CMAT©

Summary Manual

Extension of C Language:
Matrix Algebra, Statistics,
Nonlinear Optimization and Estimation
Release 9 (September 2016)

Wolfgang M. Hartmann [1]
Friedrich-Ebert-Anlage 46 / I
D-69117 Heidelberg, Germany

August 18, 2018

[1] Thanks to my dear wife Walee and my old brave Apollo 4500.

Contents

List of Tables

Chapter 1

User Software License Agreement

Please read this license carefully before using the software. By installing or using this software, you are agreeing to be bound by the terms of this license. If you do not agree to the terms of this license, either contact the author or promptly remove the software.

1. This software may not be distributed to third parties.
2. Supply of any part of this software as part of another software requires the separate prior written agreement of the CMAT managers, which may include financial terms.
3. This software is copyrighted and may not be modified, decompiled, reverse engineered, or disassembled.
4. Due acknowledgment shall be made of the use of CMAT in research reports or publications.
5. The academically priced version of this software may only be used for teaching or research.

As long as the author has a regular income, this software is distributed as non-profit and patience is expected since *"everything free comes without guarantee"*. The author is grateful for responses by users such as bug reports or proposals for improvement.

Chapter 2

Introduction

CMAT is one more interactive matrix language, similar to (but not a clone of) MATLAB[582], O-Matrix, GAUSS, SAS/IML®[?], SCILAB, EULER, OCTAVE, or YORICK. The author created CMAT to have a tool that fits his own needs and those of people like him who need a language which is numerically stable and very efficient in computer time and memory usage.

The following principles dominated the design of CMAT:

- CMAT is an interpreter and compiler. CMAT executes the programming statements in the order in which they are typed. At the same time, the statements are compiled and assembler code is written to an output file. Exceptions of this process are special compound statements, e.g. `switch` or forward `goto`'s where the execution (but not the compilation) of statements can be delayed or suspended. When the closing bracket of the compound statement or the label of the forward jump is found, the whole block of compiled code is executed. Also, when the compiled code segment is passed later again, for example after a backward `goto`, the compiled assembler code is executed. User specified functions must be defined before they can be called. Therefore, functions are always executed by running through prior compiled assembler code. When CMAT is run with standard input (interactively), the output is written to standard output by default but can be redirected to two output files, one for error messages, warnings, and notes, the other for results. If CMAT is run with an input file (in batch mode), the output is written automatically to the two files. Using the `#include` command you can redirect the program input to other files.

- The language offers easy tools to write data objects into temporary or permanent files and to access them by their name in the computations. When referring to the name of the object, CMAT remembers whether the object is stored incore or in a file.

- CMAT's syntax is similar to that of the C language. Users who know how to write C code do not need to look up a syntax manual for the scalar operations. In addition to the interpreter capability, there are, however, some important differences to C. And the language is extended to deal with vectors, matrices, tensors, data lists (containing scalars, vectors, matrices, tensors, structs, and sublists), k dimensional trees. See page 20 for more details.

- CMAT provides an easy interface for almost all subroutines available in LINPACK, ARPACK, EISPACK[780][287], and LAPACK. Additionally, it provides a set of numerical subroutines which are currently only difficult to come by, including some for sparse matrix analysis, linear and nonlinear optimization, and especially nonlinear and robust, statistical estimation methods.

Some of the results reported in this manual depend on the C compiler used and may slightly differ from those obtained by the user. CMAT was developed in two versions:

1. using the Domain C compiler (see [226]) running in BSD 4.3 version of UNIX [129].
2. using the Watcom C/C++ and Fortran 77 compilers running in Windows NT version 4.

CMAT offers an interface to *gnuplot* for graphical output. Currently there are two ways to connect to `gnuplot`:

interactively The `gnuplot { ... }` syntax permits the input of code submitted to the *gnuplot* software.

batch mode The `rc = gpbatch(gpfiles)` function executes *gnuplot* command scripts stored in one or more `.gp` files equivalent to a DOS or Linux call:
`gnuplot gpfil_1 ... gpfil_n`. The `gpfiles` input argument must be either a string scalar or vector of strings with the path names to the files. As with *gnuplot* the `"-"` "file name" refers to an interactive input section.

The user must download a current version of the *gnuplot* software from the internet and unzip it into a `gnuplot` directory parallel to the c̃mat directory.

The use of some terminals (output devices in *gnuplot*) needs downloading of additional software from the internet:

svg needs the `SVGView.exe` software, e.g. downloadable for no charge from Adobe.

epslatex needs the *graphicx* software which is downloadable for no charge from the internet.

Specificially two programs are used from BSD: YACC for parsing and LEX for lexical analysis. The PC version uses *MKS Lex & Yacc* [623]. This manuscript was typeset using LaTeXby PCTeX32 [675].

The author of CMAT would like to thank the following scientists and friends for contributing software and advice:

- ARPACK Team (Lehouq, Sorensen, and Yang)
- LAPACK team, especially J. Dongarra and Z. Bai
- Michael Berry, University of Tennessee
- Åke Björck and Mikael Adlers, Linköping Universitet, Sweden
- Paul Boggs, National Institute of Standards
- Michael Browne, Ohio State University
- Frank Bretz, Hannover University, Germany
- Andreas Christmann, University of Dortmund
- William J. Cook, University of Waterloo, Canada
- Ove Edlund, LuleåUniversity of Technology, Sweden
- John J. Forrest, Cbc and Clp developer retired from IBM Watson
- Alan Genz, Washington State University

- Robert Hartwig, NC State University, Raleigh
- Don Hedeker, University of Illinois, Chicago
- Harry Joe, University of British Columbia
- Linda Kaufman, Lucent Technologies
- Jan de Leeuw, University of California, Los Angeles
- Bernd Luevelsmeyer, HEITEC Erlangen
- Kaj Madsen and H. Nielsen, Technical University of Denmark
- Alberto Maydeu-Olivares, Barcelona and Madrid
- Olvi Mangasarian, University of Wisconsin, Madison
- Rod McDonald, University of Illinois, Champaign
- Jorge Moré, Argonne National Laboratory
- Stephen Nash, George Mason University
- Esmond Ng, Oak Ridge National Laboratory
- Mustafa Pinar, Bilkent University, Turkey
- Michael J. D. Powell, University of Cambridge, UK
- Thomas Ragg, quantiom (programmer of frontend)
- Jim Ramsay, Mc Gill University, Montreal, Canada
- Peter Rousseeuw, University of Antwerp, Belgium
- William W. Rozeboom, University of Alberta, Canada
- Michael Saunders, Stanford University
- Robert Schnabel, University of Colorado
- Paul Somerville, University of Central Florida
- Peter Spelucci, Technical University Darmstadt, Germany
- Yoshio Takane, McGill University, Montreal
- Gail Towne, Winston-Salem NC
- Stephen Wright, Argonne National Laboratory

Any bugs in CMAT should be contributed entirely to its author. Although the author of CMAT is employed at a major software company, the design and implementation of CMAT was done solely in his spare time, and there is no other connection to his employers product than his experience as a programmer. CMAT contains some modules which look very similar to software the author developed as an employee for this software company. However, the code here is developed completely independend. The first version of CMAT was developed on an Apollo 4500 computer purchased at an employee sale in 1993 for the amount of $ 10. Money was needed also for

1. Mortice-Kern Yacc and Lex software.
2. The Latex compiler by PC TeX.
3. The Watcom C/C++ and FORTRAN 77 compilers with documents, which are free of charge in the meantime.
4. Some newer PC hardware and computer supplies, like CD ROM and backup drives and cartridges. Power supply.

Chapter 3

Installing and Running CMAT

3.1 Installing CMAT in MS Windows

3.1.1 Use of the Installer

The installer writes all necessary files into a `cmat` directory and installs Tcl TK for the frontend.

1. Run the file install.exe which is on the CD ROM.
2. Your screen will turn blue and all necessary files are copied to a directory tree `c:\cmat` or another location of your choice.
3. After the file copying process you must define a path for opening the frontend program:
   (a) The frontend program is `c:\cmat\mytest\cmat-fe.tcl`
   (b) Use the *Open with ...* box to define the program `c:\cmat\tcltk\bin\wish83.exe` for opening the frontend file `cmat-fe.tcl`
4. Now the frontend display should be on your screen for the first time. For future use you should create a shortcut icon to `c:\cmat\mytest\cmat-fe.tcl`

See the `README.txt` for a few more details.

3.1.2 Manual Installation in MS Windows

The CD contains the following directory structure:

1. cmat\com: with subdirectory: cmat\com\main: contains the Windows executable `cmat.exe` together with a number of MS Windows DLL's which are needed for running CMAT:

   `xarpack.dll` Arpack ([509])

   `xfsrc.dll` lots of Fortran (public domain) code

   `xlapack.dll` real Lapack subroutines (Fortran version, see [15]) `zlapack.dll` complex Lapack subroutines (Fortran version)

`xfsrc.dll` subroutine library

`xfsrc2.dll` subroutine library

`xfsrc3.dll` subroutine library

`xqhull.dll` subroutine library

`cmatm0.m` contains storable summary information about the message file `cmatm.m`. The content of this file is read into memory during the initialization of CMAT.

`cmatm.m` contains the compact message file as it is used during the CMAT run. The content of this file is not stored into RAM.

See the following sections about the two kinds of CMAT execution: interactively and in batch mode.

2. cmat\doc: contains the following files of the manual:

   `cm_all.pdf` contains the main user manual

   `cm_sum.pdf` contains a short summary

   `cm_tut.tex` contains the tutorial part

   `cm_ref.tex` contains the reference part

   `cm_det.tex` contains the *Details* chapter and the chapter with *Some Useful CMAT Modules* and a chapter with a collection of more or less famous *Data Sets*.

   Note that `cm_all.pdf` contains all information, whereas the other files contain only parts of `cm_all.pdf`. This directory also contains the corresponding `...dvi` files.

3. cmat\save: contains data objects created by the `obj2fil()` function. The names of such files can be used like variables in statements.

4. cmat\tcltk: contains the binary Tcl and Tk files for running the frontend.

5. cmat\mytest: contains the TclTk script of the user frontend. If you are not pleased with its function you may enhance this code. Otherwise this is an almost empty directory designed to be your working playpen. Of course you may create directories of your own choice. However, working in a directory one level beneath the `cmat` directory is preferrable for finding the executable `cmat.exe` in `cmat\com\main`.

6. cmat\test: contains a number of files for testing cmat. The .inp files contain the input statements for CMAT, the .log files contain the log output, and the .txt files contain the text output.

7. cmat\tgplt: contains a number of files for some `gnuplot` applications

8. cmat\tmicro: contains a number of files for the analysis of microchip data

9. cmat\tnlp: contains a number of files for testing the `lp`, `qp`, and `nlp` functions of cmat.

10. cmat\tsem: contains a number of files for testing the `sem` function of cmat.

You are highly recommended to keep the directory structure as it is, especially if you intend to change the message files. For using the Tcl/TK frontend you should perform the the last steps which were described in the section above (3.1.1).

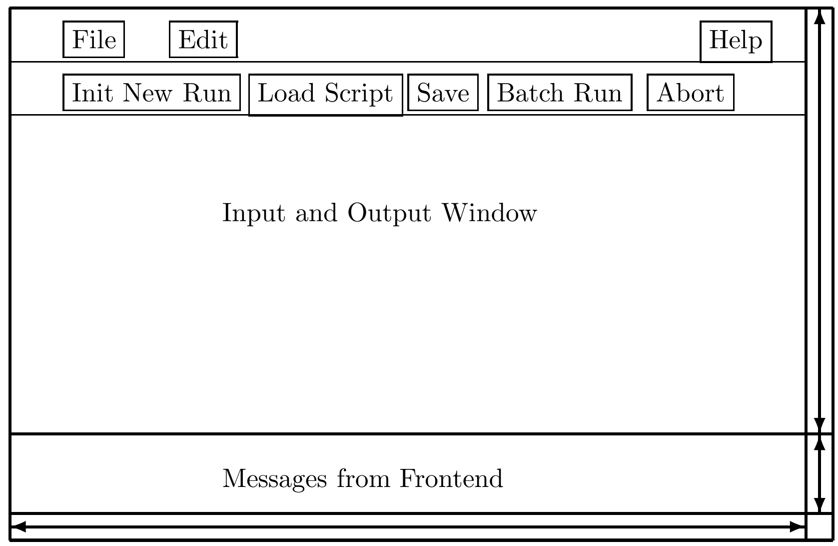

Figure 3.1: Description of Frontend

3.2 Execution of CMAT

The ideal environment to run CMAT is an operating system that provides both, an editor and a shell, e.g. UNIX or Linux. The PC version contains a frontend written in *Tcl TK* which makes interactive and batch processing easier. There are alternatives for the PC, if you don't want to use our frontend:

- For batch processing, even MS DOS would be sufficient, for the interactive processing one of the DOS Extender could be necessary.
- The developers preferred way is to use GNU NTEmacs (working in Windows NT and 95) together with BASH. Working with tools like *MKS Toolkit* would be possible too.

The Linux and Unix operating systems with command line input are more appropriate for CMAT. There will be such versions later in 2003.

3.2.1 Execute CMAT Interactively

The PC version of CMAT comes with a small frontend (GUI) since the command line input with MS DOS is not very comfortable and not many people know how to work with GNU NTEmacs and BASH. The Installer program loads not only the Tcl TK code for the frontend, it will also install the Tcl and TK software itself if it is not already installed at the users computer.

The frontend window is divided into two parts. The upper part is for command line input and all output which normally is written to the `.log` and `.lst` file during batch processing. The lower part contains only output in form of messages from the frontend program. The upper part of the input window contains a row of menus and a row of buttons. The following menus are available:

File tis menu contains only the choices `Load`, `Save`, `UF_Kill`, and `Exit`.

Edit this menu only contains `Clear`.

Help this menu contains choices to open online help documents (summary, tutorial, reference, detail, and complete users manual) and it contains an `About` window button.

The second row contains the following buttons:

Init New Run this will clean the input window and start another CMAT invocation

Load Script this will open a browser window permitting the input of a script which is immediately executed. The input and output is visible in the upper window.

Save for saving the content of the input window into a file,

Batch Run this will open a browser window permitting the input of a file name which is immediately executed. The output is written to the `...log` and `...lst` files.

Exit for exiting the application.

You can either type CMAT input into the input window, or you can cut and past using `CTRL x`, `CTRL c`, and `CTRL v` keys or you can load a script using the corresponding button. In addition the following keys can be used for moving the cursor and view in the display:

Page Up, Page Down for moving cursor and view one page up or down in the output.

Home, End for moving the cursor to the begin or end of the current line.

CTRL Home, CTRL End for moving the cursor to the first or last line of the output window.

To run CMAT interactively with command line input and standard output (e.g. by using NTEmacs with BASH oder some kind of DOS extender) you need to invoke CMAT by typing `cmat` into the command line and start the program input after an opening compound statement { on a new input line. CMAT is terminated either with an `exit;` statement or by closing the initial compound statement with }. Therefore, the shortest CMAT program is

```
cmat
{ ; }
```

. When passing through the input statements the first time, the statements are compiled and executed simultaneously, with the following exceptions, where execution but not the compilation is postponed:

forward jump : This is a `goto` statement with a target label later in the code. The execution is delayed until the target label is found.

if statement : The execution is delayed until the next statement is started. If the `if` statement is connected with a compound statement, the execution is performed until the compilation of the first statement after the closing brace } is started.

switch statement : The execution is delayed until the first statement after the related compound statement is found.

for and while statement : This is equivalent to the `if` statement.

conditional operator : The execution of the `?` and `:` conditional statement is delayed until the next statement is started.

In later passes the compiled assembler code is executed. By default the output is directed to standard output but can be redirected using the `logfile(`*filename*`)` or `txtfile(`*filename*`)` standard functions. Calling the two functions without an input argument *filename* redirects the output to standard output. During the program input, an `#include "`*pathname*`"` can redirect the standard input to the input file *pathname*. This feature specifically permits the execution of CMAT commands in an ASCI file which was generated during the run. After executing the commands in the included input file, the program input automatically returns to the standard input or the file from where the last `#include` command was issued. Note, that the length of the *pathname* argument is restricted (see page 19).

Online help is available only when working with the Frontend. Otherwise the corresponding .pdf files in `cmat\doc` must be opened by the user. The document files make use of the *hyperref* package which permits bookmarks.

3.2.2 Execute CMAT in Batch Mode

You may use the Tcl TK frontend which is delivered with CMAT, however you can also execute CMAT statements inside an ASCII file *fname.inp* by typing `cmat fname`. The input file *fname.inp* must then start with an opening { brace of a compound statement. Like in interactive mode, `#include "`*pathname*`"` commands redirect the program input to other files specified in the pathname. This feature makes it possible to execute CMAT code which was written to an ASCI file earlier at the same run. By default all output is directed to logfile *fname.log* and the output file *fname.txt*. If there exist already a file with the same name *fname.log* (resp. *fname.txt*) the old files are renamed into *fname.blg* (resp. *fname.bxt*) to prevent the loss of old results. However, like in interactive mode, the output can be redirected to other files or standard output using the `logfile()` and `txtfile()` functions.

3.2.3 Utility Files

During it's execution CMAT generates a number of utility files. To enable simultaneous batch runs of `cmat` the file names are extended by an integer number which is the job process ID *pid*.

__opro__... created with every execution of `cmat`

__memb__... created with every execution of `cmat`

__data__... created with every execution of `cmat`

__scra__... created with every execution of `cmat`

_uf_... prefix of utility files created by specific functions.

These are temporary files which normally are deleted during a normal exit of `cmat`, e.g. using the closing }. However, as a result of aborted runs a number of those files may remain in your working directory, e.g. `mytest`. Those are being deleted at the begin of the next execution of `cmat`. They could also be deleted by running the batch file `uf_kill`. The frontend has a specific command for that in the `File` pulldown menu.

Chapter 4

Restrictions and Comparisons

4.1 Some Shortcomings

Due to the limited time and financial resources of the author we have to report the following list of shortcomings in CMAT:

1. Other Operating Systems: At an early stage, CMAT was developed in UNIX using BSD Lex and Yacc. A host.h header file still defines most of the differences, i.e. path names, functions with variable arguments, DLL declarations. Therefore, it should be quite easy to make running versions for Unix, Linux, and the Mac.
2. Dynamic Binding: It should be possible to link object code of subroutines written in C, C++, or Fortran to the executable. Specifically for the Windows OS, the linking of user defined subroutine DLL's should be possible.
3. String processing: Not many operations with string arguments are available at this time. However, most of the string processing subroutines of the standard C runtime library are available.
4. Arrays with more than 2 dimensions (tensors) are now supported but some tensor operations, especially with tensors containing string data, may still have some problems.
5. Not all index operations try to preserve prior assigned row and column names.

4.2 Length Restrictions

The following are constants, which may be easily changed for a more customary version of CMAT:

- Maximum length of variable, function, or file names is 32 chars.
- Maximum length of row and column names is 32 chars.
- Maximum length of row and column labels is 80 chars.
- Maximum path length of directories (used in the arguments of some functions) is 256 characters.

- Maximum length of an input token is 1024 characters. Note, that this normally restricts the length of input string data (embedded into quotes). There is NO restriction other than the available memory for the internal length of string data which must be provided to Yacc and Lex. Note, that the length comments is not restricted.
- A maximum of 10 lookup directories may be specified.

4.3 Differences with C Language

Most of the syntax of CMAT is compatible to that of the C language when working with scalar data types. However, there are a few exceptions.

1. Differences when working with scalars:
    - The scope keywords `extern` and `static` are not implemented. Only inside functions are variables locally defined. However, variables can be easily reset, freed, or reallocated, e.g. when using type declarations or simple assignments.
    - In CMAT variable types do not need to be declared, but variables can be casted to specific types. If type declarations are used then the declared variables are reallocated and initialized. A variable should be casted if only its type but not its value should be changed.
    - CMAT has only one integer type (which is always long int) and only one floating point data type referred to as `real`, which is internally double precision.
    - CMAT permits the complex data type and easier ways of string processing.
    - CMAT provides additional casting like `complex`, `conj`, and `imag`.
    - There is obviously no need for pointers in CMAT.
    - Subroutine definitions can appear anywhere in the code but must be defined before being called.
    - Input and returned arguments in subroutines are separated in CMAT. For multiple returns, the return statement in the function definition must have a list of variable names in parentheses, e.g. `return(a,b,c);`. The function can then be called in the form `<a,b,c> = f(d,e,f);`, where arguments `d,e,f` are inputs and arguments `a,b,c` are returned.
    - Function calls may contain a `global` clause. The `global` clause could be necessary when functions are called by other functions and must have a specific number of input arguments.
    - The CMAT language provides the additional `link`, `option`, and `poptn` keywords.
    - In CMAT many of the standard functions have default argument settings and some of the standard functions return different types of results.
    - Some of the standard functions were not implemented because there was either no need (e.g. memory allocation) or they did not seem to be important enough in an early release.
2. Differences when working with vectors, matrices, and tensors:
    - Data objects in CMAT can be scalars, vectors, matrices, tensors, kD trees, and single indexed lists of those objects. Data lists may contain the following data objects:
        - numerical (int, real, complex) and string scalars
        - vectors and matrices
        - tensors

  - (sub-)lists
- In CMAT matrix entries $a_{i,j}$ are referred to in the form `a[i,j]` and not like `a[i][j]` as in C.
- A tensor `t[i,j,...]` is indexed in the same way.
- Tensors, matrices and vectors may be initialized in type declarations, e.g. `real a[5,15]=1.;` is a valid initialization of the 5×15 matrix $\mathbf{A}$.
- Lists are initialized in the form `list l[5];` specifying the name and an initial length of the list. Each entry then is initialed to a scalar with missing value.
- Structs are declared using the keyword `struct`. Entries of structs are referred to by compound names, where the struct name is separated from the entry name by a dot. E.g. `a.b.c` means that struct `a` has a struct entry `b`, and substruct `b` has an enyry `c`.
- The additional `free` keyword can be used to free memory of strings, vectors, matrices, tensors, kD trees, or entire lists which are no longer needed in the code.
- CMAT provides additional casting like `herm`, `symm`, and `tri2sym` to matrix objects.

Chapter 5

Summary of Operators, Keywords, and Functions

5.1 General Remarks

The list of functions in CMAT contains many of the standard functions of a C compiler library (see [357], chapter 11). However, many of these functions were extended to both, complex arithmetic and matrix and tensor notation. In a few cases the functions had to be modified for practical purposes. For example, many functions have a minimum and a maximum number of arguments, and arguments inbetween those two ranges may be set to missing values permitting internal default settings of those arguments.

Some of the traditional one-argument functions (like `abs`) are trivially extended to matrix notation, so that the scalar function is applied to every element of the object and the result is an object of the same dimension. Some of the traditional 2-argument functions (like `atan2` or `fmod`) are extended in the following most trivial way:

- If one of the two arguments is scalar and the other argument is a matrix or vector, then the function is performed using the scalar and each entry of the other operand. The result is of the same size as the matrix or vector argument. The function is performed only on numeric data, string data are not processed.
- If both arguments are vectors or matrices, then both must have the same dimension (same number of rows and columns). The function is performed pairwise with corresponding pairs of entries of the two operands.

When the traditional 2-argument functions `max` and `min` are being called with only one vector, matrix, or tensor argument, it will return the entry of the data object which has the largest resp. smallest value.

Some of the functions can deal with missing values (coded as a period) in an intelligent way. Usually this is mentioned in the `Restrictions` part of each function in the reference manual. Usually, a missing value is returned if one of the non-default arguments is a scalar with missing value. Note, that for a number of functions the run time option SECOND will execute an alternative implementation not necessarily one related to the LAPACK software product and not necessarily only for linear algebra functions. If you choose for some reason to select the SECOND option for a specific function call, you are adviced to turn off this option afterward by specifying the PRIME option since there is still some not fully tested code among the alternatives.

Some of the larger functions permit the input of options vectors or two-column options matrices as input arguments. Each entry of the options vector corresponds to a specific option and must be set to a missing value to permit the default value to be used. The default value must not necessarily be zero. Therefore it is highly recommended to initialize an options vector with missing values like `optvec = const(10,1,.);` to avoid that unexpected zero settings are being used.

