GMDR User Manual

GMDR software Beta 0.9

Updated March 2011

As an open source project, the source code of GMDR is published and made available to the public, enabling anyone to copy, modify and redistribute the source code. The most updated source code is available at http://sourceforge.net/projects/gmdr/.

Its SVN repository is https://gmdr.svn.sourceforge.net/svnroot/gmdr.

The most up-to-date GMDR software is distributed at http://www.ssg.uab.edu/gmdr/.

GMDR is developed and maintained by

Guo-Bo Chen, Ph.D.
Section on Statistical Genetics
Department of Biostatistics
University of Alabama at Birmingham
Contact bchen@ms.soph.uab.edu
Phone: 205-975-9263

Xiang-Yang Lou, Ph.D.
Section on Statistical Genetics
Department of Biostatistics
University of Alabama at Birmingham
Contact xlou@ms.soph.uab.edu
Phone: 205-975-9145

Yan Lei, M.S.
Department of Anatomy and Neurobiology
College of Medicine
University of Tennessee Health Science Center

INTRODUCTION

The determination of gene-by-gene and gene-by-environment interactions has long been one of the greatest challenges in genetics. The traditional methods are typically inadequate because of the problem referred to as the "curse of dimensionality." Recent combinatorial approaches, such as the multifactor dimensionality reduction (MDR) method, the combinatorial partitioning method, and the restricted partition method, have a straightforward correspondence to the concept of the phenotypic landscape that unifies biological, statistical genetics, and evolutionary theories. However, the existing approaches have several limitations, such as not allowing for covariates that restrict their practical use.

The Generalized MDR (GMDR) method permits adjustment for discrete and quantitative covariates and is applicable to both dichotomous and continuous phenotypes in various population-based study designs. The GMDR method, which, compared with the MDR method, can use score statistics to handle both quantitative and dichotomous traits and permits adjustment for covariates. To serve the users better, we have further enhanced the interface under the concept of "what you see is what you get" – a prevalent idea in software engineering, and thus the analysis which should satisfy average requirement for researchers can be accomplished through a graphical user interface (GUI). Furthermore, a Perl script is developed and included for advanced analysis, such as *p*-value evaluation.

SYSTEM REQUIREMENT

GMDR can run smoothly in any operating systems, such as Windows, Mac OS, Linux, in which the Java Virtual Machine and Perl are supported.

Java: Java SE Runtime Environment 1.6 or later.

Perl: Perl v5.10 or later.

RAM: ≥ 512M.

DATA FORMAT

Marker File

Genetic marker data can come in different formats, usually denoted by either a single genotype, or by a pair of alleles. GMDR can read marker files if they are organized as one of the two formats described below.

When the marker file is organized in the genotype format, it should look like this:

```
SNP1   SNP2   SNP3   class

1      2      1      0

0      2      1      0

…

2      2      1      1

0      2      1      1
```

The first row in the file is the header line which is composed of the names of SNPs and the status for each subject, and the columns are delimited by white space. **The class is coded 0 for unaffected, or 1 for affected**.

If each marker is represented by a pair of alleles, the file should look like this:

```
SNP1   SNP2   SNP3   class

0  0   1  1   0  1   0

0  0   1  1   0  1   0

…

1  1   1  1   1  0   1

0  0   1  1   0  1   1
```

For both formats shown above, GMDR eventually converts them to genotypes:

```
SNP1   SNP2   SNP3   class

00     11     01     0
```

00	11	01	0
…			
11	11	01	1
00	11	01	1

It should be noted that in this conversion, GMDR ignores phasing information, such as the pair of alleles, highlighted, converted genotype of which is “01”.

IMPORTANT GMDR cannot recognize a marker file in which both genotypes and alleles are used.

HINT For both of the accepted marker file formats, alleles can be represented by any character, (1,2,3, 4 or A, T, G, C, for example) .

Missing data is denoted by “.”. When in the allele format, if one of the alleles is missing, denoted by “. 1” or “1 .”, and “. .” if both are missing, GMDR always treats the whole locus as missing.

Phenotype file

GMDR closely relies on the score statistics which can either be built in GMDR or imported from other package. To facility score calculation, GMDR supports the inclusion of one or more covariates. Lining up with the marker file is very important for the phenotype file. With the first row as the header line, a phenotype file reads like:

Salt1	status
3.63	1
2.91	0
3.34	1
-1.40	0
5.81	1
0.70	0
1.56	1

Missing data is denoted by “.”.

Sample files

A marker file, 3uppercase_letter_marker.txt, and a phenotype file, 3uppercase_letter_cov_phe.txt are included in the package. In the example marker file, there are 1000 subjects of which 500 are affected and 500 are unaffected, and for each subject 10 markers are genotyped. In the phenotype file, two phenotypes, salt, and class, are included.

