

MDR User's Guide

Version 1.1.0

Jason Moore, Ph.D¹
Principal Investigator

Nate Barney, B.S.
Developer

Peter Andrews²
Developer

¹Jason.H.Moore@Dartmouth.edu

²Peter.C.Andrews@Dartmouth.edu

Contents

1	Introduction	2
1.1	What is MDR?	2
1.2	Further Reading	2
1.3	Obtaining the Software	2
1.4	Minimum System Requirements	2
1.5	Starting the program	3
1.6	Data Format	3
2	Algorithm Overview	4
2.1	Core Algorithm	4
2.1.1	MDR Models	4
2.1.2	Attribute Construction	4
2.1.3	Classification	4
2.2	Cross-Validation	4
3	MDR Analysis	5
3.1	Loading a Data File	5
3.2	Configuring the Analysis	5
3.2.1	Analysis Configuration	5
3.2.2	Search Method Configuration	6
3.2.2.1	Exhaustive	6
3.2.2.2	Forced	6
3.2.2.3	Random	6
3.3	Running the Analysis	7
3.4	Analysis Results	7
3.4.1	Summary Table	7
3.4.2	Graphical Model	7
3.4.3	Best Model	7
3.4.4	If-Then Rules	7
3.4.5	CV Results	7
3.4.6	Dendrogram	7
3.4.7	Fitness Landscape	7
4	Data Filtering	8
5	Attribute Construction	9

Chapter 1

Introduction

1.1 What is MDR?

Multifactor Dimensionality Reduction (MDR) is a nonparametric and genetic model-free alternative to logistic regression for detecting and characterising nonlinear interactions among discrete genetic and environmental attributes. The MDR method combines attribute selection, attribute construction, classification, cross-validation, and visualization to provide a comprehensive and powerful data mining approach to detecting, characterizing, and interpreting nonlinear interactions.

1.2 Further Reading

- For a recent review of the MDR method and its application to real data, see [1].
- For new ideas about incorporating MDR into a flexible computational framework for detecting interactions, see [2].
- For additional information about the method and available software, see [3]
- For our online discussion about gene-gene interactions, see [4]

1.3 Obtaining the Software

MDR is available as an open-source (GPL) software package. It is a cross-platform program written entirely in Java. It is available from the MDR web site [3]. You may also contact Dr. Jason Moore for a copy of the software or source code if you experience difficulties downloading it from the web site. Development of this project was funded by NIH grant AI59694 (PI - Moore).

1.4 Minimum System Requirements

- Java Runtime Environment, version 5.0 or higher [5]
- 1 GHz Processor
- 256 MB Ram
- 800x600 screen resolution

1.5 Starting the program

After unpacking the archive, there will be a file called `mdr.jar`. Under most operating systems, simply double-clicking this file will be sufficient to start the program. However, there are reasons a user may wish to start the program from the command line. To do so, open a command shell and navigate to the directory containing `mdr.jar`. Issue the command:

```
java -jar mdr.jar
```

MDR can consume a large amount of memory, depending on its configuration. If MDR needs more memory than is initially allocated to the Java Runtime Environment, issue a command like this:

```
java -Xmx1024M -jar mdr.jar
```

This command will tell the JRE to set the maximum heap size to 1,024 MB, or 1 GB. This value may be modified as appropriate.

There are also command-line options available for MDR itself, if the user wishes to run in batch mode with no graphical user interface. These options are described in a later section of this document.

1.6 Data Format

MDR expects data files to be in a specific format. The file `MDR-SampleData.txt`, included in the distribution, provides an example. The definition of the format is as follows:

- All fields are tab-delimited.
- The first line contains a header row. This row assigns a label to each column of data. Labels should not contain whitespace.
- Each following line contains a data row. Data values may be any string value which does not contain whitespace.
- The right-most column of data is the class, or status, column. The data values for this column must be 1, to represent "Affected" or "Case" status, or 0, to represent "Unaffected" or "Control" status. No other values are allowed.

Chapter 2

Algorithm Overview

2.1 Core Algorithm

2.1.1 MDR Models

2.1.2 Attribute Construction

2.1.3 Classification

2.2 Cross-Validation

Chapter 3

MDR Analysis

3.1 Loading a Data File

The first step in any MDR analysis is to load a data file. First, format your data as described in Section 1.6. Then, start the program. You will see a button labelled **Load Datafile**. Clicking this button brings up a file browser, which allows you to navigate to and select your data file.