5.2 Tables of Operators and Functions

5.2.1 Keywords

Keywords	
Keyword	**Short Description**
`break`	terminates loop or case in switch statement
`case`	case label in `switch` statement
`char`	cast to char data type
`complex`	cast to complex data type
`conj`	cast to conjugate complex
`continue`	continues with next loop in `for` for `while`
`default`	default label in switch statement
`else`	introduces the alternative to `if`
`exit`	terminates excution of CMAT
`for`	introduces loop statement
`free`	free memory belonging to a variable
`function`	introduces function definition
`global`	defines global arguments in function definition
`gnuplot`	start of *gnuplot* script
`goto`	executes jump to a label
`gpend`	end of *gnuplot* script
`herm`	cast matrix to be Hermitian
`if`	introduces conditional statement
`imag`	cast for imaginary part of complex number
`int`	cast for integer data type
`link`	specifying label as identifier for `return` jump
`list`	defining data lists (array of objects)
`option`	used for specifying runtime options
`poptn`	used for specifying runtime print options
`print`	generic `print` statement
`psd`	cast matrix to be positive semi-definite
`rename`	renaming variable or user function
`real`	cast data type to floating point real
`return`	returns either to function call or link statement
`sizeof`	returns size of object (`sizeof` is no function)
`struct`	defining data structs
`switch`	for conditional jump to a set of case labels
`symm`	cast matrix to be symmetric
`tri2sym`	cast lower or upper triangular matrix to symmetric
`void`	may be used for type of function definition
`while`	introduces loop statement

5.2.2 Operators

Operators	
Op	**Meaning**
/* . */	comment
( )	grouping terms in expression
	function parameters, arguments
[]	matrix literal
[””]	modified matrix literal
<>	function returns
{ }	compound statement
Unary Arithmetic Operators	
+	positive sign
−	negative sign
!	logical negation
˜	one’s complement
++	increment
−−	decrement
‘	transpose
.‘	nonconjugate transpose
Binary Arithmetic Operators	
+	addition
−	subtraction
∗	multiplication
/	forward division
\	backward division
%	remainder int division
∗∗	power
@	Kronecker product
.∗	elementwise multiplication
./	elementwise forward division
.\	elementwise backward division
. ∗ ∗	elementwise power
Shift, Relational, and Logical	
<<	left shift
>>	right shift
&	bitwise-and operation
|	bitwise-inclusive-or operation
ˆ	bitwise-exclusive-or operation
&&	logical-and operation
||	logical-or operation
<	less-than relation
>	greater-than relation
<=	less-than-or-equal relation
>=	greater-than-or-equal relation
==	equal-to relation
! =	not-equal-to relation
. <	elementwise less-than relation
. >	elementwise greater-than relation
. <=	elementwise less-than-or-equal-to rel.
. >=	elementwise greater-than-or-equal-to rel.
. ==	elementwise equal-to relation
.! =	elementwise not-equal-to relation
? :	Conditional Operator

Operators	
Op	**Meaning**
Concatenation Operators	
− >	horizontal from left to right
< −	horizontal from right to left
| >	vertical from top to bottom
< |	vertical from bottom to top
\ >	diagonal from left top to right bottom
< /	diagonal from left bottom to right top
Assignment Operators	
+ =	addition assignment
− =	subtraction assignment
∗ =	multiplication assignment
/ =	forward division assignment
\ =	backward division assignment
%=	remainder assignment
@=	Kronecker product assignment
<<=	left-shift assignment
>>=	right-shift assignment
&=	bitwise-and assignment
| =	bitwise-or assignment
ˆ=	bitwise-exclusive-or assignment
Subscript Reduction Operators	
+	sum
∗	product
| + |	sum of absolute values
| ∗ |	product of absolute values
∗∗	sum-of-squares
<>	maximum value
><	minimum value
<:>	index of maximum value
>:<	index of minimum value
< | >	maximum absolute value
> | <	minimum absolute value
<! >	index of max. abs. value
>! <	index of min. abs. value
Sorting and Ranking	
<	ascending values
>	descending values
<:	ranks of ascending values
>:	ranks of descending values
< |	ascending absolute values
> |	descending absolute values
<!	ranks of ascending absolute values
>!	ranks of descending absolute values
˜	permute randomly

5.2.3 Constants and Information

Constants and Information		
Function	**Spec**	**Short Description**
`attrib`(z $<$, "*prop*" $>$)	prop=	returns attributes of object
	"nopr"	do not print table of object attributes
	"name"	name of object (first argument)
	"otyp"	object type
	"dtyp"	data type
	"styp"	storage form
	"nrow"	number of rows (same as nrow(z) function)
	"ncol"	number of cols (same as ncol(z) function)
	"lbw"	lower band width
	"ubw"	upper band width
	"slen"	(maximum) length of strings
	"nstr"	number of strings
	"nmis"	number of missing values in object
	"nzer"	number of nonzero values in object
	"vmin"	smallest value
	"vmax"	largest value
	"nrm2"	2-norm (Frobenius or Euclidean) of object
	"rcond"	reciprocal condition of matrix
	"det"	determinant of matrix
`all`(a)		check for all entries nonzero
`any`(a)		check for any nonzero elements
b = `bmi`($w\_kg, h\_met <, bmi <, optn >>$)		*body-mass-index*
b = `branks`(a)		return tied and bivariate ranks of `a`
b = `chngtxt`($a, old, new <, numb >$)		change text in string data to other text
$clab$ = `clabel`(a)		return column labels `clab` of `a`
b = `clabel`($a, clab$)		assign labels `clab` to columns of `a`
$cnam$ = `cname`(a)		return column names `cnam` of `a`
b = `cname`($a, cnam$)		assign names `cnam` to columns of `a`
a = `cons`($< nr, nc, val, type >$)	type=	creates constant matrix
	'g'	full rectangular matrix (default)
	'd'	diagonal matrix
	'l'	lower triangular matrix
	'u'	upper triangular matrix
t = `const`($nsiz <, val >$)		creates constant tensor
b = `convert`($a, sopt$)	sopt=	conversion
		temperature (Fahrenheit, Celsius)
		metric $\leftrightarrow$ English
B = `cperm`($A <, perm >$)		permute columns of matrix

Constants and Information (Contd.)		
Function	**Spec**	**Short Description**
d = `date`($< sopt >$)	sopt=	returning date (depends on computer clock)
	missing	returns date in string form
	"ymd"	year*10000 + month*100 + day
	"year"	current year
	"month"	current month
	"day"	current day
`decrypt`($ofil$,$ifil$ $<, pwd <, optn >>$)		decryption of encrypted files and directories
$ostr$ = `decryp2`($istr$ $<, pwd <, optn >>$)		decryption of encrypted string objects
r = `deg2rad`(d)		conversion from degrees to radians
d = `diag`($a <, k >$)		creates (sub/super) diagonal matrix
b = `dia2vec`($a <, k >$)		moves (sub/super) diagonal from matrix to vector
i = `dim`(a)		returns the number of dimensions of scalar, vector, matrix, tensor
b = `dimlabel`(a,lab)		allocate dimension labels
lab = `dimlabel`(a)		fetch dimension labels
b = `dimname`(a,nam)		allocate dimension names
nam = `dimname`(a)		fetch dimension names
`encrypt`($ofil$,$ifil$ $<, pwd <, optn >>$)		encryption of files and directories (incl. subdirectories)
$ostr$ = `encryp2`($istr$ $<, pwd <, optn >>$)		encryption of string objects
`help`("$name$" $<$,"s" $>$)		opening *CMAT Reference Manual*
		at "name" using *Acrobat Reader*
i = `ide`(n)		creates $n \times n$ identity matrix
d = `insert`(a,b, $row <, col >$)		insert object b into object a
$tree$ = `kdtcrt`($x <, optn >$)		create kD tree
$< inds, dist, pnts >$ =		find nearest neighbor nodes of kD tree
= `kdtnea`($tree$,y $<, optn >$)		
$< nfnd, inds, dist, pnts >$ =		find nodes of kD tree in neighborhood balls
= `kdtrng`($tree$,y, $radius <, optn >$)		
`indx` = `loc`($a <, crit <, val >>$)	crit	returns index locations
	"nz"	nonzeros and nonmissing values
	"ms"	missing values
	"eq"	equal to zero or value
	"ne"	not equal to zero or value
	"gt"	greater than zero or value
	"ge"	greater or equal than zero or value
	"lt"	smaller than zero or value
	"le"	smaller or equal zero or value
b = `lstlabel`(a,$lab <, ind >$)		allocate entry labels of list
lab = `lstlabel`($a <, ind >$)		fetch entry labels of list
b = `lstname`(a,$nam <, ind >$)		allocate entry names of list
nam = `lstname`($a <, ind >$)		fetch entry names of list

Constants and Information (Contd.)		
Function	**Spec**	**Short Description**
`macon(`$<$ *prop* $>$`)`	prop=	generic and machine constants
	"nopr"	do not print table of machine constants
	"mlng"	largest long integer representation, i.e. $2^{mand} - 1$
	"mint"	largest integer representation
	"msht"	largest short integer representation
	"mand"	number of (base) digits of mantissa
	"bexp"	largest exponent before overflow
	"sexp"	smallest exponent before gradual underflow
	"ieee"	=1: computer has IEEE rounding properties, =0: no
	"base"	base of computer arithmetics
	"meps"	machine epsilon, smallest $\epsilon > 0$ with $fl(1. - \epsilon) < 1.$
	"mbig"	largest double float representation, i.e. $(base^{bexp}) * (1 - meps)$
	"msma"	smallest double float representation, i.e. $base^{(sexp-1)}$
	"savr"	value of save reciprocal, i.e. $1/savr$ does not overflow
	"l10b"	base 10 logarithm of largest double value
	"l10s"	base 10 logarithm of smallest double value
	"lnbig"	natural logarithm of largest double value
	"lnsma"	natural logarithm of smallest double value
	"pi"	$\Pi = 3.14159265358979324$
	"e"	Euler $e = 2.71828182845904523$
	"gamma"	Euler-Mascheroni $\gamma = .577215664901532861$
	"ib"	currently used index base
	"ls"	line size specification
	"ps"	page size specification
	"ifiw"	specified field width for output of integer
	"dfiw"	specified field width for output of double float
	"sing"	singularity threshold
	"spar"	range for sparsity ($0 <= spar <= 1$)
	"relz"	relative zero threshold
to `= mat2ten(`*ml*,*nind* $<$, *ordr* $>$`)`		move list of matrices into tensor
d `= mdy(m,d,y,`$<$ *sopt* $>$`)`	sopt=	obtain number of days from date input
	"year2"	two digits year input assuming 19th century date (def.)
	"y1900"	as "year2" but subtracts 1900 * 365 days
	"year4"	four digit year input
	"dm2y2"	as "year2" but day and month input is swapped
	"dm2y4"	as "year4" but day and month input is swapped

Constants and Information (Contd.)		
Function	**Spec**	**Short Description**
m = `mrand`($kind <, a, b >$)	kind=	creates multivariate random matrix
	"mnor"	multivariate normal $\mathcal{N}(\mu, \Sigma)$
	"tcau"	multivariate $\sqcup(\mu, \Sigma, df)$ with $df > 0$
	"pear"	multivariate Pearson with $c \in [-1, 1]$
	"khin"	multivariate Khintchine with $df > 0$
	"mnom"	multinomial for scalar n and r vector p
	"unis"	uniformly distributed inside n dimensional sphere
	"unos"	uniformly distributed on n dimensional sphere
	"unie"	uniformly distributed inside n dimensional unit cube
	"unoe"	uniformly distributed on n dimensional unit cube
$indx$ = `order`($vec1, vec2, \ldots$)		hierarchical ranking
$nxtprm$ =	sopt=	obtains n next permutations or combinations
= `permute`($sopt, n, curind <, upprng >$)	"perm"	for permutations (no replications)
	"comb"	for combinations (with replications)
$next$ =		permutations and combinations
`permcomb`($sopt, nfol, n, k <, istart >$)		with or without replications
id = `pid()`		returns interger ID of current process
$nams$= `prefname`($pref, nind$)		returns a vector of prefix names
d = `rad2deg`(r)		conversion from radians to degrees
a = `rand`($< nr, nc, type, \ldots >$	type=	creates random matrix
$< "dist" \| rank, \ldots >$)	'g'	rectangular `rand`($nr, nc, 'r', "dist", \ldots$)
	'd'	diagonal `rand`($n, n, 'r', "dist", \ldots$)
	'u'	upper triangular `rand`($n, n, 'r', "dist", \ldots$)
	'l'	lower triangular `rand`($n, n, 'r', "dist", \ldots$)
	's'	symmetric `rand`($n, n, 'r', "dist", \ldots$)
	'r'	rectangular `rand`($nr, nc, 'r' <, rank >$)
	'e'	symmetric `rand`($n, n, 'e' <, rank <, elo, ehi >>$)
	'o'	column orthogonal `rand`($nr, nc, 'o'$)
	dist=	see table at *Probability Functions* below
t = `randt`($nsiz <, dist, \ldots >$		creates random tensor
	dist=	see table at *Probability Functions* below
v = `randperm`(n)		creates a vector of the integers $1, \ldots, n$ randomly permuted

Constants and Information (Contd.)		
Function	**Spec**	**Short Description**
b = `ranktie`(a)		ranking entries of vector a with tied ranks averaged
$< nr, nc >$ = `rccount`($a <, val <, rel >>$)		count specific entries in rows and columns
d = `remove`(a,$inds$)		remove indexed entries from object a
b = `replace`($a, old, new <, rel >$)		change some values of matrix to other values
$rlab$ = `rlabel`(a)		return row labels `rlab` of `a`
b= `rlabel`(a,$rlab$)		assign labels `rlab` to rows of `a`
$rnam$= `rname`(a)		return column labels `clab` of `a`
b = `rname`(a,$rnam$)		assign names `rnam` to rows of `a`
B = `rperm`($A <, perm >$)		permute rows of matrix
b = `shape`($a <, nrb ><, ncb >$)		changes row and column number of matrix
$< c, ind >$ = `setdiff`(a,b)		set difference of `a` with `b`
$< c, index >$ = `setisect`(a,b)		set intersect of `a` with `b`
c = `setmembr`(a,b)		binary membership of `a` in `b`
$< c, index >$ = `setunion`(a,b)		set union of `a` and `b`
$< c, ind >$ = `setxor`(a,b)		set xor of `a` and `b`
a = `size`($a <, ind >$)		returns the size of the dimensions of vector, matrix, tensor
`sleep`($isec$)		suspend execution for `isec` seconds
$< ind, bmat, rank >$ = `sortrow`($amat$		sorting rows of a matrix w.r.t.
$<, key <, optn >>$)		key formed by a set of columns
b = `sortp`(a,ind)		partial sorting wrt. index
irc = `spawn`($path, args <, optn >$)		execute child process
c = `spmat`($nr, nc, rind, cind, val <, sopt >$		creates sparse matrix
`srand`($seed <, sopt >$)		initializes uniform random generators
$< irc, str >$ = `system`($commands$)		execute shell command and optional
		save output into string object
s = `time`($< sopt >$)	sopt=	returning time (depends on computer clock)
	"dtim"	returns daytime hour*3600 + minute*60 + second
	"hour"	current hour
	"min"	current minute
	"sec"	current second (default)
	"clock"	number processor clock ticks since starting job
ml = `ten2mat`($tens <, ordr >$)		move tensor into list of matrices
v = `ten2vec`($tens <, ordr >$)		move tensor to single vector
to = `tenperm`($tens$,$ordr$)		permutes the dimensions of a tensor
b = `tri2vec`($a <, opt >$)		moves (sub/super) triangle from matrix to vector
$tens$ = `vec2ten`(vec,$dims$)		move vector to tensor
$< c, i1, i2 >$ = `unique`(a)		unique entries of of `a`
b = `vec2tri`($a <, nc <, opt >>$)		moves entries of a vector into lower, upper, or symmetric matrix

5.2.4 Input and Output

In- and Output Functions	
Function	**Short Description**
a = csvread("filepath" <, optn >)	read comma separated (CSV) values files
error(mess)	print error message into the log
export(a,"fname","ftyp" <, optn >)	export (write) data set
"ftyp" =	"mat", "sas", "spss", "splus", "stata", "mtb"
irc = fclose(< fid >)	close one or all open files
irc = feof(fid)	test for end-of-file mark
irc = ferror(fid <, iclear >)	test for file read/write error
a = fil2obj("filepath")	reads object from file into memory
v = filestat("filepath")	returns a vector of file statistics
fnam = fnampid("filepath" <, "ext" >)	returns concatenation of filename with PID
fid = fopen("filepath","permission")	open file for read/write
nbyt = fprintf(fid,"format" <, a, ... >)	format data for string output into file
a = fread(fid,dtyp <, nr <, nc >>)	binary input from file into data
irc = fremove("filepath")	remove some file
irc = frename("oldfil","newfil")	rename file "oldfil" to "newfil"
a = fscanf(fid,"format" <, nr <, nc >>)	format string read from file into data
irc = fseek(fid <, offset <, type >>)	move to byte location in open file
nbyt = ftell(fid)	tell current file location
nbyt = fwrite(fid,a <, dtyp >)	binary output of data to file
a = generead("filepath", resp, sep <, optn <, scod >>)	input of comma and other separated file
	input of special micro array data sets
< a, b, ... > = import("fname","ftyp")	import (read) data set(s)
"ftyp" =	"mat", "sas", "spss", "splus", "stata", "mtb"
libname(rnk,"dirpath")	specifies or despecifies library
logfile("filepath")	redirects .log output to file
lstmem(< i >)	prints table of memory allocations
lststk(< i >)	prints table of execution stack
lstvar(< i >)	prints stack of symbol tables
obj2fil(a,"filepath")	writes object from memory to file
printf(format,arg1, arg2, ...)	prints formatted scalars and objects
s = pritfile("filepath" <, optn >)	transform text file into vector of strings
rewind(< fid >)	rewind one or all open files
a = rspfile("filepath" < "sep" <, optn >>)	read sparse data set into object
sound(ifreq <, octav <, durat >>)	beep with frequ and duration
sprintf(format,arg1, arg2, ...)	formats data into string
a = sscanf(str, format <, nr <, nc >>)	reads scalars and objects
txtfile("filepath")	redirects .txt output to file
warning(mess)	print warning message into the log
nr = wspfile(a,"filepath" < "sep" <, optn >>)	writes sparse data set from object
irc = zip7("command")	run the *7zip* program in
irc = zip7("what","archive" <, file <, option >>)	cmat_util directory

5.2.5 Elementary Math and String Processing

Elementary Math Functions	
Function	**Short Description**
y=abs(z)	absolute value or complex modulus $\|z\|$
y=acos(z)	inverse trigonometric cosine
y=acosh(z)	inverse hyperbolic cosine
y=asin(z)	inverse trigonometric sine
y=asinh(z)	inverse hyperbolic sine
y=atan(z)	inverse trigonometric tangent
y=atan2(x,y)	trigonometric tangent
y=atanh(z)	inverse hyperbolic tangent
y=ceil(z)	rounds to an integer upward to $+\infty$
y=cos(z)	trigonometric cosine
y=cosh(z)	hyperbolic cosine function
y=exp(z)	exponential function
y=fix(z)	rounds to an integer toward 0
y=floor(z)	rounds to an integer downward to $-\infty$
r=fmod(x,y)	remainder r of $x = ky + r$
y=ldexp(x,i)	returns xj^i for *radix* $j = 2$
y=log(z)	natural logarithm function
y=log2(z)	base-2 logarithm
y=log10(z)	base-10 logarithm function
y=max($a <, b >$)	maximum in a or between a and b
$< y, ind >$=max($a, b, ...$)	maximum value and index location
$< y, ind >$=maxn(a, n)	n largest values and index locations
y=min($a <, b >$)	minimum in a or between a and b
$< y, ind >$=min($a, b, ...$)	minimum value and index location
$< y, ind >$=minn(a, n)	n smallest values and index locations
y=pow(z,c)	power function z^c
i=rand()	pseudorandom generator
y=round($z <, ndig >$)	rounds toward the nearest integer
y=sign(z)	signum function ($y = 0$ for $x = 0; y = 1$ for $x > 0; y = -1$ for $x < 0$)
y=sign2(a,b)	signum function ($y = \|a\|$ for $b \geq 0; y = -\|a\|$ for $b < 0$)
y=sign4($a1, a2, a3, b$)	signum function ($y = a1$ for $b < 0; y = a2$ for $b = 0; y = a3$ for $b > 0$)
y=sin(z)	trigonometric sine function
y=sinh(z)	hyperbolic sine function
i=sizeof(z)	number of bytes used for the storage of z
y=sqrt(z)	square root $\sqrt{z}$
y=tan(z)	trigonometric tangent function
y=tanh(z)	hyperbolic tangent function

String Processing Functions	
Function	**Short Description**
int =`atoi`(s)	converts string `s` to integer
real=`atof`(s)	converts string `s` to float
b = `byte`(a)	transfers integers (0,..,255) into ASCI chars
b = `noblanks`($a <, str >$)	removes leading and trailing blanks from string data
str = `strcat`($s1, s2 <, n >$)	concatenate `s2` to the end of `s1`
str = `strchr`(s,c)	search `s` for first occurence of `c`
int = `strcmp`($s1, s2 <, n >$)	compare `s1` with `s2` lexicographically
str = `strcpy`($s1, s2 <, n >$)	overwriting copy of `s2` to `s1`
int = `strcspn`(s, set)	search `s` for first occurence of char included in `set`
int = `strlen`(s)	return length of string `s`
str = `strlwr`(s)	convert string to lower case
int = `strpos`(s,c)	search `s` for first occurence of `c`
str = `strrchr`(s,c)	search `s` for last occurence of `c`
str = `strrev`(s)	reverse string
int = `strrpos`(s,c)	search `s` for last occurence of `c`
int = `strspn`(s, set)	search `s` for first occurence of char **not** included in `set`
str = `strupr`(s)	convert string to upper case

Note, that all `str...()` functions have been extended to vector, matrix, and tensor arguments.

5.2.6 Advanced Math

Advanced Math Functions		
Function	**Spec**	**Short Description**
v = besseli($x <, alpha <, "esc" >>$)		Bessel I function
v = besselj($x <, alpha <, "esc" >>$)		Bessel J function
v = bessely($x <, alpha <, "esc" >>$)		Bessel Y function
v = besselk($x <, alpha <, "esc" >>$)		Bessel K function
y = dawson(x)		Dawson integral
v = ellipi($kind, x <, y >$)		complete elliptic integral
v = ellinc($kind, x, y, z <, \rho >$)		incomplete elliptic integral
y = expint($x <, "esc" \vert "one" >$)		exponential integral
y = factrl($a <, b <, c >>$)		factorial
v = ferdirc(ord, x)		Ferri-Dirac integral
v = ferdiri(ord, x, b)		incomplete Ferri-Dirac integral
y = fft($x <, optn >$)		fast Fourier transform
y = ffti($x <, optn >$)		inverse fast Fourier transform
$< g, j, c, h >$ = fider($f, x <, sopt <, par <, grad >>>$)	sopt =	derivatives by finite differences
	"grad"	returns gradient
	"jaco"	returns Jacobian
	"crpj"	returns cross product Jacobian
	"hess"	returns Hessian
	"forw"	uses forward difference formula
	"cent"	uses central difference formula
	"expo"	uses Richardson extrapolation
	"fsum"	applies on sum of functions
	"flsq"	applies on sum of squares of functions
$xmin$ = fmin($func, xest$	sopt=	minimization of univariate function
$<, range <, sopt <, par <, grad >>>>$)	"bre"	method by Brent
z = fresnel(x)		Fresnel integral
$zero$ = fzero($func, xest$	sopt=	zero of univariate function
$<, range <, sopt <, par <, grad >>>>$)	"bre"	method by Brent
	"b+d"	method by Bus and Dekker
	"mul"	method by Muller
y = gcd(a, b)		greatest common divisor
$< gof, circ >$ = hamilton($indx, vert <, optn >$)		find (all or some) Hamilton circuits
$< gof, circ >$ = hamilton($adja <, optn >$)		
y = horner($coef, xval <, opt >$)		evaluate the Horner scheme
$< f1, d1, d0 >$ = intpol($x0, f0, x1 <, sopt >$)	sopt=	univariate interpolation
	"lin"	linear
	"cub"	cubic
	"spl"	spline

Advanced Math Functions (Contd.)		
Function	**Spec**	**Short Description**
$< gof, xind >$ `= knapsack(`$prof, wgt, cap <, optn >$`)`		one- and multi-dimensional Knapsack problem
`res = latlong(`$task, pnt1 <, pnt2 <, optn >>$`)`		functions of Latitude and Longitude data
$< gof, post >$ `=` `= locat1(`$vwgt, fwgt <, optn >$`)`		multifacility location problem with rectilinear distance
$< gof, xind, yind, ofun, lm >$ `=` `= locatn(`$cmat <, par <, dvec >>$`)`		assigning K optimal locations among $n > K$ potential locations for servicing m clients.