IMPUTATION FOR MISSING GENOTYPES AND PHENOTYPES

When GMDR reads a missing marker, it is imputed proportional to the frequencies of the internal reference panel (i.e the frequency of the markers observed at this locus).

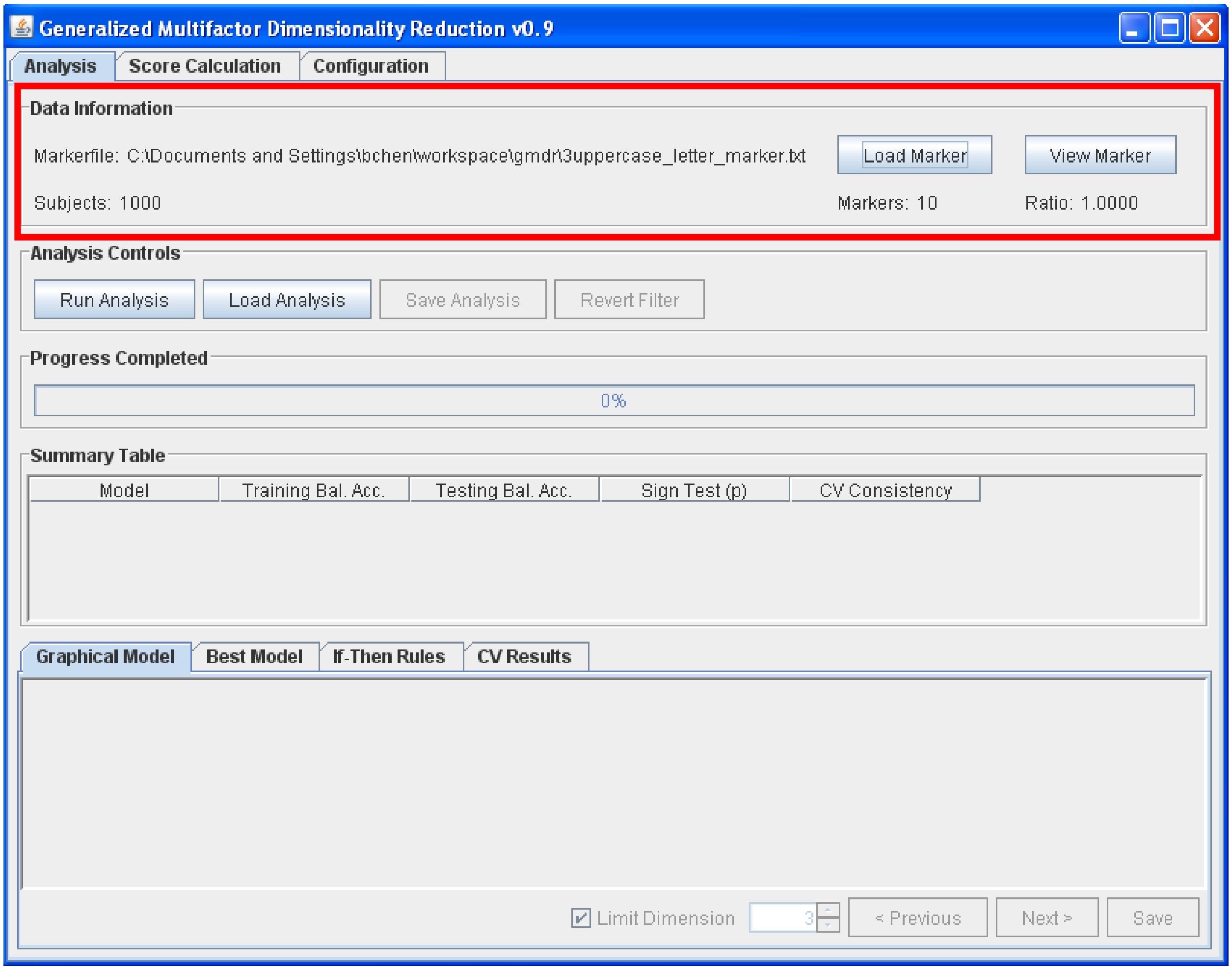
For phenotypic data, if the column of the data is selected as the response, depending on the method used for building score, the missing values are imputed randomly with binary variable, 0 or 1, if logistic regression is used, or imputed with the mean if linear regression is used. When a variable is used as a predictor, its missing data is simply imputed with the mean.

