

WinEDMA User's Guide

Version 1.0.1 beta

Copyright © 2002

Release date: March 31, 2002

Theodore M. Cole III
Department of Basic Medical Science
School of Medicine
University of Missouri – Kansas City
2411 Holmes St.
Kansas City, MO 64108
USA

E-mail: colet@umkc.edu
WinEDMA WWW: <http://c.faculty.umkc.edu/colet>
EDMA WWW: <http://faith.med.jhmi.edu>

NOTE

WinEDMA version 1.0.1 beta (release date March 31, 2002) is a *test release* of both the executable program and the accompanying documentation. A user of this release might better be called a “tester.” Some of the procedures are not yet available, and the corresponding portions of the manual are not yet included. For procedures that are available for use, a great deal of care has been taken to check the program for correct function. However, the tester must be aware that programming errors may still be present, and these may affect both the program’s runtime performance and the correctness of program output. Therefore, no warranties are either expressed or implied, and the tester is advised to use the program *at his or her own risk!*

The point of making a beta-level release of *WinEDMA* is so the developer can receive input from testers regarding bugs, suggestions for improvements in the user interface, adequacy and clarity of the documentation, and so forth. Testers will be able to provide this information via the Internet beginning in April 2002.

CITING USE OF THIS PROGRAM

If this program is used in preparing a published study, please provide the following citation:

Cole, TM III (2002) *WinEDMA*: Software for Euclidean distance matrix analysis.
Version 1.0.1 beta. Kansas City: University of Missouri – Kansas City School of Medicine.

Mention of where the program can be obtained (c.faculty.umkc.edu/colet or faith.med.jhmi.edu) would also be greatly appreciated.

Contents

In the following list of *WinEDMA* procedures and help topics, “available” labels indicate procedures that can be used in the current program version (*WinEDMA* version 1.0.1 beta, release date March 31, 2002). Procedures labeled “under construction” are untested or not yet complete.

1.	Installation	1
2.	Hardware and Operating System Requirements	2
3.	General Organization of <i>WinEDMA</i>	3
	The About Page	3
	A Note about On-Line Help	4
	The Procedure Pages	4
	Other Pages	6
	User-Interface Basics	7
	Navigating the User Interface	9
	Running an Analysis	
4.	Input File Formatting	
	Program Limitations	
	The File Header	
	Missing Data	
	What If File Headers Disagree?	
	Sample Files	
5.	Subset Files	
	The File Header	
	Sample Files	
	A Note about Program-Supplied Landmark Labels	
6.	First Principles of Euclidean Distance Matrix Analysis	
	Some Initial Definitions	
	Landmark Data	
	The “Nature Space”	
	Coordinate-System Invariance and More Precise Definitions	
7.	The ESTIMATION Procedure (<i>available</i>)	
	What the Procedure Does	

	User Input
	Understanding the Printed Output
8.	The FORM Procedure (<i>available</i>)
	What the Procedure Does
	User Input
	Understanding the Printed Output.....
	Understanding the Graphics Output.....
9.	The INFLUENTIAL Procedure (<i>available</i>)
	What the Procedure Does
	User Input
	Understanding the Graphics Output
10.	The GROWTH Procedure (<i>available</i>)
	What the Procedure Does
	User Input
	Understanding the Printed Output
	Understanding the Graphics Output
11.	The SHAPE Procedure (<i>available</i>)
	What the Procedure Does
	User Input
	Understanding the Printed Output
	Understanding the Graphics Output
12.	The ASYMMETRY Procedure (<i>under construction</i>)
13.	The R&S Procedure (<i>under construction</i>)
14.	The ORDINATION Procedure (<i>available</i>)
	What the Procedure Does
	User Input
	Understanding the Printed Output
	Understanding the Graphics Output
15.	The CLUSTER Procedure (<i>available</i>)
	What the Procedure Does
	User Input
	Understanding the Printed Output

	Understanding the Graphics Output
16.	The CLASSIFICATION Procedure (<i>under construction</i>)
17.	The RANDOM Procedure (<i>available</i>)
	What the Procedure Does
	User Input
	Understanding the Printed Output
18.	The PHYLOGENY Procedure (<i>available</i>)
19.	The HYPOTHETICAL Procedure (<i>available</i>)
	What the Procedure Does
	User Input
	Understanding the Printed Output
20.	The CLIQUE Procedure (<i>available</i>)
	What the Procedure Does
	User Input
	Understanding the Printed Output
21.	The FUZZY Procedure (<i>under construction</i>)
22.	The EDITOR (<i>available</i>)
23.	The UTILITIES Procedures
	The Outliers Utility (<i>under construction</i>)
	The PermSub Utility (<i>under construction</i>)
	The Convert Utility (<i>under construction</i>)
	The Screen Utility (<i>under construction</i>)
	The Scale Utility (<i>under construction</i>)
	The ModTest Utility (<i>under construction</i>)
	The MeasErr Utility (<i>under construction</i>)
	The Header Utility (<i>under construction</i>)
	The DBIO Utility (<i>under construction</i>)
24.	Program Configuration (<i>under construction</i>)
25.	Error Messages
26.	Bibliography
	Methodological
	Applications

	Other References
27.	Distribution and User Support
28.	Future Plans
	APPENDIX: Updates and bug fixes

1. Installation.

Test versions of *WinEDMA* (version 1.0.1 beta; release March 31, 2002) that are supplied via the Internet or on diskette come archived in “zip” format. Zip formatting is a standard method for file archiving that accomplishes two things: 1) it collects the program, documentation, and example files in a single file (EDMA.zip), which simplifies file transfer; and 2) it compresses the archived files, which speeds downloading or, alternatively, allows everything to fit on a single floppy disk.

Test-version users should follow these steps to install *WinEDMA*. *Note that if this process is repeated in the future, with an updated version of the archive, the existing copies of the program, the documentation files, and the sample files will be overwritten! Other files, such as the user’s own data, will be unaffected.*

Step 1:

Make a main *WinEDMA* directory on the computer’s hard drive (e.g., C:\WinEDMA) and either download or copy the EDMAzipped.exe file to that directory. In the current version, this directory will be the default for input and output files.

Step 2:

Double-click on the EDMAzipped.exe icon. The self-executing file will automatically “unzip” into the executable program file (WinEDMA.exe) and a folder (subdirectory) called “WinEDMA.doc” that contains the program documentation. No matter what the main *WinEDMA* directory is called, *the current version of the program requires that the documentation subdirectory be called “WinEDMA.doc”*. There will be another folder (subdirectory) called “Samples”, which contains a few examples of data files and subset files.

Step 3:

Open the WinEDMA.doc folder. It contains a file called WinEDMA.pdf, which is an Adobe Acrobat file. This file can be read and printed using the Adobe Acrobat Reader, which can be downloaded at not cost from www.adobe.com. The user should print the file and read the initial sections of it before trying to use the program.

2. Hardware and Operating System Requirements.

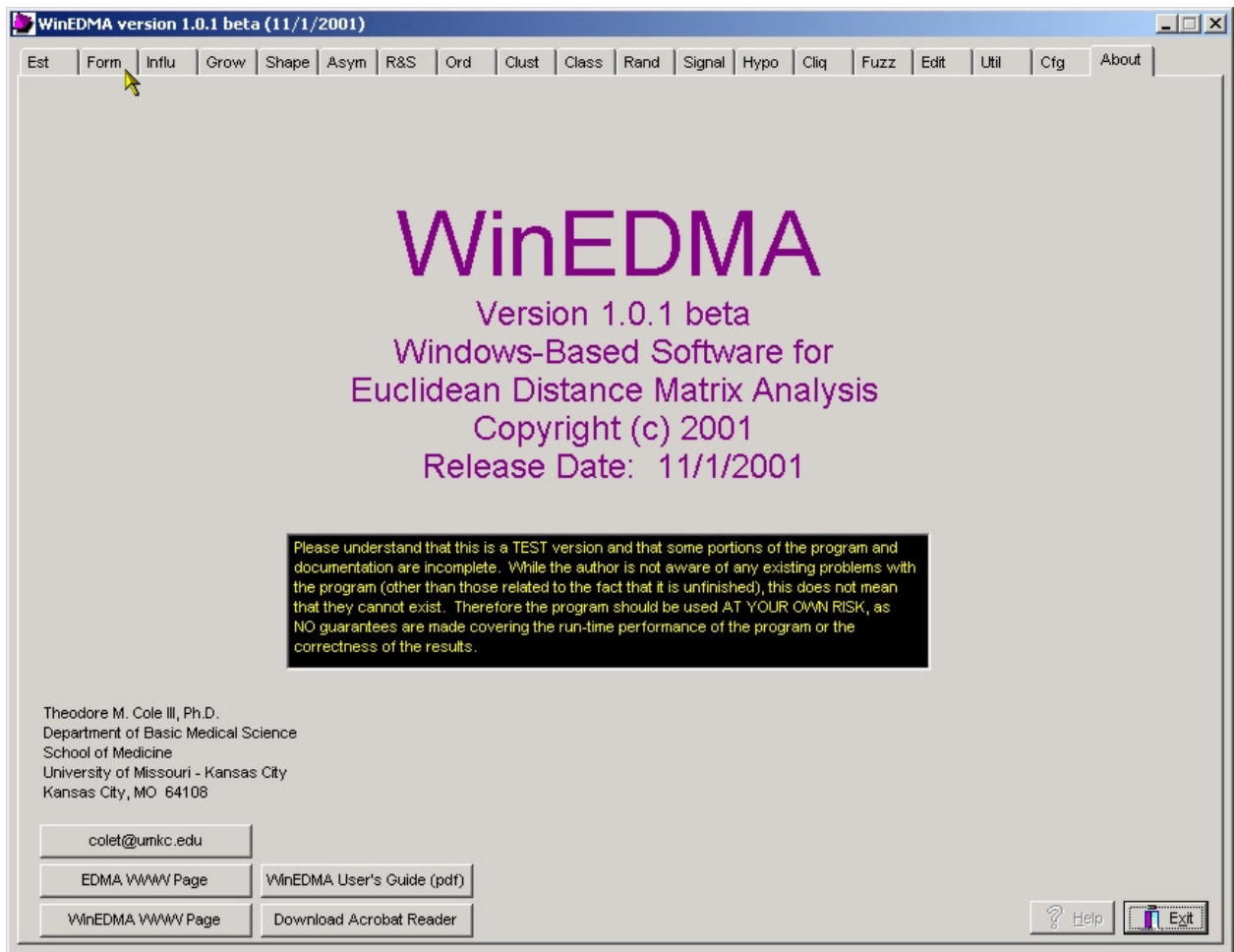
- IBM-compatible computer with 486 or greater processor
- Minimum of 16 MB of RAM (at least 64 MB is preferred)
- Minimum of 5 MB of free hard-disk space
- Microsoft Windows 95/98/Me/2000/NT

3. General Organization of *WinEDMA*.

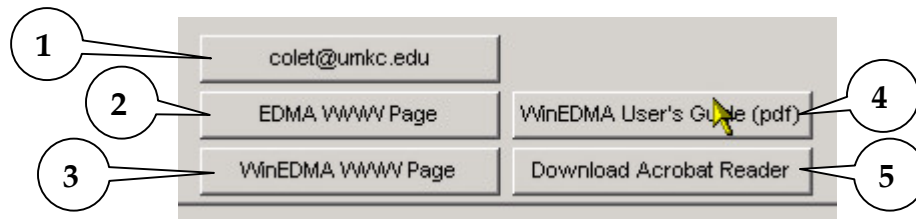
There are different components to the *WinEDMA* program that correspond to different types of analyses (*e.g.*, form comparison, growth comparison, shape comparison, etc.). Each type of analysis has its own page on the user interface. The different pages are arranged like folders in a filing cabinet. Each page is accessible by using the mouse to click on the corresponding tab along the top. When a tab is clicked, the selected page moves to the front.

The About Page

The *About* page is the page that first appears when *WinEDMA* is executed. It serves to identify the program, its version, and its release data, as well as providing access to the program documentation and e-mail access to user support.



At the lower left-hand corner of the page is a selection of buttons:



1. Sends e-mail to the author (with the user's default e-mail program)
2. Accesses the main EDMA web page, maintained by Joan Richtsmeier's lab
3. Accesses the *WinEDMA* web page, maintained by the author
4. Opens the *WinEDMA User's Guide* (this document) using the Adobe Acrobat Reader
5. Accesses the download page for the latest version of the Adobe Acrobat Reader

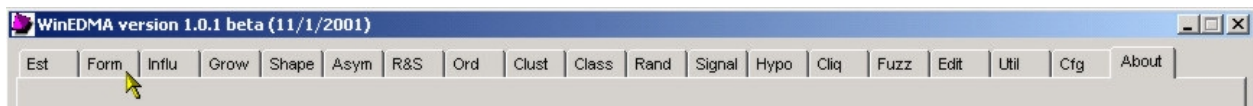
At the lower right-hand corner of the page are two other buttons. The first is the *Help* button, which provides access to program documentation (*under construction*). The *Exit* button shuts the program down.

A Note about On-Line Help:

In this manual, the description of each procedure is organized into three sections: 1) *What the Procedure Does*; 2) *User Input*; and 3) *Understanding the Output*. The on-line documentation is not intended to be a complete restatement of all of the information found here. Instead, it is intended to provide the user with necessary information while the program is running. Therefore, the on-line help reproduces only the *User Input* sections of each procedure, as well as abbreviated versions (*i.e.*, with limited examples) of the sections on formatting input and subset files.

The Procedure Pages

The Procedure pages that are currently in place are as follows (from left to right across the top of the *WinEDMA* window):



Est: Performs the ESTIMATION procedure. Reads a set of landmark coordinate data and calculates sample statistics, including the XY(Z) coordinates of the mean form, the mean form matrix, and the among-landmark variance-covariance matrix (Lele, 1993). Each estimate can be saved in its own output file.

Form: Performs the FORM procedure. Reads two sets of landmark coordinate data (numerator and denominator samples) and performs a form-difference analysis (Lele & Richtsmeier, 1991). Output includes sample statistics, the form-difference matrix, the *T*-statistic for form-difference testing (with optional bootstrap distribution), and (optional)

marginal confidence intervals for the form-difference matrix elements. The form-difference matrix can be saved its own output file.

Influ: Performs the INFLUENTIAL procedure. Reads a form-, growth-, or shape-difference matrix and use Cole & Richtsmeier's (1998) graphic approach to the analysis of influential landmarks (Lele & Richtsmeier, 1992).