$< x, lm, rp, duals, sens >$ `=`
`= lp(`$"meth", c, lau <, lbub <, optn <, xint >>>$`)`
`= lp(`$"meth", "path" <, optn <, xint >>>$`)`
	meth
	
	"lps"
	"pcx"
	"con"
	"clp"
		minimize or maximize linear function
		
		using `lpsolve()` (Berkelaar et al., 2004)
		using interior point PCx algorithm (Czyzyk et al, 1997)
		using LPASL continuation method (Madsen & Pinar, 1993)
		using Clp code by J. J. Forrest

$< x, rp, duals, sens >$ `= lpassign(`$cost <, optn >$`)`		linear assignment problem
$< x, rp, duals, sens >$ `=` `= lptransp(`$cost, rrhs, crhs <, optn >$`)`		linear transportation problem

$< c, lau, lubc >$ `=`
`= mpsread(`$fpath <, optn >$`)`
$< prob, row, col, rhs, rng, bnd >$ `=`
`= mpswrite(`$strvec, c <, lau <, lubc >>$`)`
		reading MPS file for LP
		
		writing MPS file for LP

$< ilnk, dlnk >$ `= mstdis(`$x <, "meth" <, optn >>$`)`	"meth" "rohlf" "whitn"	minimum (maximum) spanning tree based on distances method by Rohlf (1978) method by Whitney
$< gof, pred >$ `= mstgra(`$indx, vert, cost <, optn >$`)` $< gof, pred >$ `= mstgra(`$adja, cost <, optn >$`)`		find minimum spanning tree based on graph data

Advanced Math Functions (Contd.)		
Function	**Spec**	**Short Description**
$< xr, rp, der1 >$ `= nle(`$func, x0$	tech=	solve system of nonlinear equations
$<, sopt <, par, < jac >>>$`)`		for options settings see reference manual
$< xr, rp, der1, der2, acon, dpro, jhnlc >$	tech=	minimize or maximize nonlinear function
`= nlp(`$func, x0 <, optn <, lbub <, lau$	”QADPEN”	Quadratic Penalty Algorithm
$<, nlcon <, grad <, hess < jcon >>>>>>>$`)`	”DQNSQP”	Quasi-Newton SQP Method
	”VMCWD”	Powell’s original VMCWD method
	”TRUREG”	Trust-Region Method
	”NEWRAP”	Line Search Newton-Raphson Method
	”NRRIDG”	Ridge Newton-Raphson Method
	”DBLDOG”	Double-Dogleg Methods (DBFGS,DDFP)
	”QUANEW”	Quasi-Newton Methods (DBFGS,DDFP,BFGS,DFP)
	”POWBLC”	Powell’s BFGS for Linear Constraints
	”LMQUAN”	Limited Memory Quasi-Newton Methods
	”CONGRA”	Conjugate Gradient Methods
		with versions: PB, FR, PR, CD, BM, SFR, SPR
	”NONDIF”	Nondifferential Subgradient Methods (BT)
		with different versions
	”NMSIMP”	Nelder-Mead Simplex Method
	”COBYLA”	Constrained Optimization by Linear Approximation
	”LINCOA”	Linear constrained Optimization Algorithm
	”BOBYQA”	Bound constrained Optimization by
		Quadratic Approximation
	”UOBYQA”	Unconstrained Optimization by
		Quadratic Approximation (default version=1)
		version=0: ”NEWUOA” modified ”UOBYQA”
	”SIMANN”	Simulated Annealing (Global Opt.)
	”GENALG”	Genetic Algorithm (only maximization)
	”LEVMAR”	Levenberg-Marquardt Method
	”HYQUAN”	Hybrid Quasi-Newton Methods (DBFGS, DDFP)
	”NL1REG”	Nonlinear L_1 Regression
	”NLIREG”	Nonlinear L_∞ Regression
	”MINMAX”	Nonlinear MinMax Optimization
	”NONE”	do not perform any optimization
$< y, t >$ `= ode(`$func, tab, y0$	optn=	ordinary differential equations
$<, par <, root <, afun <, jacm >>>$`)`	par[4]=1	Adams method (nonstiff)
	par[4]=2	Gear’s method (stiff)
	par[4]=3	switch dynamically Gear’s or Adams
	par[5]=0	$\frac{dy_i}{dt} = f_i(y,t), \quad i = 1, \ldots, n$
	par[5]=1	$\mathbf{A}\frac{dy}{dt} = f(y,t)$
	par[5]=2	$\mathbf{a_i}\frac{dy_i}{dt} = f_i(y,t), \; i = 1, \ldots, m$
		$0 = f_i(y,t), \quad i = m+1, \ldots, n$

Advanced Math Functions (Contd.)		
Function	**Spec**	**Short Description**
$< xr, lm, rp >$ = pcx($"mpsfil" <, keyopt >$)		PCx algorithm for linear programming
$< coef, sse >$ = polyfit(x,y,$d <, opt >$)		fit polynomials of degree d
z = polynom($a <, sopt <, x >>$)	sopt "zero" "eval" "deri" "coef"	operations on polynomial find zeros evaluate polynomial evaluate first derivative return coefficients for given zeros
y = polyval($coef$,$x <, opt >$)		evaluate polynomials at x
$< b, e >$ = primfact(x)		prime factors of x
$< xr, rp >$ = qp($mat <, vec$ $<, lau, <, x0 <, optn <, lbub >>>>>$)	tech= "QPNUSP" "QPRASP" "QPPOGI" "QPTRON" "QPMANP" "QPBARR"	minimize or maximize quadratic function Null Space (Active Set) Method Range Space (Active Set) Method Goldfarb and Idnani Algorithm by Powell Lin-Moré Trust-Region (TRON) Method (only BC) Madsen-Nielsen-Pinar Algorithm (only BC) Interior Point (Barrier) Algorithm (only BC)
$< H, c, y >$ = qptst($n <, par >$)		create Moré-Toraldo QP test problem
$area$ = quad($func$,$ab <, optn >$)		quadrature of function for options settings see reference manual
$< area, d0 >$ = quadip($x0, f0, ab <, sopt >$)	sopt "lin" "cub" "spl"	quadrature of interpolation linear cubic spline
$< area, aerr, nfun >$ = quadirgw($func$,$ndim <, optn >$)		quadrature with Gaussian weights over infinite region adaptive and stochastic method
$< area, aerr, nfun >$ = quadsimp($func$,$vertx <, optn >$)		multivariate vector quadrature over simplex region
$< area, aerr, nfun >$ = = quadvec($func$,$ab <, optn >$)		multivariate vector quadrature over rectangular region adaptive and stochastic method
$< gof, path, len >$ = = splnet($indx$,$vert, cost, brack <, optn >$) $< gof, path, len >$ = splnet($adja, cost, brack <, optn >$)		find shortest path length between two nodes of a network
r = ssq(v)		sum of squares of entries
$< length, tour, dist >$ = = tsp($meth, norm, data <, optn <, intour >>$)		traveling salesman problem

Matrix Functions		
Function	**Spec**	**Short Description**
$< val, vec, res, scl >=$	sopt	ARPACK Functions: Matrix spec.
$= \texttt{arpack}(a, sopt, nv, ncv <, optn <, b >>)$	"sev","gsev"	EVD: real symmetric
	"nev","gnev"	EVD: real unsymmetric
	"zev","gzev"	EVD: complex unsymmetric
$< val, lvec, rvec, res, scl >=$	sopt	ARPACK Functions: Matrix spec.
$= \texttt{arpack}(a, sopt, nv, ncv <, optn >)$	"svd"	SVD: selected values (vectors)
$< val, vec, res, scl > \texttt{=}$	sopt	ARPACK Functions
$= \texttt{arpack}(op\_fun, sopt, nv, ncv <, optn >)$		operator function specification
$< val, lvec, rvec, res, scl > \texttt{=}$	sopt	ARPACK Functions
$= \texttt{arpack}(op\_fun, sopt, nv, ncv <, optn >)$	"svd"	SVD operator function specification
$< l, p, r, rcon > \texttt{=} \texttt{chold}(a, sopt)$	sopt	Cholesky decomposition
	"piv"	perform pivoting
	"eng"	use Ng-Peyton-Liu sparse minimum degree ordering
	"add"	use Amestoy-Davis-Duff sparse minimum degree ordering
	"gmw"	Gill-Murray-Wright modified Cholesky D.
	"esc"	Eskow-Schnabel modified Cholesky D.
	"mor"	Morè modified Cholesky Decomposition
$< q, r, v, pi, lind > \texttt{=} \texttt{cod}(a <, b <, sopt >>)$		complete orthogonal decomposition
$< x, lind > \texttt{=} \texttt{codapp}(a, b <, sopt >)$		minimum length solution of rank deficient LSQ
$r \texttt{=} \texttt{cond}(a <, sopt >)$	sopt	condition of matrix
	"svd"	use singular value decomposition
	"est"	use iterative estimation
$b \texttt{=} \texttt{cumprd}(a <, sopt >)$	sopt	cumulative product
	"lr"	left-right
	"ud"	upper-down
$b \texttt{=} \texttt{cumsum}(a <, sopt >)$	sopt	cumulative sum
	"lr"	left-right
	"ud"	upper-down
$r \texttt{=} \texttt{det}(a)$		determinant of matrix
$d \texttt{=} \texttt{design}(v <, opt >)$		(stat.) design matrix
$d \texttt{=} \texttt{diag}(a <, k >)$		creates (sub/super) diagonal matrix
$v \texttt{=} \texttt{dia2vec}(a <, k >)$		moves (sub/super) diagonal from matrix to vector
$b \texttt{=} \texttt{echelon}(a)$		row echelon normal form
$< eval, rvec, lvec > \texttt{=} \texttt{eig}(a$	sopt1	eigenvalues and eigenvectors
$<, sopt1 <, sopt2 <, vl <, vu >>>>)$	"impql"	implicit QL method
	"bisec"	bisection method
	"ratql"	rational QL method
	"jacob"	Jacobian method
	"dicon"	divide-and-conquer
	sopt2	
	"nobal"	do not scale nor permute
	"nosca"	do not scale
	"noper"	do not permute

Matrix Functions (Contd.)		
Function	**Spec**	**Short Description**
$< gof, Aica, Wica, Scor, Eval, Evec >$ `=` `= fica(`$data, optn <, Eini$ $<, Vini <, Agues >>>$`)`		(fast) independent component analysis
b `= flip(`$a <, sopt >$`)`	sopt "lr" "ud"	flip entries in matrix left-right upside-down
$< eval, rvec, lvec >$ `= geig(`$a, b <, type >$`)`	type= 1 2 3	generalized eigenvalue problem $\mathbf{AZ} = \mathbf{\Lambda BZ}$ $\mathbf{ABZ} = \mathbf{\Lambda Z}$ $\mathbf{BAZ} = \mathbf{\Lambda Z}$
$< x, y >$ `= glm(`a, b, d`)`		$\min_x \parallel y \parallel_2$ subject to $\mathbf{A}x + \mathbf{B}y = d$
$< eval, form_a, form_b, rvec, lvec >$ `=` `= gschur(`a, b`)`		generalized Schur decomposition
$< sig_a, sig_b, v, r, u_a, u_b >$ `= gsvd(`a, b`)`		generalized singular value decomposition
c `= hankel(`$a <, p >$`)`		create Hankel matrix
c `= hdprod(`a, b`)`		horizontal direct product
u `= hhold(`$v <, ind >$`)`		compute Householder transform
b `= hnform(`a`)`		Hermite normal form
b `= inv(`a`)`		inverse of matrix
b `= invupd(`$a, v \vert\vert s$`)` c `= invupd(`a, v, s`)`		update of inverse matrix update of inverse matrix
$< x, norm >$ `= ldp(`$a, b <, optn >$`)`	sopt=	linear distance programming
$< x, nrnk, (sval \vert rcon) >$ `=` `lls(`$a, b <, sopt <, optn >>$`)`	sopt= "svd" "cod" "qrd" "lqd" "evd" "gel" "itr"	$\min_x \parallel \mathbf{A}x_j - b_j \parallel_2, \; j = 1, \ldots, p$ use singular value decomposition use complete orthogonal decompostion use QR decomposition use LQ decomposition use eigenvalue decomposition (symmetric $\mathbf{A}$) use Gaussian elimination with normal equations use iterative method specified by `optn` e.g. LSQR by Paige and Saunders; CGNR, CGNE
x `= lsolv(`$a, b <, sopt <, optn >>$`)`	sopt= "svd" "cod" "qrd" "lqd" "evd" "gel" "itr"	$\min_x \parallel \mathbf{A}x_j - b_j \parallel_2, \; j = 1, \ldots, p$ use singular value decomposition use complete orthogonal decompostion use QR decomposition use LQ decomposition use eigenvalue decomposition (symmetric $\mathbf{A}$) use Gaussian elimination (sparse or dense) use iterative method specified by `optn` SYMMLQ, CG, MINRES; LSQR; CGsqu, BiCG, BICGstab, CGNR, QMR, GMRES
x `= lse(`a, b, c, d`)`		$\min_x \parallel \mathbf{A}x - c \parallel_2$ subject to $\mathbf{B}x = d$
$< l, u, pi >$ `= lud(`a`)`		LU decomposition

Matrix Functions (Contd.)		
Function	**Spec**	**Short Description**
$< dist, ldet, rmu, rcov > =$ $= \texttt{mahalanobis}(x <, mu <, cov <, optn >>>)$		Mahalanobis distance (whole matrix or only diagonal)
$i = \texttt{ncol}(a)$		number of columns of matrix
$< w, h > = \texttt{nnmf}(a, k < optn <, uini <, wini >>>)$ $< w, h, d > = \texttt{nnmf}(a, k < optn <, uini <, wini >>>)$		nonnegative matrix factorization
$r = \texttt{norm}(a <, p >)$		vector or matrix norm
$i = \texttt{nrow}(a)$		number of rows of matrix
$z = \texttt{nullsp}(x <, sopt <, tol >>)$	sopt= "svd" "qrd"	null space of matrix use singular value decomposition use QR decomposition
$coeff = \texttt{ortpol}(a <, maxdeg < wgt >>)$		coefficients of orthogonal polynomials
$< u, r, v > = \texttt{ortvec}(a <, q >)$		compute vector orthogonal to columns
$< b, nrnk, sval \mid rcon > = \texttt{pinv}(a <, sopt >)$	sopt= "svd" "cod" "evd"	 use singular value decomposition use complete orthogonal decomposition use eigenvalue decomposition
$n = \texttt{profile}(a)$ $< b, perm > = \texttt{profred}(a)$		computes profile of matrix reduces profile of matrix
$< q, r, pi > = \texttt{qrd}(a <, ind >)$		QR decomposition
$z = \texttt{rangesp}(x <, sopt <, tol >>)$	sopt= "svd" "qrd"	range space of matrix use singular value decomposition use QR decomposition
$z = \texttt{rank}(a <, sopt <, tol >>)$	sopt=	rank of matrix
	"svd" "qrd"	use (dense) singular value decomposition use (dense or sparse) QR decomposition
$r = \texttt{rcond}(a <, sopt >)$	sopt= "svd" "est"	reciprocal condition of matrix use singular value decomposition use iterative estimation
$< form_t, vec_z, eval > = \texttt{schur}(a)$		Schur decomposition
$< gof, D, X, Y, Rho > =$ $\texttt{sdd}(data, nfac <, optn <, wgt >>)$		semi discrete decomposition
$< sval, v, u > = \texttt{svd}(a <, p \mid "eco" <, optn >>)$	sopt= "eco"	singular value decomposition economic version (small **U** or **V**)
$< s, v, u > =$ $\texttt{svdtrip}(a, "meth" <, optn >)$ $< s, v, u > =$ $\texttt{svdtrip}(funa, "meth" <, optn <, funata >>)$	"meth" "bls","las",... "meth" "bls","las",...	compute (s, v, u) triplets of largest singular singular values and vectors compute (s, v, u) triplets of largest singular singular values and vectors
$< s2, v2, u2 > = \texttt{svdupd}(a, b <, s0, v0, u0 >)$		rank r update of SVD
$b = \texttt{sweep}(a <, p >)$		sweep matrix
$z = \texttt{sylve}(a, b, c <, s >)$		solve Sylvester equation
$< y, perm > = \texttt{tarjan}(x)$		permute row/col Tarjan's algorithm

Matrix Functions (Contd.)		
Function	**Spec**	**Short Description**
$tens \| scal \| vec$ = `tentvec(`$tens, vec <, n >$`)`		multiply tensor with vector
`tens = tentmat(`$tens, mat <, n >$`)`		multiply tensor with matrix
$tens \| scal$ = `tentten(`$aten, bten <, n >$`)`		multiply tensor with tensor
c = `toeplitz(`a`)`		create Toeplitz matrix
b = `toeplitz(`a,"dur"`)`		solve Yule-Walker equations using Durbin algorithm
b = `toeplitz(`a,"lev", b`)`		linear system with Toeplitz matrix Levinson algorithm
b = `toeplitz(`a,"tre"`)`		invert Toeplitz matrix using Trench algorithm
r = `trace(`a`)`		trace of matrix
$< b, c, s >$ = `tridod(`a, v`)`		Rank-1 downdate of Cholesky factor
$< b, c, s >$ = `triupd(`a, v`)`		Rank-1 update of Cholesky factor

5.2.7 Statistics

Statistical Functions		
Function	**Spec**	**Short Description**
$< powr, gvec, gmat >$ = `acss`($stim, dgrp$ $<, optn <, nisac >>>$)		animal carcinogenicity power and sample size Peto test, power computed by Weibull MC
$< scor, vars, rprm, cprm, sprm >$ = `anacor`($a, optn$)		correspondence analysis of contingency table
$< scor, vars >$ = `anaprof`($a, optn$)		correspondence analysis of profile data
$< auc, tab, xopt, cov >$ = = `auroc`($data <, optn <, popt <, x0 > . >$)		area under the receiver operating curve with as. standard error
$< gof, coef, covm, pred >$ = = `bidimreg`($type, y, x, <, optn >$)		bidimensional regression
$< c, ase, conf, acov >$ = `bivar`($a <, sopt1 <, sopt2 <, ipar >>>$)	sopt1= "scp" "cov" "corr" "spea" sopt2= "par" "inv" "pinv"	bivariate functions SSCP (scalar product) matrix covariance matrix Pearson correlation matrix Spearman correlation matrix partial correlation inverse of covariance matrix pseudoinverse of covariance matrix
$< gof, bstv, stat, hist, his2 >$ = `boruta`($data, modl <, optn <, class <, cwgt >$)		Boruta (feature selection) algorithm (based on Random Forest modeling)
$< gof, eval, xload, yload, xscor, yscor, xcqua, ycqua >$ = `canals`($xydata, scale, xind, yind, optn$)		canonical correlation (nonmetric Gifi version)
$< gof, sqcc, cstd, craw, swth, sbtw >$ = `cancor`($data, optn, xind, yind$)		 canonical correlation
$< u, v, d >$ = `centroid`($a, k < optn >$)		centroid factorization
$< gof, est, resi, cov, mod1, mod2, boci >$ = = `cfa`($data, optn, <, patt <, scal <, wgt$ $<, xini <, targ <, wtrg > . >$)		confirmatory factor analysis
$< memb, crit, mu, sigma, pi, asemu, asesig, asepi >$ = = `clmix`($x, optn <, par3 <, par4 <, par5 >>>$)		Fitting mixtures of normal and t components

Statistical Functions (Contd.)		
Function	**Spec**	**Short Description**
$< est, serr, conf, pval >$ = `conting(`$a, sopt <, par >$`)`	sopt=	association of contingency tables
	"corr"	Pearson correlation
	"spea"	Spearman rank correlation
	"polchor"	polychoric correlations
	"gamma"	Goodman-Kruskal gamma
	"taub"	Kendall's tau_b
	"tauc"	Kendall's tau_c
	"somcr"	Somer CR
	"somrc"	Somer RC
	"lambcr"	lambda CR
	"lambrc"	lambda RC
	"lambsym"	symmetric lambda
	"unccr"	uncertainty CR
	"uncrc"	uncertainty RC
	"uncsym"	symmetric Uncertainty
	"exrisk"	exact risk
	"oddrat"	odds ratio
$< cont, ntot >$ = = `contsim(`$rsum <, csum <, ntab >>$`)`		simulate contingency table with specified row and column sums
$< cov, dlta, fcov, scov, smu > =$ = `covshr(`$xdat, sopt, <, par <, rind >>$`)`	sopt=	shrink covariance matrix
	"corr"	average correlation method
	"mark"	market return method
	"diag"	diagonalmethod
	"twop"	two parameter method
$< est, ecov, ocov >$ = `delta(`$xstar,$ $fest, fopt <, opt <, egrd >>$`)`		Delta method
$< gof, estim, yxprd >$ = `demreg(`$data <, opt >$`)`		(univar.) Deming regression
d = `dist(`$x <, sopt <, optn <, scal >>>$`)`	sopt=	distance matrix
	"L2"	Euclidean (L_2) Distances
	"L1"	City-Block (L_1) Distances
	"Li"	Maximum (L_∞) Distances
	"Gower"	Gower Dissimilarities
$< cov, mu, b >$ = `emcov(`$a <, par >$`)`		estimate covariance matrix and mean vector when data have missing values

Statistical Functions (Contd.)		
Function	**Spec**	**Short Description**
$< y, t, c >$ `=`	sopt	rotate factors to simple structure
`= frotate(`$x <, sopt <, par <, targ <, weig >>>>$`)`	"crafer"	Crawford-Ferguson family
	"varmax"	Varimax rotation
	"quamax"	Quartimax rotation
	"equmax"	Equamax rotation
	"parmax"	Parsimax rotation
	"facpar"	Factor Parsimony rotation
	"bentlr"	Bentler rotation criterion
	"minent"	Minimum Entropy rotation
	"tandm1"	orthogonal Tandem 1 rotation
	"tandm2"	orthogonal Tandem 2 rotation
	"infmax"	orthogonal Infomax rotation
	"mccamm"	orthogonal McCammon rotation
	"oblmin"	Direct Oblimin family
	"quamin"	Direct Quartimin
	"biqmin"	Bi-Quartimin ritation
	"covmin"	Covarimin rotation
	"simmax"	Simplimax oblique rotation
	"oblmax"	Oblimax rotation
	"geomin"	Geomin rotation
	"promax"	Promax oblique rotation
	"targt1"	Target rotation
	"targt2"	(partially specified) Target rotation
$< gov, est, resi, cov >$		factor analysis (exploratory)
`= factor(`$data, optn <, wgt <, init <, prior >>>$`)`		
$< beta, shrnk, yptrn, yptst >$		Garotte (Breiman, 1993)
`= garotte(`$trn, model, optn <, class <, test >>$`))`		nonnegative linear regression

Statistical Functions (Contd.)		
Function	**Spec**	**Short Description**
$< gof, parm, ncov, rcov, yhat >$ = = $\texttt{gee}(data, model <, optn, <, class <, rstr$ $<, xini >>>)$		generalized estimation of equations
$< gof, parm, ase, conf, cov, typ1, typ3, yhat, roc >$ = $\texttt{glim}(data, model <, optn, <, class$ $<, xini <, contr >>>>)$		generalized linear models
$< gof, parm, ase, conf, cov, typ1, typ3, yhat, theta, roc >$ = $\texttt{glmixd}(data, model <, optn <, class <, random$ $<, xini <, contr >>>>)$		mixed generalized linear models (random effects; ordinal and nominal response)
$< gof, parm, ase, conf, cov, typ1, typ3, resi >$ = $\texttt{glmod}(data, model <, optn, <, class <, cont >>>)$		general linear model
gof = $\texttt{hbaddtst}(data, < optn >)$ $< gof, hyp, ci >$ = $\texttt{hbanova}(data <, freq <, optn >>)$ gof = $\texttt{hbbartlett}(data <, freq <, optn >>)$ gof = $\texttt{hbcovar}(ydata, wdata <, freq <, optn >>)$ $< gof, fsep, yhat, zhat >$ = = $\texttt{hbdiscrim}(data, idc <, optn <, zdat >>)$ $< gof, parm, yhat, zhat, test >$ = = $\texttt{hblreg}(xdat, ydat <, optn <, zdat <, cont >>>)$ gof = $\texttt{hbltst}(xdat, ydat, freq <, optn >)$ $< gof, parm, yhat >$ = = $\texttt{hblqrg}(xdat, ydat, mpow, freq, <, optn >)$ $< gof, parm >$ = $\texttt{hbrcmp}(x1, y1, x2, y2 <, optn >)$ gof = $\texttt{hbscheffe}(data, cont <, freq <, optn >>)$ $< ttest, mom, ftest >$ = $\texttt{hbttest}(data1, data2 <, optn >)$		Test for additivity by Tukey (Nollau, 1975) Nine ANOVA models I and II (Nollau, 1975) Bartlett test (Nollau, 1975) two models for covariance analysis with one covariable discriminance analysis (Nollau, 1975) linear least squares regression (Nollau, 1975) linearity test by R. A. Fisher (Nollau, 1975) Simple polynomial regression (Nollau, 1975) Compare coefficients of two linear regressions Scheffè (1959) test (Nollau, 1975). uni- and bivariate t test and Welch's test
$hist$ = $\texttt{histogrm}(a, k, <, optn >)$		obtain k histogram
$< gof, eval, discr, obscor, cquant, stquant >$ = $\texttt{homals}(a, optn)$		homogeneity analysis of multimonial data
$< gof, parm >$ = $\texttt{hotell}(x, mu0 <, optn >)$		standard/robust one-sample Hotellings Test
`yhat` = $\texttt{isoreg}(yxw <, optn >)$		(weighted) isotone regression
$< gof, est, cov >$ = $\texttt{irtml}(data,$ $optn <, tini <, bini <, bc >>>)$		Maximum Likelihood IRT (various algorithms)
$< gof, scal, loev, pairs > =$ = $\texttt{irtms}(data, optn)$		(nonparametric) Mokken scale IRT by Hardouin (2007)

Statistical Functions (Contd.)		
Function	**Spec**	**Short Description**
$< jack, boot >$ =	task=	jackknife and bootstrap
= jboot($data$,"$task$", $usfun <, optn >$)	"all"	perform all methods
	"jack"	Jackknife
	"norm"	normal
	"perc"	percentile
	"hybr"	hybrid
	"bc"	bias corrected
	"bca"	bias corrected accelerated
	"stud"	Studentized, bootstrap-t
$< gof, dens, mesh >$ =		1- and 2-dimensional Gaussian
= kde($data, optn$)		kernel density estimation
$< gof, beta, yptrn, yptst >$		LARS and related methods
= lars($trn, model, optn <, class <, test >>$)	type=	L_1 and L_2 constrained linear regression
	1: lars	least angle reagression (Tibshirani et al., 1996)
	2: lasso	Lasso (default) (Tibshirani et al., 1996)
	3: stage	forward stagewise (Tibshirani et al., 1996)
	4: rdige	Ridge Regression;
	5: lars-en	Elastic Net (Zou & Hastie, 2003);
	6: ust	Univariate Soft Thresholding (Donoho et al., 1995)
$< coef, null >$ = lda($sym <, par >$)		analysis of linear dependencies
$< gof, best, parm, yptr, yptt >$ = lrallv($msel, trn,$		all variable subset linear regression
$model, optn <, class <, aov16 <, grpv <, tst >>>>$)		
$< gof, parm, yptr, yptt >$ = lrforw($trn,$		stepwise forward linear regression
$model, optn <, class <, aov16 <, grpv <, tst >>>>$)		variable selection for large applications
v = mad($a <, optn >$)		median absolute deviation (MAD)
$parm$ = mardia($a <, sopt <, optn >>$)	sopt=	mv skewness and kurtosis with p values
	"mvk"	multivariate kurtosis
	"mvs"	multivariate skewness
	"all"	multivariate kurtosis and kurtosis
$< gof, conf, resi, wgt, ... >$ =	meth	multidimensional scaling
= mds($meth, data, optn <, xini, ... >$)	"unis"	Method for unidimensional scaling by de Leeuw()
		exact (all combinations) method for small problems
	"torg"	Torgerson metric MDS method (related to `anacor()`)
		(used as start solution for iterative algorithms)
	"alscal"	Takane, Young, & de Leeuw (1977) metric MDS method
		see also de Leeuw (2012)
	"kyst"	Kruskal-Young-Shepard-Torgerson method (1978)
	"smacof"	metric and nonmetric MDS method by de Leeuw (1984, 1994, 20
		Mair, and Groenen (early work also by Heiser)
	"multsc"	maximum likelihood method by Ramsay (1977, 1978)
	"maxlik"	maximum likelihood method similar to MULTISCALE
		by Ramsay (1977, 1978)
	"indscal"	3-way method by Carroll and Chang (1970)
	"sumscal"	3-way method by de Leeuw and Pruzansky (1978)
	"cospa"	3-way method by Schoenemann (1972)
$< gof, xconf, yconf, resi, ... >$ =	meth	multidimensional unfolding
= mdu($meth, data, optn <, xini <, yini, ... >$)	"torg"	Torgerson metric method (related to `svd()`)
		(used as start solution for iterative algorithms)
	"alscal"	Takane, Young, & de Leeuw (1977) metric method
		see also de Leeuw (2012)
	"smacof"	metric and nonmetric MDU method by de Leeuw, Mair, and Gr
		(early work also by Heiser)
v = median($a <, optn >$)		median