If the user cannot assess the influence of missing data, it is not encouraged to leave the missing data to GMDR. We expect the imputation implemented in GMDR reduces annoying software crashes caused by tiny missing data which will not influence the analysis result too much. Even though GMDR imputes missing data, other more professional imputation tools, such as HaploIHP, MACH, are preferable to impute the missing markers whenever possible.

START GMDR

The GMDR graphic user interface can be turned on either by double clicking the icon of GMDR or by typing “java –jar gmdr.jar” at the terminal. When it is started, it looks like the following figures. As the function of each button is almost self-explained, we only present ones which may be not that intuitive to users. We will also exemplify the usage of GMDR by illustrating the analysis of the example files.

ANALYSIS TAB



Markerfile: C:\Documents and Settings\bchen\workspace\gmdr\3uppercase_letter_marker.txt

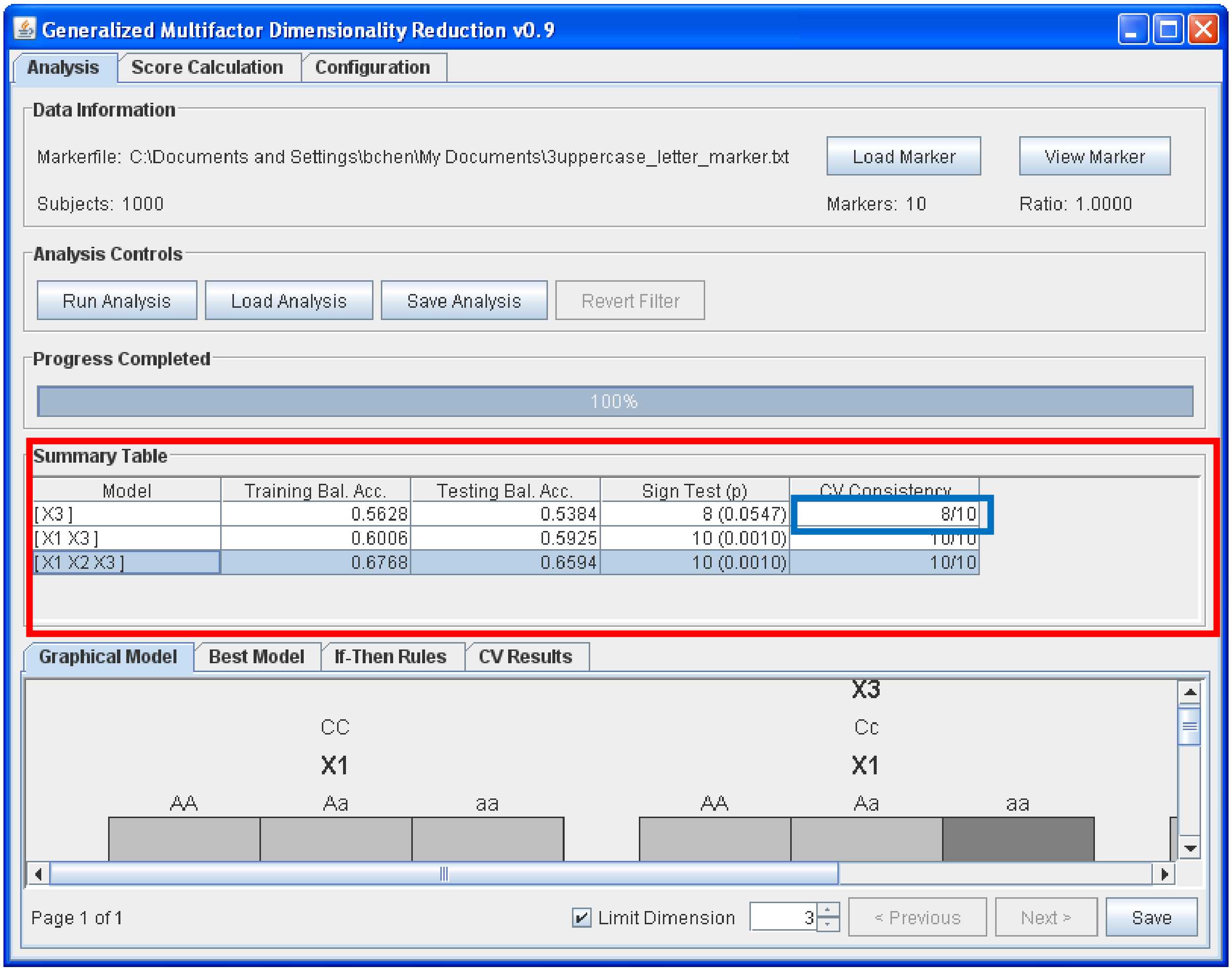
Subjects: 1000　Markers: 10　Ratio: 1.0000

0%

☑ Limit Dimension 3

This tab is the one of the two major domains in GMDR. After the file 3uppercase_letter_marker.txt is loaded, this tab looks like the snapshot above. The Data Information frame summarizes the location and the content of the marker file, which has 1000 subjects, 10 genetic markers, and the ratio of affected and unaffected is 1.

