of Significance. Biometrika 1986, 73 (3), 751–754.
16. Raudvere, U.; Kolberg, L.; Kuzmin, I.; Arak, T.; Adler, P.; Peterson, H.; Vilo, J. G: Profiler: A Web Server for Functional Enrichment Analysis and Conversions of Gene Lists (2019 Update). Nucleic Acids Res. 2019, 47 (W1), W191–W198.