Statistical Functions (Contd.)		
Function	**Spec**	**Short Description**
$< gof, parm, cov, resi > \texttt{ =}$		mixed effects
$\texttt{= mixregv(}data, modl, optn, covar <, rand <, errvar >>\texttt{)}$		location-scale regression
$< x, r, c > \texttt{ = mpolish(}a <, optn >\texttt{)}$		mean and median polish
$< gof, lik, est, fpbar, BF > \texttt{ = mucomp(...)}$		test group and order restrictions on means
$< gof, BF > \texttt{ = mucomp(}data, nres, modl <, optn >\texttt{)}$		(confirmatory ANOVA)
$< gof, stat, parm > \texttt{ =}$		multiple comparison of $K > 2$ means
$\texttt{= multcomp(}imet, g, x <, optn <, con <, nams >>>\texttt{)}$		
$< prob, stat > \texttt{ = mvntest(}sopt, x <, optn >\texttt{)}$	sopt=	testing for multivariate normality
	"mark"	Mardia's test for multivariate kurtosis
	"mard"	Mardia's test for multivariate skewness
	"maro"	Mardia & Foster (1983) omnibus test
	"wmin"	Wang and Hwang (2011) test
	"roys"	Royston (1983) W test
	"hezi"	Henze-Zirkler (1990) test
	"q123"	Small's $Q1$, $Q2$, $Q3$
	"doha"	Doornik-Hansen Omnibus test
	"szek"	Szekely-Rizzo (2005) test
	"mudh"	Mudholkar-McDermott-Srivastava
$< gof, parm, ase, conf, cov > \texttt{ = nlreg(}func, x0$	optn	nonlinear L_p norm regression
$<, optn <, nlpopt <, lbub <, lau <, nlcon$		with Wald, PL, and Jackknife CI's
$<, grad <, hess <, jcon > .. >)$	L_2	least squares
	L_1	L_1 or LAV regression
	L_∞	Chebychev regression (no CI's)
	L_p	general L_P with $1 \leq p \leq 10$
$model, optn <, class <, tun <, kfun <, test >>\texttt{)}$		
$< gof, est, resi, cov >$		binary factor analysis
$\texttt{= noharm(}data, optn <, guess <, init >>\texttt{)}$		
$< parm, delt, ase, conf, cov > \texttt{ = odr(}func, x0, xdat$		nonlinear orthogonal distance regression
$<, ydat <, optn <, fjac <, d0$		(incl. OLS regr. w. mult. response)
$<, we <, wd <, bfix <, xfix > ... >\texttt{)}$		
$< oind, oval, crit > \texttt{ = outlier(}a, sopt <, optn >>\texttt{)}$	sopt=	univariate outlier detection
	"chisq"	χ^2 method
	"sig_3"	3-Sigma rule
	"tukey"	Tukey method
	"chauv"	Chauvenet method
	"grubb"	Grubbs (1969) test
	"thomp"	Thompson τ test
	"dixon"	Dixon Q test
$< mkur, madi > \texttt{ = outlmd(}a <, sopt1 <, opt1 >>\texttt{)}$	sopt1=	multidimensional outlier detection
	"mvk"	multivariate kurtosis
$< loc, scat, dist > \texttt{ = outlmd(}a <, sopt2 <, opt2 >>\texttt{)}$	sopt2=	multidimensional outlier detection
	"mve"	minimum volume ellipsoid
	"mcd"	minimum covariance determinant
$< gof, eval, cquan, load, scor > \texttt{ =}$		K set canonical correlation analysis
$\texttt{= overals(}data, vtyp, sets, optn\texttt{)}$		(Gifi version)
$< psym, pind > \texttt{ = partial(}asym, ipar <, par >\texttt{)}$		partial covariances or correlations

Statistical Functions (Contd.)		
Function	**Spec**	**Short Description**
$< gof, eval, evec, comp >$ = = `pca`($data, optn <, targ <, wgt >>$)		principal component analysis (PCA) (various column and row oriented algorithms)
$< rms, parm, yprd, vrms, trms, tprd >$ = = `pls`($xtrn, yind, xind <, icmp <, optn <, xtst >>>$)		partial least squares (PLS) and principal components regression (PCR)
$< corr, tau, ccov, tcov, ctco, gof >$ = = `polychor`($data <, ind <, optn >>$)		polychoric correlations with thresholds and asymptotic covariances
$< gof, eval, obscor, cquant, loada, scora, loadb, scorb >$ = `primals`($a, optn$)		onedimensional homogeneity analysis of multimonial data
$< gof, eval, load, scor, catq, single, multi > =$ = `princals`($data, scale, optn$)		principal component analysis of categorical data (Gifi, 1990)
$< d, rot, c >$ = `procrust`($a, b <, par <, nrow >>$)		orthogonal Procrustes problem
$< plan, planb >$ = `promep`($levs <, optn >$)		MinPROMEP: partially replicated minimal orthogonal main-effect plans
$quant$ = `quantile`($a, k \mid prob <, optn >$)		obtain k quantile nine different types like in R function
$< gof, misc, pred, gini, prox, outl, intact >$ = = `rafprd`($infor, data, modl <, optn <, class <, cwgt > . >$)		Random Forest prediction (scoring) (based on the model obtained by `ranfor()`)
$< gof, misc, pred, gini, prox, outl, inact, tmis, tprd, tgini >$ = = `ranfor`($data, modl <, optn <, class <, cwgt <, test > . >$)		Random Forest modeling (Breiman) Regression and Classification
$< gof, beta, ase, conf, cov, res >$ = `reg`($data, model$ $<, sopt <, optn <, class >>>$)	sopt= "l_2" "l_1" "l_inf" "l_p" "odis" "lms" "lts" "hub"	linear regression analysis least squares regression L_1 or LAV regression L_∞ or Chebychev regression L_p regression $p \geq 1$ orthogonal distance regression least median squares regression least trimmed median regression some Huber regression methods
$< gof, area, diff, cova, covd >$ = = `roccomp`($yvec, xmat <, contr <, optn >>$)		comparison of areas under ROC curve
$nsamp$ = `sampallo`($data, vubc <, optn <, cost >>$)		optimal sample allocation in strata
$samp$ = `sample`($nsmp, nobs \mid prob <, optn >$)		equal or unequal probability sampling with or without replacement
$< ind, smp >$ = `sampmd(data<,optn>)`		maximum distance sampling from data set

Statistical Functions (Contd.)		
Function	**Spec**	**Short Description**
$< gof, parm, gcvtab, cov, restrn, restst >$ =		Smoothly Clipped Absolute Deviations
= scad($meth, xtrn, lamb, frac, model, optn$	"meth"	method for variable selection (Fan and Li, 2001)
$<, class <, xtst >>$)	"lse"	linear least squares regression
	"phr"	proportional hazards regression
	"svm"	support vector machines regression
$vest$ = scalpha($data <, optn >$)		sample coefficient alpha (Cronbach, 1951)
$< nval, infl, lrgof, malin >$ = screetst($eval, optn$)		scree test for eigen and singular values
$< gof, klc, eklc, spm, spr, spc, wspm, wspr, wspc >$ =		DOE search probability
= sdcspm($D <, rho <, optn >>$)		measures of Design matrices
$< nxtgen, nxtforb >$ = selc($inides, forbid, nlev, optn$)		sequential elimination of level combinations
$< gof, est, resi, toteff, indeff >$ = sem($data, model, optn$		(mean and) covariance structure analysis
$<, parms <, wmat <, lc <, repar <, nlc, nlcb >>>>>$)		and (linear) structural equations
$< gof, parm, shape, ase, cov >$ =		robust MANOVA based on spatial signs
= sgmanova($data, modl <, class <, optn >>$)		
$< parm, stder, conf, cov >$ = simex($data, model, vardis$		simulation extrapolation
$<, optn <, class >>$)		
$< gof, parm, atst, ptst, res >$ = sir($a, model$	sopt=	sliced inverse regression
$<,$ "$sopt$" $<, optn <, slic <, class >>>>$)	"sir"	sliced inverse regression
	"save"	sliced average variance estimation
	"rphd"	principal Hessian direction (residuals)
	"yphd"	principal Hessian direction (response)
	"qphd"	quadratic principal Hessian direction
$< z, locscal >$ = stand($x <, sopt <, optn <, ls >>>$)		(columnwise) standardization
	sopt=	(location,scale)
	mea(n)	(mean,1)
	med(ian)	(median,1)
	sum	(0,sum)
	euc(len)	(0,Euclidean length)
	ust(d)	(0,Uncorrected Std. Deviation)
	std	(mean,Standard Deviation)
	ran(ge)	(minimum,range)
	mid(range)	(midrange,range/2)
	max(abs)	(0,max absolute value)
	iqr	(median,interquartile range)
	mad	(median,abs. deviation from median)
	biw(c)	iweight 1-step M-estim,biweight A-estimate)
	hub(er)(c)	(Huber 1-step M-estimate,Huber A estimate)
	wav(e)(c)	(Wave 1-step M-estimate,Wave A estimate)
	agk(p)	(mean,AGK (ACECLUS) estimate)
	spa(cing)(p)	(mid-minimum-spacing,minimum spacing)
	lpm(p)	(*Lp* location,*Lp* scale (FASTCLUS))
	inp	(input,input)
	rev	(reverse use of input,reverse use of input)

Statistical Functions (Contd.)		
Function	**Spec**	**Short Description**
$< ttest, mom, ftest > = \texttt{ttest}(data1, data2 <, optn >)$		uni- and bivariate t test and Welch's test
$< pval, cint, kwal > = \texttt{wilcox}(xdata, ydata <, optn >)$		Wilcoxon rank sum test (Mann-Whitney test) and Wilcoxon signed rank test
$< c, ase, conf > = \texttt{univar}(a <, sopt < optn >>)$	sopt=	univariate function
	”min”	minimum value
	”max”	maximum value
	”rng”	range
	”ari”	arithemetic mean
	”med”	median
	”ust”	uncorrected standard deviation
	”var”	variance
	”std”	standard deviation
	”mad”	median absolute deviation (MAD)
	”sma”	scaled median absolute deviation
	”s_n”	robust scale S_n (Rousseeuw)
	”q_n”	robust scale Q_n (Rousseeuw)
	”ske”	skewness
	”kur”	kurtosis
	”qu1”	first quartile
	”qu3”	third quartile
	”iqr”	interquartile range
	”qua”	all three quartiles
	”loo”	L_∞ (maxabs) norm
	”lpm”	L_p norm, $p \geq 1$
	”fsp”	fourth spread (Hoaglin, 1983)
	”biw”	Tukey's biweight location
	”bis”	Tukey's biweight scal
	”hub”	Huber's location (Goodall, 1983) for $k > 0$
	”hus”	Huber's scale (Iglewicz, 1983) for $k > 0$
	”wav”	Andrew's wave location (Goodall, 1983) for $c > 0$
	”was”	Andrew's wave scale (Iglewicz, 1983) for $c > 0$
	”msp”	Minimum spacing location (Sarle, 1995) for $0 < p < 1$.
	”mss”	Minimum spacing scale (Sarle, 1995) for $0 < p < 1$.
$< effrep, outrep, cordat > = \texttt{urd1out}(X, y <, optn >)$		unreplicated factorial designs with single outlier
$< gof, est, conf > =$ $= \texttt{xctlog}(data, model, exct <, optn < class >>)$		exact logistic regression by MCMC method

5.2.8 Data Mining

Data Mining Functions		
Function	**Spec**	**Short Description**
$< gof, asso, nset > = \texttt{assoc}(cust, data, <, optn <, supp >>)$		associations of items
$< weights, add1, add2 > = \texttt{cluster}(x, sopt$	sopt=	cluster methods
$<, optn <, scal >>)$	"Agnes"	Agglomerative (Hierarchical) Nesting
	"Clara"	Clustering Large Applications
	"Diana"	Divisive (Hierarchical) Analysis
	"Fanny"	Fuzzy Analysis
	"Mona"	Monothetic Analysis
	"Pam"	Partitioning Around Medoids
$< gof, ccc, sv > = \texttt{cuclcr}(data, cstr$		cubic cluster criterion
$<, optn <, sv <, rsq >>>)$		
$< xful, scal > = \texttt{impute}(xmis, sopt < optn$	sopt=	imputation of missing values
$<, class <, bounds >>>)$	"scalar"	impute scalar constant
	"randuni"	uniform random in column range
	"randnrm"	columnwise (ν, σ) normal random
	"colmean"	constant column mean
	"mindist"	rowwise minimum distance
	"knearn"	rowwise K nearest neighbor
	"linreg"	columnwise linear regression
	"simpls"	columnwise linear PLS (SIMPLS)
	"krnpls"	columnwise linear PLS (Kernel PLS)
$< gof, bmat, ecor, caval, cqord, cstat > =$		classic item tree analysis
$= \texttt{ita}(data, optn)$		
$< gof, bmat, eqor, diff, cqord, cstat > =$		induct. item tree analysis
$= \texttt{ita}(data, optn)$		
$< gof, alpha, beta, wgts, resi > =$		multivariate SVM
$= \texttt{mvsvm}(xtrn, ytrn <, optn <, kfun >>)$		
$< gof, parm, fit, tabs, stat, scor, tscor > =$		nonlinear data mining
$= \texttt{nlfit}(train, modl <, optn <, class <, fun1 <, fun2$		choosing from a set of nonlinear functions
$<, actf <, link <, test > .. >)$		
$< gof, scor, fit, tabs > =$		scoring for `nlfit()`
$= \texttt{nlfitprd}(data, parm, stat, modl <, optn <, class$		with new data set
$<, actf <, link > .. >)$		
$< gof, pc, eval > =$		(nonlinear) Kernel PCA
$= \texttt{nlkpca}(x, optn <, class <, kfun >>)$		
$< alfa, sres, vres, yptr, yptt > = \texttt{nlkpls}(trn,$		(nonlinear) Kernel PLS
$< gof, tree, cltrn, prtrn, cltst, prtst > =$		recursive partitioning (chaid)
$= \texttt{recupar}(trn, modl, optn, ord <, nom <, flt <, test > . >)$		(similar to SAS `treedisc` macro)
$< gof, rules > = \texttt{rules}(asso <, optn >)$		rules in associations of items
$< gof, osequ > = \texttt{sequ}(cust, visit, data, asso <, optn <, supp >>)$		sequences of items
$< gof, beta, yprd, errt > =$		stochastic matching pursuit and
$= \texttt{smp}(data, tau, model <, optn, < class >>)$		componentwise Gibbs sampler

Data Mining Functions		
Function	**Spec**	**Short Description**
$< gof, beta, resi > =$		relevance vector machines
$= \texttt{rvm}(xtrn, ytrn <, optn <, kfun >>)$		(Tipping,2001, Herbrich, 2002)
$< gof, theta, cmat, bvec, yprd, ftrn, yptt, ftst > =$		SM support vector machines:
$= \texttt{smsvm}(task, train, lamb, model, optn,$	"task"	multicategory classification
$class <, test, <, t0 <, x0 <, kfun >>>>)$	"msvm"	multicategory SVM by Lee and Wahba
	"ssvm"	structured multicategory SVM by Y. Lee
$< gof, scal, xwgt, ywgt, nass,$		Self Organizing Maps
$tmtrn, cwtrn, ptrn, mtrn, tmtst, cwtst, ptst, mtst > \ =$		Kohonen Maps
$= \texttt{som}(xtrn, ytrn <, optn <, neus <, epos <, xtst <, ytst > . >)$		CP-ANN, SKN, XYF
$< gof, nodes, levmap, yptr, yptt > =$		binary tree regression
$= \texttt{split}(trn, model, optn <, class <, tst >>)$		with binary response
$< alfa, sres, vres, yptr, yptt, plan, tunerr, tunzer > =$		support vector machines:
$= \texttt{svm}(train, model <, optn <, class <, x0 <, kfun <, test > . >)$	"imet"	classification and regression
	"FQP"	full QP method
	"DQP"	decomposed QP method (shrinking)
	"LSVM"	Lagrangian SVM
	"ASVM"	Active SVM
	"PSVM"	Proximal SVM
	"SSVM"	Smooth (and Reduced) SVM
	"SMO"	Sequential Minimal Optimization
$< alfa, sres, vres, yptr, yptt, plan, tunerr, tunzer > =$		SVM feature selection
$= \texttt{svmfsm}(train, model <, optn <, class <, x0 <, kfun <, test > . >)$		for SVM classification and regression
$sym = \texttt{svmmat}(data, model <, optn <, class >>)$	kind	computes SVM kernel matrix
	"line"	linear function
	"poly"	polynomial function
	"rbf"	Gaussian radial basis function ($[0,1]$)
	"rbf2"	mod. Gaussian radial basis function ($[0,1]$)
	"rbfcs"	mod. Gaussian radial basis function ($[0,1]$)
	"erbf"	exponential radial basis function
	"tanh"	sigmoid function (same as "sigm")
	"sigm"	sigmoid function (same as "tanh")
	"four"	Fourier function ($[-\pi/2, \pi/2]$)
	"spli"	spline function ($[0,1]$)
	"anov"	anova function
	"curv"	curvspline function ($[0,1]$)
	"bspl"	Bspline function ($[0,1]$)
	"anob"	anova spline function ($[0,1]$)

Data Mining Functions		
Function	**Spec**	**Short Description**
$< yptr, vec >$ = `svmpred`($test, alfa, train, model$ $<, optn <, class <, kfun >>>$)		computes SVM predicted values (score test data set)
$< alfa, sres, vres, yptr, yptt, plan >$ = `svmstw`($train,$ $model <, optn <, class <, test >>>$)		SVM stepwise feature selection for SVM classification and regression
$< gof, est, tree, scor, struct, trace >$ = = `varclus`($data, optn <, ingrp >$)		clustering variables
$< gof, mod >$ = `varsel`($data, optn, yind, xind$)		multiple variable selection
$< gof, mod >$ = `varsel`($data, optn, model <, class >$)		multiple variable selection

5.2.9 Survival Methods for the Analysis of Censored Data

Survival Functions		
Function	**Spec**	**Short Description**
$< gof, curv, dase >$ = `survcurv`($sopt$,		(Adjusted) Survival Curves
$data, modl, optn <, clas >$)	sopt	
	"adjaal"	adjusted Aalen's additive model
	"adjcox"	adjusted Cox's proportional hazards model
	"aalen"	Aaalen: based on Cox PH estimates
	"tsiatis"	Tsiatis: based on Cox PH estimates
	"breslow"	Breslow: based on Cox PH estimates
	"kapme"	Kaplan-Meier: based on Cox PH estimates
	"kalpr"	Kalbfleisch-Prentice: based on Cox PH estimates
	"green"	Greenwood: based on Cox PH estimates
	"exact"	Exact: based on Cox PH estimates
	"fleha"	Fleming-Harrington: based on Cox PH estimates
	"efron"	Efron: based on Cox PH estimates
	"kapme2"	Kaplan-Meier: not based on any model
	"fleha2"	Fleming-Harrington: not based on any model
	"fh2"	FH2: not based on any model
$< gof, res >$ = `survprd`($parm, covm, sopt$,		Survival Regression Prediction
$data, modl, optn <, clas >$)	sopt	scoring of test data
$< gof, parm, cov, res, tres >$ = `survreg`($sopt$,		Survival Regression
$data, modl, optn <, clas <, test >>$)	sopt	
	"aalen"	Aalen's additive model
	"phcox"	Cox's proportional hazards model
	"extrem"	regression with extreme distribution
	"logist"	regression with logistic distribution
	"gauss"	regression with Gaussian distribution
	"weibul"	regression with Weibull distribution
	"loglog"	regression with loglogistic distribution
	"lognor"	regression with lognormal distribution
	"expon"	regression with exponential distribution
		(this is Weibull with fixed scale=1)
	"rayle"	Rayleigh distribution
		(this is Weibull with fixed scale of .5)

5.2.10 Analysis of Micro Array Data

Micro Array Functions		
Function	**Spec**	**Short Description**
a = `generead`($fpath$,$resp$, sep <, $optn$ <, $scod$ >>)		input of comma and other separated file input of special micro array data sets
< gof, $parms$, $dnew$, mu > = = `affvsn`($data$,$optn$ <, ref >)		Variance Stabilizing Normalization
< gof, $dnew$ > = `affrma`($data$,$optn$ <, ref >)		RMA method for normalizing microarray data

5.2.11 Time Series

Time Series Functions		
Function	**Spec**	**Short Description**
$< gof, coef, resi, forc >$ = = `arima`($data$,$pord$ $<, optn >$)		AutoRegressive Integrated Moving Average (ARIMA) algorithm
$< gof, est, root, ase, cis, cov, sco >$ = = `arma`(y $<, x <, ar <, ma <, optn <, p0 > . >$)		autoregressive moving-average method
$< gof, yhat >$ = = `armafore`(np,nh, est, y $<, x <, ar <, ma <, optn > . >$)		forecasting step of the autoregressive moving-average method
$< coef, err >$ = `armcov`($data$, ord $<, optn >$)		modified covariance method
$< gof, est, root, ase, nwcov, cov, resi >$ = = `arhet`(y, $lagbnd$ $<, optn >$)		heterogeneous autoregressive method
$< prob, stat >$ = = `berkow`(y $<, optn <,$ "$dist$" $<, ... >>$)		Berkowitz testing for distributions like `kstest()`
$< coef, err >$ = `burg`($data$, ord $<, optn >$)		Burg's method for moving average coefficients
$< ccov, cmu >$ = `cndcov`($data$, zmu, $isel$, $optn$)		conditional covariance matrix
b = `covlag`(a, k)		sequence of lagged cross product matrices
$< gof, est, ase, cis, cov, sco >$ = = `garch`($meth$, y, x, o, q, p $<, optn < x0 >>$)		ARCH, GARCH, TARCH, AVARCH, ZARCH, APARCH, EGARCH, AGARCH, NAGARCH, IGARCH, FIGARCH
$< gof, cov, phi >$ = `mburg`(x, $lags$ $<, optn <, cov >>$		modified Burg algorithm
xyp = `mempsd`($coef$,$resi$ $<, optn >$)		power spectrum of autoregressive filter
xyp = `pwelch`($data$ $<, optn >$)		power spectrum of a time series by Welch
`forec` = `tslocfor`($xdat$,$order$ $<, optn >$)		forecasting using local model
$< relerr, inderr >$ = = `tsloctst`($xdat$,$order$ $<, optn >$)		error estimation for local model
acv = `tsmeas`($xdat$,"$alcdis$", b $<, optn >$)	"alcdis"	algorithmic complexity: partitions of equal distance
acv = `tsmeas`($xdat$,"$alcpro$", b $<, optn >$)	"alcpro"	algorithmic complexity: partitions with same probability
$< mse, nmse, nrmse, cc >$ = = `tsmeas`($xdat$,"$arfit$", m $<, optn >$)	"arfit"	statistical errors of fit at lead times
$< mse, nmse, nrmse, cc >$ = = `tsmeas`($xdat$,"$arprd$", m $<, optn >$)	"arprd"	statistical errors of prediction at lead times
$< bic, cumbic >$ = = `tsmeas`($xdat$,"$bicorr$", tau $<, optn >$)	"bicorr"	bicorrelation
$cordim$ = `tsmeas`($xdat$,"$cordim$", tau, m, s $<, optn >$)	"cordim"	correlation dimension
$< cdim, csum, cent >$ = = `tsmeas`($xdat$,"$cordi2$" $<, optn >$)	"cordi2"	correlation sum, entropy, and dimension
$corent$ = `tsmeas`($xdat$,"$corent$", tau, m, r $<, optn >$)	"corent"	approximate correlation entropy
$corrad$ = `tsmeas`($xdat$,"$corrad$", tau, m, r $<, optn >$)	"corrad"	radii of given correlation sums
$corsum$ = `tsmeas`($xdat$,"$corsum$", tau, m, r $<, optn >$)	"corsum"	correlation sum
$detfl$ = `tsmeas`($xdat$,"$detfl$" $<, optn >$)	"detfl"	detrended fluctuation analysis
$< stat, pval, cval, lags, resi >$ = `tsmeas`($xdat$,"$diful$" $<, optn >$)	"diful"	(augmented) Dickey-Fuller testing (with automatic lag selection)

Time Series Functions (Contd.)		
Function	**Spec**	**Short Description**
$< stat, pval > = \texttt{tsmeas}(xdat, "grang", tau <, optn >)$	"grang"	Granger causality testing for different types of inference
$< est, ase, r2s, cov, resi > = \texttt{tsmeas}(xdat, "arvec", tau <, optn >)$	"arvec"	vector AR modeling (homo- and heteroskedastic)
$< est, ase, r2s, cov, resi > = \texttt{tsmeas}(xdat, "iresp", tau <, optn >)$	"iresp"	impulse response modeling for specified value of lead
$< ebnd, psv, freq > = = \texttt{tsmeas}(xdat, "eband", band <, optn >)$	"eband"	energy in frequency band
$< ym, ymn, tima, tmimi, dmima > = = \texttt{tsmeas}(xdat, "exfea", f, nsmp <, optn >)$	"exfea"	local extreme values for a window
$< xmin, xmax > = \texttt{tsmeas}(xdat, "extrem" <, optn >)$	"extrem"	maxima and/or minima of a component
$falsnn = \texttt{tsmeas}(xdat, "falsnn", tau, m <, optn >)$	"falsnn"	percentage of false nearest neighbors
$fann = \texttt{tsmeas}(xdat, "falsn2" <, optn >)$	"falsn2"	fraction of false nearest neighbors
$< fsle, ulen, npnt > = = \texttt{tsmeas}(xdat, "fslexp" <, optn >)$	"fslexp"	finite size Lyapunov exponent
$< mob, comp > = \texttt{tsmeas}(xdat, "hjoer" <, optn >)$	"hjoer"	Hjoerth parameters mobility and complexity
$hurst = \texttt{tsmeas}(xdat, "hurst" <, optn >)$	"hurst"	Hurst exponent
$< cor, cum, dec, zer > = = \texttt{tsmeas}(xdat, "kenda", tau <, optn >)$	"kenda"	Kendall autocorrelation
$< mse, nmse, nrmse, cc > = \texttt{tsmeas}(xdat, "larfit", tau, m, nn, t <, optn >)$	"larfit"	in-sample direct predictions with a local model
$< mse, nmse, nrmse, cc > = = \texttt{tsmeas}(xdat, "larprd", tau, m, nn, t <, optn >)$	"larprd"	direct predictions with a local model
$< stat, prob > = = \texttt{tsmeas}(xdat, "ljung" <, optn >)$	"ljung"	Ljung-Box Test for serial correlation
$< stat, prob > = \texttt{tsmeas}(xdat, "lmtst" <, optn >)$	"lmtst"	LM Test for serial correlation
$lyap = \texttt{tsmeas}(xdat, "lyapk" <, optn >)$	"lyapk"	largest Lyapunov exponent (Kantz, 1994)
$lyap = \texttt{tsmeas}(xdat, "lyapr" <, optn >)$	"lyapr"	largest Lyapunov exponent (Rosenstein et al., 1993)
$< medf, psv, freq > = = \texttt{tsmeas}(xdat, "medfr" <, optn >)$	"medfr"	median frequency in range
$< mut, cummut, minmut > = = \texttt{tsmeas}(xdat, "mutdis", tau, b <, optn >)$	"mutdis"	minimum of mutual information
$< mut, cummut, minmut > = = \texttt{tsmeas}(xdat, "mutpro", tau, b <, optn >)$	"mutpro"	minimum of mutual information
$rmut = \texttt{tsmeas}(xdat, "mutual" <, optn >)$	"mutual"	time delayed mutual information
$nwcov = \texttt{tsmeas}(xdat, "nwcov" <, optn >)$	"nwcov"	Newey-West covariance matrix
$< gof, est, cov, yhat > = = \texttt{tsmeas}(ydat, "nwreg", xdat <, optn >)$	"nwreg"	Newey-West regression with HAC standard errors