After marker data are loaded and score data are calculated or input, the multifactor dimensionality reduction analysis can be implemented by clicking the button of “Run Analysis”. When no score is selected in the Score Calculation tab — which will soon be introduced – GMDR calculates the default score based on the ratio of affected to unaffected individuals in the sample. The previous analysis can also be reloaded to avoid redundant computation through the button of “Load Analysis”.



Generalized Multifactor Dimensionality Reduction v0.9

Markerfile: C:\Documents and Settings\bchen\My Documents\3uppercase_letter_marker.txt

Subjects: 1000 Markers: 10 Ratio: 1.0000

Summary Table

Model	Training Bal. Acc.	Testing Bal. Acc.	Sign Test (p)	CV Consistency
[X3]	0.5628	0.5384	8 (0.0547)	8/10
[X1 X3]	0.6006	0.5925	10 (0.0010)	10/10
[X1 X2 X3]	0.6768	0.6594	10 (0.0010)	10/10

Page 1 of 1 ☑ Limit Dimension 3

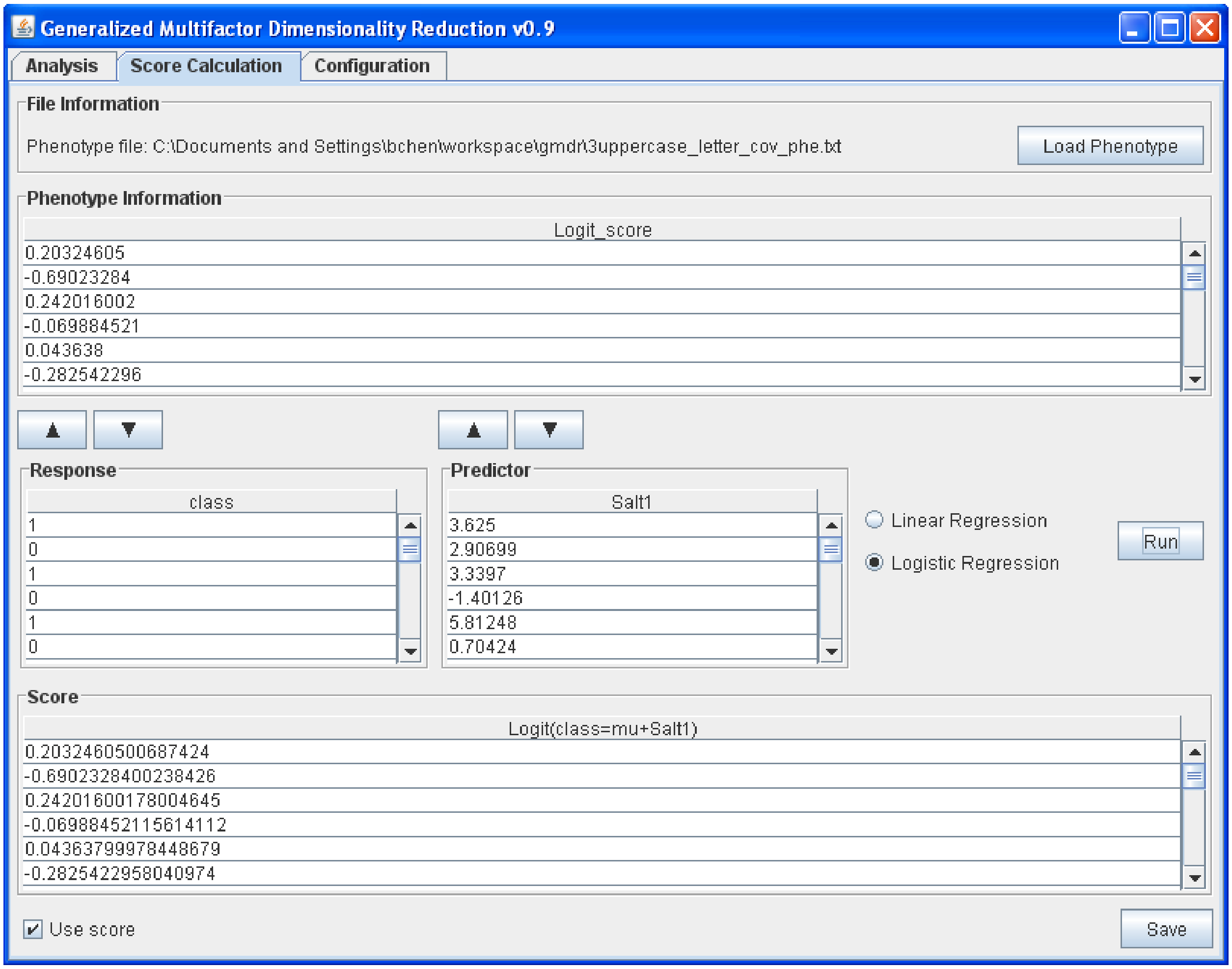
As soon as completing multifactor dimensionality reduction analysis, the Summary Table lists the test statistics, such as Training Balanced Accuracy, Testing Balanced Accuracy, Sign Test, calculated as a summarization of the "winners" in K-fold cross validation. Other statistics can be found in the "Best Model", "CV Results" tabs of the window near the bottom.