Grow: Performs the GROWTH procedure. Reads four sets of landmark coordinate data (numerator and denominator samples, each stratified into older and younger subsamples) and performs a growth-difference analysis (Richtsmeier & Lele, 1993). Output includes sample statistics, the sample growth matrices, the growth-difference matrix, the G-statistic for growth-difference testing (with optional bootstrap distribution), and (optional) marginal confidence intervals for the growth-difference matrix elements. The growth matrices and growth-difference matrix can be saved their own output files.

Shape: Performs the SHAPE procedure. Reads two sets of landmark coordinate data and performs a shape-difference analysis (Lele & Cole, 1996). Sample mean forms are scaled according to a user-specified scaling factor. Output includes sample statistics, the sample form and shape matrices, the shape-difference matrix, a scale-difference statistic, the Z-statistic for shape-difference testing (with optional bootstrap distribution), and (optional) marginal confidence intervals for the shape-difference matrix elements. The shape-difference matrix can be saved to its own output file.

Asym: Performs the ASYMMETRY procedure, which is under construction. Reads a set of landmark coordinate data and analyzes patterns of directional asymmetry, fluctuating asymmetry, or antisymmetry (Palmer, 1996).

R&S: Performs the R&S procedure, which is under construction. Reads two or more sets of landmark coordinate data and analyzes shape variation following Rao & Suryawanshi's (1996,1998; Rao, 2000) methods.

Ord: Performs the ORDINATION procedure. Reads one or more sets of landmark coordinate data and computes a dissimilarity matrix that can measure form, shape, growth, or dimorphism differences. This matrix is used as the basis for ordination (metric scaling) in the form of a principal coordinates analysis. Output includes the dissimilarity matrix, the scaling factor (if applicable) for each observation, eigenvalues and eigenvectors (principal-axis scores), histograms of scores, and (optional) correlations between scores and interlandmark distances.

Clust: Performs the CLUSTERING procedure, which is under construction. Reads a set of landmark coordinate data or a dissimilarity matrix and performs a variety of hierarchical (UPGMA, single-linkage, complete-linkage, and neighbor-joining algorithms, as well as consensus dendrograms) and nonhierarchical cluster (minimum-spanning tree and K-means) analyses. The dissimilarities can be based on form, shape, growth, or dimorphism. Dendrogram plots can be output either to a printer or in a variety of graphics formats.

Class: Performs the CLASSIFICATION procedure, which is under construction. Reads landmark coordinate data for one or more observations with unknown sample affinities. Compares each unknown to a set of reference samples, and computes the probability that each unknown belongs to each of the samples. Uses various similarity criteria and cross-validation to compute the probabilities.

Rand: Performs the RANDOM procedure. Reads files of sample estimates (e.g., the XY(Z) coordinates of the mean form and the among-landmark variance-covariance matrix) and generates random data under a multivariate-normal perturbation model (Lele, 1993; Lele & Cole, 1996). Output an XYZ-format set of randomly-generated landmarks.

Signal: Performs the SIGNAL procedure, which is under construction. Reads a directory file (containing the names of different XYZ files), computes a dissimilarity matrix based on form, growth, or shape, and then compares a dendrogram constructed from the matrix with a cladogram topology supplied by the user (Cole et al., 1999, 2002).

Hypo: Performs the HYPOTHETICAL procedure. Reads a starting form (in XYZ or MXYZ format) and a transformation matrix (form-difference, growth-difference, or user-specified). A hypothetical form (Richtsmeier & Lele, 1993) is then generated, at a scale specified by the user. The dimensionality of the hypothetical form is checked, and the coordinates of the hypothetical form are output to a file.

Cliq: Performs the CLIQUE procedure. Reads coordinates of two objects and finds landmark subsets where the shapes are most similar (Cole et al., 1998, 1999), where “similarity” is measured relative to a user-specified tolerance level. Output includes information about the membership of landmarks in cliques.

Fuzz: Performs the FUZZY procedure, which is under construction. Reads a file of XY(Z) coordinates containing repeated measurements for some or all landmarks. Computes and outputs the MLE of the mean for “fuzzy” landmarks (Valeri et al., 1998). Can also be used to analyze measurement error in ordinary landmarks (Corner et al., 1992), although the *MeasErr* utility will soon be available for that purpose.

Other Pages:

Edit: A text editor that allows viewing, editing, and printing of text files while *WinEDMA* is in use.

Util: A series of utilities, primarily for the manipulation of data files. These include:

Outliers: For detection of outlying observations and landmarks.

PermSub: For applying a subset file to a data file, resulting in a new file with fewer landmarks.

Convert: For converting *WinEDMA* files to formats used with other morphometrics software (e.g., *Morpheus* or *Morphometrika*) and vice-versa.

Screen: For screening a data set to find observations with missing values (which are not allowed for many EDMA procedures).

Scale: For scaling observations in a data set according to some user-specified scaling factor.

ModTest: For assessing the suitability of a multivariate-normal perturbation model to the description of an empirical data set.

MeasErr: For analysis of measurement error, where repeated measurements have been collected from a stationary object.

Header: For standardizing header formats (e.g., landmark and observation labels) across multiple data files.

DBIO: For transferring data between *WinEDMA* format and various database Formats.

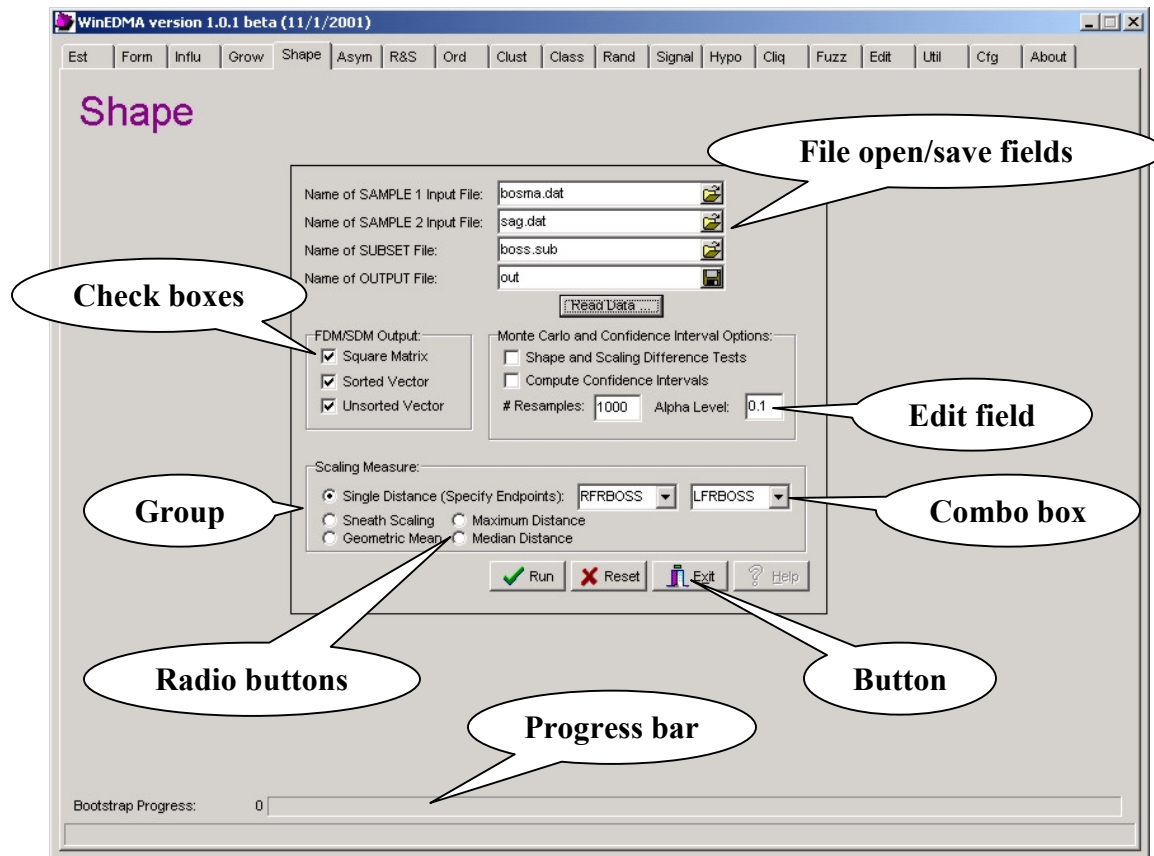
MScribe: For collection of landmark coordinate data using the MicroScribe Digitizer (Immersion Corporation).

Cfg: A page of configuration parameters for specifying program and help-file directories, formatting output files, etc.

User-Interface Basics:

Experienced users of Windows-based software may not need to consult this section; however, less-experienced users might find it helpful.

Throughout the documentation, reference will be made to edit fields, check boxes, radio buttons, and other controls in the user interface. To make references to these controls clearer, examples are illustrated here:



Buttons: Buttons are pressed to perform an action such as reading data, running a procedure, opening a dialog box (see below), asking for help, or exiting the program. The actions of specific buttons are described below, under "Running an Analysis."

Edit fields: Edit fields are rectangular areas that accept input from the keyboard. Once text is entered into an edit field, nothing else needs to be done (in other words, pressing *Enter* is not necessary).

File open/save fields: These are edit fields with built-in buttons, used for entering file names. The buttons either have a file icon or a disk icon, for opening or saving files, respectively:

The user can type file names directly into the edit field, or can select files from dialog boxes (see below) that will open when the associated buttons are pressed.

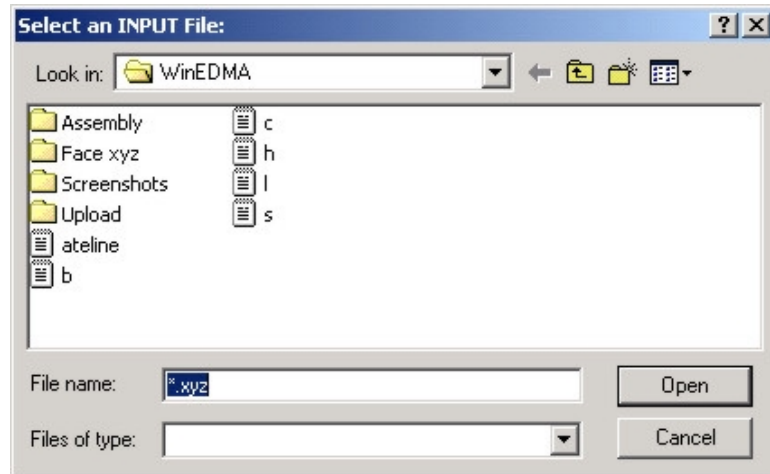
Groups: A group is a collection of controls that is enclosed in an “engraved” border. The controls are related to a single task, such as setting parameters for bootstrapping or file output.

Check boxes: Check boxes direct the program to perform optional analyses and outputs. A check box is a square that is empty when not selected and filled with a check mark when selected. In some cases, a check box is contained in a group with other controls, and the implementation of those controls depends on the state of the check box. For example, in the “Bootstrap options” group on the Form page, information about the reference sample (radio buttons) and the number of resamples (edit field) are immaterial if the “Perform bootstrap” check box is not selected. It is possible to focus on a check box and select/deselect it with the keyboard, but it is more convenient to click on it using the left mouse button.

Radio buttons: Radio buttons are used to select among multiple options when only one of the options can be implemented at once. For example, in the “Bootstrap options” box on the Form page, the reference sample can be either “Numerator” or “Denominator”, but not both. The radio buttons are controlled by clicking on them using the left mouse button.

Combo boxes: Combo boxes have an edit field for typing and a down arrow to the right. In WinEDMA, combo boxes ordinarily provide the user with a way of selecting landmarks (e.g., as endpoints of a distance used as a scaling measure). The user may type a landmark name into the edit field, or may use the down arrow to access a list of choices. The user may scroll through the list and then make a choice using the left mouse button. The list of choices will always be generated automatically by the program.

Dialogs: Dialogs are graphic interfaces that are most often used to allow the WinEDMA user to select file names. They can display all of the files in the current directory. They can also display a limited selection of files, according to the file type (=extension), as in the following example where only “.xyz” files are listed. Different extension “filters” can be specified using the “Files of type” combo box. The user can select a file by clicking on its name with the left mouse button or by typing in the “File name” field. The *Open* button is then clicked and the file is read.



Other dialog boxes (within the EDITOR) allow the user to control printing and select fonts. Still others are used for controlling graphics output.

Messages: Messages will occasionally appear while the program is in use. They require the user to click an “OK” button before doing anything else. Messages always appear to let the user know when an analysis has finished running. Otherwise, messages usually alert the user to errors.

Progress bars: Progress bars are not really “controls”, as they cannot be manipulated by the user. They are simply used to give the user some visual feedback as to the progress of an ongoing analysis. For some of the analyses, the computations are fast enough that a progress bar is unnecessary, so they are only active during computationally-intensive analyses such as bootstrapping and calculation of confidence intervals.

Navigating the User Interface:

If a control on the user interface is ready to receive some input from the user (typing with the keyboard or a mouse click), that part is said to have the *focus*. Only one item can have the focus at a time. To change the focus, the user may use the *Tab* key when moving from one control to another (e.g., between edit fields, between radio buttons, etc.). The “tab order” of a page’s controls is set so the focus moves in a logical progression. The focus cannot be moved to a control that is disabled. Pressing the *Enter* key will not move the focus to the next control; however, if a button (e.g., *Run*, *Reset*, or *Exit*) has the focus, pressing *Enter* has the same effect as clicking the button with the mouse.

To exit the program, use the *Exit* button on any of the procedure pages.

Running an Analysis:

When a user selects a procedure page (e.g., *Form*, *Grow*, etc.), some of the interface controls will be immediately accessible, while others will be disabled and will appear “grayed.” Generally speaking, the controls that are immediately accessible include file open/save edit fields and the *Read Data...* button. The other controls that are enabled at this point are the *Reset* button, which erases the edit fields and resets all of the other controls to their (disabled) default values, and the *Exit* button, which exits the program completely.

When the file names have been specified, the user clicks the *Read Data...* button to load the data into memory. When the data are successfully read, the remainder of the controls are enabled, and may be set to suit the user. Finally, the *Run* button is pressed to execute the procedure. When the procedure is complete, the user is notified. The page is then disabled, the edit fields are erased, and all other controls are set to their defaults and disabled.

Note that the output file is not initialized until the *Run* button is pressed. That means that simply reading the data and adjusting the program options will not overwrite a previously written file of the same name, losing the information in that file.

4. Input File Formatting.

With this release of *WinEDMA* (version 1.0.1 beta), there are specific types of files [other than arrays of $XY(Z)$ coordinates] that can be output by some program components and input into others. For example, the ESTIMATION procedure can be used to estimate the mean form and the variance-covariance matrices from a sample of raw data. These estimates are written to files that can then be read by the RANDOM procedure, so that the variation can be visualized using randomly-generated data. Some of these files can be used with the *EDMASketch* program, which is currently being developed.