Time Series Functions (Contd.)		
Function	**Spec**	**Short Description**
$< pacor, ase >$ = = tsmeas($xdat$,"$pacor$", $tau <, optn >$)	"pacor"	partial correlations with robust asymptotic standard errors
$< cor, cum, dec, zer >$ = = tsmeas($xdat$,"$pears$", $tau <, optn >$)	"pears"	Pearson autocorrelation
$renyi$ = tsmeas($xdat$,"$renent$" $<, optn >$)	"renent"	Renyi entropy of Qth order
$< cor, cum, dec, zer >$ = = tsmeas($xdat$,"$spear$", $tau <, optn >$)	"spear"	Spearman autocorrelation
$corr$ = tsmeas($xdat$,"$xcorr1$" $<, optn >$)	"xcorr1"	autocorrelation
$corr$ = tsmeas($xdat$,"$xcorr2$" $<, optn >$)	"xcorr2"	cross correlations among time series data
$ydat$ = tstrans($xdat$,"$sopt$" $<, optn <, add >>$)	"sopt"	transformations for time series data
	"aaft"	Amplitude Adjusted Fourier Transform Theiler et al (1992)
	"baki"	Baxter-King filtering
	"boxc"	Box-Cox transform depending on λ
	"four"	Fourier Transform surrogate
	"gaus"	marginal cumulative function to Gaussian
	"hopr"	Hodrick-Prescott filtering
	"iaaf"	Iterated Amplitude Adjusted Fourier Schreiber & Schmitz (1996)
	"line"	transform linearly to the interval $[0, 1]$
	"lagd"	lag difference transform depending on lag(s)
	"logd"	log difference transform depending on lag(s)
	"l121"	simple (iterated) 1-2-1 filter (see TISEAN)
	"norm"	transform to zero mean and unit standard deviation
	"notc"	Notch filter (see TISEAN)
	"perm"	random permuted data surrogate
	"sago"	Savitzky-Golay filter (see TISEAN)
	"stap"	statistically transformed AR process Kugiumtzis (2002b)
	"unif"	marginal cumulative function to Uniform in $[0, 1]$
$< gof, newdata >$ = = x11($data <, optn <, inidate >>$)		seasonal adjustment with X11 algorithm

5.2.12 Probability and Combinatorics

Probability Functions		
Function	**Spec**	**Short Description**
p = `adprob(n,d,"vers")`		Probability of Anderson Darling CDF
p = `adtest(x,"vers")`		Anderson-Darling test (uniform and normal)
$< v, der >$ = `airy`$(z <, "esc" >)$		Airy A_i function
$< v, der >$ = `airybi`$(z <, "esc" >)$		Airy B_i function
$< prob, stat >$ = `berkow`$(a, optn, "dist" < ... >)$	dist	Berkowitz test for univariate distr.
		(for distributions see `kstest()`
$< y, der1, der2 >$ = `beta`(p,q)		Beta function
$< y, der1, der2 >$ = `betaln`(p,q)		log Beta function
$< y, der1, der2 >$ = `betainc`(p,q,x)		incomplete Beta function
y = `betamis`(p,x,a,b)		Beta function (m.v.)
y = `binomis`$(p,s,prob,n)$		Binomial function (m.v.)
p = `binorm`(h,k,r)		Bivariate Normal Distribution
v = `cdf23`$("dist", lim, corr <, par >)$	"dist"	bi- and trivariate CDF functions
	"norm"	normal distribution
	"t"	t distribution
p = `cdf`$(dist, quant <, par_1, \ldots, par_k >)$	dist	cumulative density function
("bern",x, p)	"bern"	Bernoulli distribution
("beta",$x, a, b <, l <, u >>$)	"beta"	Beta distribution
("bin",$s, prob, n$)	"bin"	Binomial distribution
("cau",$x, <, \mu <, \sigma >>$)	"cau"	Cauchy distribution
("chis",$x, df <, nc >$)	"chis"	(noncentral) ChiSquare distribution
("expo",$x <, \sigma >$)	"expo"	Exponential distribution
("f",$x, ndf, ddf <, nc >$)	"f"	(noncentral) F distribution
("gam",$x, shape <, scale >$)	"gam"	Gamma distribution
("gaus",$x <, \mu <, \sigma >>$)	"gaus"	Gauss (Normal) distribution
("geom",m, p)	"geom"	Geometric distribution
("hypg",$x, m, k, n <, r >$)	"hypg"	Hypergeometric distribution
("igau",$x, shape$)	"igau"	Inverse Gauss (Wald) distribution
("lapl",$x <, \mu <, \sigma >>$)	"lapl"	Laplace distribution
("logi",$x <, \mu <, \sigma >>$)	"logi"	Logistic distribution
("logn",$x <, \mu <, \sigma >>$)	"logn"	LogNormal distribution
("negb",$x, prob, n$)	"negb"	Negative Binomial distribution
("norm",$x <, \mu <, \sigma >>$)	"norm"	Normal (Gauss) distribution
("pare",$x, a <, k >$)	"pare"	Pareto distribution
("pois",n, λ)	"pois"	Poisson distribution
("t",$t, df <, nc >$)	"t"	t distribution
("unif",$x <, l <, u >>$)	"unif"	Uniform distribution
("wald",x, d)	"wald"	Wald (Inverse Gauss) distribution
("weib",x,shape<,scale>)	"weib"	Weibull distribution
v = `cdfmv`$("dist", ilim <, mu <, cov <, par >>>)$	"dist"	multivariate CDF functions
	"norm"	normal distribution
	"t"	t distribution

Probability Functions (Contd.)		
Function	**Spec**	**Short Description**
$y = \texttt{chimis}(p,x,df <,nc>)$		ChiSquare function (m.v.)
$c = \texttt{combn}(x <,n>)$		generates all combinations of x taken n at a time.
$c = \texttt{combn2}(x)$		generates all combinations of x taken $n = 2$ at a time
$p = \texttt{diehd}(func <,optn>)$	test "gcd" "bda" "gor"	test of uniform random generators greatest common denominator test birthday spacings test gorilla test (extensive "monkey" test)
$r = \texttt{dixonr}(sopt,i,j,n,r\|R\|\alpha <,opt>)$		pdf, cdf, and critical values of Dixon's r
$d = \texttt{dmnom}(x,prob)$		computes the density of multinomial distributed points
$v = \texttt{erf}(x)$ $v = \texttt{erfcom}(x <,"esc">)$ $v = \texttt{erfinv}(x)$		error function complementary error function inverse error function
$< p,dens,err > = \texttt{fgen}(y,nu,wgt <,par>)$		cdf of generalized F distribution
$y = \texttt{fmis}(p,f,ndf,ddf <,nc>)$		F function (m.v.)
$y = \texttt{gammamis}(p,x,shape,scale)$		Gamma function (m.v.)
$< y,der1,der2 > = \texttt{gamma}(x)$ $< y,der1,der2 > = \texttt{gammaln}(x)$ $< y,der1,der2 > = \texttt{gammainc}(p,x)$		Gamma function log Gamma function incomplete Gamma function
$z = \texttt{hcube}(x <,tran <,scal >>)$		generates all points on a hypercube lattice.
$quant$ = $= \texttt{icdfmv}(dist,prob,mu <,sigma <,par >>)$		inverse CDF (percent point function) for multivariate normal and t distribution
$< prob,stat >$ = $= \texttt{jarbera}(y <,optn>)$		Jarque-Bera test for univariate normal dist. (skewness and kurtosis)
$p = \texttt{ksprob}(n,d <,sopt>)$	sopt "bar" "sle" "fap" "mtw"	probability of Kolmogorov CDF and the complementary problem compute complementary problem Simard & L'Ecuyer algorithm fast version of Simard & L'Ecuyer Marsaglia-Tsang-Wang algorithm

Probability Functions (Contd.)		
Function	**Spec**	**Short Description**
$< prob, stat >$ = `kstest(`a,$optn$,"$dist$" $< ... >$`)`	dist	Kolmogorov-Smirnov test for univariate distr.
	"bern"	Bernnoulli distribution
	"beta"	Beta distribution
	"bin"	Binomial distribution
	"cau"	Cauchy distribution
	"chis"	Chisquare distribution
	"expo"	Exponential distribution
	"f"	(noncentral) F distribution
	"gam"	Gamma distribution
	"gaus"	Gauss distribution
	"geom"	Geometric distribution
	"hypg"	Hypergeometric distribution
	"igau"	inverse Gauss distribution
	"lapl"	Laplace distribution
	"logi"	Logistic distribution
	"logn"	Lognormal distribution
	"negb"	negative Binomial distribution
	"nor"	normal distribution
	"par"	Pareto distribution
	"pois"	Poisson distribution
	"t"	(noncentral) t distribution
	"unif"	uniform distribution
$< conf, vol >$ = `mnprop(`$v <, alpha <, meth >>$`)`	meth	CIs for multinomial proportions
	"sison"	compute Sison-Glaz intervals
	"goodman"	compute Goodman intervals
`mrand(`$kind <, a, b >$`)`	kind=	creates multivariate random matrix
	"mnor"	multivariate normal $\mathcal{N}(\mu, \Sigma)$
	"mnom"	multinomial for scalar n and r vector p
	"unis"	uniformly distributed inside n dimensional sphere
	"unos"	uniformly distributed on n dimensional sphere
	"unie"	uniformly distributed inside n dimensional unit cube
	"unoe"	uniformly distributed on n dimensional unit cube
$< vec, contr >$ = `multst(`$options$`)`		multiple testing and simultaneous confidence intervals
vec = `mvelps(`$cov, mu, u, c, optn$`)`		multivariate normal and multivariate t probabilities over ellipsoidal regions noncentral F and χ^2 distribution
y = `nbinmis(`$p, s, prob, n$`)`		Negative Binomial function (m.v.)
n = `nsimplex(`p, n`)`		computes the number of points on a (p, n) simplex number p-part compositions of n
y = `normis(`p, x, μ, σ`)`		Normal function (m.v.)
T = `owenst(`h, a`)`		Owen's T function
$pout$ = `padjust(`$imet$, $pin <, par <, covm >>$`)`	imet=	multivariate p adjustment
	bon	Bonferroni
	hol	Holm
	hom	Hommel
	hoc	Hochberg
	bho	Benjamini & Hochberg
	bye	Benjamini & Yekutieli

Probability Functions (Contd.)		
dens = `pdf`($dist, quant < par_1, \ldots, par_k >$)	dist	probability density function
("bern",x, p)	"bern"	Bernoulli distribution
("beta",$x, a, b <, l <, u >>$)	"beta"	Beta distribution
("bin",s, p, n)	"bin"	Binomial distribution
("cau",$x <, \mu <, \sigma >>$)	"cau"	Cauchy distribution
("chis",$x, df <, nc >$)	"chis"	(noncentral) ChiSquare distribution
("expo",$x <, \sigma >$)	"expo"	Exponential distribution
("f",$f, ndf, ddf <, nc >$)	"f"	(noncentral) F distribution
("gam",$x, shape <, scale >$)	"gam"	Gamma distribution
("gaus",$x <, \mu <, \sigma >>$)	"gaus"	Gauss (Normal) distribution
("geom",m, p)	"geom"	Geometric distribution
("hypg",$x, m, k, n <, r >$)	"hypg"	Hypergeometric distribution
("igau",$x, shape$)	"igau"	Inverse Gauss (Wald) distribution
("lapl",x$<, \mu <, \sigma >>$)	"lapl"	Laplace distribution
("logi",x$<, \mu <, \sigma >>$)	"logi"	Logistic distribution
("logn",x$<, \mu <, \sigma >>$)	"logn"	LogNormal distribution
("negb",m, p, n)	"negb"	Negative Binomial distribution
("norm",$x <, \mu <, \sigma >>$)	"norm"	Normal (Gauss) distribution
("pare",$x, a <, k >$)	"pare"	Pareto distribution
("pois",n, λ)	"pois"	Poisson distribution
("t",$t, df <, nc >$)	"t"	t distribution
("unif",$x <, < l >, < u >>$)	"unif"	Uniform distribution
("wald",x, d)	"wald"	Wald (Inverse Gauss) distribution
("weib",$x, shape <, scale >$)	"weib"	Weibull distribution
v = `pdfmv`("*dist*",$quant, mu, cov <, par >$)	"dist"	multivariate density functions
	"norm"	normal distribution
	"t"	t distribution
dens =		probability density function for
= `pdfmv`($dist, q <, mu <, sigma <, optn >>>$)		multivariate normal and t
y = `poismis`(p, s, λ)		Poisson function (m.v.)
pts = `pppd`($type, mu, sig, skew, kurt <, ipri >$)		percentage points of Pearson distribs.
`rand`($< nr, nc, type, \ldots >$	type=	creates random matrix
$< dist \mid rank, \ldots >$)	'g'	rectangular `rand`($nr, nc, 'g', "dist", \ldots$)
	'd'	diagonal `rand`($n, n, 'd', "dist", \ldots$)
	'u'	upper triangular `rand`($n, n, 'u', "dist", \ldots$)
	'l'	lower triangular `rand`($n, n, 'l', "dist", \ldots$)
	's'	symmetric `rand`($n, n, 's', "dist", \ldots$)
	'r'	rectangular `rand`($nr, nc, 'r' <, rank >$)
	'e'	symmetric `rand`($n, n, 'e' <, rank <, elo, ehi >>$)
	'o'	column orthogonal `rand`($nr, nc, 'o'$)
	dist=	see table below

Probability Functions (Contd.)		
p = `randisc`($nsmp$,"bin", $n, p <, imet <, ipri >>$) p = `randisc`($nsmp$,"hyp", $n, m, N <, imet <, ipri >>$) p = `randisc`($nsmp$,"poi", $lambda <, imet <, ipri >>$)		binary discrete random variate hypergeometric discrete random variate Poisson discrete random variate
z = `rmult`(n,p)		multinomial random generator
$< prob, stat >$ = = `shapwilk`($y <, optn >$)		Shapiro-Wilks test for univariate normal and Shapiro-Francia test
$dist$ = `simdid`($a, optn$)		obtain similarities or distances of discrete distribs.
y = `tmis`($p, t, df <, pr >$)		T function (m.v.)
$< est, ci >$ = `xctbinom`($x, n, p <, optn >$)		exact Binomial test
$p1$ = `xctbip1`($powr, ssiz, p0 <, optn >$)		$p(alt)$ of exact Binomial test
pow = `xctbipow`($ssiz, p0, p1 <, optn >$)		power of exact Binomial test
$size$ = `xctbissz`($powr, p0, p1 <, optn >$)		sample size for exact Binomial test
pow = `xctfipow`($x, or <, optn >$)		power of exact Fisher test
$< est, ci >$ = `xctfishr`($x, or <, optn >$)		exact Fisher test
$size$ = `xctfissz`($x, or <, optn >$)		sample size for exact Fisher test
$pval$ = `xcthybr`($tab <, optn >$)		hybrid exact Fisher test
$< est, ci >$ = `xctmcnem`($x, or <, optn >$)		exact McNemar test
$< est, ci >$ = `xctpoiss`($x, T, r <, optn >$)		exact Poisson test
$pval$ = `xctsimu`($tab <, optn >$)		MC simu. of exact Fisher test
z = `xsimplex`(p, n)		generates all points of a (p, n) simplex number p-part compositions of n
$< r1, r2, r3, r4 >$ = `zoverw`($sopt, muz$, $sigz, muw, sigw, rho <, r <, opt >>$		probability and density of z/w for normal z and w

5.2.13 Random Generators

Uniform Random Generators		
Distr.	**Add. Arg.**	**Description**
"icmp"	a, b	uniform RNG, very bad 16 bit version in Watcom C Compiler, int version
"iuni"	a, b	uniform with lower range a and upper range b, int version
		Mooore, RAND Corporation, see Fishman, p. 605
"iacm"	a, b	uniform random generator by Schrage (1979) in ACM, int version
		this is not a good choice
"ikis"	a, b	uniform random generator KISS by Marsaglia & Tsang, int version
"iecu"	a, b	Tausworthe uniform random generator by L'Ecuyer (1996), int version
"iec2"	a, b	Tausworthe uniform random generator by L'Ecuyer (1996), int version
"imwc"	a, b	multiply-with-carry RNG (Marsaglia, 2003), period 2^{128}, int version
"ix128"	a, b	XOR RNG (Marsaglia, 2003), period 2^{128}, int version
"iwow"	a, b	modified XOR RNG (Marsaglia, 2003), period $2^{192} - 2^{32}$, int version
"imet"	a, b	Mersenne-Twister (Matsumoto & Nishimura, 1998), int version
"iase"	a, b	uniform AES RNG (Hellakalek & Wegenkittel, 2003), int version
"igfs"	a, b	uniform GFSR4 RNG (Ziff, 1998), int version
"ilux"	a, b	uniform RANLUX RNG (Lüscher, 1994), int version
"itsh"	a, b	twin source hexadecimal (Richarson,2011), int version
"its7"	a, b	twin source base 256 (Richarson,2011), int version
"itss"	a, b	single source base 256 (Richarson,2011), int version
"its4"	a, b	twin source base 256 (Richarson,2011), int version
"dcmp"	a, b	uniform with lower range a and upper range b
		very bad 16 bit version in Watcom C Compiler, real version
"duni"	a, b	uniform with lower range a and upper range b
		Mooore, RAND Corporation, see Fishman, p. 605, real version
"dacm"	a, b	uniform random generator by Schrage (1979) in ACM, real version
		this is not a good choice
"dkis"	a, b	uniform random generator KISS by Marsaglia & Tsang, real version
"decu"	a, b	Tausworthe uniform random generator by L'Ecuyer (1996), real version
"dmwc"	a, b	multiply-with-carry RNG (Marsaglia, 2003), period 2^{128}, real version
"dx128"	a, b	XOR RNG (Marsaglia, 2003), period 2^{128}, real version
"dwow"	a, b	modified XOR RNG (Marsaglia, 2003), period $2^{192} - 2^{32}$, real version
"imet"	a, b	Mersenne-Twister (Matsumoto & Nishimura, 1998), real version
"dase"	a, b	uniform AES RNG (Hellakalek & Wegenkittel, 2003), real version
"dgfs"	a, b	uniform GFSR4 RNG (Ziff, 1998), real version
"dlux"	a, b	uniform RANLUX RNG (Lüscher, 1994), real version
"dtsh"	a, b	twin source hexadecimal (Richarson,2011), real version
"dts7"	a, b	twin source base 256 (Richarson,2011), real version
"dtss"	a, b	single source base 256 (Richarson,2011), real version
"dts4"	a, b	twin source base 256 (Richarson,2011), real version

Random Generator Distributions		
Distr.	**Add. Arg.**	**Description**
"beta"	α, β	Beta, $\mathcal{BE}(\alpha,\beta)$, with $\alpha > 0$ and $\beta > 0$; Randlib version (Brown et al. 1997, [118])
"bet2"	α, β	Beta, $\mathcal{BE}(\alpha,\beta)$, with $\alpha > 0$ and $\beta > 0$ version by Fishman (1996, [260])
"bino"	n, p	Binomial, $\mathcal{B}(n,p)$, with $n = 1, 2, \ldots$ and $0 < p < 1$ Randlib version (Brown et al. 1997, [118])
"bin2"	n, p	Binomial, $\mathcal{B}(n,p)$, with $n = 1, 2, \ldots$ and $0 < p < 1$ version by Fishman (1996, [260])
"cau1"	α, β	noncentral Cauchy, $\mathcal{C}(\alpha,\beta)$, with $-\infty < \alpha < \infty$ and $\beta > 0$ version by Fishman p. 192 (1996, [260])
"cau2"	α, β	noncentral Cauchy, $\mathcal{C}(\alpha,\beta)$, with $-\infty < \alpha < \infty$ and $\beta > 0$ version by Fishman p. 187 (1996, [260])
"chis"	df	chi square with $df > 0$ Randlib version (Brown et al. 1997, [118])
"exex"	λ	Double Exponential, $\mathcal{DE}(\lambda)$, version by Fishman p. 192 (1996, [260])
"expo"	λ	Exponential, $\mathcal{E}(\lambda)$, $\lambda > 0$ Randlib version (Brown et al. 1997, [118])
"exp2"	λ	Exponential, $\mathcal{E}(\lambda)$, $\lambda > 0$ *ziggurat* method by Marsaglia & Tsang (2000, [572])
"exp3"	λ	Exponential, $\mathcal{E}(\lambda)$, $\lambda > 0$ version by Fishman p. 192 (1996, [260])
"exp4"	λ	Exponential, $\mathcal{E}(\lambda)$, $\lambda > 0$ version by Fishman p. 189 (1996, [260])

Random Generator Distributions (Contd.)		
Distr.	**Add. Arg.**	**Description**
"frch"	α	Fréchet, $\mathcal{FR}(\alpha)$, with $\alpha > 0$
		version by Zielinski(), p.106
"fsnd"	α, β	Snedecor's F, $\mathcal{F}(\alpha,\beta)$, with $\alpha > 0$ and $\beta > 0$
		Randlib version (Brown et al. 1997, [118])
"fsn2"	α, β	Snedecor's F, $\mathcal{F}(\alpha,\beta)$, with $\alpha > 0$ and $\beta > 0$
		version by Fishman p. 208 (1996, [260])
"gamm"	α, β	Gamma, $\mathcal{G}(\alpha,\beta)$, with $\alpha > 0$ and $\beta > 0$
		Randlib version (Brown et al. 1997, [118])
"gam2"	α, β	Gamma, $\mathcal{G}(\alpha,\beta)$, with $\alpha > 0$ and $\beta > 0$
		version by Fishman p. 193 (1996, [260])
"geom"	p	Geometric, $\mathcal{G}\rceil(p)$, with $0 < p < 1$
"hyge"	α, β, n	Hypergeometric, $\mathcal{H}(\alpha,\beta,n)$ with $\alpha > 0$, $\beta > 0$, and $1 \leq n \leq \alpha + \beta$
		version by Fishman p. 218-220 (1996, [260])
"logn"	μ, σ	Lognormal, $\mathcal{LN}(\mu,\sigma)$, with mean μ and standard deviation $\sigma > 0$
		version by Fishman (1996, [260])
"ncch"	df,$nonc$	noncentral chi square with $df > 0$
		Randlib version (Brown et al. 1997, [118])
"ncfs"	α, β,$nonc$	noncentral F, $\mathcal{F}(\alpha,\beta,nonc)$, with $\alpha > 0$ and $\beta > 0$
		Randlib version (Brown et al. 1997, [118])
"negb"	r, p	negative Binomial, $\mathcal{NB}(r,p)$, with $r > 0$ and $0 < p < 1$
		Randlib version (Brown et al. 1997, [118])
"neg2"	r, p	negative Binomial, $\mathcal{NB}(r,p)$, with $r > 0$ and $0 < p < 1$
		version by Fishman p. 222 (1996, [260])
"norm"	μ, σ	Normal, $\mathcal{N}(\mu,\sigma)$, with mean μ and standard deviation $\sigma > 0$
		Randlib version (Brown et al. 1997, [118])
"nor2"	μ, σ	Normal, $\mathcal{N}(\mu,\sigma)$, with mean μ and standard deviation $\sigma > 0$
		ziggurat method by Marsaglia & Tsang (2000, [572])
"nor3"	μ, σ	Normal, $\mathcal{N}(\mu,\sigma)$, with mean μ and standard deviation $\sigma > 0$
		version by Fishman p. 190 (1996, [260])
"nor4"	μ, σ	Normal, $\mathcal{N}(\mu,\sigma)$, with mean μ and standard deviation $\sigma > 0$
		version by Fishman p. 191 (1996, [260])
"pois"	λ	Poisson, $\mathcal{P}(\lambda)$, with $\lambda > 0$
		Randlib version (Brown et al. 1997, [118])
"poi2"	λ	Poisson, $\mathcal{P}(\lambda)$, with $\lambda > 0$
		version by Fishman p. 214 (1996, [260])
"poi3"	λ	Poisson, $\mathcal{P}(\lambda)$, with $\lambda > 0$

Random Generator Distributions (Contd.)		
Distr.	**Add. Arg.**	**Description**
"rayl"	σ	Rayleigh with $\sigma > 0$ (equivalent to Rice when $\mu = 0$) version by Zielinski(), p.106
"rice"	μ, σ	Rice with mean μ and $\sigma > 0$ version by Zielinski(), p.106
"tabl"	$[p_1, \ldots, p_n]$	tabled probability distribution with given table $0 \le p_1 \le p_2 \le \ldots \le p_n \le 1$
"stud"	n	Student's t, $\mathcal{S}(n)$, with $n = 1, 2, \ldots$ version by Fishman p. 207 (1996, [260])
"tria"	h	Triangular distribution $0 < h < 1$
"univ"	μ, σ	uniform with mean μ and standard deviation $\sigma > 0$
"weib"	α, λ	Weibull, $\mathcal{W}(\alpha, \lambda)$, with $\alpha > 0$ and $\lambda > 0$

The keywords `"iuni", "iacm", "ikis", "iecu"` generate corresponding uniform integer random numbers in $[0, MACLONG]$.