After loading the file, MDR will display some statistics about it. The following fields will be displayed at the top of the main tab:

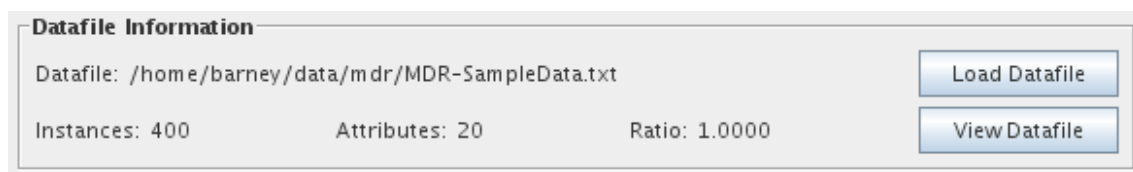


Figure 3.1: Data file information section

Datafile: The full path of the current data file.

Instances: The number of instances¹ in the data set.

Attributes: The number of attributes² in the data set. This does not include the class attribute.

Ratio: The ratio of affected instances to unaffected instances.

The **View Datafile** button will, when clicked, display the data set in a separate window. This allows visual inspection to ensure the data is correct. This display is read-only. To make changes, you will have to edit the data file in a text editor and reload it in MDR.

3.2 Configuring the Analysis

Now that you've got a data set loaded, the next step is to configure the program to perform the analysis. At the top of the window, there are several tabs. Click the **Configuration** tab. There are two sections of controls on this tab: **Analysis Configuration** and **Search Method Configuration**.

3.2.1 Analysis Configuration

This section contains controls that configure options global to all MDR analyses. In order, they are:

¹or rows, or individuals, etc.

²or columns, or variables, etc.



Analysis Configuration

Random Seed:

Attribute Count Range: :

Cross-Validation Count:

Paired Analysis: ☐

Tie Cells:

Compute Fitness Landscape: ☐

Figure 3.2: Analysis configuration controls

Random Seed: The seed used to initialize the random number generator. When a given seed is provided, the analysis should always return exactly the same results. This is only used when partitioning a data set for cross-validation. (Unless you're using the Random Search method. See Section 3.2.2.3.) Change the seed if you wish to try a different partition of the data.

Attribute Count Range: The minimum number of attributes to consider together, as well as the maximum number. MDR will analyze all combinations of attributes for each value from the minimum number, up to and including the maximum number.

Cross-Validation Count: The number of intervals into which to divide the data, for the purposes of cross-validation. Setting this to 1 disables cross-validation.

Paired Analysis: If this is checked, MDR alters the way it partitions the data set, to accomodate a paired data set. It makes sure adjacent case/control pairs always end up in the same cross-validation interval. This option is not available if the data set is not organized in such a way as to support it.

Tie Cells: Sometimes, during an MDR analysis, a model cell contains numbers of cases and controls which are exactly proportional to the numbers in the entire data set. This is called a Tie Cell. MDR cannot decide what to do in these cases, so the user must select whether these cells should be called Affected, Unaffected, or Unknown. The default is Affected, in order to minimize the false negative rate.

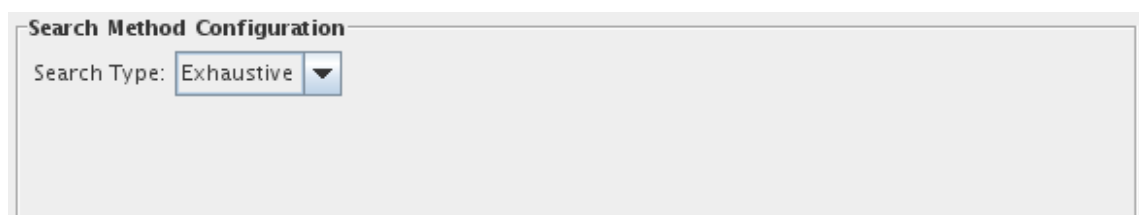
Compute Fitness Landscape: The fitness landscape is a display of all models considered during the run, and the associated fitness (**balanced accuracy** For each class, some information) of each. This can consume a good deal of memory, so it is off by default. If you wish to see this information, check this box.

3.2.2 Search Method Configuration

There are three search methods available for driving MDR.

3.2.2.1 Exhaustive

For each attribute count specified, exhaustively examine each combination of attributes. This search method has no options.