IMPORTANT Unless CV Consistency equals K, the user should be cautious when citing the test statistics which are listed in GMDR. For example, for the one-order model, in 10-fold cross-validation, as "X3" won 8 times, and "X2" won twice, the test statistics were eventually calculated based on 8 "X3" models and 2 "X2" models. Consequently, the test statistics for model "X3" were biased. In contrast, for the three-order models, as the model "X1 X2 X3" always won all 10 rounds of cross validation, the test statistics were unbiased. And it is similar to the model "X1 X3".

IMPORTANT To get the accurate test statistics for the best multilocus model, we recommend that users run the Perl script, which will be introduced in the section for Permutation test using the Perl script.

HINT To confirm whether the test statistics are biased, the best way is to check the CV consistency and the count of cross-validation rounds, as illustrated in the blue box. A difference implies biased test statistics.

SCORE CALCULATION TAB



Phenotype file: C:\Documents and Settings\bchen\workspace\gmdr\3uppercase_letter_cov_phe.txt

Logit_score
0.20324605
-0.69023284
0.242016002
-0.069884521
0.043638
-0.282542296

class
1
0
1
0
1
0

Salt1
3.625
2.90699
3.3397
-1.40126
5.81248
0.70424

○ Linear Regression

● Logistic Regression

Logit(class=mu+Salt1)
0.2032460500687424
-0.6902328400238426
0.24201600178004645
-0.06988452115614112
0.04363799978448679
-0.2825422958040974

☑ Use score

For the Generalized Multifactor Dimensionality Reduction method, which, compared with the MDR method, is advanced in that it uses the score statistic to handle both quantitative and dichotomous traits and permits adjustment for covariates. When the phenotype file has been loaded, its content shows in the phenotype information table. Score calculation can then be implemented easily by dragging the variables into their respective niches, say the response table and the predictor's table. Depending on the distribution of the selected response variable, the user can employ either the linear or the logistic regression to build score statistics with selected variables. In general the tab can serve the users in three schemes in terms of score calculation.

SCHEME 1 builds a score statistic with adjustment for the covariates. Having selected the response variable, the user can further drag other phenotypes into the "Predictor" table.