5.2.14 Plotting

Plotting Functionality		
Function	**Spec**	**Short Description**
$< vol, box >$ = `boundbox`(xy)		minimum bounding box
$< gof, vert, offs, cent, neigh, norm >$ = = `convhull`$(x <, optn <, thresh <, bounds <, feapnt > . >)$		compute convex hulls (dim=2,3,...) (see `qhull` software)
$< gof, vert, offs, cent, neigh >$ = = `delaunay`$(x <, optn <, thresh <, bounds > . >)$		Delaunay triangulation (dim=2,3,...) (see `qhull` software)
rc = `gpbatch`$(gpfils)$		invoking gnuplot with input file(s)
`gnuplot ... gpend`		executing gnuplot interactively
`histplot`$(y <, titl <, optn <, fpath >>>)$		printer plotting of histograms
$< gof, yhat, ytst >$ = `loess`$(ytrn, xtrn,$ $optn <, wgt <, para <, drsq <, xtst >>>>)$		multivariate robust locally weighted regression (Cleveland)
$< xyp, res >$ = `lowess`$(xy <, optn >)$		(robust) locally weighted regression
$< vol, box >$ = `maxempty`$(xy <, xybc >)$		maximum empty box
$< gof, ap, xpa >$ = `propurs`$(data, nsol <, optn <, wgt >>)$		projection pursuit (Friedman and Tukey)
$< gof, vert, offs, cent, neigh >$ = = `voronoi`$(x <, optn <, thresh <, bounds > . >)$		Voronoi diagrams (dim=2,3,...) (see `qhull` software)
`xplot`$(x <, titl <, optn <, labl <, fpath >>>>)$		univariate printer plotting
`xyplot`$(y, x <, titl <, optn <, labl <, fpath >>>>)$		printer plotting of (x, y) diagrams

5.2.15 Runtime Options

Runtime Options		
Option	**Spec**	**Short Description**
C_FIELDW=	*int*	field width of complex numbers with `print` statement
CENTER		center output of object with `print` statement
DEBUG=	*string*	only for debugging purposes
DECIMALS=	*int*	number of decimals real numbers with `print` statement.
ECHO		Echo input in .log output (default)
F_FIELDW=	*int*	field width of real numbers with `print` statement
I_FIELDW=	*int*	field width of integer numbers with `print` statement
INDBASE=	*int*	(also: IB) defines the lower index range (def. IB=1)
LINESIZE=	*int*	(also: LS) maximum number of chars online (def. LS=68)
NALLOC	*int*	set number of memory allocations for `lstmem` function
NAME		include name of the variable with `print` statement
NOCENTER		left flushed output with `print` statement (default)
NODEBUG		turning off the DEBUG option
NOECHO		suppress echo input in .log output
NONAME		opposite of NAME option (default)
NOPRINT		suppress all output to the .lst file
NOWSC		suppress work space compression (memory problem)
OPT_BS=	*int*	optimal block size used by LAPACK (def. OPT_BS=64)
PAGESIZE=	*int*	(also: PS) maximum number of lines printed on page (def. PS=60)
PRIME		return to prime version of subroutines (default)
PRINT		permits output to the .lst file after NOPRINT option (default)
RANDUNI		uniform RNG from RAND Corporation is used (SAS, IBM) (default)
RANDKISS		uniform RNG KISS by Marsaglia & Tsang (2002) (good)
RANDLECU		uniform RNG by L'Ecuyer (1999) is used (very good)
RANDLEC2		uniform RNG by L'Ecuyer (1999) is used (very good)
RANDMER		uniform Mersenne-Twister RNG (Matsumoto & Nishimura, 1998) (good)
RANDAES		uniform AES RNG (Hellakalek & Wegenkittel, 2003) (good)
RANDGFSR		uniform GFSR4 RNG (Ziff, 1998) (good)
RANDLUX		uniform RANLUX RNG (Lüscher, 1994) (good)
RANDXOR32		uniform RNG XOR32 is used, period $2^{32}-1$
RANDXOR64		uniform RNG XOR64 is used, 64bit, period $2^{64}-1$
RANDXORWOW		uniform RNG XORWOW is used, period $2^{192}-2^{32}$ (very good)
RANDXOR128		uniform RNG XOR128 is used, period $2^{128}-1$ (good)
RANDMWC3		uniform RNG MWC3 is used, period $2^{128}-1$ (good)
RANDCMP		uniform RNG of the host compiler is used (bad, period $2^{16}-1$)
RANDACM		uniform RNG by Schrage in ACM TOMS (1979), not good
RELZERO=	*real*	relative zero criterion
SECOND		run second version instead of prime version
SEED=	*int*	(re-) initialize the *seed* of `rand()`
SING=	*real*	criterion for singularity test (def. SING=1e-8)
SPRANGE=	*real*	sparsity range which (def. SPRANGE=.5)
SYMCRIT=	*real*	criterion for detection of symmetry (def. $\sqrt{meps}$)
USEUTF=	*int*	decide between work space or utility file (def. USEUTF=1000000)
WSC		(re-) permit (default) work space compression

Chapter 6

The Bibliography

Bibliography

[1] Abbasi, S. & Shaheen, F. (2008), "Faster generation of normal and exponential variates using the ziggurat method", Manuscript submitted to *JSS*.

[2] Abebe, A., Daniels, J., McKean, J.W., & Kapenga, J.A. (2001), *Statistics and Data Analysis*, `http://www.stat.wmich.edu/s160/book/`

[3] Abramowitz, M. & Stegun, I.A. (1972), *Handbook of Mathematical Functions*, Dover Publications, Inc., New York.

[4] Adlers, M. (1998), *Sparse Least Squares Problems with Box Constraints*, Linköping: Linköping University, Sweden.

[5] Agrawal, R., Imielski, T., & Swami, A (1993), "Mining association rules between sets of items in large databases"; Proceedings of the *ACM SIGMOD Conference on Management of Data*, p. 207-216.

[6] Agresti, A. (1996), *An Introduction to Categorical Data Analysis*, New York: John Wiley & Sons.

[7] Agresti, A. (2002), *Categorical Data Analysis*, Second Edition, New York: John Wiley & Sons.

[8] Ahn, H., Moon, H., Kim, S., & Kodell, R.L. (2002), "A Newton-based approach for attributing tumor lethality in animal carcinogenicity studies", *Computational Statistics and Data Analysis*, **38**, 263-283.

[9] Akcin, H. & Zhang, X. (2010), "A SAS Macro for direct adjusted survival curves based on Aalen's model"; *JSS*.

[10] Al-Baali, M. & Fletcher, R. (1985), "Variational Methods for Nonlinear Least Squares", *J. Oper. Res. Soc.*, **36**, 405-421.

[11] Al-Baali, M. & Fletcher, R. (1986), "An Efficient Line Search for Nonlinear Least Squares", *J. Optimiz. Theory Appl.*, **48**, 359-377.

[12] Al-Subaihi, A.A. (2002), "Variable Selection in Multivariable Regression Using SAS/IML", *JSS*, 2002.

[13] Amestoy, Davis, & Duff, I.S. (1996), "An approximate minimum degree ordering algorithm", *SIAM J. Matrix Analysis and Applic.* **17**, 886-905.

[14] Anderberg, M.R. (1973), *Cluster Analysis for Applications*, New York: Academic Press, Inc.

[15] Anderson, E., Bai, Z., Bischof, C., Demmel, J., Dongarra, J. , Du Croz, J., Greenbaum, A., Hammarling, S., McKenney, A., Ostrouchov, S., & Sorensen, D., (1995), *LAPACK User's Guide*, SIAM, Philadelphia, PA.

[16] Anderson, T. W. & Darling, D. A. (1954), "A test of goodness of fit", *Journal of the American Statistical Association*, **49**, 765-769.

[17] Andrei, N. (2007), “Scaled memoryless BFGS preconditioned conjugate gradient algorithm for unconstrained optimization”; *Optimization Methods and Software*, **22**, 561-571.

[18] Andrews, D.F., Bickel, P.J., Hampel, F.R., Huber, P.J., Rogers, W.H., Tukey, J.W. (1972), *Robust Estimation of Location: Survey and Advances*, Princeton NJ: Princeton University Press.

[19] Andrews, D.W.K. & Fair, R.C. (1988), ”Inference in Nonlinear Econometric Models with Structural Change,” *Review of Economic Studies*, **55**, 615-640.

[20] Anraku, K. (1999), “An Information Criterion for Parameters under a simple order restriction”, *Biometrika*, **86**, 141-152.

[21] Appelgate, D., Bixby, R., Chvatal, V. & Cook, W.(2006), *Concorde TSP Solver*, `http://www.tsp.gatech.edu/concorde`.

[22] Appelgate, D., Bixby, R., Chvatal, V. & Cook, W.(2000), “TSP cuts which do not conform to the template paradigm” in M. Junger & D. Naddef (eds.): *Computational Combinatorial Optimization, Optimal or Provably Near-Optimal Solutions*, *Lecture Notes in Computer Science*, Vol. 2241, pp. 261-304, London: Springer Verlag.

[23] Appelgate, D., Cook, W. & Rohe, A.(2003), “Chained Lin-Kernighan for large traveling salesman problems”, *INFORMS Journal on Computing*, **15**, 82-92.

[24] Aranda-Ordaz, F.J. (1981), ”On two families of transformations to additivity for binary response data,” *Biometrika*, **68**, 357-364.

[25] Archer, C.O. & Jennrich, R.I. (1974), “Standard errors for rotated factor loadings”; *Psychometrika*, **38**, 581-592.

[26] Armstrong, R.D. & Kung, D.S. (1979), “Algorithm AS 135: Min-Max Estimates for a Linear Multiple Regression Problem”, Appl. Statist. **28**, 93-100.

[27] Axelsson, O. (1996), *Iterative Solution Methods*, Cambridge University Press, Cambridge.

[28] Azzalini, A. & Capitanio, A. (1999), “Statistical applications of the multivariate skew-normal distribution”, *Journal Roy. Statist. Soc. B* **61**, part 3.

[29] Azzalini, A. & Dalla Valle, A. (1996), “The multivariate skew-normal distribution”, *Biometrika*, **83**, 715-726.

[30] Baker, F.B. (1992), *Item Response Theory: Parameter Estimation Techniques*, Marcel Dekker.

[31] Ballabio, D., Consonni, V., & Todeschini, R. (2009), “The Kohonen and CP-ANN toolbox: a collection of MATLAB modules of Self Organizing Maps and Counterpropagation Artificial Neural Networks”; *Chemometrics and Intelligent Laboratory Systems*, **98**, 115-122.

[32] Ballabio, D., Consonni, V., Vasighi, M., & Kompany-Zareh, M. (2011), “Genetic algorithm for architecture optimisation of counter-propagation artificial neural networks”; *Chemometrics and Intelligent Laboratory Systems*, **105**, 56-64.

[33] Ballabio, D. & Vasighi, M. (2011), “A MATLAB toolbox for self organizing maps and derived supervised neural network learning strategies”; *JSS*, 2011.

[34] Bamber, D. (1975), “The area above the ordinal dominance graph and the area below the receiver operating graph”, *J. Math. Psych.*, **12**, 387-415.

[35] Bao, X. (2007), “Mining Transaction/Order Data using SAS Enterprise Miner Association Mode”; *SAS Global Forum*, Paper 132-2007.

[36] Barber, C.B., Dobkin, D.P., & Hubdanpaa, H.T. (1996), “The quickhull algorithm for convex hulls”; *ACM Trans. on Mathematical Software.*

[37] Barnett, V. & Lewis, T. (1978), *Outlier in Statistical Data*, New York: John Wiley & Sons.

[38] Barrodale, I. & Philips, F.D.K. (1975), “An improved algorithm for discrete Chebyshev linear approximation”, in Hartnell, B.L., Williams, H.C (Eds.): *Proc. Fourth Manitoba Conf. on Numerical Mathematics*, Winnipeg 1975, 177-190.

[39] Barrodale, I. & Roberts, F.D.K. (1973), “An improved algorithm for discrete l_1 linear approximation”, *SIAM Journal Numerical Analysis*, **10**, 839-848.

[40] Barrodale, I. & Roberts, F.D.K. (1974), “Algorithm 478: Solution of an overdetermined system of equations in the $l1$-norm”,

[41] Bartels, R.E. & Stewart, G.W. (1972), “Solution of the equation AX + Xb = C”, Comm. ACM, **15**, 820-26.

[42] Bartolucci, F. (2005), “Clustering univariate observations via mixtures of unimodal normal mixtures”; *Journal of Classification*, **22**, 203-219.

[43] Bates, D.M. & Watts, D.G. (1988), *Nonlinear Regression Analysis and Its Applications*, New York: John Wiley & Sons.

[44] Baxter, M. and King, R.G. (1999), “Measuring business cycles: Approximate band-pass filters for economic time series”; *The Review of Economics and Statistics*, **81**, 575-593.

[45] Beale, E.M.L. (1972), ”A Derivation of Conjugate Gradients”, in *Numerical Methods for Nonlinear Optimization*, F. A. Lootsma (ed.), London: Academic Press.

[46] Bellman, R. (1995), *Introduction to Matrix Analysis*, SIAM, Philadelphia, PA.

[47] Ben-Israel, A. & Greville, T.N. (1974), *Generalized Inverses: Theory and Applications*, New York: John Wiley & Sons.

[48] Benjamini, Y. & Hochberg, Y. (1995), “Controlling the false discovery rate: a practical and powerful approach to multiple testing”, *Journal of the Royal Statistical Society B*, **57**, 289-300

[49] Benjamini, Y. & Liu, W. (1999), “A step-down multiple hypotheses testing procedure that controls the false discovery rate under interdependence”, *Journal of Statistical Planning and Inference*, **82**, 163-170.

[50] Benjamini, Y. & Yekutieli, D. (2001), “The control of the false discovery rate in multiple testing under dependency”, *Annals of Statistics*, **29**1165-1188.

[51] Benner, A., Ittrich, C., & Mansmann, U. (2006), “Predicting survival using microarray gene expression data,” Technical Report, *German Cancer Research Institute*, Heidelberg, Germany.

[52] Benner, A. (2006), “Survival analysis in high dimensions,” Technical Report, *German Cancer Research Institute*, Heidelberg, Germany.

[53] Bennett, K.P. (1999), “Combining support vector and mathematical programming methods for classification”; in: B. Schölkopf, C. Burges, & A. Smola (eds.): *Advances in Kernel Methods - Support Vector Machines*, pp. 307-326; Cambridge, MA: MIT Press.

[54] Bennett, K. P. & Embrechts, M. J.(2003), "An optimization perspective on kernel partial least squares regression", , .

[55] Bennett, K.P. & Mangasarian, O.L. (1992), "Robust linear programming discrimination of two linearly inseprable sets", *Optimization Software and Methods*, **1**, 23-34.

[56] Bentler, P.M. (1983), "Some Contributions to Efficient Statistics in Structural Models: Specification and Estimation of Moment Structures", *Psychometrika*, **48**,493-517.

[57] Bentler, P.M. (1989): *EQS, Structural Equations, Program Manual*, Program Version 3.0, Los Angeles: BMDP Statistical Software, Inc.

[58] Bentler, P.M. & Bonett, D.G. (1980), "Significance Tests and Goodness of Fit in the Analysis of Covariance Structures", *Psychological Bulletin*, **88**, 588-606.

[59] Bentler, P.M. & Weeks, D.G. (1980), "Linear Structural Equations with Latent Variables", *Psychometrika*, **45**, 289-308.

[60] Bentler, P.M. & Weeks, D.G. (1982), "Multivariate Analysis with Latent Variables", in *Handbook of Statistics, Vol. 2*, eds. P.R. Krishnaiah and L.N. Kanal, North Holland Publishing Company.

[61] Berkelaar, M., Eikland, K., Notebaert, P. (2004), *lp_solve (alternatively lpsolve)*, Open source (Mixed-Integer) Linear Programming system, Version 5.5.

[62] Berkowitz, J. (2001), "Testing density forecasts, with applications to risk management"; *Journal of Business and Economic Statistics*, **19**, 465-474.

[63] Bernstein, P. L. (1992), *Capital Ideas: The Improbable Origins of Modern Wall Street*, New York: The Free Press.

[64] Berry, M. B. & Browne, M. (1999), *Understanding Search Engines*, Philadelphia: SIAM.

[65] Berry, M. B. & Liang, M. (1992), "Large scale singular value computations", *International Journal of Supercomputer Applications*, **6**, pp. 13-49.

[66] Berry, M.J.A. & Linoff, G. (1997), *Data Mining for marketing, Sales, and Customer Support*, New York: J. Wiley and Sons, Inc.

[67] Bernaards, C.A. & Jennrich, R. A. (2004), "Gradient projection algorithms and software for arbitrary rotation criteria in factor analysis"; `http://www.stat.ucla.edu/research`; submitted for Publication.

[68] Betts, J. T. (1977), "An accelerated multiplier method for nonlinear programming", *Journal of Optimization Theory and Applications*, **21**, 137-174.

[69] Bi, J., Bennett, K.P., Embrechts, M., Breneman, C.M., & Song, M. (2002), "Dimensionality reduction via sparse support vector machines"; *Journal of Machine Learning*, **1**, 1-48.

[70] Billingsley, P. (1986), *Probability and Measure*, Second Edition, New York: J. Wiley.

[71] Birgin, E.G. & Martinez, J.M. (2001), "A spectral conjugate gradient method for unconstrained optimization"; *Appl. Math. Optim.*, **43**, 117-128.

[72] Bishop, Y.M., Fienberg, S. & Holland, P.W. (1975), *Discrete Multivariate Analysis*, Cambridge: MIT Press.

[73] Björck, A. (1996), *Numerical Methods for Least Squares Problems*, SIAM, Philadelphia, PA.

[74] Blaker, H. (2000), "Confidence curves and improved exact confidence intervals for discrete distributions"; *Canadian Journal of Statistics*, **28**, 783-798.

[75] Blanchard, G. & Roquain, E. (2008), "Two simple sufficient conditions for FDR control", *Electronic Journal of Statistics*, **2**, 963-992.

[76] Bock, R.D. (1972), "Estimating item parameters and latent ability when responses are scored in two or more nominal categories"; *Psychometrika*, **37**, 29-51.

[77] Bock, R.D. & Aitkin, M. (1981), "Marginal maximum likelihood estimation of item parameters: Application of an EM algorithm"; *Psychometrika*, **46**, 443-459.

[78] Boggs, P.T., Byrd, R.H., & Schnabel, R.B. (1987), "A stable and efficient algorithm for nonlinear orthogonal distance regression", *SIAM J. Sci. Stat. Comput.*, **8**, 1052-1078.

[79] Boggs, P.T., Byrd, R.H., Donaldson, J.R. & Schnabel, R.B. (1989), "Algorithm 676 - ODRPACK: Software for Weighted Orthogonal Distance Regression", *ACM TOMS*, **15**, 348-364.

[80] Boggs, P.T., Byrd, R.H., Rogers, J.E., & Schnabel, R.B. (1992), *Users ́ Reference Guide for ODRPACK Version 2.01*, Technical Report NISTIR 92-4834, National Institute of Standards and Technology, Gaithersburg MD.

[81] Boik, R. J. & Robison-Cox, J.F. (1997), "Derivatives of the Incomplete Beta Function", paper submitted to *Journal of Statistical Software*.

[82] Bollen, K.A. (1986), "Sample Size and Bentler and Bonett's Nonormed Fit Index", *Psychometrika*, **51**, 375-377.

[83] Bollen, K.A. (1987), "Total, direct, and indirect effects in structural equation models", in: *Sociological Methodology*, C. C. Clogg (ed.), Washington, DC: American Sociological Association.

[84] Bollen, K.A. (1989 a), "A New Incremental Fit Index for General Structural Equation Models", *Sociological Methods and Research*, **17**, 303-316.

[85] Bollen, K.A. (1989 b), *Structural Equations with Latent Variables*, New York: John Wiley & Sons, Inc.

[86] Bollen, K.A. & Stine, R.A. (1990), "Direct and indirect effects: classical and bootstrap estimates of variablity", in: *Sociological Methodology*, C. Clogg (ed.), Washington, DC: American Sociological Association.

[87] Bollen, K.A. & Stine, R.A. (1992), "Bootstrapping goodness-of-fit measures in structural equation models", in: *Testing Structural Equation Models*, Bollen, K.A. & J.S. Long (eds.), Newbury Park: Sage.

[88] Bollerslev, T. (1986), "Generalized Autoregressive Conditional Heteroskedasticity," *Journal of Econometrics*, **31**, 307-327.

[89] Bolstad, B.M., Irizarry, R.A., Astrand, M, and Speed, T.P. (2003) "A Comparison of Normalization Methods for High Density Oligonucleotide Array Data Based on Bias and Variance"; *Bioinformatics*, **19**, 2, 185-193.

[90] Bonnett, D.G., Woodward, J.A., & Randall, R.L. (2002), "Estimating p-values for Mardia's coefficients of multivariate skewness and kurtosis", *Computational Statistics*, **17**, 117-122.

[91] de Boor, C. (1978), *A Practical Guide to Splines*, Berlin: Springer Verlag.

[92] Borchers, H. W. (2013), Package *adagio*, in CRAN.

[93] Borg, I. & Groenen, P. (2005), *Modern Multidimensional Scaling: Theory and Applications*; Berlin: Springer Verlag.

[94] Borg, I., Groenen, P.J.F., & Mair, P. (2010), “Multidimensionale Skalierung”, Muenchen: Hampp Verlag.

[95] Botev, Z. I., Grotowski, J. F., and Kroese, D. P. (2010), ”Kernel Density Estimation Via Diffusion”, *Annals of Statistics*, **38**, 2916-2957.

[96] Bouaricha, A. & Moré, J.J. (1995), “Impact of Partial Separability on Large-scale Optimization”, Technical Report MCS-P487-0195, Argonne National Laboratory, Argonne.

[97] Bowman, K.O. & Shenton, L.R. (1979), “Approximate percentage points of Pearson distributions”; *Biometrica*, **66**, 147-151.

[98] Bracken, J. & McCormick, G.P. (1968), *Selected Applications of Nonlinear Programming*, New York: John Wiley & Sons.

[99] Bradley, P.S. & Mangasarian, O.L. (1998), “Feature selection via concave minimization and support vector machines,” in: J. Shavlik (ed.), *Machine Learning Proceedings of the Fifteenth International Conference*, 82-90, San Francisco: Morgan Kaufmann.

[100] Bradley, P.S. & Mangasarian, O.L. (1998), “Massive Data Discrimination via Linear Support Vector Machines”, Technical Report 98-05, Data Mining Institute, University of Wisconsin, Madison, Wisconsin.

[101] Bradley, P.S., Mangasarian, O.L., & Musicant, D.R. (1999), “Optimization methods in massive datasets,” in: J. Abello, P.M. Pardalos, and M.G.C. Resende (eds), *Handbook of Massive Datasets*, Dordrecht, NL: Kluwer Academic Press.

[102] Brand, E.M. (2002), “Incremental singular value decomposition of uncertain data with missing values”; *European Conference on Computer Vision (ECCV)*, 2350 : 707-720.

[103] Breiman, L. (1993), “Better subset selection using the non-negative garotte”; *Technical Report*, Univ. of California, Berkeley.

[104] Breiman, L. (2001), “Random Forests”, *Machine Learning*, **45**, 5-32.

[105] Breiman, L. & Cutler, A. (2001), “Random Forests”, Technical Report `www.stat.berkeley.edu/ breiman`.

[106] Breiman, L. & Cutler, A. (2002), “How to use Survival Forests (SPDV1)”, Technical Report `www.stat.berkeley.edu/ breiman`.

[107] Breiman, L., Friedman, J.H., Olshen, R.A., & Stone, C.H. (1984), *Classification and Regression Trees*, Wadsworth, Belmont CA.

[108] Brent, R. (1973), *Algorithms for Minimization Without Derivatives*, Prentice Hall, Inc.

[109] Breslow, N. E. & Day, N. E. (1980), *Statistical Methods of Cancer Research; Vol. I: The analysis of Case-Control Studies*, IARC Scientific Publications, IARC Lyon.

[110] Breslow, N. (2005), “Case-Control Study, Two-phase”, in P. Armitage (ed.), *Encyclopedia of Biostatistics*, pp. 734-741, New York: J. Wiley & Sons.

[111] Bretz, F. (1999), *Powerful Modifications of William’s Test on Trend*, Dissertation, Dept. of Bioinformatics, University of Hannover.

[112] Bretz, F., Hsu, J.C., Pinheiro, J.C. & Liu, Y. (2008), “Dose finding - a challenge in statistics”, *Biometrical Journal*, **50**, 480-504.

[113] Bretz, F., Hothorn, T. & Westfall, P. (2011), *Multiple Comparisons Using R*, Boca Raton, London, New York: CRC Press.

[114] Bretz, F. & Hothorn, L.A. (1999), "Testing dose-response relationships with a priori unknown, possibly non-monotonic shapes", Technical Report, Dept. of Bioinformatics, University of Hannover.

[115] Brockwell, P.J., Dahlhaus, R., & Trindade, A.A. (2002), "Modified Burg algorithms for multivariate subset autoregression", Technical Report 2002-015, Dept. of Statistics, Univeristy of Florida.

[116] Bronstein, I.N. & Semendjajew, K.A. (1966), *Taschenbuch der Mathematik*, B.G. Teubner, Leipzig.

[117] Brown, B.W., Lovato, J., & Russell, K. (1997) *DCDFLIB: Library of Fortran Routines for Cumulatice Distribution Functions, Inverses, and Other Parameters*, Dept. of Biomathematics, University of Texas, Houston.

[118] Brown, B.W., Lovato, J., & Russell, K., Venier, J. (1997) *RANDLIB: Library of Fortran Routines for Random Number Generation*, Dept. of Biomathematics, University of Texas, Houston.

[119] Brown, R. G. (2009), "Dieharder: A random number test suite"; `www.phy.duke.edu/~rgb/General/dieharder.php`

[120] Browne, M.W. (1974), "Generalized Least Squares Estimators in the Analysis of Covariance Structures", *South African Statistical Journal*, **8**, 1 - 24.

[121] Browne, M.W. (1982), "Covariance Structures", in *Topics in Multivariate Analyses*, ed. D.M. Hawkins, Cambridge University Press.

[122] Browne, M. W. (1984), "Asymptotically Distribution-free Methods for the Analysis of Covariance Structures", *Br. J. math. statist. Psychol.*, **37**, 62-83.

[123] Browne, M. W., (1992), "Circumplex Models for Correlation Matrices", Psychometrika, **57**, 469-497.

[124] Browne, M. W. (2001), "An overview of analytic rotation in exploratory factor analysis"; *Multivariate Behavioral Research*, **36**, 111-150.

[125] Browne, M. W. & Cudeck, R. (1993), "Alternative Ways of Assessing Model Fit", in: *Testing Structural Equation Models*, eds. K. A. Bollen & S. Long, Newbury Park: SAGE Publications.

[126] Browne, M. W. & Du Toit, S.H.C. (1992), "Automated Fitting of Nonstandard Models", *Multivariate Behavioral Research*, **27**, 269-300.