Program Limitations:

Previous versions of *WinEDMA* were limited to 200 landmarks and 1000 observations. However, this version of *WinEDMA* uses dynamic arrays for the storage of nearly all data sets and intermediate results. With dynamic arrays, the maximum limits for numbers of landmarks and observations are limited only by the available RAM on the user's computer.

The File Header:

The data in every type of *WinEDMA* input file is preceded by a set of lines collectively called the *file header*. Each header has three required components, each placed on a separate line of text (with no blank lines between):

Line 1: This is an identifying *title* for the input file (limited to a single line of text). It will be copied to output files (where it can be modified using the text editor), so that the user can keep track of the data that have been included in each analysis.

Line 2: This line contains a *file type code* (consisting of two to four letters) that identifies the contents of a file (e.g., whether it is a mean form, a set of raw data, a growth matrix, etc.). The codes are not case-sensitive. The file types are as follows (N = number of observations; K = number of landmarks; D = number of dimensions):

- *DDM* - a $K \times K$ dimorphism-difference matrix
- *DM* - a $K \times K$ dimorphism matrix
- *DIR* - a directory file, containing the file names for different XYZ files to be included in a single analysis (e.g., classification or phylogenetic analyses)
- *DIST* - an $N \times N$ matrix of dissimilarities among observations
- *DVCV* - a $D \times D$ variance-covariance matrix (among coordinate axes)
- *FDM* - a $K \times K$ form-difference matrix

- *FM* - a $K \times K$ form matrix.
- *FUZZ* - a "stacked" file containing two- or three-dimensional landmark coordinate data, with repeated measurements of one or more landmarks.
- *FVCV* - a full-rank $K \times K$ variance-covariance matrix (among landmarks).
- *GDM* - a $K \times K$ growth-difference matrix.
- *GM* - a $K \times K$ growth matrix.
- *MXYZ* - the two- or three-dimensional coordinates of a mean form.
- *PAIR* - a pairing file for left- and right-side landmarks for studies of asymmetry.
- *SVCV* - a singular $K \times K$ variance-covariance matrix (among landmarks).
- *TRAN* - a general $K \times K$ transformation matrix for hypothetical forms.
- *TREE* - an $N \times N$ matrix describing a phenogram/cladogram topology.
- *TRI* - a file specifying triangles, for use with the R&S procedure.
- *UNK* - an XYZ-format file containing coordinate data for unknown specimens to be classified.
- *XYZ* - a "stacked" file containing two- or three-dimensional landmark coordinate data. *This is the type of data file that is most often used by WinEDMA.*

Line 3: This is the *parameter line*, which specifies the size of the matrix (or matrices) to be read. In the case of an XYZ file, it also indicates the sample size. In addition, the parameter line specifies whether labels are provided for the landmarks and/or the observations (see below). The specific formatting of the parameter line depends on the file type. Here are the basic parameter-line formats:

- *XYZ, MXYZ, and UNK Files:* The parameter line has three integers (separated by spaces), that specify: a) *K*: the number of landmarks per observation; b) *D*: the dimensionality of the objects (either 2 or 3); and c) *N*: the number of observations in the file. For example, if the data consist of eight three-dimensional landmarks for 25 specimens, the parameter line will be

8 3 25

An MXYZ file, which contains the coordinates of a sample mean form, has essentially the same format as an XYZ file, except there is no N parameter. Thus the parameter line for the mean of the example just given would be

8 3

- *Square symmetric matrices:* Square symmetric matrices with dimensions equal to K (the number of landmarks per observation) will all have K as their only parameter. Specifically, this is the case for files of type DM, DDM, FM, FDM, GM, GDM, FVCV, SVCV, TRANS, and TREE. Similarly, the format for files of type DVCV have only a single parameter, which is equal to the dimensionality (D). Finally, matrices of type DIST and TREE have a single parameter, which is the number of observations/taxa (N), while DIR files have a single parameter (F) that indicates the number of files to be analyzed.

Landmark and observation labels: In the above example for an XYZ parameter line, only information about K , D , and N is provided (*i.e.*, 8, 3, and 25, respectively). Nothing is mentioned about labels, so default labeling is provided in the output. The default landmark labels are the landmark numbers, preceded by 'LND_' (*e.g.*, LND_1, LND_2, etc.). for the second, etc. Similarly, the default observation label is the observation number, preceded by 'OBS_' (*e.g.*, OBS_1, OBS_2, etc.). However, it is usually more useful to provide meaningful labels to the landmarks, the observations, or both. If the user wants to specify labels for the landmarks, the K parameter should be directly followed by an 'L'. The case doesn't matter, but 'l' (el) can easily be misread by the user as '1' (one), so capitals are recommended. For example, suppose the user wants to supply labels for the eight landmarks in the example above. The proper parameter-line format is then

8L 3 25

If landmark labels are specified by the user, they are listed on the line immediately following the parameter line. A *WinEDMA* input file should never contain *any* blank lines. Landmark names can contain any ASCII number or character, but cannot contain spaces. They are limited to eight characters in length. The landmark names are separated by spaces and, if there are a lot of them, they can be placed on more than one line. Continuing with the example, suppose the landmarks are craniometric landmarks. The title line, file type, parameter line and landmark labels might look like this:

```
Craniometric data for adult chimps
XYZ
8L 3 25
Nasion Nasale Lambda Bregma Asterion Inion Basion Opisthio
```

Note how, for the eighth landmark, the name "Opisthion" required truncation to eight characters. Because line-wrapping is permitted, this format is equivalent:

```
Craniometric data for adult chimps
XYZ
8L 3 25
Nasion Nasale Lambda
Bregma Asterion Inion
Basion Opisthio
```

User-specified observation labels are slightly more complicated, because the label placement can take two forms. On one hand, the observation labels can simply be listed following the parameter line and the (optional) user-specified landmark labels. On the other hand, the user may prefer to associate observations and their labels more directly (see below). To specify that the *N* observations have labels that are listed prior to any of the XYZ data, an 'L' immediately follows the *N* parameter in the parameter line. For example:

```
8 3 25L
Male1 Male2 Male3 Male4 Male5 Male6 Male7 Male8 Male9 Male10
Female1 Female2 Female3 Female4 Female5 Female6 Female7 Female8
Female9 Female10 Female11 Female12 Female13 NoSex1 NoSex2
```

If both landmark names are specified and observation names are listed before the data, the parameter lines and labels should look like this:

```
Craniometric data for adult chimps
XYZ
8L 3 25L
Nasion Nasale Lambda Bregma Asterion Inion Basion Opisthio
Male1 Male2 Male3 Male4 Male5 Male6 Male7 Male8 Male9 Male10
Female1 Female2 Female3 Female4 Female5 Female6 Female7 Female8
Female9 Female10 Female11 Female12 Female13 NoSex1 NoSex2
```

Note how the observation labels immediately follow the landmark labels, with no blank lines. The rules for observation labels are the same as those for landmark labels: eight-space maximum length; all ASCII characters; no spaces within a label; labels separated by spaces, with line-wrapping permitted.

It is often more convenient to associate observation labels more closely with their respective landmark data. To associate observations and their labels more directly, an 'A' can be placed directly after the *N* parameter:

```
8 3 25A
```

In this case, the label for observation 1 is placed on a line by itself, followed immediately by the corresponding data, followed by the label for observation 2 and its data, and so forth. For example:

```

8 3 25A
Male1
0.143 0.194 -0.223 --> XYZ coordinates of LND_1 for Male1
0.653 -0.293 0.334
...
3.927 3.112 0.021 --> XYZ coordinates of LND_8 for Male1
Male2
0.156 0.203 -0.111 --> XYZ coordinates for LND_1 for Male2
0.554 -0.202 0.330
...
3.886 -0.302 0.119
Male3
etc....

```

This is a useful format when assembling large data files from smaller data files or, especially, when searching for observations with outliers.

Other input-file types can also contain landmark or observation labels. Continuing with the above example, suppose the user outputs a mean form matrix for the sample. The header for the file will look like this, assuming the landmark labels are user-specified:

```

Mean form matrix: Craniometric data for adult chimps
FM
8L
Nasion Nasale Lambda Bregma Asterion Inion Basion Opisthio

```

What if the labels were not user-specified in the XYZ files? The header for the mean form matrix would look like this:

```

Mean form matrix: Craniometric data for adult chimps
FM
8L
LND_1 LND_2 LND_3 LND_4 LND_5 LND_6 LND_7 LND_8

```

Note that, while there was no 'L' associated with the K parameter in the user's XYZ file, *WinEDMA* adds an 'L' here because it gave the landmarks names.

If the user computed a matrix of dissimilarities among the 25 observations in the sample (assuming the observation labels are user-specified), the title, file-type, and parameter lines will look like this:

```

Dissimilarity matrix: Craniometric data for adult chimps
DIST
25L
Male1 Male2 Male3 Male4 Male5 Male6 Male7 Male8 Male9 Male10
Female1 Female2 Female3 Female4 Female5 Female6 Female7 Female8
Female9 Female10 Female11 Female12 Female13 NoSex1 NoSex2

```

The Landmark Data:

Of course, each *WinEDMA* file will follow the parameter line and optional labels with numeric data, whether they be XYZ coordinates or elements of matrices output by *WinEDMA* (e.g., for types FM, FDM, FVCV, etc.). In any case, the data directly follow the header, with no intervening blank lines. Similarly, there should be no blank lines within the data. All numbers are separated by spaces (no commas!). For coordinate data, the XY(Z) coordinates for each landmark will appear on a separate line (the way that they are output by most digitizing programs), so that each observation in the examples above would consist of eight rows of XYZ coordinates.

Square Symmetric Matrices:

If the input file contains a square, symmetric matrix (e.g., for types FM, FDM, FVCV, etc.), the program will read either the entire matrix or only the diagonal and below-diagonal elements. Therefore, the following two form matrices are equivalent.

FIGURE

Missing Data:

The current release of *WinEDMA* **does not** support missing data, and a specimen must have a full complement of landmarks to be included in an analysis. If it is important to include an incomplete specimen (e.g., a fossil), the problem of missing landmarks can be handled using a subset file (see below). The missing landmarks should be indicated by an alphanumeric identifier that cannot be mistaken for data. For example,

```
0.323 0.541 -.984
9999 9999 9999    ← missing landmark
-1.243 1.228 3.503
```

or

```
0.323 0.541 -.984
? ? ?          ← missing landmark
-1.243 1.228 3.503
```

or

```
0.323 0.541 -.984
```

. . . ← missing landmark
-1.243 1.228 3.503

The user should be consistent in using the same indicators of missing data each time, because this will be necessary once the program is capable of handling missing data. The missing-data indicator will be specified on the *Configuration* page of the user interface. The default indicator will be 9999. For further information about automated file editing and missing data, see the *Screen* utility.

What if Header Formats Disagree?

Suppose a user runs the FORM procedure for two data sets where the landmark and/or observation labels are set up differently. For example, one data set may have no user-specified landmark labels, while the second may have them. At the moment, the program will use the header for the file whose name is entered *last* (i.e., closest to the bottom of the user interface). For example, in the GROWTH procedure, the header format for the “younger denominator” file will be the one that is used in the program output. In the future, the user will be able to standardize the header formats for a collection of files using the *Header* utility (under construction).

Of course, any two data files to be included in the same analysis must agree in the number of landmarks (*K*) and the number of dimensions (*D*). Otherwise, the analysis cannot be carried out. Finally, the landmarks must always be in the same order for all of the observations and samples considered.

Other File Types:

There are other file types that are specific to certain analyses, including the ESTIMATION, ORDINATION, CLUSTER, CLASSIFICATION, and CLIQUE procedures. To learn about these types, the relevant procedure pages should be consulted.

Sample Files:

To see more examples of input-file formatting, the user can consult the sample data files provided with *WinEDMA*.

5. Subset Files.

Subset files were created to decrease the number of XYZ files that a *WinEDMA* user has to manage. Suppose a user has collected a series of 100 craniometric landmarks from all parts of the skull, but is interested in only two or three small regions for a particular study. Instead of making two or three smaller files for each sample (by making copies of the whole files and manually editing them down, so only the landmarks of interest remain), it is much easier to use subset files. The user creates a subset file (usually no more than four lines of text) for each small analysis, leaving single copies of large data files intact.

The File Header:

The header of every *WinEDMA* subset file has four required features (and no optional features):

Line 1: This is an identifying title for the subset file (limited to a single line of text). It will be copied to output files whenever the subset file is used. The title should be a descriptor of the landmarks included in the subset, so that there is no confusion about what landmarks were actually analyzed (*e.g.*, "Subset with palate landmarks only", if the larger landmark set describes the whole skull). Because the title lines from both the XYZ file and the subset file are reported, information in the subset file does not need to repeat information about the sample itself, only about the subset membership.

Line 2: This line contains a file type code, identifying it as a subset file. In every case, subset files must be identified by the file type "SUB" (not case-sensitive).

Line 3: This is the parameter line, which consists of two items, separated by a space. The first is a single letter called the *subset instruction*, which is either "K" or "D" (not case-sensitive). "K" stands for "keep" and "D" stands for "delete". The second item is an integer called the *subset size*, which indicates how many landmarks are included. The choice of keeping or deleting should be the one that minimizes typing. For example, if the user has data files with 100 landmarks, but is only interested in ten, it is far easier to use the keep instruction, followed by a set of ten integers, as opposed to the delete instruction and a set of 90 integers. If the user were interested in a subset of 90 landmarks out of 100, it would be far simpler to use the delete instruction, followed by ten integers.

Line 4: This line consists of a series of integers (separated by spaces) describing the *subset membership*. The number of integers equals the subset size (in Line 3), and the integers describe the landmarks that should be either kept or deleted, according to the subset instruction. The integers describe the order of landmarks in the data file. See below for some examples.

A Note about Program-Supplied Landmark Labels and Subsets:

If the user does not specify landmark labels, *WinEDMA* creates them for use in program output (see **Input File Formatting** for details). The user should note that these labels are assigned prior to the application of a subset. For example, LND_5 is the fifth landmark in the XYZ file, not the fifth landmark in the subset membership line. If the user specifies landmark labels, the labeling of output is the same whether or not a subset file is used.

Examples:

In this example, only three landmarks (numbers 2, 5, and 7 in the XYZ file) are retained:

```
Triangle 2,5,7 out of a large landmark dataset
SUB
K 3
2 5 7
```

The first line is the title line, the second indicates that the file type is SUB, the third specifies that the user wants to keep three landmarks, and the fourth line specifies that the second, fifth, and seven landmarks from the XY(Z) files are to be analyzed.