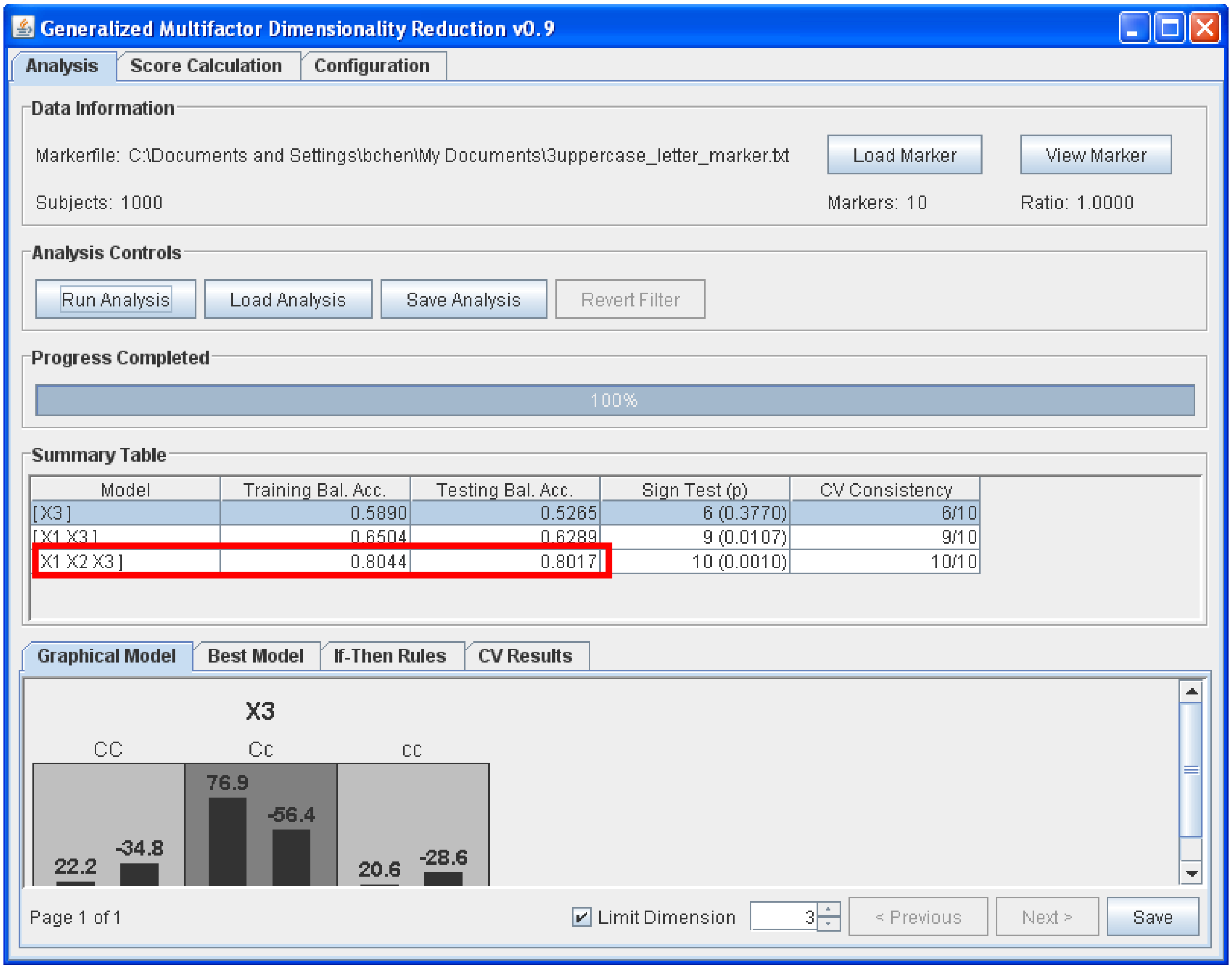
Depending on the distribution of the selected response variable, either linear or logistic regression models can be used to build the score statistic. As illustrated above, we used “class” as the response, and “Salt1” as the predictor, and built the score statistic with a logistic regression model because the response is dichotomous. The title of the score “Logit (class = mu + Salt1)”, which was generated automatically by GMDR, indicates how the score was built: the score was built in the logistic regression model in which class was the response and Salt1 was the covariate.

SCHEME 2 builds a score statistic without adjustment for any covariates. Having selected the response, the user can leave the “Predictor” table empty if without adjustment. Depending on the distribution of the response, the linear regression model or the logistic regression model can be used to build the score. With the example phenotype file, if the user drags the “class” into the response table, chooses “logistic regression”, and clicks “Run”, it builds the score without adjustment.

HINT If one uses “class” as the response without adjustment for covariates, the analysis result should be the same as the one that does not use a score statistic.

SCHEME 3 imports a built score statistic into GMDR. If the user would like to build the score statistic with methods that are not covered in GMDR, it is very simple to import it. Under this scheme, when a user drags the built score into the response table, it is automatically considered the score statistic by GMDR. In the example phenotype file, the column named “logit_score” is a built score statistic, which is the same as the one built in scheme 1.

IMPORTANT The score statistic will not be used until “Use score” is checked. When “Use score” is checked, GMDR runs the analysis with the calculated score statistic. The snapshot below illustrated the result when the calculated score statistic is used. It indicated an enhanced estimate of the test statistics that both Training Balanced Accuracy and Testing Balanced Accuracy are increased. If “Use score” is unchecked, GMDR runs the analysis with the default score which is calculated using the affected status which is specified in the last column in the marker file.



Data Information

Markerfile: C:\Documents and Settings\bchen\My Documents\3uppercase_letter_marker.txt