[127] Browne, M.W. & Shapiro, A. (1986), "The Asymptotic Covariance Matrix of Sample Correlation Coefficients under General Conditions", *Linear Algebra and its Applications*, **82**, 169-176.

[128] Brownlee, K.A. (1965), *Statistical Theory and Methodology in Science and Engineering*; New York: John Wiley & Sons.

[129] *BSD UNIX Programmer's Manual* (1989), Hewlett-Packard Comp., Palo Alto, CA.

[130] Bunch, J.R. & Kaufman, L. (1977), "Some stable methods for calculating inertia and solving symmetric linear systems", *Math. Comp.* **31**, 163-179.

[131] Burg, J. P. (1968), "A new analysis technique for time series data"; *NATO Advanced Study on Signal Processing with Emphasis on Underwater Acoustics*, Enschede, netherlands, August, 1968.

[132] Burnham, K. (1989), "Numerical Survival Rate Estimation for Capture-Recapture Models Using SAS PROC NLIN", in: L.McDonald, B. Manly, J. Lockwood, & J. Logan (Eds.): *Estimation and Analysis of Insect Populations*, Lecture Notes in Statistics **55**, Springer Verlag, Berlin-Heidelberg-New York.

[133] Bus, J.C.P. & Dekker, T.J. (1975), "Two efficient algorithms with guaranteed convergence for finding a zero of a function", *ACM TOMS*, **1**, 330-345.

[134] Butler, R.W., Davies, P.L., & Jhun, M. (1993), "Asymptotics for the Minimum Covariance Determinant Estimator", *The Annals of Statistics*, **21**, 1385-1400.

[135] Byrd, R.H., Lu, P., Nocedal, J., & Zhu, C. (1995), "A limited memory algorithm for bound constrained optimization", *SIAM Journal Scientific Computation*, **16**, 1190-1208.

[136] Byrd, R., Nocedal, J., & Schnabel,R. (1994) "Representations of Quasi-Newton Matrices and their use in Limited Memory Methods", *Mathematical Programming*, **63** no. 4, 129-156.

[137] Cai, L., Maydeu-Olivares, A., Coffman, D.L. & Thissen, D. (2006), "Limited information goodness of fit testing of item response theory models for sparse 2^p tables"; *British Journal of Mathematical and Statistical Psychology*, **59**, 173-194.

[138] Cain, K.C. & Breslow, N.E. (1988), "Logistic regression analysis and efficient design for two-stage studies", *American Journal of Epidemiology*, **128**, 1198-1206.

[139] Carbon, C.C. (2011), "BiDimRegression: A Matlab toolbox for calculating bidimensional regressions"; JSS, 2011.

[140] Carey, V., Zeger, S.L., & Diggle, K.Y., (1993) *Modelling multivariate binary data with alternating logistic regressions*, *Biometrika*, **80**, 517-526.

[141] Carlson, B.C. & Notis, E.M. (1979), "Algorithm 577: Algorithms for Incomplete Elliptic Integrals", *ACM TOMS*, **7**, 398-403.

[142] Carpaneto, G., Dell'Amico, M., & Toth, P. (1995), "A branch-and-bound algorithm for large scale asymmetric traveling salesman problems", Algorithm 750, *Transactions on Mathematical Software*, **21**, 410-415.

[143] Carroll, R., Gail, M., Lubin, J. (1993), "Case-control studies with errors in covariates", *Journal of the American Stat. Association*, **88**, 185-199.

[144] Carroll, R.A., Ruppert, D. & Stefanski, L.A. (1995), *Measurement error in Nonlinear Models*, London: Chapman and Hall.

[145] Carroll, R.A., Küchenhoff, H., Lombard, F., & Stefanski, L.A. (1996), "Asymptotics for the SIMEX estimator in nonlinear measurement error models", *JASA*, **91**, 242-25.

[146] Carroll, J.D. & Chang, J.J. (1970), "Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition"; *Psychometrika*, **35**, 283-320.

[147] Carroll, R., Gail, M., Lubin, J. (1993), "Case-control studies with errors in covariates", *Journal of the American Stat. Association*, **88**, 185-199.

[148] Catral, M., Han, L., Neumann, M. & Plemmons, R. J. (2004), "On reduced rank nonnegative matrix factorization for symmetric nonnegative matrices", *Linear Algebra and Applications*, **393**, 107-127.

[149] Cattell, R.B. (1966), "The scree test for the number of factors", *Multivariate Behavioral Research*, **1**, 245-276.

[150] Cattell, R.B. & Vogelmann, S. (1977), "A comprehensive trial of the scree and KG criteria for determining the number of factors", *Multivariate Behavioral Research*, **12**, 289-325.

[151] Chamberlain, R.M., Powell, M.J.D., Lemarechal, C., & Pedersen, H.C. (1982), "The watchdog technique for forcing convergence in algorithms for constrained optimization", *Mathematical Programming*, **16**, 1-17.

[152] Chan, T. F. (1982a), "An improved algorithm for computing the singular value decomposition", *ACM TOMS*, **8**, 72-83.

[153] Chan, T. F. (1982b), “Algorithm 581: An improved algorithm for computing the singular value decomposition”, *ACM TOMS*, **8**, 84-88.

[154] Chan, T. F. (1987), “Rank revealing QR factorizations”, *Linear Algebra and Its Applic.*, **88/89**, 67-82.

[155] Chang, C.M. (2003), “MinPROMEP: Generation of partially replicated minimal orthogonal main-effect plans using a novel algorithm”, *JSS*, 2003.

[156] Chang, C.M. (2003), “Construction of partially replicated minimal orthogonal main-effect plans using a general procedure”, *Utilitas Mathematica*, 2003.

[157] Chasalow, S. (2005), “The combinat Package”; Technical Report for R functions. See `CRAN` website.

[158] Chatterjee, S. & Price, B. (1977), *Regression Analysis by Example*, New York: John Wiley & Sons.

[159] Chauvenet, W. (1863), *A Manual of Spherical and Practical Astronomy*, Dover N.Y. 1960, 474-566.

[160] Chen, R.-B., Chu, C.-H., and Weng, J.-Z. (2010), “A stochastic matching pursuit (SMP) MATLAB toolbox for Bayesian variable selection”; *JSS*, 2010.

[161] Cheung, T. Y. (1980), “Multifacility location problem with rectilinear distance by the minimum cut approach”, *ACM Trans. Math. Software*, **6**, 387-390.

[162] Christensen, R., Pearson, L.M., & Johnson, W. (1992), “Case deletion diagnostics for mixed models”, *Technometrics*, **34**, 38-45.

[163] Chu, M. T. & Funderlik, R. E. (2002), “The centroid decomposition: Relationships between discrete variational decompositions and SVD’s”; *SIAM Journal Matrix Anal. Appl.*, **23**, 1025-1044.

[164] Chung, J,K., Kannappan, P.L., Ng, C.T. & Sahoo, P.K. (1989), “Measure of distance between probability distributions”; *Journal of Mathem. Analysis and Applications*, **138**, 280-292.

[165] Clarke, G.P.Y. (1987), ”Approximate Confidence Limits for a Parameter Function in Nonlinear Regression”, *JASA*, **82**, 221-230.

[166] Clauß, G. & Ebner, H. (1970), *Grundlagen der Statistik für Psychologen, Pädagogen und Soziologen*, Frankfurt a.M. und Zürich, Verlag Harri Deutsch.

[167] Cleveland, W.S. (1979), “Robust locally weighted regression and smoothing scatterplots”; *JASA*, **74**, 829-836.

[168] Cleveland, W.S., Grosse, E. & Shyu, W.M. (1992), “Local regression models”; in: *Statistical Models in S*, eds. J.M. Chambers and T.J. Hastie; New York: Chapman & Hall.

[169] Cody, W.J. (1969), “Rational Chebyshev approximation for the error function”, *Mathematics of Computation*, **23**, 631-637.

[170] Cody, W.J. (1993), “Algorithm 715”, *ACM TOMS*, **19**, 22-32.

[171] Cody, W.J. & Waite, W. (1980), *Software Manual for the Elementary Functions*, Prentice Hall, Inc., Englewood Cliffs, NJ.

[172] Coleman, T.F., Garbow, B.S., & Moré, J.J. (1984), “Software for Estimating Sparse Jacobian Matrices”, *ACM TOMS*, **10**, 329-345.

[173] Coleman, T.F., Garbow, B.S., & Moré, J.J. (1985), “Software for Estimating Sparse Jacobian Matrices”, *ACM TOMS*, **11**, 363-377.

[174] Conn, A. R. & Gould, N.I.M (1984), "On the location of directions of finite descent for nonlinear programming algorithms", *SIAM Journal Numerical Analysis*, **21**, 1162-1179.

[175] Cook, R.D. (1998), *Regression Graphics*, New York: Wiley & Sons.

[176] Cook, J. R. & Stefanski, L.A. (1994), "Simulation Extrapolation in Parametric Measurement Error Models", JASA **89**, 1314-1328.

[177] Cook, R.D. & Weisberg, S. (1990), "Confidence curves for nonlinear regression", *JASA*, **85**, 544-551.

[178] Cook, R.D & Weisberg, S. (1994), *An Introduction to Regression Graphics*, New York: John Wiley & Sons.

[179] Cook, R.D. & Weisberg, S. (1999), *Applied Regression Including Computing and Graphics*, New York: Wiley. & Sons

[180] Copenhaver, T.W. and Mielke, P.W. (1977), "Quantit analysis: a quantal assay refinement," *Biometrics*, **33**, 175-187.

[181] Cormen, T.H., Leiserson, C.E., & Rivest, R.L. (1997), *Introduction to Algorithms*, Cambridge: MIT Press.

[182] Cornuejols, G. & Tütüncü, R. (2006), *Optimization Methods in Finance*, Pittsburgh PA: Carnegie Mellon University.

[183] Cox, C. (1998), "Delta Method", *Encyclopedia of Biostatistics*, eds. Armitage, P. & Colton, T., 1125-1127, New York: J. Wiley.

[184] Cox, D.R. & Hinkley, D.V. (1974), *Theoretical Statistics*, London: Chapman and Hall.

[185] Chamberlain, R.M.; Powell, M.J.D.; Lemarechal, C.; & Pedersen, H.C. (1982), "The watchdog technique for forcing convergence in algorithms for constrained optimization", *Mathematical Programming*, **16**, 1-17.

[186] Cooley, W.W. & Lohnes, P.R. (1971), *Multivariate Data Analysis*, New York: John Wiley & Sons, Inc.

[187] Cramer, J.S. (1986), *Econometric Applications of Maximum Likelihood Methods*, Cambridge: Cambridge University Press.

[188] Cristianni, N. & Shawe-Taylor, J. (2000), *Support Vector Machines and other kernel-based learning methods*, Cambridge University Press, Cambridge.

[189] Croes, G. A. (1958), "A method for solving traveling-salesman problems", *Operations Research*, **6**, 791-812.

[190] Cronbach, L.J. (1951), "Coefficient alpha and the internal structure of tests"; *Psychometrika*, **16**, 297-334.

[191] Croux, C., Filzmoser, P., Pison, G. & Rousseeuw (2004), "Fitting multiplicative models by robust alternating regressions"; Technical Report.

[192] Cudeck, R. & Browne, M.W. (1984), "Cross-validation of covariance structures", *Multivariate Behavioral Research*, **18** 62-83.

[193] Czyzyk, J., Mehrotra, S., Wagner, M. & Wright, S.J. (1997) "PCx User Guide (Version 1.1)", Technical Report OTC 96/01, Office of Computational and Technology Research, US Dept. of Energy.

[194] Dale, J.C. (1985), "A Bivariate Discrete Model of Changing Colour in Blackbirds", in *Statistics in Ornithology*, Morgan, B.J.T. and North, P.M. (eds.) Berlin: Springer-Verlag.

[195] Dale, J.C. (1986), "Global Cross-Ratio Models for Bivariate, Discrete, Ordered Responses", *Biometrics*, **42**, 909-917.

[196] Dalenius, T. (1957), *Sampling in Sweden. Contributions to Methods and Theories of Sample Survey Practice*; Stockholm: Almqvist & Wiksells.

[197] Daniel, J.W., Gragg, W.B. , Kaufman, L. & Stewart, G.W. (1976), "Reorthogonalization and Stable Algorithms for Updating the Gram-Schmidt QR Factorization", Math. Comp. **30**, 772-795.

[198] Davies, L. (1992), "The Asymptotics of Rousseeuw's Minimum Volume Ellipsoid Estimator", *The Annals of Statistics*, **20**, 1828-1843.

[199] Davison, A.C. & Hinkley, D.V. (1997), *Bootstrap Methods and their Application*, Cambridge: Cambridge University Press.

[200] Davison, A.C., Hinkley, D.V., & Schechtman, E. (1986), " Efficient bootstrap simulation", *Biometrika*, **73**, 555-566.

[201] Dayal, B.S. & McGregor, J.F. (1997), "Improved PLS Algorithms", *Journal of Chemometrics*, **11**, 73-85.

[202] Deak, I. (1980), "Three Digit Accurate Multiple Normal Probabilities", *Numer. Math.*, **35**, 369-380.

[203] De Berg, M., Cheong, O., & van Kreveld, M. (2008). *Computational Geometry: Algorithms and Applications*, New York: Springer, 2008.

[204] DeJong, S. (1993), "SIMPLS: An alternative approach to partial least squares regression", *Chemometrics and Intelligent Laboratory Systems*, **18**, 251-263.

[205] DeJong, S. & ter Braak, C.J.F. (1994), *Journal of Chemometrics*, **8**, 169-174.

[206] de Leeuw, J. (1977), "Applications of convex analysis to multidimensional scaling"; in: Barra, Brodeau, Romier, & van Cutsem (eds.): *Recent Developments in Statistics*, Amsterdam: North Holland Publishing Company.

[207] de Leeuw, J. (1983), "Models and Methods for the Analysis of Correlation Coefficients", *Journal of Econometrics*, **22**, 113-137.

[208] de Leeuw, J. (1984), "Canonical Analysis of Categorical Data"; Leiden: DSWO Press.

[209] de Leeuw, J. (2005), "SMACOF in R", Technical Report.

[210] de Leeuw, J. (2012), "ALSCAL in R", Technical Report.

[211] de Leeuw, J. (1994), "Block relaxation methods in statistics"; in Bock, Lenski, & Richter (ed.): *Information Systems and Data Analysis*, Berlin: Springer Verlag.

[212] de Leeuw, J. (2007), see the following website for a set of R programs: `http://gifi.stat.ucla.edu/psychoR/`

[213] de Leeuw, J. & Heiser, W. (1977), "Convergence of correction-matrix algorithms for multidimensional scaling"; in J. C. Lingoes (ed.): *Geometric Representations of Relational Data*, Ann Arbor, MI: Mathesis Press.

[214] de Leeuw, J. & Mair, P. (2009), "Multidimensional scaling using majorization: The R package smacof", *Journal of Statistical Software* **3**, 1-30. `www.jstatsoft.org/v31/i03/`

[215] de Leeuw, J. & Pruzansky, S. (1978), "A new computational method to fit the weighted Euclidean distance model"; *Psychometrika*, **43**, 479-490.

[216] de Leeuw, J., Young, F. W. & Takane, Y. (1976), "Additive structure in qualitative data: An alternating least squares method with optimal scaling features"; *Psychometrika*, **41**, 471-503.

[217] DeLong E.R., DeLong D.M., & Clarke-Pearson, D.L. (1988), "Comparing the areas under two or more correlated receiver operating characteristic curves: A nonparametric approach," *Biometrics*, **44**, pp. 837-845.

[218] Dennis, J.E., Gay, D.M., & Welsch, R.E. (1981), "An Adaptive Nonlinear Least-Squares Algorithm", *ACM Trans. Math. Software*, **7**, 348-368.

[219] Dennis, J.E. & Mei, H.H.W. (1979), "Two new unconstrained optimization algorithms which use function and gradient values", *J. Optim. Theory Appl.*, **28**, 453-482.

[220] Dennis, J.E. & Schnabel, R.B. (1983), *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, New Jersey: Prentice-Hall.

[221] Diggle, P.J., Liang, K.Y., & Zeger, S.L. (1994) *Analysis of Longitudinal Data*, Oxford: Clarendon Clarendon Press, 1994.

[222] Ding, C., He, X., & Simon, H. D. (2005), "On the equivalence of nonnegative matrix factorization and spectral clustering", *Proc. SIAM Data Mining Conf.*, 2005.

[223] Ding, C., Peng, T.L., & Park, H. (2006), "Orthogonal nonnegative matrix tri-factorizations for clustering", .

[224] Dixon, W. J. (1950), "Analysis of extreme values"; *Annals of Mathematical Statistics*, **21**, 488-506.

[225] Dixon, W. J. (1951), "Ratios involving extreme values"; *Annals of Mathematical Statistics*, **22**, 68-78.

[226] *DOMAIN C Library (CLIB) Reference* (1985), Apollo Computer Inc., Chelmsford, MA.

[227] Dongarra, J.J., Bunch, J.R., Moler, C.B., & Stewart, G.W. (1979), *LINPACK User's Guide*, SIAM, Philadelphia, PA.

[228] Doornik, J. A. & Hansen, H. (2008): "An omnibus test for univariate and multivariate normality", *Oxford Bulletin of Economics and Statistics*, **70**, 927-939.

[229] Draper, N.R. & Smith, H. (1981), *Applied Regression Analysis*, New York: J. Wiley.

[230] Du, D.-Z. & Pardalos, P. M. (1993), *Network Optimization Problems - Algorithms, Applications, and Complexity*, Singapore: World Scientific.

[231] Duff, I.S. & Reid, J.K. (1978), *An implementation of Tarjan's algorithm for the block triangularization of a matrix*, ACM TOMS, **4**, 137-147.

[232] Duncan, G.T. (1978), "An Empirical Study of jackknife-constructed confidence regions in nonlinear regression", *Technometrics*, **20**, 123-129.

[233] Dunnett, C. W. (1980), "Pairwise multiple comparisons in the unequal variance case", *Journal of the American Statistical Association*, **75**, 796-800.

[234] Ecker, J. G. & Kupferschid, M. (1988), *Introduction to Operations Research*, (Reprint of 1991), Malabar, FL: Krieger Publishing Company

[235] Eckert, R. E. (1994), Purdue University, W. Lafayette, Personal Communication.

[236] Edlund, O. (1999), *Solution of Linear Programming and Non-Linear Regression Problems Using Linear M-Estimation Methods*, LuleåUniversity of Technology, Sweden.

[237] Efron, B. (1994), *The Jackknife, the Bootstrap, and Other Resampling Methods*, Philadelphia: SIAM.

[238] Efron, B. & Tibshirani, R.J. (1993), *An Introduction to the Bootstrap*, New York: Chapman & Hall.

[239] Efron, B., Hastie, T., Johnstone, I. & Tibshirani, R. (2002), “Least Angle Regression”, *The Annals of Statistics*, **32**, 407-499.

[240] Einarsson, H. (1998), “Algorithms for General non-differentiable Optimization Problems”; Master of Science Thesis, IMM, Technical University of Denmark.

[241] Einarsson, H. & Madsen, K. (1999), “Sequential linear programming for non-differentiable optimization”; Paper presented at the *SIAM Meeting on Optimization*, Atlanta 1999.

[242] Elton, E. and Gruber, M. (1981), *Modern Portfolio Theory and Investment Analysis*, New York: John Wiley & Sons, Inc.

[243] Emerson, P.L. (1968), “Numerical construction of orthogonal polynomials from a general recurrence formula”, *Biometrics*, **24**, 695-701.

[244] Engle, R.F. (1982), ”Autoregressive Conditional Heteroskedasticity with Estimates of the Variance of UK Inflation”, *Econometrica*, **50**, 987-1008.

[245] Eskow, E. & Schnabel, R.B. (1991), “Algorithm 695: Software for a New Modified Cholesky Factorization”, *ACM Trans. Math. Software*, **17**, 306-312.

[246] Esmailzadeh, N. (2013), “Two level search designs in Matlab”, paper and software submitted to JSS.

[247] Everitt, B.S. (1984), *An Introduction to Latent Variable Methods*, London: Chapman & Hall.

[248] Everitt, B.S. (1996), *A Handbook of Statistical Analyses Using SAS*, London: Chapman & Hall.

[249] Fan, J., & Li, R. (2001), “Variable selection via nonconcave penalized likelihood and its oracle properties,” *JASA*, **96**, pp. 1348-1360.

[250] Fan, J., & Li, R. (2002), “Variable selection for Cox’s proportional hazards model and frailty model,” *The Annals of Statistics*, **30**, pp. 74-99.

[251] Fay, M. P. (2010), “Two-sided exact tests and matching confidence intervals for discrete data”, *R Journal*, **2**, 53-58.

[252] Ferguson, T. (1996), *A Course in Large Sample Theory*, London: Chapman & Hall.

[253] Ferson, W.E. and Harvey, C.R. (1992), ”Seasonality and Consumption-Based Asset Pricing,” *Journal of Finance*, **47**, 511-552.

[254] Fig, M. (2010), Matlab toolbox for permutations and combinations.

[255] Finney, D.J. (1947), “The estimation from individual records of the relationship between dose and quantal response”, *Biometrika*, **34**, 320-334.

[256] Fisher, R. A. (1935), “The logic of inductive inference”, *Journal of the Royal Statistical Society, Series A*, 39-54.

[257] Fisher, R.A. (1936), “The use of multiple measurements in taxonomic problems”, *Annals of Eugenics*, **7**, part II, pp. 179-188.

[258] Fisher, R. A. (1962), “Confidence limits for a cross-product ratio”, *Australian Journal of Statistics*, **4**, 41.

[259] Fisher, R. A. (1970), *Statistical Methods for Research Workers*, Oliver & Boyd.

[260] Fishman, G.S. (1996), *Monte Carlo: Concepts, Algorithms, and Applications*, New York: Springer Verlag.

[261] Fishman, G.S. & Moore, L.R. (1986), "An exhaustive analysis of multiplicative congruential random number generators with modulus 2**31-1", *SIAM Journal on Scientific and Statistical Computation*, **7**, p.24-45.

[262] Fletcher, R. (1987), *Practical Methods of Optimization*, 2nd Ed., Chichester: John Wiley & Sons.

[263] Fletcher, R. & Powell, M.J.D. (1963), "A Rapidly Convergent Descent Method for Minimization", *Computer Journal*, **6**, 163-168.

[264] Fletcher, R. & Xu, C. (1987), "Hybrid Methods for Nonlinear Least Squares", *Journal of Numerical Analysis*, **7**, 371-389.

[265] Floudas, C.A. & Pardalos, P.M. (1990), *A Collection of Test Problems for Constrained Global Optimization Algorithms*; *Lecture Notes in Computer Science*, **455**, Berlin: Springer Verlag.

[266] Forrest, J. J. & Lougee-Helmer, R. (2014), *Cbc*, `jjforre@us.ibm.com`, `robinlh@us.ibm.com`.

[267] Forrest, J. J., de la Nuez, D., & Lougee-Helmer, R. (2014), *Clp*, `http://www.tsp.gatech.edu/concorde`.

[268] Forsythe, G.E., Malcolm, M.A., & Moler, C.B. (1977), *Computer Methods for Mathematical Computations*, Englewood-Cliffs, NJ: Prentice Hall, 1977.

[269] Foster, D.P. & Stine, R.A. (2004), "Variable selection in Data Mining: Building a predictive model for bankruptcy"; JASA, **99**, 303-313.

[270] Fox, T., Hinkley, D. & Larntz, K. (1980), "Jackknifing in Nonlinear Regression", *Technometrics*, **22**, 29-33.

[271] Frank, I. & Friedman, J. (1993), "A statistical view of some chemometrics regression tools"; *Technometrics*, **35**, 109-148.

[272] Fraser, C. (1980), *COSAN User's Guide*, Toronto: The Ontario Institute for Studies in Education.

[273] Fraser, C. & McDonald, R.P. (1988), "NOHARM: Least squares item factor analysis", *Multivariate Behavioral Research*, **23**, 267-269.

[274] Fredman, M.L., Johnson, D.S., McGeoch, L.A. & Ostheimer, G. (1995), "Data structures for Traveling Salesmen", *J. Algorithms*, **18**, 432-479.

[275] Friedman, A., & Kohler, B. (2003), "Bidimensional regression: Assessing the configural similarity and accuracy of cognitive maps and other two-dimensional data sets", *Psychological Methods*, **8**, 468-491.

[276] Friedman, J.H., Bentley, J.L., Finkel, R.A. (1977), "An algorithm for finding best matches in logarithmic expected time", em ACM Transactions Math. Softw., p. 209-226, `http://doi.acm.org/10.1145/355744.355745`

[277] Friedman, J.H., & Tukey, J.W. (1974), "A projection pursuit algorithm for exploratory data analysis", J. Amer. Stat. Assoc. **62**, 1159-1178.

[278] Friedman, J. H. & Stuetzle, W. (1981), "Projection pursuit regression", *JASA*, **76**, 817-823.

[279] Fritsch, F.N. & Carlson, R.E. (1980), "Monotone piecewise cubic interpolation", *SIAM Journal Numerical Analysis*, **17**, 238-246.

[280] Froehlich, H., & Zell, A., (2004), "Feature subset selection for support vector machines by incremental regularized risk minimization", in: *The International Joint Conference on Neural Networks (IJCNN)*, **3**, 2041-2046.

[281] Fuller, W.A. (1987), *Measurement Error Models*, New York: J. Wiley & Sons.

[282] Fung, G. & Mangasarian, O.L. (2000), "Proximal Support Vector Machines", Technical Report, Data Mining Institute, University of Wisconsin, Madison, Wisconsin.

[283] Fung, G. & Mangasarian, O.L. (2002), "Finite Newton method for Lagrangian support vector machine classification"; Technical Report 02-01, Data Mining Institute, Computer Sciences Dep., Univ. of Wisconsin, Madison, Wisconsin, 2002.

[284] Fung, G. & Mangasarian, O.L. (2003), "A Feature Selection Newton Method for Support Vector Machine Classification", *Computational Optimization and Applications*, 1-18.

[285] Gaevert, H., Hurri, J., Saerelae, J. & Hyvaerinen, A. (2005) see `http://www.cis.hut.fi/projects/ica/fastica/` for version 5

[286] Gallant, A. R. (1987), *Nonlinear Statistical Models*, New York: John Wiley & Sons, Inc.