In a second example, suppose that the third landmark in a data set is problematic because it is sometimes missing. It could be deleted from the analysis with the following subset file:

```
Subset to drop LND_3
SUB
D 1
3
```

Applying subset information permanently:

There may occasionally be cases where an investigator finds that a certain subset file is always used for a collection of data files. For example, there may be a landmark that is problematic because it is nearly always missing or is subject to a high degree of measurement error for some reason. Instead of specifying a subset file to delete the landmark every time an analysis is run, it might better to use the subset file to permanently delete the landmark from the data files. To save a lot of editing time, the user can perform this task using the *PermSub* utility (see below).

6. First Principles of Euclidean Distance Matrix Analysis.

Some Initial Definitions:

Landmark Data:

The “Nature Space”:

Coordinate-System Invariance and More Precise Definitions:

7. The ESTIMATION Procedure.

What the Procedure Does:

The ESTIMATION procedure reads a sample of N observations, each with K landmarks in D dimensions. The procedure calculates the sample mean form, which can be expressed either as a $K \times D$ matrix of landmark coordinates (*i.e.*, in the same format as the raw data) or as a $K \times K$ form matrix. The among-landmark variance-covariance matrix is output as a $K \times K$ matrix. The user has the option of saving these estimates to files for use with other *WinEDMA* procedures or other programs. The ESTIMATION procedure can accept a file with a single observation, if there is an interest in producing either a rotated set of coordinates or a form matrix.

In more technical terms, the procedure uses Lele's (1993) method of calculating the mean form and the among-landmark variance-covariance matrix. The estimation method assumes the following model of within-sample variation, called the *general perturbation model* (Goodall, 1991; Lele, 1993):

$$\mathbf{X}_i = (\mathbf{M} + \mathbf{E}_i)\mathbf{\Gamma}_i + \mathbf{t}_i$$

The description of the model that follows is adapted from Lele (1993) and from Lele & Cole (1996). For a relatively non-technical discussion, see Lele (1999). The $K \times D$ matrix \mathbf{X}_i represents the coordinates of K landmarks of observation i , as they are digitized (*i.e.*, this matrix represents the raw data for the i th observation). \mathbf{M} is the $K \times D$ matrix of the population mean's coordinates. The matrix \mathbf{E}_i represents the *actual* residual differences in form between the i th observation and the population mean. These deviations can be due to either genetic or environmental causes, and may be of considerable biological interest.

The remaining parameters are called *nuisance parameters*. The values of these parameters are entirely arbitrary and are, therefore, of no biological interest. $\mathbf{\Gamma}_i$ is a $D \times D$ matrix that describes how the i th observation is arbitrarily rotated relative to the mean. Arbitrary translations of the i th observation relative to the mean are described by the $K \times 1$ vector \mathbf{t}_i .

Because of the unavoidable presence of nuisance parameters in raw data, the ESTIMATION procedure uses a method that is *invariant* to these parameters (Lele & Richtsmeier, 1990; Lele, 1993, 1999; Lele & Cole, 1996; Lele & McCulloch, 2001). The first step is to compute the mean *form matrix* for the sample.

Estimation of a sample's mean-form matrix is not as straightforward as simply adding the individual observations' form matrices together and dividing by the sample size, because this leads to a biased estimate (Lele, 1993). Instead, the following steps are taken. Let $e_{ij,m}$ be the *squared distance* between landmarks i and j for observation m . [Note that while this section is adapted from Lele (1993) and from Lele & Cole (1996), the subscripts have been modified to be consistent with other EDMA publications.] The sample mean for this squared distance is

$$\bar{e} = \frac{1}{n} \sum_{m=1}^n e_{ij,m}$$

The variance of the squared distances is

$$s^2_{ij} = \frac{1}{n} \sum_{m=1}^n (e_{ij,m} - \bar{e}_{ij})^2$$

Now, if $D=2$ (that is, the objects measured are all two-dimensional), then the following quantities are calculated:

$$\hat{\delta}_{ij} = (\bar{e}_{ij}^2 - s_{ij}^2)^{0.5}$$

Note that the variance of the *squared* distances is subtracted from the *squared* mean of *squared* distances. Therefore, when the square root of the quantity is taken, the result is still in terms of *squared* distances.

If $D=3$, then a slightly different formula is used:

$$\hat{\delta}_{ij} = (\bar{e}_{ij}^2 - \frac{3}{2} s_{ij}^2)^{0.5}$$

Note that the quantities in parentheses are not guaranteed to be positive for any given sample, which leads to computational problems when the square roots are taken. The implications of this problem are discussed below, under *Understanding the Output*. However, these problems are generally rare in real data sets.

The first estimate that is obtained for the sample mean form ($\hat{\mathbf{M}}$) is the *mean form matrix*, where the elements are simply calculated as:

$$\mathbf{FM}(\hat{\mathbf{M}})_{ij} = (\hat{\delta}_{ij})^{0.5}$$

Now the elements are back in the scale of linear measurements. This is what is output by the ESTIMATION procedure when the mean form is output as a form matrix (see below).

In addition to computing a sample's mean form in form-matrix terms, the *coordinates* of the mean form can also be computed. This is done by subjecting the mean

form matrix to principal coordinates analysis (Gower, 1966; Lele, 1993; Lele & Cole, 1996).

An estimate of Σ_K^* is obtained as follows. Each observation \mathbf{X}_i is column-centered by subtracting the column means (*i.e.*, the means of the x , y , and z coordinates) for that observation. This step centers each observation at the origin of the coordinate system. Call the column-centered observation \mathbf{X}_i^C . Like the raw observations \mathbf{X}_i , each column-centered observation is a $K \times D$ matrix

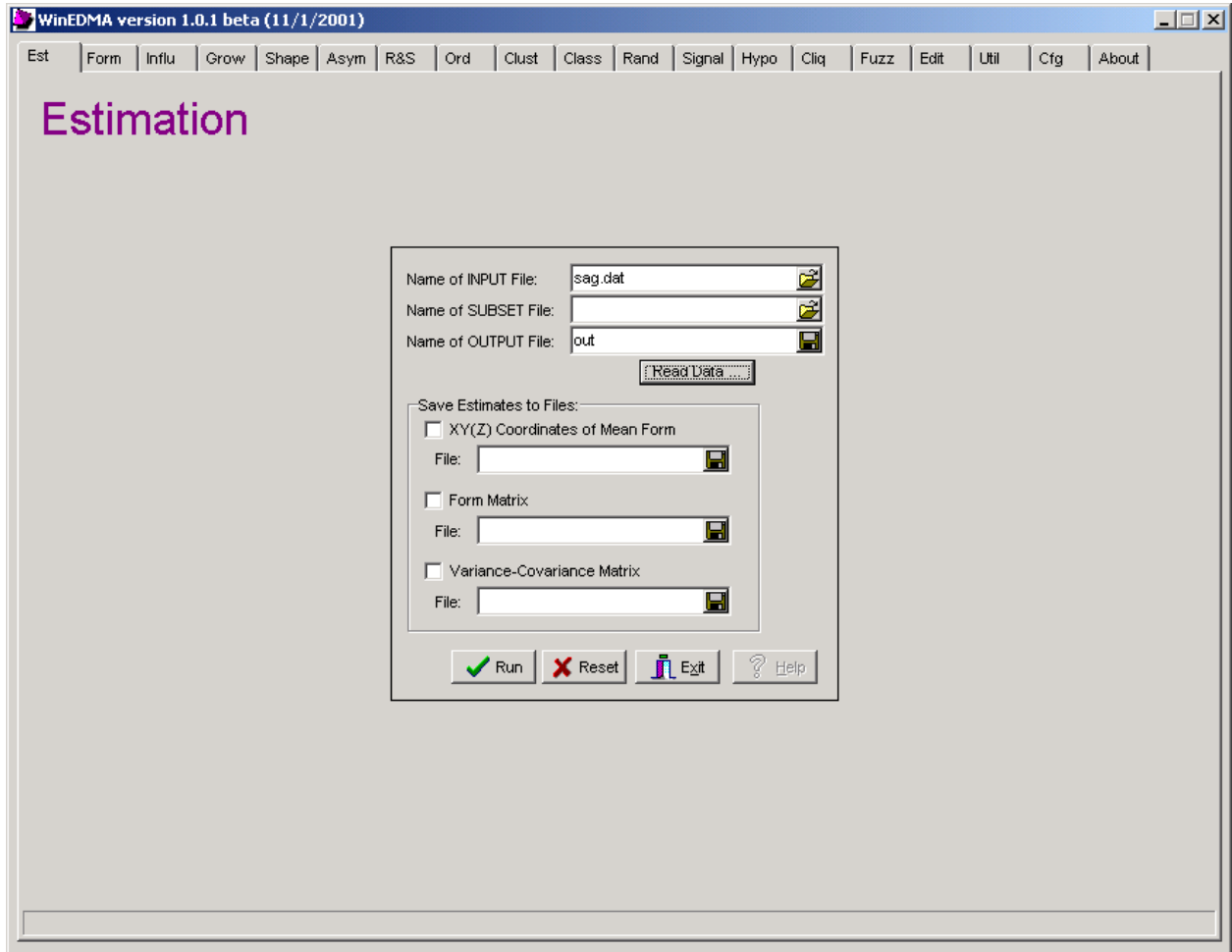
Finally, there are occasionally instances where one or more of the diagonal elements of Σ_K^* may be negative (so that the matrix has a rank less than $K-1$ and is not positive semidefinite). This implies that one or more landmarks have negative variances, which is obviously not possible by definition. This is a computational phenomenon that occurs when the variance in one or more landmarks is very small (close to zero). It also may occur in the presence of very large (regardless of sign) covariances among landmarks. The remedy to this problem is to find the positive semidefinite matrix that is *most similar* to the estimate of Σ_K^* , and using that matrix as a further-improved estimator. Let us denote the eigenvalues of Σ_K^* as $\lambda_1 \dots \lambda_K$ and the corresponding eigenvectors as $\mathbf{e}_1 \dots \mathbf{e}_K$. Because the improved estimator is not positive semidefinite, one or more of its smallest eigenvalues will be negative. To obtain the further-improved estimator, the negative eigenvalues are set to zero and the matrix is reconstructed:

$$\Sigma_K^* (\text{further improved}) =$$

Whether this step is necessary or not, it is always taken whenever *WinEDMA* estimates Σ_K^* (whatever the procedure). If it is not needed, the estimate may experience a nearly imperceptible change. However, if it is needed, it will make an important difference.

There is an important point to make about estimating mean forms and variance-covariance matrices with *subsets* of larger data sets. Suppose there

User Input:



Name of INPUT File (required):

Usually a file of type XYZ. However, MXYZ files are also accepted, if the user has the XY(Z) coordinates of a mean form and wants to compute the form matrix of the mean.

Name of SUBSET File (optional):

A file of type SUB. If this field is left blank, no subset is used. Otherwise, the program uses the specified file. There is no other control to determine whether a subset is used.

Name of OUTPUT File (required):

Output files are generated by the program. *For the moment, the program does not ask the user's permission to overwrite existing files of the same name, but does so automatically.*

Save Estimates to Files (optional):

This box contains a series of options for saving parameter estimates to files (for plotting, cutting and pasting into documents, or using with other *WinEDMA* modules). Three types of parameters may be saved: the two- or three-dimensional coordinates of the estimated mean form, the form matrix of the mean, and the singular variance-covariance matrix among landmarks (Σ_K^*). For the form-difference and variance-covariance matrices, only the diagonal and below-diagonal elements are reported. If the boxes are checked, indicating that the user wants to save one or more estimates, file names must be supplied. The user can use the title fields to specify title lines for the output files; otherwise, (not very descriptive) default titles are supplied. Finally, landmark labels are included in all estimate files. They are the same labels that appear in the output file.

Understanding the Printed Output:

Understanding the Graphics Output:

8. The FORM Procedure.

What the Procedure Does:

The section on the ESTIMATION procedure outlined the procedure for describing a sample's mean form in a way that is coordinate-system invariant. The FORM procedure provides a coordinate-system-invariant method of *comparing* the mean forms (sizes and shapes). The FORM procedure provides the user with a method for determining whether the mean forms of two samples are significantly different. More importantly, the procedure can provide confidence intervals that allow the user to localize form differences and determine *how* the samples differ.

The following discussion is adapted from Lele & Richtsmeier (1991), where EDMA was first formally described. Suppose there are two samples, called *A* and *B*, that are measured with *K* landmarks in *D* dimensions.

The two form matrices are compared using a *form-difference matrix (FDM)*. Like the form matrices that enter into its calculation, the **FDM** is square and symmetric ($K \times K$). It is defined as follows:

$$\mathbf{FDM}(A, B)_{ij} = \frac{\mathbf{FM}(A)_{ij}}{\mathbf{FM}(B)_{ij}}$$

for all $i, j = 1 \dots K$, with the convention that $0/0 = 0$ (*i.e.*, the diagonal elements are zero by definition). For two landmarks *i* and *j*, $\mathbf{FDM}(A, B)_{ij}$ is the mean distance between *i* and *j* for sample *A*, divided by the mean of the same distance for sample *B*. Because the **FDM** is symmetric, all of the information about form differences can be gained from examining the below-diagonal elements.

If all of the below-diagonals are 1.0, then *A* and *B* have exactly the same *form*. Not only are they the same scale, but their proportions are identical. If all of the below-diagonal elements are the same but are not equal to 1.0, then *A* and *B* differ in *scale*, but not in *shape*. For example, if all of the below-diagonals are 2.0, then *A* is twice as large as *B* for all distances; however, the proportions of *A* and *B* are identical. If the below-diagonals are heterogeneous, then *A* and *B* have different *shapes* or proportions.