Subjects: 1000

Markers: 10

Ratio: 1.0000

Analysis Controls

Progress Completed

100%

Summary Table

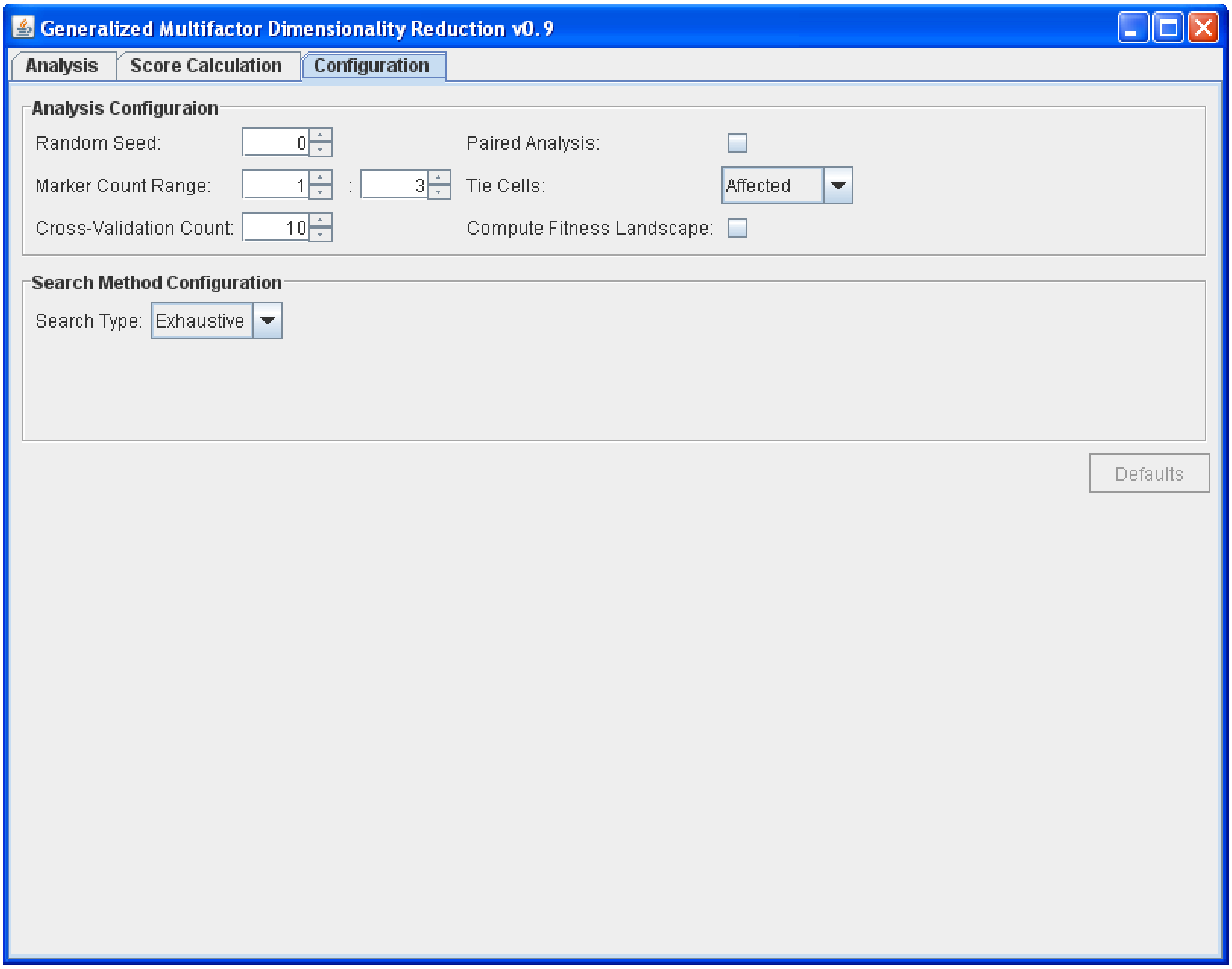
Model	Training Bal. Acc.	Testing Bal. Acc.	Sign Test (p)	CV Consistency
[X3]	0.5890	0.5265	6 (0.3770)	6/10
[X1 X3]	0.6504	0.6289	9 (0.0107)	9/10
X1 X2 X3]	0.8044	0.8017	10 (0.0010)	10/10

Page 1 of 1

☑ Limit Dimension 3

The user can save the calculated score by clicking the “Save” button. Once a calculated score is saved, the score can be reused as an imported score, as described in scheme 3.

CONFIGURATION



Analysis Configuraion

Random Seed: 0

Marker Count Range: 1 : 3

Cross-Validation Count: 10

Paired Analysis: □

Tie Cells: Affected

Compute Fitness Landscape: □

Search Method Configuration

Search Type: Exhaustive

The cross-validation count determines how many subdivisions the sample partitions, and a change of the random seed carries out an alternative partitioning of the sample. Each subdivision serves as the testing set in cross-validation. If the sample size is less than 500, we suggest 5-fold cross-validation; if the sample size is greater than 1,000, we suggest 10-fold cross-validation. Unless the random seed and the number of subdivisions are constant, analysis results may differ.

Paired Analysis: if the dataset is organized pair by pair, such as in discordant sib pair design, checking this option enables every pair of subjects allocated in the same subdivision. This option enables analysis of a discordant sib pair design.