Search Method Configuration

Search Type:

Figure 3.3: Exhaustive search method configuration

3.2.2.2 Forced

Examine only one attribute combination. The combination must be specified in the provided text field as a comma-separated list of attribute labels. The labels are case-sensitive.

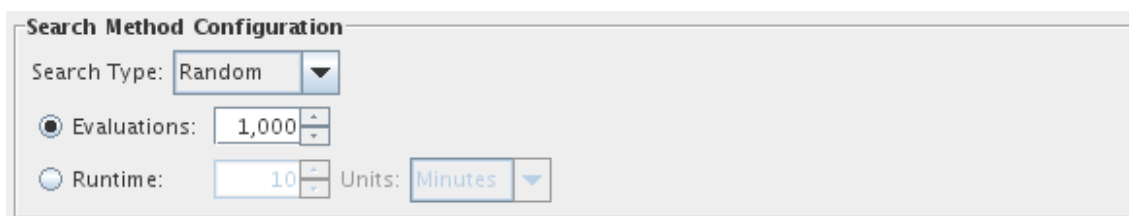


The image shows a 'Search Method Configuration' dialog box. It has a title bar with the text 'Search Method Configuration'. Inside, there is a 'Search Type:' label followed by a dropdown menu set to 'Forced'. Below this is a 'Forced Attribute Combination:' label followed by a text input field containing the text 'X1,X6,X8'.

Figure 3.4: Forced search method configuration

3.2.2.3 Random

For each attribute count specified, examine random combinations. There are two options here.



The image shows a 'Search Method Configuration' dialog box. It has a title bar with the text 'Search Method Configuration'. Inside, there is a 'Search Type:' label followed by a dropdown menu set to 'Random'. Below this are two radio button options. The first is 'Evaluations:' with a text input field set to '1,000'. The second is 'Runtime:' with a text input field set to '10' and a 'Units:' dropdown menu set to 'Minutes'.

Figure 3.5: Random search method configuration

Evaluations: For each attribute count specified, evaluate a given number of random combinations.

Runtime: For each attribute count specified, evaluate random combinations for a given amount of time.

3.3 Running the Analysis

Once you've loaded a data set and set the configuration appropriately, the next step is to start the analysis. Switch back to the **Analysis** tab, and click the **Run Analysis** button. The button will change to **Stop Analysis**, and other controls will become disabled. The analysis will begin, and the progress bar will begin to move to the right.

The analysis can be stopped at any time by clicking the **Stop Analysis** button, but MDR cannot resume a stopped analysis. The results that were available when the analysis was stopped will continue to be available.

3.4 Analysis Results

3.4.1 Summary Table

When each attribute count finishes, information about the best model discovered for that attribute count is entered as a row in the summary table. The fields are described below.

Highlighting a row in the summary table causes the results tabs below to display detailed information for that row. (Some tabs display information about the entire run, and those tabs will remain empty until the run finishes.)

Summary Table				
Model	Training Bal. Acc.	Testing Bal. Acc.	Sign Test (p)	CV Consistency
X1	0.5539	0.5175	7 (0.1719)	9/10
X1 X8	0.6083	0.5650	9 (0.0107)	8/10
X1 X6 X8	0.8725	0.8702	10 (0.0010)	10/10
X1 X2 X6 X8	0.8756	0.8489	10 (0.0010)	9/10

Figure 3.6: Analysis summary table

Model The attributes that participated in the best model discovered.

Training Bal. Acc. The average training **balanced accuracy** across all **cross-validation** intervals. (If cross-validation is not being used, this column is simply the balanced accuracy for the model over the entire data set.)

Testing Bal. Acc.

Sign Test (p)

CV Consistency

3.4.2 Graphical Model

3.4.3 Best Model

3.4.4 If-Then Rules

3.4.5 CV Results

3.4.6 Dendrogram

3.4.7 Fitness Landscape

Chapter 4

Data Filtering

Chapter 5

Attribute Construction

Bibliography

- [1] Moore JH. Computational analysis of gene-gene interactions using multifactor dimensionality reduction. *Expert review of molecular diagnostics*, 4(6):795–803, November 2004.
- [2] Moore JH, et al. A flexible computational framework for detecting, characterizing, and interpreting statistical patterns of epistasis in genetic studies of human disease susceptibility. *Journal of Theoretical Biology*, 241(2):252–61, July 2006.
- [3] <http://www.multifactordimensionalityreduction.org/>.
- [4] <http://compgen.blogspot.com/>.
- [5] <http://www.java.com/>.