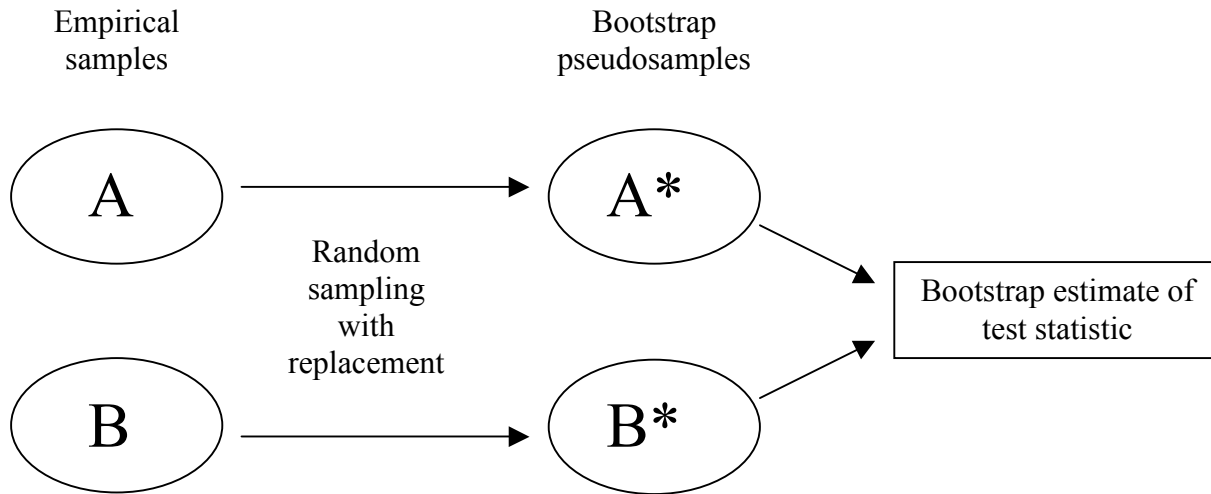
Suppose the below-diagonals of an **FDM** are heterogeneous. How does an investigator decide whether this is a statistically significant difference? The comparison of two form matrices using a form-difference matrix is an inherently multivariate problem, but it is possible to summarize the *degree* of form difference in terms of a single statistic. While a number of different statistics are feasible, Lele & Richtsmeier (1991) developed the following statistic:

$$T = \frac{\max[\mathbf{FDM}(A, B)]}{\min[\mathbf{FDM}(A, B)]}$$

With many familiar statistical tests, an investigator calculates a test statistic (Student's t , for example) and compares the statistic to a table of critical values. If the observed statistic exceeds the critical value, the null hypothesis is rejected. Otherwise, the null hypothesis cannot be rejected. Many of the statistics used with EDMA do not have conveniently tabled critical values; Lele & Richtsmeier's T statistic is no exception. Instead, an alternative method for assessing the significance of a given value of T must be used: *bootstrapping*. Bootstrapping (Efron & Tibshirani, 1986, 1993; Davies & Hinkley, 1996) is a computer-intensive method of using statistics that have distributions that are either unknown or computationally unwieldy.

There are two basic types of bootstrapping: *nonparametric* and *parametric*.

In Lele & Richtsmeier's (1991) original EDMA paper, the nonparametric bootstrap resampling strategy for form-difference testing works as follows, and is illustrated here:



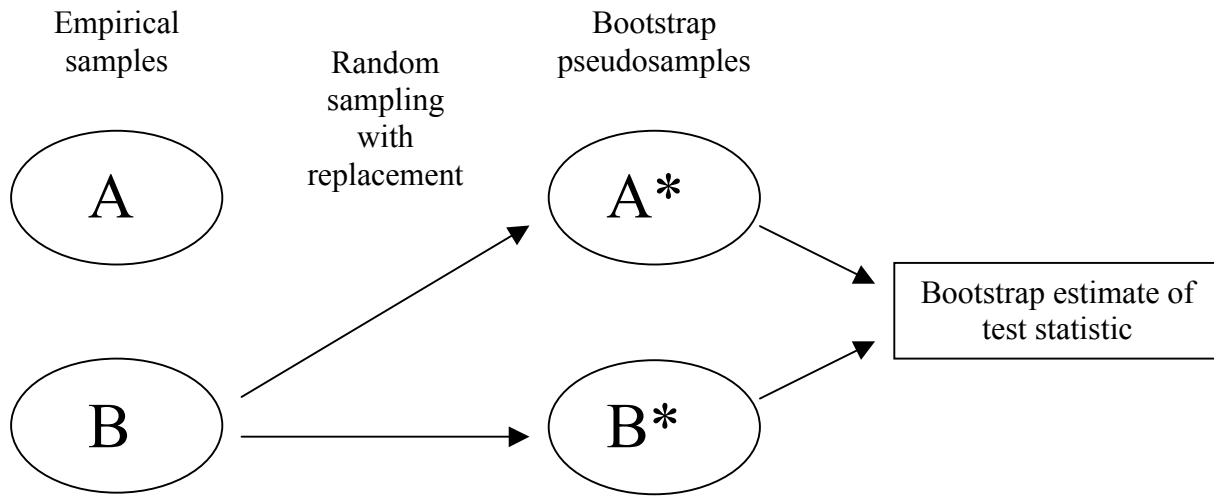
Step 1: Suppose there are two empirical samples, called A and B , with sample sizes n_A and n_B , respectively. Compute the mean forms and compare them using the T statistic.

Step 2: Generate a *pseudosample* called A^* by selecting observations from sample A randomly and with replacement. The “random” part of the sampling strategy states that each observation has an equal chance (1 in n_A) of being selected. The “with

replacement” part means that an observation can be selected more than once, so that some observations may be selected multiple times, while others may not be selected at all. Repeat the same process with sample B , generating a pseudosample called B^* . Using the pseudosamples, compute mean forms and a form-difference matrix. Then compute a bootstrap version of the test statistic T , called T^* .

Step 3: Repeat Steps 3 a large number of times (e.g., 100 to 1000), saving T^* each time.

This resampling strategy is still used for the computation of confidence intervals for form and growth differences (Lele & Richtsmeier, 1995), which are discussed below. However, for the purposes of *testing* for overall differences in form or growth, the resampling strategy was modified by Richtsmeier & Lele (1993). The modified strategy is shown here:



Here, the empirical sample B is the source of observations for *both* pseudosamples A^* and B^* . In the *WinEDMA* user interfaces for the FORM and GROWTH procedures, it is referred to as the *reference sample*.

The statistical details for preferring this resampling strategy are subtle, and the interested user should consult Richtsmeier & Lele (1993).

User Input:

Name of NUMERATOR/DENOMINATOR Input File (required):

Usually files of type XYZ. However, MXYZ files are also accepted, if the user has the XY(Z) coordinates of mean forms and wants simply to compute the form matrices and form-difference matrix.

Subset file name (optional):

A file of type SUB. If this field is left blank, no subset is used. Otherwise, the program uses the specified file. There is no other control to determine whether a subset is used.

Output file name (required):

Output files are generated by the program. *For the moment, the program does not ask the user's permission to overwrite existing files of the same name, but does so automatically.*

Form-difference matrix output options (optional):

These check boxes allow the user to specify how the form-difference matrix should be printed in the output. If no boxes are checked, the matrix is not printed at all (e.g., if the user is only interested in testing). With each of the options, only the below-diagonal elements of the matrix are printed.

Bootstrap options (optional):

The controls in this box set the options for form-difference testing (Lele & Richtsmeier, 1991). The check box specifies that testing should be performed. The reference-sample selection allows specification of which sample should be used as the source of bootstrap pseudosamples (Richtsmeier & Lele, 1993). Finally, there is a field for specifying the number of pseudosamples to be used for testing (generally 100 to 1000). This field cannot be left blank if testing is performed.

Confidence interval options (optional):

The controls in this box set the options for estimation of marginal confidence intervals for the elements of the form-difference matrix (Lele & Richtsmeier, 1995). The check box specifies that the intervals should be estimated. The resampling-method radio buttons allow the user to choose whether the intervals should be estimated nonparametrically (Lele & Richtsmeier, 1992) or parametrically (Lele & Cole, 1996). There are then fields for specifying the number of pseudosamples to be used (generally more than used in testing -- 300 to 1000) and the alpha-level for the confidence intervals (e.g., 0.1 for 90% intervals or 0.05 for 95% intervals). These fields cannot be left blank if confidence intervals are estimated.

Saving the form-difference matrix to a file (optional):

This box contains options for saving the form-difference matrix to a file (e.g., for cutting and pasting into a document or using with other *WinEDMA* procedures). If the box is checked, indicating that the user wants to save the matrix, a file name must be supplied. The user can use the title field to specify a title line for the output file; otherwise, a (not very descriptive) default title line is supplied. Landmark labels are included in the file header; they are the same labels that appear in the output file.

Understanding the Printed Output:

Understanding the Graphics Output:

9. The INFLUENTIAL Procedure.

What the Procedure Does:

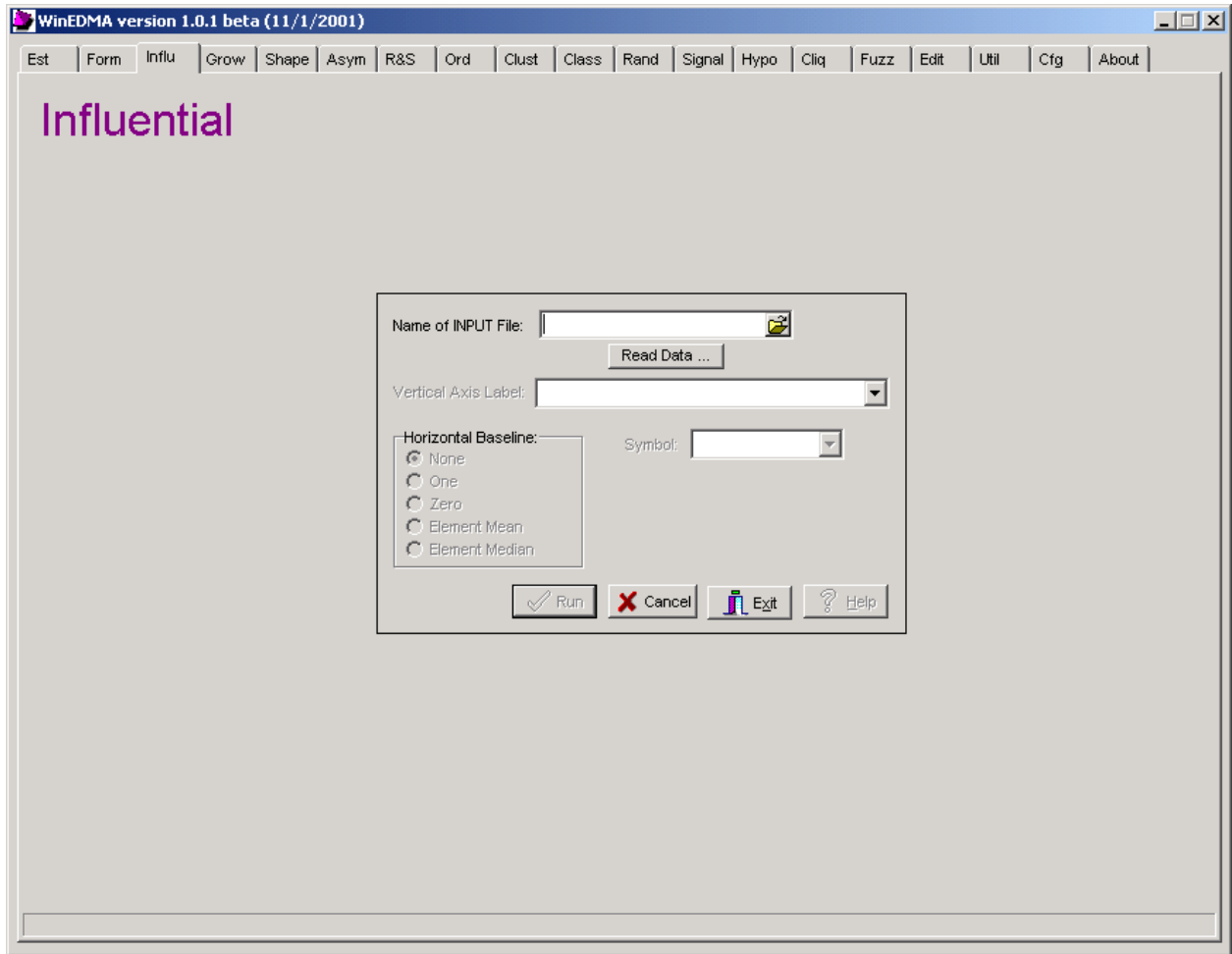
Few morphometric studies are limited to the simple demonstration that two mean forms or shapes are significantly different. What is usually of greater interest is the description of *how* they are different, so that the numerical results can be interpreted in the context of morphology (Corruccini, 1978). Therefore, there is often an interest in finding the measurements that are “most responsible” for the between-group differences. When Lele & Richtsmeier (1992) described this idea in terms of landmark data, they coined the term “influential landmark.”

Lele & Richtsmeier’s (1992) procedure for identifying influential landmarks is as follows:

The *INFLUENTIAL* procedure takes a slightly different, graphics-oriented approach to the identification of influential landmarks (Cole & Richtsmeier, 1998). The graphic method is a very time-efficient way of identifying influential landmarks quickly, even when the number of landmarks is large. For example, in an analysis of sexual dimorphism in macaque skulls, Cole & Richtsmeier (1998) used 35 landmarks, with a total of 595 interlandmark distances. The results of the graphic analysis look like this:

Even though the form-difference matrix was fairly large in this case, the graphic method allows the investigator to identify the anatomical regions of greatest sex difference in a matter of a few seconds. They are landmarks 9-12 (the premaxilla) and xx-xx (the sphenobasilar synchondrosis).

User Input:



Name of INPUT File (required): The input file contains a square matrix that describes a two-sample comparison. It can be a form- (FDM), growth- (GDM), shape- (SDM), or dimorphism-difference (DDM) matrix.

Title for Plot:

Vertical Axis Label (optional): The program automatically determines the type of matrix that has been input by reading the file header. Given the type of matrix, it will supply one of the following labels:

Matrix type	Vertical axis label
FDM	
GDM	
SDM	
DDM	

The Vertical Axis Label combo box allows the user to specify some label other than the ones provided by the program.

Horizontal Baseline (optional):

The Horizontal Baseline radio group gives the user the option to superimpose a horizontal line on the plot at a specified vertical-axis level. A horizontal baseline is often helpful in providing a visual indication of the expected values of equal forms or shapes.

Symbol:

Available symbols include: circles (the default), diamonds, up triangles, down triangles, crosses, and diagonal crosses.

Jitter Symbols:

When many distances are analyzed, the distributions of the points may become very dense in places, so that many symbols overlap. To make densely-packed symbols appear more distinctly, the symbols can be “jittered” slightly by adding small amounts of random noise to their horizontal-axis values (*i.e.*, to the landmark numbers). The vertical-axis values remain unaffected. Jittering is especially effective in combination with the small dot symbol.

Understanding the Graphics Output:

Data Note: The macaque data (male and female adult means) used in the example above are provided as part of the sample data included with *WinEDMA* (*male.xyz* and *female.xyz*).