HINT Although the random seed and the count of subdivisions changes the GMDR result, it is recommended to try different random seeds because only a significant association will emerge constantly, irrespective of the variation of the partitioning.

PERMUTATION TEST USING THE PERL SCRIPT

Although the features of GMDR such as model-free and nonparametric methods introduce great flexibility to detect multilocus models, they also lead to unknown distributions of the test statistic. Consequently, *p*-values of the test statistics cannot be evaluated analytically. As the "rule of thumb", permutation is employed in evaluating the *p*-values of statistics in GMDR.

To facilitate the evaluation of the *p*-value for a test statistic, a Perl script, which is interactive and easy to use, has been developed and accompanies the GMDR software. After switching to the directory where the script is, it can be run by typing "perl GMDR_permutatin.pl". Nevertheless, we still give some guidelines which, we think, may help the users run the script smoothly.

The fold for cross-validation: If the sample size is less than 500, 5-fold CV is suggested; or 10-fold if the sample size is greater than 1,000.

The replication for permutation: the replication determines the accuracy of the nominal *p*-value assessed by permutation. If the user is going to get a nominal *p*-value less than 0.001, 1,000 replications are required; to get a nominal *p*-value less than 0.0001, 10,000 replications are required, and so forth.

HINT A couple of ancillary files showing up after the script finished the analysis. The one whose name begins with an underscore (lowercase??) character is the score statistic file which is shuffled to generate the random score statistic in permutation, and the one whose name begins with "GMDR_" saves the result calculated from the permutation procedure.

COMMAND LINE OPTIONS

Although the GUI and the Perl script facilitate GMDR for most routine analyses, the command line options offer further flexibility in customizing the analysis. We highlight some basic usages here by demonstrating in working examples.

SCHEME 1 Run GMDR using the default score

```
java –jar gmdr.jar –min=1 –max=3 3uppercase_letter_marker.txt
```

SCHEME 2 Run GMDR using a built score statistic

```
java –jar gmdr.jar –min=1 –max=3 3uppercase_letter_marker.txt –score=3 –
scorefile=3uppercase_letter_cov_phe.txt
```

It tells GMDR to detect a multilocus model from 1 to 3 by using the score statistic which is the listed in the third column, logit_score, in 3uppercase_letter_cov_phe.txt.

SCHEME 3 Run GMDR while building a score statistic

```
java –jar gmdr.jar –min=1 –max=3 3uppercase_letter_marker.txt –response=2 –predictor=1 –
family=B –phefile=scorefile=3uppercase_letter_cov_phe.txt
```

It tells GMDR to detect multilocus model from 1 to 3 after building the score statistic by using the logistic regression model in which the second column, class, and the first column, Salt1, in 3uppercase_letter_cov_phe.txt serve as the response and the predictor, respectively. “–family=B” tells the response variable following a binomial distribution and GMDR automatically chooses the logistic regression to build the score statistic. If the response variable follows a continuous distribution, one should set -family=C, and then the linear regression is used to build the score statistic.

SCHEME 4 Build a score statistic without adjustment for covariates

```
java –jar gmdr.jar –min=1 –max=3 3uppercase_letter_marker.txt –response=2 –family=B –
phefile=scorefile=3uppercase_letter_cov_phe.txt
```

Everything is the same except that “-predictor=” is excluded.

Other options

-cv=<int>, the default is 10.

-paired, if the dataset is sib pair and each sib pair is organized in a pair.

These two options have been described in the section on Configuration

-forced_search=<comma-separated markers list>, if the user wants to evaluate a specific set of markers, this option might help.

For more options, please check

ACKNOWLEDGEMENTS

GMDR is developed on the original MDR software version 07. The original MDR Java source code was downloaded from Epistasis.org. We greatly appreciate Dr. Jason Moore and his colleagues at the Dartmouth Medical School, for making their MDR Java source code available for this project. This work was funded in part by the National Institutes of Health Grant R01 DA025095.

REFERENCES

Xiang-Yang Lou, Guo-Bo Chen, Lei Yan, Jennie Z. Ma, Jun Zhu, Robert C. Elston, and Ming D. Li. A Generalized Combinatorial Approach for Detecting Gene-by-Gene and Gene-by-Environment Interactions with Application to Nicotine Dependence. Am J Hum Genet 2007, 80:1125-37.

Chen, G. B., Y. Xu, H. M. Xu, M. D. Li, J. Zhu, and X. Y. Lou, 2011 Practical and theoretical considerations in study design for detecting gene-gene interactions using MDR and GMDR approaches. PLoS ONE 6:e16981.