10. The GROWTH Procedure.

What the Procedure Does:

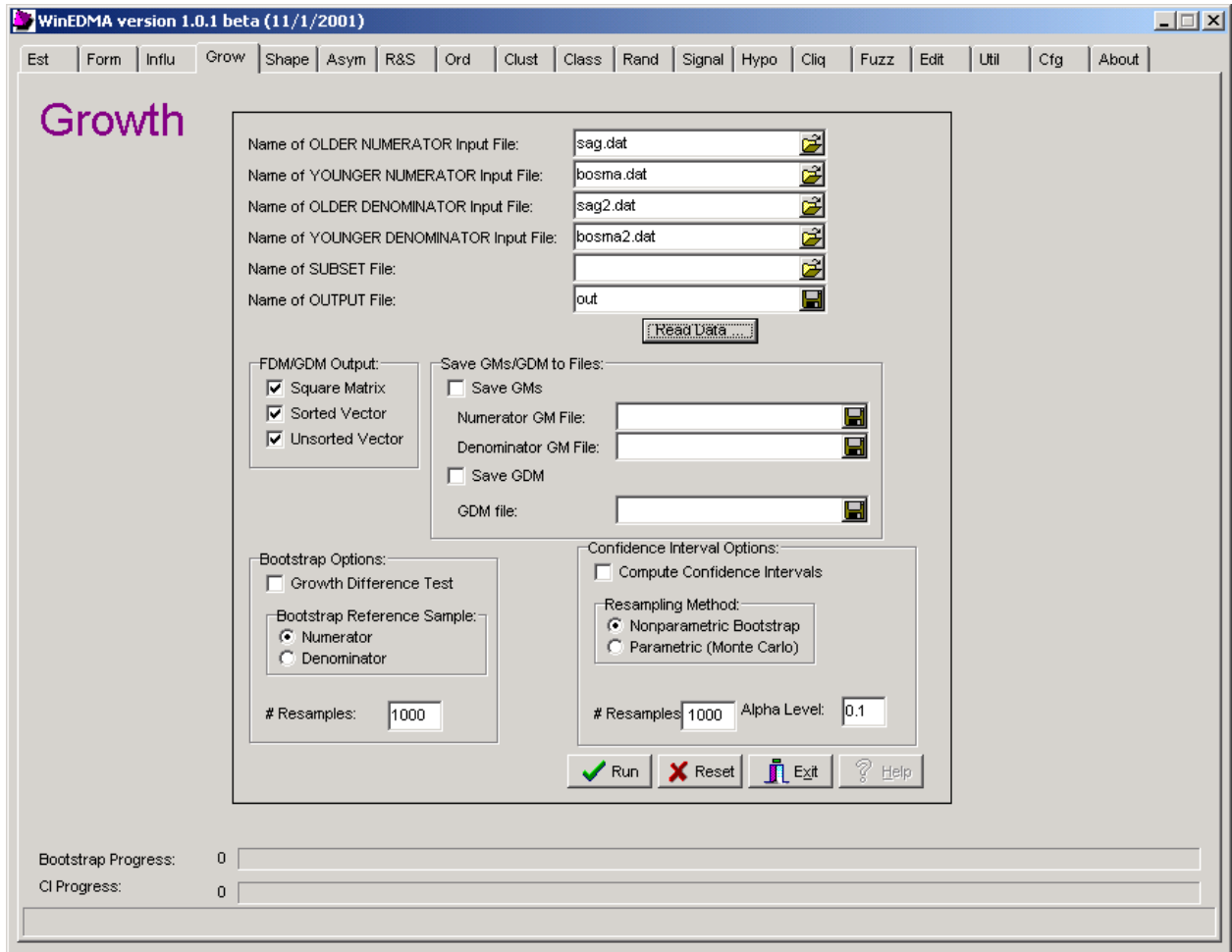
The EDMA procedures for describing and comparing growth are extensions of Lele & Richtsmeier's (1991) method for studying form. The growth methods are developed in papers by Corner & Richtsmeier (199x), Richtsmeier et al. (1992, 1993), and Richtsmeier & Lele (1993). These studies assume that a researcher is studying growth in *cross-sectional samples* (Cock, 1966) that are stratified into age groups or developmental stages. Suppose a researcher is studying growth in a species and wants to describe the pattern of growth that occurs between two developmental stages (called 1 and 2) that are indicated, for example, by tooth eruptions. The form change over this interval can be described by a growth matrix (**GM**), which is simply a special case of a form matrix:

For all $ii,j = 1 \dots K$, with the convention that $0/0 = 0$ (for the diagonal elements). By convention, the older sample goes in the numerator of the growth matrix, while the younger sample goes in the denominator. Because older specimens are generally larger than younger ones, the below-diagonal elements of the **GM** tend to be greater than 1.0.

Suppose that a researcher then wants to *compare* growth patterns between one and two years in two species, called *A* and *B*. The one-year-old and two-year-old samples of *A* are called *A1* and *A2*, respectively. Similarly, the similarly-aged samples for *B* are called *B1* and *B2*. As described above, growth matrices for both species can be constructed to describe their respective form changes over the same growth interval:

In most published applications of these methods (*e.g.*, Richtsmeier *et al.*, 1992, 1993; Richtsmeier & Lele, 1993), the samples are stratified into ordinal age stages. However, the same methods can be used

User input:



*Input file names for older and younger numerator and denominator samples (**required**):*

Usually files of type XYZ. However, MXYZ files are also accepted, if the user has the XY(Z) coordinates of mean forms and wants simply to compute the form matrices, growth matrices, and growth-difference matrix.

Subset file name (optional):

A file of type SUB. If this field is left blank, no subset is used. Otherwise, the program uses the specified file. There is no other control to determine whether a subset is used.

*Output file name (**required**):*

Output files are generated by the program. *For the moment, the program does not ask the user's permission to overwrite existing files of the same name, but does so automatically.*

Form-difference matrix output options (optional):

These check boxes allow the user to specify how the form-difference matrix should be printed in the output. If no boxes are checked, the matrix is not printed at all (e.g., if the user is only interested in testing). With each of the options, only the below-diagonal elements of the matrix are printed.

Bootstrap options (optional):

The controls in this box set the options for form-difference testing (Lele & Richtsmeier, 1991). The check box specifies that testing should be performed. The reference-sample selection allows specification of which sample should be used as the source of bootstrap pseudosamples (Richtsmeier & Lele, 1993). Finally, there is a field for specifying the number of pseudosamples to be used for testing (generally 100 to 1000). This field cannot be left blank if testing is performed.

Confidence interval options (optional):

The controls in this box set the options for estimation of marginal confidence intervals for the elements of the form-difference matrix (Lele & Richtsmeier, 1995). The check box specifies that the intervals should be estimated. The resampling-method radio buttons allow the user to choose whether the intervals should be estimated nonparametrically (Lele & Richtsmeier, 1992) or parametrically (Lele & Cole, 1996). There are then fields for specifying the number of pseudosamples to be used (generally more than used in testing -- 300 to 1000) and the alpha-level for the confidence intervals (e.g., 0.1 for 90% intervals or 0.05 for 95% intervals). These fields cannot be left blank if confidence intervals are estimated.

Saving the form-difference matrix to a file (optional):

This box contains options for saving the form-difference matrix to a file (e.g., for cutting and pasting into a document or using with other *WinEDMA* procedures). If the box is checked, indicating that the user wants to save the matrix, a file name must be supplied. The user can use the title field to specify a title line for the output file; otherwise, a (not very descriptive) default title line is supplied. Landmark labels are included in the file header; they are the same labels that appear in the output file.

Understanding the Printed Output:

Understanding the Graphics Output:

11. The SHAPE Procedure.

What the Procedure Does:

User Input:

WinEDMA version 1.0.1 beta (11/1/2001)

Est Form Influ Grow Shape Asym R&S Ord Clust Class Rand Signal Hypo Cliq Fuzz Edit Util Cfg About

Shape

Name of SAMPLE 1 Input File: sag.dat

Name of SAMPLE 2 Input File: bosma.dat

Name of SUBSET File:

Name of OUTPUT File: out

Read Data

FDM/SDM Output:

- ☒ Square Matrix
- ☒ Sorted Vector
- ☒ Unsorted Vector

Monte Carlo and Confidence Interval Options:

- ☐ Shape and Scaling Difference Tests
- ☐ Compute Confidence Intervals

Resamples: 1000 Alpha Level: 0.1

Scaling Measure:

- ☒ Single Distance (Specify Endpoints): REAMS RAST
- ☐ Sneath Scaling
- ☐ Geometric Mean
- ☐ Maximum Distance
- ☐ Median Distance

Run Reset Exit Help

Bootstrap Progress: 0

Input file names for numerator and denominator samples (**required**):

Usually a file of type XYZ. However, MXYZ files are also accepted, if the user has the XY(Z) coordinates of a mean form and wants to compute the form matrix of the mean.

Subset file name (*optional*):

A file of type SUB. If this field is left blank, no subset is used. Otherwise, the program uses the specified file. There is no other control to determine whether a subset is used.

Output file name (**required**):

Output files are generated by the program. For the moment, the program does not ask the user's permission to overwrite existing files of the same name, but does so automatically.

Form-difference matrix output options (*optional*):

These check boxes allow the user to specify how the form-difference matrix should be printed in the output. If no boxes are checked, the matrix is not printed at all

(e.g., if the user is only interested in testing). With each of the options, only the below-diagonal elements of the matrix are printed.

Bootstrap options (optional):

The controls in this box set the options for form-difference testing (Lele & Richtsmeier, 1991). The check box specifies that testing should be performed. The reference-sample selection allows specification of which sample should be used as the source of bootstrap pseudosamples (Richtsmeier & Lele, 1993). Finally, there is a field for specifying the number of pseudosamples to be used for testing (generally 100 to 1000). This field cannot be left blank if testing is performed.

Confidence interval options (optional):

The controls in this box set the options for estimation of marginal confidence intervals for the elements of the form-difference matrix (Lele & Richtsmeier, 1995). The check box specifies that the intervals should be estimated. The resampling-method radio buttons allow the user to choose whether the intervals should be estimated nonparametrically (Lele & Richtsmeier, 1992) or parametrically (Lele & Cole, 1996). There are then fields for specifying the number of pseudosamples to be used (generally more than used in testing -- 300 to 1000) and the alpha-level for the confidence intervals (e.g., 0.1 for 90% intervals or 0.05 for 95% intervals). These fields cannot be left blank if confidence intervals are estimated.

Saving the form-difference matrix to a file (optional):

This box contains options for saving the form-difference matrix to a file (e.g., for cutting and pasting into a document or using with other *WinEDMA* procedures). If the box is checked, indicating that the user wants to save the matrix, a file name must be supplied. The user can use the title field to specify a title line for the output file; otherwise, a (not very descriptive) default title line is supplied. Landmark labels are included in the file header; they are the same labels that appear in the output file.

Understanding the Printed Output:

Understanding the Graphics Output:

12. The ASYMMETRY Procedure.

What the Procedure Does:

Many complex organisms exhibit bilateral symmetry, where the right and left halves of the body are mirror images reflected across a midline plane.

To introduce the *ASYMMETRY* procedure, some definitions are first necessary. There are several different types of asymmetry, and these are defined on two bases. First, asymmetries are defined in terms of how pronounced they are. They can be either *conspicuous* or not. While the definition of “conspicuous” is certainly not hard and fast, a useful definition of a conspicuous asymmetry is one that is detectable by the naked eye, without the need for precise measurement. Asymmetries that are “not conspicuous” are far more subtle. As an extreme example, flounders are conspicuously asymmetric, because one can easily see that a flounder has both of its eyes on one side of its head. In contrast, normal human faces appear symmetric, although asymmetries can always be found if the scale of measurement is fine enough. The asymmetry of a normal human face might therefore be considered “not conspicuous.”

User Input:

Understanding the Printed Output:

Understanding the Graphics Output:

13. The R&S Procedure.

The R&S procedure is under construction. Documentation will be provided when the procedure becomes available.

14. The ORDINATION Procedure.

What the Procedure Does:

Ordination methods are frequently used in studies of taxonomy, ontogeny, and functional morphology.

User Input:

The screenshot shows the 'Ordination' dialog box in WinEDMA. The title bar reads 'WinEDMA version 1.0.1 beta (11/1/2001)'. The menu bar includes: Est, Form, Inflx, Grow, Shape, Asym, R&S, Ord, Clust, Class, Rand, Signal, Hypo, Clq, Fuzz, Edit, URI, Cfg, About. The dialog box has three tabs: 'Input an XYZ File', 'Input a Matrix or Matrices', and 'Calculate a Matrix'. The 'Input an XYZ File' tab is active. It contains the following fields and options:

- 'Name of XY(Z) Input File:' with a text box containing 'sag.dat' and a file selection icon.
- 'Name of SUBSET File:' with a text box and a file selection icon.
- A checkbox for 'Paired Observations' which is unchecked.
- 'Name of OUTPUT File:' with a text box containing 'out' and a file selection icon.
- A 'Read Data...' button.
- 'Scaling Measure:' section with radio buttons: 'NONE' (selected), 'Single Distance (Specify Endpoints):' (with 'REAMS' and 'RAST' dropdowns), 'Sneath Scaling', 'Maximum Distance', 'Geometric Mean', and 'Median Distance'.
- 'Output Options:' section with checkboxes for 'Form Matrices', 'Shape Matrices', and 'Report Correlations'.
- 'Number of Axes to Retain (A=All):' with a text box containing '3'.
- Buttons at the bottom: 'Run' (green checkmark), 'Reset' (red X), 'Exit' (purple icon), and 'Help' (question mark).

This close-up shows the 'Input an XYZ File' tab. It includes the 'Name of XY(Z) Input File:' field with 'sag.dat', the 'Name of SUBSET File:' field, the 'Paired Observations' checkbox, and the 'Read Data...' button.

This close-up shows the 'Input a Matrix or Matrices' tab. It includes the 'Name of Matrix Input or List File:' field with 'sag.fdm' and a file selection icon.

This close-up shows the 'Calculate a Matrix' tab. It includes the 'Name of XY(Z) Input File:' field with 'sag.dat', the 'Name of SUBSET File:' field, the 'Paired Observations' checkbox, and the 'Read Data...' button.

Input file names for numerator and denominator samples (required):

Usually a file of type XYZ. However, MXYZ files are also accepted, if the user has the XY(Z) coordinates of a mean form and wants to compute the form matrix of the mean.

Subset file name (optional):

A file of type SUB. If this field is left blank, no subset is used. Otherwise, the program uses the specified file. There is no other control to determine whether a subset is used.

Output file name (required):

Output files are generated by the program. *For the moment, the program does not ask the user's permission to overwrite existing files of the same name, but does so automatically.*

Form-difference matrix output options (optional):

These check boxes allow the user to specify how the form-difference matrix should be printed in the output. If no boxes are checked, the matrix is not printed at all (e.g., if the user is only interested in testing). With each of the options, only the below-diagonal elements of the matrix are printed.

Bootstrap options (optional):

The controls in this box set the options for form-difference testing (Lele & Richtsmeier, 1991). The check box specifies that testing should be performed. The reference-sample selection allows specification of which sample should be used as the source of bootstrap pseudosamples (Richtsmeier & Lele, 1993). Finally, there is a field for specifying the number of pseudosamples to be used for testing (generally 100 to 1000). This field cannot be left blank if testing is performed.

Confidence interval options (optional):

The controls in this box set the options for estimation of marginal confidence intervals for the elements of the form-difference matrix (Lele & Richtsmeier, 1995). The check box specifies that the intervals should be estimated. The resampling-method radio buttons allow the user to choose whether the intervals should be estimated nonparametrically (Lele & Richtsmeier, 1992) or parametrically (Lele & Cole, 1996). There are then fields for specifying the number of pseudosamples to be used (generally more than used in testing -- 300 to 1000) and the alpha-level for the confidence intervals (e.g., 0.1 for 90% intervals or 0.05 for 95% intervals). These fields cannot be left blank if confidence intervals are estimated.

Saving the form-difference matrix to a file (optional):

This box contains options for saving the form-difference matrix to a file (e.g., for cutting and pasting into a document or using with other WinEDMA procedures). If the box is checked, indicating that the user wants to save the matrix, a file name must be supplied. The user can use the title field to specify a title line for

the output file; otherwise, a (not very descriptive) default title line is supplied. Landmark labels are included in the file header; they are the same labels that appear in the output file.

Understanding the Printed Output:

Understanding the Graphics Output:

15. The CLUSTERING Procedure.

What the Procedure Does:

Clustering methods are types of exploratory data analyses that are used to discover the presence of “natural groups” within a set of observations (Johnson & Wichern, 1982). When clustering methods are applied to morphometric data, the purpose is to discover groups of observations that are similar to each other in form or shape. There are two basic categories of clustering algorithms: *hierarchical* and *nonhierarchical*. The categories differ in the assumptions that they make about how the similarities and differences among data are structured or patterned.

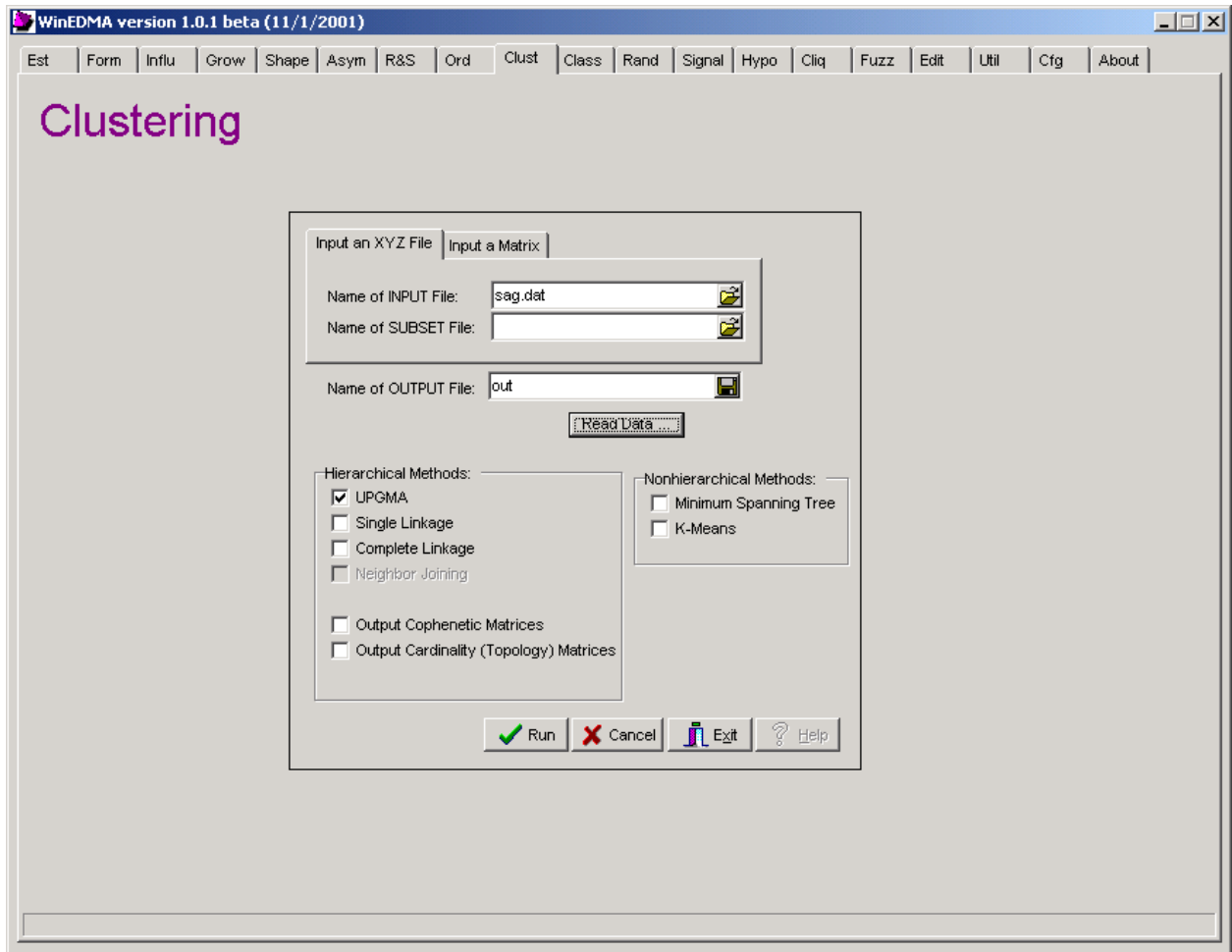
All of the clustering methods used with *WinEDMA* are based on the same measurement of *dissimilarity*. While many interlandmark distances may be used to quantify the difference between two observations,

WinEDMA allows the user to perform two basic types of cluster analysis: *hierarchical* and *nonhierarchical*. The difference between the types lies in a very basic assumption about the relationships between the (as yet undiscovered) groups of observations.

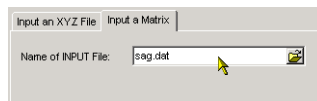
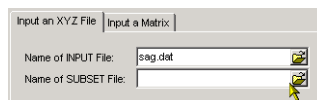
There are three different types of hierarchical clustering methods that can be performed using *WinEDMA*: 1) unweighted pair-group arithmetic averaging (UPGMA); 2) single-linkage analysis; and 3) complete linkage analysis. Of these, UPGMA is probably the most widely-used in morphometrics and comparative biology.

There are two types of nonhierarchical methods that *WinEDMA* can perform: 1) K-means clustering; and 2) construction of minimum-spanning trees.

User Input:



Name of the INPUT File (required):



Name of the SUBSET File:

Name of the OUTPUT File (required):

Heirarchical Methods:

Nonheirarchical Methods:

Understanding the Printed Output:

Understanding the Graphics Output:

16. The CLASSIFICATION Procedure.

What the Procedure Does:

Suppose an observer has measured an observation with an unknown taxonomic affinity, but there is reason to suspect that the unknown belongs to a group for which a reference sample has already been measured.

User Input:

Understanding the Printed Output:

17. The RANDOM Procedure.

What the Procedure Does:

User Input:

Name of the MEAN File (required):

A file of type MXYZ.

Name of the VCV File (required):

A file of type FVCV or SVCV, depending on whether the among-landmarks variance-covariance matrix is full-rank ($\text{rank}=K$) or singular ($\text{rank}<K$), respectively. A user-specified variance-covariance matrix of unknown rank can be input as a file of type UVCV and the program will determine the rank and the appropriate method for data generation.

Name of the DATA File (required):

A file of type XYZ, containing the data generated under the model.

Name of the OUTPUT File (required):

A file that records information about the parameters used to generate the data....
Note that this file does *not* contain the generated data.

Sample Size (required):

The number of observations to be randomly generated under the model.

Title for Generated Data:

Report MLE for Mean Coordinates and VCV Matrix for Generated Data:

Landmark Labels:

Specifies whether the header of the DATA file will include landmark labels. If landmark labels are to be included, the user has the option of reading them from the header of the MEAN file or letting the program create them in the LND_X format.

Observation Labels:

Specifies whether the DATA file will include observation labels created by the program in the OBS_X format. If observation labels are to be created, the user has the option of listing them all in the header or associating them with the observations (see **Input File Formatting**).

Understanding the Printed Output:

18. The PHYLOGENY Procedure.

19. The HYPOTHETICAL Procedure.

What the Procedure Does:

The HYPOTHETICAL procedure is used to obtain hypothetical forms, following Richtsmeier *et al.* (1992) and Richtsmeier & Lele (1993). The user inputs the XY(Z) coordinates of a *starting form*, along with a $K \times K$ *transformation matrix*. This matrix transforms the original coordinate data into a new set of coordinate data. The transformation itself is very simple. Suppose the XY(Z) coordinates of a specimen called *A* are input, along with a transformation matrix called **T**. Examples of transformation matrices are given below. Compute a form matrix for *A*, called **FM(A)**. **FM(A)** and **T** are both symmetric $K \times K$ matrices. We can compute a *hypothetical form matrix*, called **FM(Hyp)** by direct (elementwise) multiplication of the starting form by the transformation:

$$\mathbf{FM(Hyp)}_{ij} = \mathbf{FM(A)}_{ij} \otimes \mathbf{T}$$

for all $i, j = 1 \dots K$.

User Input:

Understanding the Printed Output:

Understanding the Graphics Output:

20. The CLIQUE Procedure.

What the Procedure Does:

User Input:

Name of mean file (required):

The mean form for random data generation must be contained in a file of type MXYZ, which is either user-specified or is output by another procedure (e.g., ESTIMATION).

Name of VCV file (required):

The variance-covariance structure must be contained in a file of type SVCV or FVCV, which denote singular and full-rank matrices, respectively.

Understanding the Printed Output:

21. The FUZZY Procedure.

The FUZZY procedure is under construction. Documentation will be provided when the procedure becomes available.

22. The EDITOR.

The *WinEDMA* Editor allows the user to read, modify, format, or print text (ASCII) files while the program is running. It is useful for fixing errors in data or subset files and for browsing output files. The Editor is controlled using toolbar buttons, the mouse, and an assortment of keyboard commands.

*When learning to use the Editor, the user should make **backups** of the files used, to prevent loss or damage of data. In other words, don't practice with files you care about!*

Enabling and Disabling the Editor:

When the user starts *WinEDMA*, the default state of the editor is “disabled”. The disabled state is indicated by a dark gray color in the text window. In this condition, text cannot be entered. Note also that only a few of the toolbar buttons are accessible. There are two ways of enabling the Editor. First, the user can create a new text file. Second, the user can open an existing file. In either case, the background color will change to the default window color under the user's Windows configuration (usually white), typing will be possible, and the toolbar buttons will become accessible. When the user has finished using the Editor, it can be disabled by pressing the Exit button on the toolbar. While it is not absolutely necessary to exit the Editor before using other components of the program, it is highly recommended. Note that saving an open file does not disable the Editor; this feature allows the user to save work incrementally when editing large files.

Toolbar Buttons:

There are 15 buttons on the Editor toolbar, and each has a bitmap to indicate its use. Holding the mouse over a button for a few seconds will provide a short verbal description of what the button does. A few of the buttons have corresponding keyboard commands (see below). From left to right, the buttons are:

New: The New button is always enabled. If the text window is disabled, the New button will enable it (along with the other buttons on the toolbar). If the button is pressed while a file is already open, the user will be asked whether the open file should be saved. The older file will then be closed automatically and a blank text window will appear. The default file name for new files is "Untitled.txt". Currently, it is not possible to have more than one file open at a time.

Open: The Open button is always enabled. It enables the text window and the rest of the toolbar. When the button is pressed, the user will be presented with a dialog box for selecting the file. As with the New button, if the Open button is pressed while a file is already in the text window, the user will be asked about saving the older file. The new file will then automatically replace the older file in the text window.

Save: The Save button is enabled when the text window is enabled (i.e., when there is something to save). Note that pressing the Save button does not disable the Editor or clear the text window. This feature allows the user to incrementally save when working with large files for extended time periods. If the file is new, the user will be prompted for a file name to replace the default name ("Untitled.txt").

Print: The Print button is enabled when the text window is enabled (i.e., when there is something to print). When it is pressed, the user is presented with a dialog box for setting printer options and beginning a print job.

Setup: The Setup button is always enabled. When it is pressed, a dialog box opens and allows the user to set printing options (e.g., print quality, paper orientation, etc.).

Font: The Font button is enabled when the text window is enabled. When it is pressed, a dialog box allows the user to adjust the font size, typeface, and color. The default typeface is black, 8-point New Courier.

Find: The Find button is enabled when the text window is enabled. When it is pressed, a dialog box allows the user to search the open file for a string of text. Searches are optionally case-sensitive and can be performed forward or backward (relative to the cursor position).

Replace: The Replace button is enabled when the text window is enabled. When it is pressed, a dialog box opens. It is similar to the Find dialog, but has the additional function of allowing the user to replace one text string with another.

Select All: The Select All button is enabled when the text window is enabled. It is used to highlight all of the text window contents (i.e., everything in a file). It is most useful for changing the font of an existing file.

Cut: The Cut button is enabled when the text window is enabled. When it is pressed, it removes highlighted text from the text window and copies it to the clipboard (in memory). The text can then be copied or pasted elsewhere in the file. Note that the clipboard only has room for one selection at a time. If a selection is cut from the text editor, it will replace any existing clipboard contents (which will be lost).

Copy:

Paste:

Delete:

Undo: The Undo button is enabled when the text window is enabled. It simply undoes the user's last action. Note that it can also undo itself, so that it is essentially an undo/redo toggle. There is currently only one undo level available, so that only the very last change to the file can be undone.

Exit: The Exit button is always enabled. When pressed, it disables both the text window and the toolbar. If the current file has been modified since the last time it was saved, the user will be prompted to save it before the text window is disabled. If the current file is unmodified, it will simply close automatically.

Using the Mouse:

The Editor component uses a "Rich Text" object for the text window. This gives the user a "drag-and-drop" capability for moving highlighted blocks of text.

Using the Keyboard:

The following is a list of basic keyboard commands that can be used when working with the Editor. Adapted from Appendix A of Borland Pascal with Objects, 7.0: Programmer's Reference (Borland International, Inc., 1992).

Cursor keys

<i>To move...</i>	<i>Press</i>
Character left	Left arrow
Character right	Right arrow
Word left	Ctrl + Left arrow
Word right	Ctrl + Right arrow
Line up	Up arrow
Line down	Down arrow
Page up	PgUp
Page down	PgDn
Beginning of line	Home
End of line	End
Top of window	Ctrl + PgUp
Bottom of window	Ctrl + PgDn
Top of file	Ctrl + Home
Bottom of file	Ctrl + End

Insert and delete commands

<i>To...</i>	<i>Press</i>
Delete character	Del
Delete character to left	Backspace
Insert/overwrite toggle	Ins

Block commands

<i>To...</i>	<i>Press</i>
Select all	Alt + A

Copy to clipboard	Ctrl + Ins
Cut to clipboard	Shift + Del
Delete block	Ctrl + Ins
Paste from clipboard	Shift + Ins

Extending selected blocks

<i>To extend selection ...</i>	<i>Press</i>
Left one character	Shift + Left arrow
Right one character	Shift + Right arrow
End of line	Shift + End
Beginning of line	Shift + Home
Same column of next line	Shift + Down arrow
Same column of previous line	Shift + Up arrow
One page down	Shift + PgDn
One page up	Shift + PgUp
Left one word	Shift + Ctrl + Left arrow
Right one word	Shift + Ctrl + Right arrow
End of window	Shift + Ctrl + PgDn
Beginning of window	Shift + Ctrl + PgUp
End of file	Shift + Ctrl + End
Beginning of file	Shift + Ctrl + Home

Other editing commands

<i>To ...</i>	<i>Press</i>
Open a file	F3
Save a file	F2
Undo	Alt + Backspace

23. The UTILITIES Procedures.

The UTILITIES page of the user interface houses procedures for performing a variety of tasks, mostly dealing with the management of data and data files. The procedures are organized into a set of tabbed pages within the UTILITIES page:

The OUTLIERS Procedure: The OUTLIERS procedure is used to aid the user in identifying any outlying observations in a sample.

The PERMSUB Procedure: The PERMSUB procedure allows the user to permanently apply the instructions contained in a SUB file to an XYZ or MXYZ file, creating a new data file with fewer landmarks. This utility is intended to make repeated uses of a particular SUB file more convenient.

The CONVERT Procedure: The CONVERT procedure allows the user to convert *WinEDMA* files to input formats that are used with other software packages. Similarly, the procedure can be used to convert files with other formats to *WinEDMA* format. Supported formats include *Morpheus* and *GRF-ND*.

The SCREEN Procedure: The SCREEN procedure screens XYZ files for missing data. Observations with missing landmarks cannot be used in many EDM procedures, and this utility provides a convenient means of identifying those observations.

The SCALE Procedure: The SCALE procedure is used to permanently scale the data in XYZ or MXYZ files, according to some user-specified scaling variable. The scaled data are output to a new data file.

The MODTEST Procedure:

The MEASERR Procedure:

The HEADER Procedure:

The DBIO Procedure: The DBIO procedure is used to export *WinEDMA* data files to a variety of database formats supported by the Borland Database Engine (Inprise Corporation). The procedure can also be used to import landmark data from these database formats into *WinEDMA* format.

The Mscribe Procedure: The Mscribe procedure allows the user to collect landmark coordinate data using Immersion Corporation's MicroScribe digitizer. The data are saved in the form of *WinEDMA* XYZ files or as "raw" files (containing annotations, repeated measurements, etc.).

24. Program Configuration.

The user cannot alter the configuration of the current program (version 1.0.1 beta; release date November 1, 2001). The page of configuration options is under construction. Documentation will be provided when the page becomes available.

25. Error Messages.

An effort has been made to provide the *WinEDMA* user with helpful error messages. Unlike previously-distributed EDMA software, *WinEDMA* uses error-handling code that describes problems in English and allows the user to fix the problem without crashing the program. However, it is impossible to anticipate every possible error that a user might make, so incomprehensible messages and program crashes are still possible on occasion. If such an error cannot be fixed, please contact user support.

Most of the errors commonly encountered with *WinEDMA* are either: 1) errors in data file formatting; or 2) various problems with the landmark coordinate data themselves. There are also various errors associated with use of the computer itself (e.g., trying to read from an empty floppy drive or write results to a full disk). The more common errors are described here.

Errors in data formatting:

Past experience suggests that many error messages that *WinEDMA* users encounter involve errors in formatting data files correctly. These errors appear when the *Read data...* button is pressed. However, the frequency of these errors seems to diminish as the user gains experience with the program. Most problems seem to be in formatting the header. Here are some suggestions for fixing data-formatting problems:

Header:

- Remember that *WinEDMA* files *never* include any blank lines.
- Make certain that the three *required* lines of the header are present and in the correct order. The title line comes first, followed by the line with the file-type indicator, followed by the parameter line.
- Make certain that all

Data errors:

In these cases, the input files are formatted correctly (so there are no error messages when the *Read data...* button is pressed), but there are actually problems with the data themselves. The most common (and often most frustrating error) is the “Variance too large” error. It occasionally occurs during estimation of sample statistics from raw data, but usually occurs during resampling. In either case, the program aborts the procedure and the user does not receive all of the results that were requested.

The suggested first step in diagnosing “Variance too large” errors is running the data through the ESTIMATION procedure. If there is a problem here, there will also be problems with other procedures. However, the opposite is not always true. Data that are perfectly “clean” and that pass through ESTIMATION without a problem can have problems in procedures like FORM, GROWTH, and SHAPE, all of which use resampling (bootstrapping).

If the data cannot be used successfully with ESTIMATION, there are several possible causes. Some errors stem from problems in data collection. Is the total number of lines of coordinate data equal to the number of landmarks per observation times the number of observations (K times N)? If not, there may be an instance where a landmark was digitized twice or where a landmark was skipped. If the number of lines is correct, then the next step is to determine whether the landmarks were digitized in the correct order for every observation. This is most easily done by using another program to plot the raw data one specimen at a time. A highly recommended program for doing this is Dennis Slice's (199x) *GRF-ND* program (life.bio.sunysb.edu/morph), which plots labeled landmarks and (for three-dimensional data) allows the user to rotate the landmark configuration interactively. A very effective way of diagnosing problem data is through Procrustes (least-squares) superimposition of the data. Many outliers are easily spotted when the individual observations are plotted as vectors from the landmark-specific means.

Other errors:

WinEDMA's English-language error messages are sometimes followed by less-understandable "run-time" error messages. A summary of some commonly-encountered run-time errors follows:

26. Bibliography.

Methodological references:

The procedures to which each citation is related are in brackets. More general references are marked "GENERAL".

Cole, T.M. III, DeLeon, V.B., Lele, S. & Richtsmeier, J.T. 1998. (abstract) [CLIQUE]

Cole, T.M. III, DeLeon, V.B., Lele, S., & Richtsmeier, J.T. 1999. (abstract) [CLIQUE]

Cole, T.M. III & Richtsmeier, J.T. 1998. A simple method for visualization of influential landmarks when using Euclidean distance matrix analysis. *American Journal of Physical Anthropology* **107**, 273-283. [INFLUENTIAL]

Corner, B.D., Lele, S. & Richtsmeier, J.T. 1992. Measuring precision of three-dimensional landmark data. *Journal of Quantitative Anthropology* **3**, 347-359. [FUZZY]

Lele, S. 1991. Some comments on coordinate free and scale invariant methods in morphometrics. *American Journal of Physical Anthropology* **85**, 407-415. [GENERAL]

Lele, S. 1993. Euclidean distance matrix analysis (EDMA): estimation of mean form and mean form difference. *Mathematical Geology* **25**, 573-602. [GENERAL; ESTIMATION]

Lele, S. 1999. Invariance and morphometrics: a critical appraisal of statistical techniques for landmark data. In MAJ Chaplain, GD Singh & JC McLachlan (eds): *On Growth and Form: Spatio-temporal Pattern Formation in Biology*. Chichester: Wiley, pp. 325-355. [GENERAL]

Lele, S. & Cole, T.M. III. 1996. A new test for shape differences when variance-covariance matrices are unequal. *Journal of Human Evolution* **31**, 193-212. [SHAPE; ESTIMATION; RANDOM]

Lele, S. & McCulloch, C.E. 1999. Invariance and morphometrics. *Journal of the American Statistical Association* (in review). [GENERAL; ESTIMATION]

Lele, S. & Richtsmeier, J.T. 1990. Statistical models in morphometrics: are they realistic? *Systematic Zoology* **39**, 60-69. [GENERAL]

Lele, S. & Richtsmeier, J.T. 1991. Euclidean distance matrix analysis: a coordinate free approach to comparing biological shapes using landmark data. *American Journal of Physical Anthropology* **86**, 415-428. [FORM]

Lele, S. & Richtsmeier, J.T. 1992. On comparing biological shapes: detection of

- influential landmarks. *American Journal of Physical Anthropology* **87**, 49-65. [INFLUENTIAL]
- Lele, S. & Richtsmeier, J.T. 1995. Euclidean distance matrix analysis: confidence intervals for form and growth differences. *American Journal of Physical Anthropology* **98**, 73-86. [FORM; GROWTH]
- Rao CR & Suryawanshi S. 1996. [R&S]
- Rao CR & Suryawanshi S. 1998. Statistical analysis of shape through triangulation of landmarks: a study of sexual dimorphism in hominids. *Proceedings of the National Academy of Sciences, USA* **95**, 4121-4125. [R&S]
- Richtsmeier JT & Lele S. 1993. A coordinate-free approach to the analysis of growth patterns: models and theoretical considerations. *Biological Reviews* **68**, 381-411. [ESTIMATION; FORM; GROWTH; HYPOTHETICAL]
- Richtsmeier JT, Cheverud JM, Danahey SM, Corner BD & Lele S. 1993. Sexual dimorphism of ontogeny in the crab-eating macaque (*Macaca fascicularis*). *Journal of Human Evolution* **25**, 1-30. [GROWTH; HYPOTHETICAL]
- Richtsmeier JT, Corner BD, Grausz HM, Cheverud JM & Danahey SE. 1993. The role of postnatal growth pattern in the production of facial morphology. *Systematic Biology* **42**, 307-330. [GROWTH; HYPOTHETICAL]
- Richtsmeier, J.T., Cole, T.M. III, Krovitz, G.E., Valeri, C.J. & Lele, S. 1998. Preoperative morphology and development in sagittal synostosis. *Journal of Craniofacial Genetics and Developmental Biology* **18**, 64-78. [ORDINATION]
- Sneath, P.H.A. & Sokal, R.R. 1973. *Numerical Taxonomy*. San Francisco: Freeman. [ORDINATION; CLUSTERING]
- Valeri, C.J., Cole, T.M. III, Lele, S. & Richtsmeier, J.T. 1998. Capturing data from three-dimensional surfaces using fuzzy landmarks. *American Journal of Physical Anthropology* **107**, 113-124. [FUZZY]
- Applications:*
- Burrows, A.M., Richtsmeier, J.T., Mooney, M.P., Smith, T.D., Losken, H.W. & Siegel, M.I. 1999. Three-dimensional analysis of craniofacial form in a familial rabbit model of nonsyndromic coronal suture synostosis using Euclidean distance matrix analysis. *Cleft Palate – Craniofacial Journal* **36**, 196-206.
- Lague & Jungers 1999

Richtsmeier et al. 1992. Faces paper

Richtsmeier et al. 1993. Macaque paper

Richtsmeier, J.T., Cole, T.M. III, Krovitz, G.E., Valeri, C.J. & Lele, S. 1998.
Preoperative morphology and development in sagittal synostosis. *Journal of Craniofacial Genetics and Developmental Biology* **18**, 64-78.

Singh, G.D., McNamara, J.A. Jr. & Lozanoff, S. 1998. Morphometry of the midfacial complex in subjects with class III malocclusions: Procrustes, Euclidean, and cephalometric analyses. *Clinical Anatomy* **11**, 162-170.

Ungar, P.S., Walker, A & Coffing, K. 1994. Reanalysis of the Lukeino molar. *American Journal of Physical Anthropology* **94**, 165-173.

Other References:

Cock, A.G. 1966.

Davies & Hinkley 1996

Efron & Tibshirani 1991.

Efron & Tibshirani 1996.

Goodall, C.R. 1991.

Gower, J.C. 1966.

Hall & Martin

Silverman, B. 1986.

Slice, D.E. 199x.

27. Distribution and User Support.

WinEDMA is distributed at not cost via two sites on the World Wide Web. The first site is <http://c.faculty.umkc.edu/colet> and the second is <http://faith.med.jhmi.edu>.

For user support or inquiries about *WinEDMA*, send e-mail to colet@umkc.edu. The following mailing address can be used as an alternative:

Theodore M. Cole III, Ph.D.
Department of Basic Medical Science
School of Medicine
University of Missouri – Kansas City
2411 Holmes St.
Kansas City, MO 64108
USA

28. Future Plans.

There are two major developments planned for Windows-based EDMA software in the near future (not including the addition of new procedures and features to *WinEDMA*). These programs will initially be developed as standalone Windows applications, but may eventually be incorporated into *WinEDMA*.

Development 1:

A pair of programs, called *WinEDMA BatchMaker* and *WinEDMA BatchRun*, which will allow batch-mode running of EDMA analyses. The user interface for *WinEDMA BatchMaker* will be identical to the *WinEDMA* interface. However, instead of running the analysis when the "Run" button is pressed, the program will write information from the interface (e.g., file names, bootstrap options, etc.) to a text file. If additional analyses are chosen, their information is appended to that file. When the batch file is complete, it can be read by *WinEDMABatchRun* and all of the analyses may be run unattended (overnight or while the user works on something else). *The target date for a test version is June 1, 2002.*

Development 2:

A program for the use of graphics with EDMA, called *EDMASketch*. When we present the results of an EDMA study, we usually scan in a photo or a line drawing of a representative specimen, and then we add landmark markers and labels using a program like Microsoft PowerPoint or Corel PhotoPaint. We then draw in the distances that interest us (e.g., the distances associated with the extremes of a form-difference matrix or the seed triangle for a landmark clique), perhaps in different colors. With *EDMASketch*, the user will be able to import a scanned image (as a bitmap or JPEG file), and add landmark markers and labels using the mouse. The user will then be able to draw in selected distances interactively, with the ability to save the altered images, so they can be output to a printer, a slide maker, or another program (e.g., to a PowerPoint presentation). *The target date for a test version is June 1, 2002.*