[287] Garbow, B.S., Boyle, J.M., Dongarra, J.J., & Moler, C.B. (1977), *Matrix Eigensystem Routines - EISPACK Guide Extension*, *Lecture Notes in Computer Science*, vol. 51, Springer Verlag, Berlin.

[288] Gay, D.M. (1983), "Subroutines for Unconstrained Minimization", *ACM Trans. Math. Software*, **9**, 503-524.

[289] Genton, M. (2001), "Classes of kernels for machine learning: a statistics perspective", *Journal of Machine Learning Research*, **2**, 299-312.

[290] Gao, X., et al. (2008), "Nonparametric multiple comparison procedures for unbalanced one-way factorial designs", *Journal of Planning and Inference*, **77**, 2574-2591.

[291] Genz, A. (1986), "Fully Symmetric Interpolatory Rules for multiple Integrals", *SIAM Journal Numer. Analysis*, **23**, 1273-1283.

[292] Genz, A. (1991), *Computing in the 90s*, *Lecture Notes in Computer Science*, vol. 507, New York: Springer Verlag.

[293] Genz, A. (1992), "Numerical Computation of Multivariate Normal Probabilities", *J. of Computational and Graphical Stat.*, **1**, 141-149.

[294] Genz, A. (1999), "Numerical computation of critical values for multiple comparison problems", Technical Report.

[295] Genz, A. (2000), "Numerical computation of bivariate and trivariate normal probabilities", Technical Report.

[296] Genz, A. (2004), "Numerical computation of rectangular bivariate and trivariate normal and *t*-probabilities", *Statistics and Computing*, **14**, 251-260.

[297] Genz, A. & Bretz, F. (1999), "Methods for the computation of multivariate t-probabilities", Technical Report.

[298] Genz, A. & Bretz, F. (1999), "Numerical computation of multivariate t-probabilities with application to power calculation of multiple contrasts", Technical Report.

[299] Genz, A. & Bretz, F. (2002), "Methods for the computation of multivariate t-probabilities", *Journal of Computational and Graphical Statistics*, **11**, 950-971.

[300] Genz, A. & Bretz, F. (2009), *Computation of multivariate normal and t-probabilities*, Lecture Notes in Statistics, Heidelberg: Springer Verlag.

[301] Genz, A. & Kass, R. (1996), "Subregion adaptive integration of functions having a dominant peak", *J. of Computational and Graphical Stat.*,

[302] Genz, A. & Kwong, K.S. (1999), "Numerical evaluation of singular multivariate normal distributions", Technical Report.

[303] Genz, A. & Monahan, J. (1996), "Stochastic integration rules for infinite regions", *SIAM Journal Scientific Computation*,

[304] George, J.A. & Liu, J.W. (1981), *Computer Solution of Large Sparse Positive Definite Systems*, New Jersey: Prentice-Hall.

[305] George, J.A., Gilbert, J.R., & Liu, J.W.H. (1993), *Graph Theory and Sparse Computations*, Springer Verlag, New York.

[306] Ghali, W.A., Quan, H., Brant, R., van Melle, G., Norris, C.M., Faris, P.D., Galbraith, P.D. & Knudtson, M.L. (2001), "Comparison of 2 methods for calculating adjusted survival curves from proportional hazards model", *Journal of American Medical Association*, **286**, 1494-1497.

[307] Ghosh, S. & Teschmacher, L. (2002), "Comparisons of search designs using search probabilities", *Journal of Statistical Planning and Inference*, **104**, 439-458.

[308] Ghysels, E. and Hall, A. (1990), "A Test for Structural Stability of Euler Conditions Parameters Estimated via the Generalized Method of Moments Estimator," *International Economic Review*, **31**, 355-364.

[309] Gifi, A., (1990) *Nonlinear Multivariate Analysis*, Chichester: Wiley, 1990.

[310] Gilbert, J.R, Ng, E., & Peyton, B.W. (1994), "An efficient algorithm to compute row and column counts for sparse Cholesky factorization", *SIAM J. Matrix Anal. Appl.*, **15**, 1075-1091.

[311] Gill, P.E. , Murray, W., Ponceleon, D.B., & Saunders, M.A. (1992), "Preconditioners for indefinite systems arising in optimization", *SIAM J. on Matrix Analysis and Applications*, **13**, 292-311.

[312] Gill, E.P., Murray, W., Saunders, M.A., & Wright, M.H. (1983), "Computing Forward-Difference Intervals for Numerical Optimization", *SIAM Journal on Scientific and Statistical Computation*, **4**, 310-321.

[313] Gill, E.P., Murray, W., Saunders, M.A., & Wright, M.H. (1984), "Procedures for Optimization Problems with a Mixture of Bounds and General Linear Constraints", *ACM Trans. Math. Software*, **10**, 282-298.

[314] Gill, P.E., Murray, W., & Wright, M.H. (1981), *Practical Optimization*, Academic Press, New York.

[315] Gini, C. (1912), "Variabilità é mutabilita, contributo allo studio delle distribuzioni e delle relazioni statistiche", *Studi Economico - Giuridici della R. Universitá di Cagliari*, **3**, 3-159.

[316] Gleason, J.R. (1988), "Algorithms for balanced bootstrap simulations", *American Statistician*, **42**, 263-266.

[317] Goano, M. (1995), "Algorithm 745: Computation of the Complete and Incomplete Fermi-Dirac Integral", *ACM TOMS*, **21**, 221-232.

[318] Goffe, W.L., Ferrier, G.D., & Rogers, J. (1994): "Global optimization of statistical functions with simulated annealing"; *Journal of Econometrics*, **60**, 65-99.

[319] Gold, C., Holub, A. & Sollich, P., (2005), "Bayesian approach to feature selection and parameter tuning for support vector machine classifiers", *Neural Networks*, **18(5-6)**, 693-701.

[320] Goldberg, D. E. (1989), *Genetic Algorithms in Search, Optimization and Machine Learning*, Reading: Addison-Wesley.

[321] Goldberg, D. E., & Richardson, J. (1987), "Genetic Algorithms with Sharing for Multimodal Function Optimization", in: *Genetic Algorithms and their Applications*, Proceedings of the Second International Conference on Genetic Algorithms, 41-49.

[322] Goldfarb, D. & Idnani, A. (1983): "A numerically stable dual method for solving strictly convex quadratic programs"; *Mathematical Programming*, **27**, 1-33.

[323] Goldfeld, S.M., Quandt, R.E., & Trotter, H.F. (1966), "Maximisation by quadratic hill-climbing", *Econometrica*, **34**, 541-551.

[324] Golub, G., & Reinsch, C. (1970), "Singular value decomposition and least squares solution", *Numerische Mathematik*, **14**, 403-420.

[325] Golub, G., & Van Loan, C.F. (1980), "An analysis of the total least squares problem", *SIAM Journal Numerical Analysis*, 883-893.

[326] Golub, G., & Van Loan, C.F. (1989), *Matrix Computations*, John Hopkins University Press, 2nd ed., Baltimore, MD.

[327] Gonin, R. & Money, A.H. (1989), *Nonlinear L_p-norm Estimation*, New York: M. Dekker, Inc.

[328] Goodall, C. (1983), "M-estimators of location: An outline of theory"; in: Hoaglin, D.C., Mosteller, F., & Tukey, J.W.: "Understanding Robust and Exploratory Data Analysis", New York: John Wiley & Sons.

[329] Goodman, L.A. (1965), "On simultaneous confidence intervals for multinomial proportions", *Technometrics*, **7**, 247-254.

[330] Goodman, L.A. (1985), "The analysis of cross-classified data having ordered and/or unordered categories: Association models, correlation models, and asymmetry models for contingency tables with or without missing entries", *The Annals of Statistics*, **13**, 10-69.

[331] Goodnight, J.H. (1979), "A Tutorial on the SWEEP Operator", *The American Statistician*, **33**, 149-158.

[332] Gould, N.I.M. (1986), "On the accurate determination of search directions for simple differentiable penalty functions", *IMA Journal of Numerical Analysis*, **6**, 357-372.

[333] Gould, N.I.M. (1989), "On the convergence of a sequential penalty function method for constrained minimization", *SIAM Journal Numerical Analysis*, **26**, 107-128.

[334] Gower, J.C. (1971), "A general coefficient of similarity and some of its properties", Biometrics, **27**, 857-871.

[335] Grassman, R., Gramacy, R.B., & Sterratt, D.C. (2011), *Package 'geometry'*; `http://geometry.r-forge.r-project.org/`.

[336] Graybill, F.A. (1969), *Introduction to Matrices with Applications in Statistics*, Belmont, CA: Wadsworth, Inc.

[337] Green, D. & Swets, J. (1996), *Signal Detection Theory and Psychophysics*, New York: John Wiley, 45-49.

[338] Greenbaum, A. (1997), *Iterative Methods for Solving Linear Systems*, Philadelphia: SIAM.

[339] Grubbs, F. E. (1969), "Procedures for detecting outlying observations in samples", *Technometrics*, **11**, 1-21.

[340] Gupta, A. & Avron, H. (2000, 2013), *WSMP: Watson Sparse Matrix Package*, *Part I - direct solution of symmetric systems*, *Part II - direct solution of general systems*, *Part III - iterative solution of sparse systems*, version 13.11, IBM Research Division, 1101 Kitchawan Road, Yorktown Heights, NY 10598 `http://www.research.ibm.com/projects/wsmp`

[341] Guttman, L. (1953), "Image theory for the structure of quantitative variates", *Psychometrika*, **18**, 277-296.

[342] Guttman, L. (1957), "Empirical Verification of the Radex Structure of Mental Abilities and Personality Traits", *Educational and Psychological Measurement*, **17**, 391-407.

[343] Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002), "Gene selection for cancer classification using support vector machines", *Machine Learning*, **46**, 389-422, 2002.

[344] Guyon, I., & Elisseeff, A. (2003), "An introduction to variable and feature selection", *Journal of Machine Learning Research*, **3**, 1157-1182, 2003.

[345] Hägglund, G. (1982), "Factor Analysis by Instrumental Variable Methods", *Psychometrika*, **47**, 209-222.

[346] Hahsler, M. & Hornik, K. (2013), "TSP - Infrastructure for the traveling salesperson problem", Technical Report, CRAN.

[347] Hald, A. (1952), *Statistical Theory with Engineering Applications*, New York: J. Wiley.

[348] Hald, J. & Madsen, K. (1981), "Combined LP and Quasi-Newton Methods for Minimax Optimization", *Mathematical Programming*, **20**, 49-62.

[349] Hald, J. & Madsen, K. (1985), "Combined LP and Quasi-Newton Methods for Nonlinear l_1 Optimization", *SIAM Journal Numerical Analysis*, **20**, 68-80.

[350] Hall, P. (1992), *The Bootstrap and Edgeworth Expansion*, New York: Springer-Verlag.

[351] Hamers, B., Suykens, J.A.K. & Vandevalle, J. (2002), "Compactly supported RBF kernels for sparsifying the Gram matrix in LS-SVM regression models", *Proceedings ICANN 2002*, Madrid, Spain; 720-726. Technical Report, Kath. University of Leeuven.

[352] Hamilton, J.D. (1994) *Time Series Analysis*, Princeton: Princeton University Press.

[353] Hanley, J.A. & McNeil, B.J. (1982), "The meaning and use of the area under a receiving operating characteristic (ROC) curve", *Radiology*, **143**, 29-36.

[354] Hansen, L.P. (1982), "Large Sample Properties of Generalized Method of Moments Estimators," *Econometrica*, **50**,1029-1054.

[355] Hansen, L.P. & Singleton, K.J. (1982), "Generalized Instrumental Variables Estimation of Nonlinear Rational Expectations Models," *Econometrica*, **50**,1269-1280.

[356] Hansen, P. R. (2005), "A test for superior predictive ability"; *Journal of Business and Economic Statistics*, **23**, 365-380. *Mathematical Programming*, **27**, 1-33.

[357] Harbison, S.P., & Steele, G.L. (1984), *A C Reference Manual*, Prentice Hall, Englewood Cliffs, NJ.

[358] Hardouin, J.-B. (2007), "Non parametric Item Response Theory with SAS and Stata"; *JSS*, 2007.

[359] Hardouin, J.-B. & Mesbah, M. (2007), "The SAS macro-program Item Response Theory Models"; *Communications in Statistics - Simulation and Computation*, **36**, 437-453.

[360] Harman, H.H. (1976), *Modern Factor Analysis*, Chicago: University of Chicago Press.

[361] Hartigan, J. A. (1975), *Clustering Algorithms*, New York: John Wiley & Sons.

[362] Hartmann, W. (1979), *Geometrische Modelle zur Analyse empirischer Daten*, Berlin: Akademie Verlag.

[363] Hartmann, W. (1991), *The NLP Procedure: Extended User's Guide*, Releases 6.08 and 6.10; SAS Institute Inc., Cary, N.C.

[364] Hartmann, W. (1992), *Nonlinear Optimization in IML*, Releases 6.08, 6.09, 6.10; Technical Report, Cary, N.C.: SAS Institute Inc.

[365] Hartmann, W. (1994), "Using PROC NLP for Risk Minimization in Stock Portfolios", Technical Report, Cary, N.C.: SAS Institute Inc.

[366] Hartmann, W. (1995), L_1 *Norm Minimization and Robust Regression - New in IML Release 6.11*, Technical Report, Cary, N.C.: SAS Institute Inc.

[367] Hartmann, W. (1995), *The CALIS Procedure: Extended User's Guide*, Release 6.11; SAS Institute Inc., Cary, N.C.

[368] Hartmann, W. (2003), "User Guide for PROC DMNEURL"; Technical Report, Cary: SAS Institute.

[369] Hartmann, W. (2005), "Resampling methods in structural equations", in: A. Maydeu-Olivares & J.J. McArdle: *Comtemporary Psychometrics (Festschrift for Roderick P. McDonald)*, Mahwah NJ: Laurence Erlbaum.

[370] Hartmann, W. (2007), "Tensor and List Operations in CMAT"; Technical Report, CMAT, Heidelberg.

[371] Hartmann, W. (2007), "Olvi's Matlab Algorithms in CMAT"; Technical Report, CMAT, Heidelberg.

[372] Hartmann, W. (2011), "Automatic Model Improvement usinf the `cfa` Function in CMAT"; Technical Report, CMAT, Heidelberg.

[373] Hartmann, W. (2015), "Difference in p values by PROC LOGISTIC and `elrm` in R"; Technical Report, CMAT, Heidelberg.

[374] Hartmann, W. (2015), "Exact logistic Regression for small Samples"; Technical Report, CMAT, Heidelberg.

[375] Hartmann, W. (2015), "Data Objects in CMAT"; Technical Report, CMAT, Heidelberg.

[376] Hartmann, W.M., & R.E. Hartwig (1996), "Computing the Moore-Penrose Inverse for the Covariance Matrix in Constrained Nonlinear Estimation", *SIAM Journal on Optimization*, **6**, p. 727-747.

[377] Hartmann, W. M., & So, Y. (1995), "Nonlinear Least-Squares and Maximum-Likelihood Estimation Using PROC NLP and SAS/IML", Computer Technology Workshop, American Statistical Association, Joint Statistical Meeting, Orlando, 1995.

[378] *Harwell Subroutine Library: Specifications*, Vol. 1 and 2 (Release 11), ed. 2, Harwell Laboratory, Oxfordshire, UK.

[379] Hastie, T., & Tibshirani, R., (2004), "Efficient quadratic regularization for expression arrays", *Biostatistics*, **5**, 329-340.

[380] Hawkins, D.M. (1994), "The feasible solution algorithm for least trimmed squares regression", *Computational Statistics and Data Analysis*, **17**, p. 185-196.

[381] Hawkins, D.M. (1994), "The feasible solution algorithm for the minimum covariance determinant estimator in multivariate data", *Computational Statistics and Data Analysis*, **17**, p. 197-210.

[382] Hawkins, D.M. & Kass, G.V. (1982), "Automatic Interaction Detection", in Hawkins, D.M. (ed.): *Topics in Applied Multivariate Analysis*, 267-302, Cambridge Univ Press: Cambridge.

[383] Hawkins, D.M., Bradu, D., & Kass, G.V. (1984), "Location of several outliers in multiple regression data using elemental sets", *Technometrics*, **26**, 197-208.

[384] Haymond, R.E., Jarvis, J.P. & Shier, D.R. (2013), "Algorithm 613: Minimum spanning tree for moderate integer weights", *ACM TOMS*, **10**, 108-110.

[385] Hedeker, D. (2012), "MIXREGLS: a Fortran program for mixed-effects location scale analysis", *JSS*.

[386] Hedeker, D. & Gibbons, R.D. (1996), "MIXOR: a computer program for mixed-effects ordinal regression analysis"; *Computer Methods and Programs in Biomedicine*, **49**, 157-176.

[387] Hedeker, D. (1999), "MIXNO: a computer program for mixed-effects nominal regression"; *Journal of Statistical Software*, **4**.

[388] Hedeker, D., Demirtas, H., & Mermelstein, R.J. (2009), "A mixed ordinal location scale model for analysis of Ecological Momentary Assessment (EMA) data"; *Statistics and its Interface*, **2**, 391-401.

[389] Hedeker, D., Mermelstein, R.J., & Demirtas, H. (2008), "An application of a mixed-effects location scale model for analysis of Ecological Momentary Assessment (EMA) data"; *Biometrics*, **64**, 627-634.

[390] Hedeker, D., Mermelstein, R.J., & Demirtas, H. (?), "Modeling between- and within-subject variance in Ecological Momentary Assessment (EMA) data using mixed-effects location scale models"; submitted.

[391] Hegger, R., Kantz, H. & Schreiber, T. (1999), "Practical implementation of nonlinear time series methods: The TISEAN package", *CHAOS*, **9**, 413

[392] Hemker, B.T., Sijtsma, K., & Molenaar, I.W. (1995), "Selection of unidimensional scales from a multidimensional item bank in the polytomous Mokken IRT model"; *Applied Psychological Measurement*, **19**, 337-352.

[393] Heiser, W. J. (1981), *Unfolding Analysis of Proximity Data*; Leiden: Karsten Druckers, PhD Thesis.

[394] Hellakalek, P. & Wegenkittel, S. (2003), "Empirical evidence concerning AES"; *ACM Transactions on Modeling and Computer Simulation*, **13**, 322-333.

[395] Helsgaun, K. (2000), "An effective implementation of the Lin-Kernighan Traveling Salesman Heuristic", *European Journal of Operational Research*, **126**, 106-130.

[396] Helsgaun, K. (2006), "An effective implementation of K-opt moves for the Lin-Kernighan TSP Heuristic", *Datalogiske Skrifter*, **109**, Roskilde University, Denmark.

[397] Henze, N. & Zirkler, B. (1990), "A class of invariant consistent tests for multivariate normality", *Communications in Statistics: Theory and Methods*, **19**,3595-3617.

[398] Herbrich, R. (2002), *Learning Kernel Classifiers: Theory and Algorithms*, Cambridge and London: MIT Press.

[399] Higham, N.J. (1988), "FORTRAN codes for estimating the one-norm of a real or complex matrix with applications to condition estimation", *ACM Trans. Math. Soft.*, **14**, pp. 381-396.

[400] Hirjim, K.F. (2006), *Exact Analysis of Discrete Data*, New York: Chapman and Hall, CRC.

[401] Hjorth, J.S.U. (1994), *Computer Intensive Statistical Methods*, London: Chapman & Hall.

[402] Hoaglin, D.C., Mosteller, F., & Tukey, J.W. (1983), "Understanding Robust and Exploratory Data Analysis", New York: John Wiley & Sons.

[403] Hochberg, Y. (1988), "A sharper Bonferroni procedure for multiple tests of significance", *Biometrika*, **75**, 800-803.

[404] Hochberg, Y. & Tamhane, A. C. (1987), *Multiple Comparison Procedures* New York: Wiley.

[405] Hochreiter, S., Clevert, D.-A., & Obermayer, K. (2006), "A new summarization method for affymetrix probe level data"; *Bioinformatics*, **22**, 943-949.

[406] Hock, W. & Schittkowski, K. (1981), *Test Examples for Nonlinear Programming Codes*, Lecture Notes in Economics and Mathematical Systems 187, Springer Verlag, Berlin-Heidelberg-New York.

[407] Hodrick, R. J. & Prescott, E. C. (1997), "Postwar US business cycles: An empirical investigation"; *Journal of Money, Credit, and Banking*, **29**, 1-16.

[408] Hoegaerts, L., Suykens, J.A.K., Vandewalle, J., & De Moor, B., (2003), "Kernel PLS variants for regression", in *Proc. of the 11th European Symposium on Artificial Neural Networks*, Bruges, Belgium; 203-208. Technical Report, Kath. University of Leeuven.

[409] Holm, S. (1979), "A simple sequentielle rejective multiple test procedure", *Scandinavian Journal of Statistics*, **6**, 65-70.

[410] Hommel, G. (1988), "A stagewise rejective multiple test procedure based on a modified Bonferroni test", *Biometrika*, **75**, 383-386.

[411] Horan, C. B. (1969), "Multidimensional scaling: Combining observations when individuals have different perceptual structures"; *Psychometrika*, **34**, 139-165.

[412] Horn, J. L. & Engstrom, R. (1979), "Cattell's scree test in relation to Bartlett's Chi-square test and other observations on the number of factors problem; *Multivariate Behavioral Research*, **14**, 283-300.

[413] Horn, R.A. & Johnson, C.R. (1985, 1996), *Matrix Analysis*, Cambridge University Press, Cambridge.

[414] Hoyer, P. O. (2004), "Non-negative matrix factorization with sparseness constraints"; *Journal of Machine Learning Research*, **5**, 1457-1469.

[415] Hsu, C.-W. & Lin, C.-J. (1999), "A Simple Decomposition Method for Support Vector Machines", Technical Report, Dept. of Computer Science and Information Engineering; National University of Taiwan.

[416] Huang, C. (2011), "SAS vs. R in data mining", `http://www.sasanalysis.com/2011/01/why-sas-is-more-useful-t`

[417] Huber, P. (1981), *Robust Statistics*, New York: John Wiley & Sons.

[418] Huber, P. J. (1985), "Projection pursuit", *The Annals of Statistics*, **13**, 435-475.

[419] Huber, W., Heydebreck, A.v., Sueltmann, H., Poustka, A., & Vingron, M. (2002), "Variance stabilization applied to microarray data calibration and to quantification of differential expression"; *Bioinformatics*, **18**, S96-S106.

[420] Huber, W., Heydebreck, A.v., Sueltmann, H., Poustka, A., & Vingron, M. (2003), "Parameter estimation for the calibration and variance stabilization of microarray data"; *Statistical Applications in Genetics and Molecular Biology*, **2**, No. 1, Article 3.

[421] Huddleston, H.F., Claypool, P.L., & Hocking, R.R. (1970), "Optimal sample allocation to strata using convex programming"; *Applied Statistics*, **19**, 273-278.

[422] Hyvaerinen, A., Karhunen, J. & Oja, E. (2001) *Independent Component Analysis*, New York: J. Wiley & Sons.

[423] Hwang, Y.T. & Wang, C.C. (2009), "Assessing multivariate normality based on Shapiro-Wilk test", *Journal of the Chinese Statistical Association*, **47**, 143-158.

[424] Iglewicz, B. (1983), "Robust scale estimators and confidence intervals for location"; in: Hoaglin, D.C., Mosteller, F., & Tukey, J.W.: "Understanding Robust and Exploratory Data Analysis", New York: John Wiley & Sons.

[425] James, L.R., Mulaik, S.A., & Brett, J.M. (1982), *Causal Analysis: Assumptions, Models, and Data*, Beverly Hills: Sage Publications, Inc.

[426] Janert, P. K. (2009), *Gnuplot in Action: Understanding data with graphs*, Greenwich CT: Manning Publications Co.

[427] Jarque, C. M. & Bera, A. K. (1987), "A test for normality of observations and regression residuals", *International Statistical Review*, **55**, 163-172.

[428] Jeffrey, D.J. (1997), "Formulae, algorithms, and quartic extrema"; *Mathematics Magazine*, **70**, pp. 349 - 356.

[429] Jenkins, M.A. & Traub, J.F. (1972), "Zeros of a complex polynomial", ACM.

[430] Jenkins, M.A. & Traub, J.F. (1975), "Principles for testing polynomial zerofinding programs", ACM TOMS, **1**, 26-34.

[431] Jennrich, R. I. (1973), "Standard errors for obliquely rotated factor loadings"; *Psychometrika*, **38**, 593-604.

[432] Jennrich, R. I. (1974), "Simplified formulae for standard errors in maximum likelihood factor analysis"; *British Journal of Math. and Statist. Psychology*, **27**, 122-131.

[433] Jennrich, R.I. (1987), "Tableau Algorithms for Factor Analysis by Instrumental Variable Methods", *Psychometrika*, **52**, 469-476.

[434] Jennrich, R.I. & Schluchter, M.D. (1986), "Unbalanced repeated-measures models with structured covariance matrices", *Biometrics*, **42**, 805-820.

[435] Jensen, D.R. & Ramirez, D.E. (1998), "Detecting outliers with Cook's D_I statistics", *Computing Science and Statistics*, **29(1)**, 581-586.

[436] Joachims, T. (1999), "Making large-scale SVM learning practical", in B. Schölkopf, C.J.C. Burges, and A.J. Smola (eds), *Advances in Kernel Methods: Support Vector Learning*, Cambridge: MIT Press.

[437] Joe, H. & Xu, J. (2007), "The estimation method of inference for margins for multivariate models"; Technical Report.

[438] Jöreskog, K.G. (1963), *Statistical Estimation in Factor Analysis*, Stockholm: Almqvist & Wicksell.

[439] Jöreskog, K.G. (1969), "Efficient estimation in image factor analysis", Psychometrika, **34**, 51-75.

[440] Jöreskog, K.G. (1973), "A general method for estimating a linear structural equation system", in *Structural Equation Models in the Social Sciences*, eds. A.S. Goldberger & O.D. Duncan, New York: Academic Press.

[441] Jöreskog, K.G. (1978), "Structural Analysis of Covariance and Correlation Matrices", *Psychometrika*, **43**, 443-477.

[442] Jöreskog, K.G. (1982), "Analysis of Covariance Structures", in *A Second Generation of Multivariate Analysis*, ed. C. Fornell, New York: Praeger Publishers.

[443] Jöreskog, K.G. & Sörbom, D. (1979), *Advances in Factor Analysis and Structural Equation Modeling*, Cambridge MA: Abt Books.

[444] Jöreskog, K.G. & Sörbom, D. (1988), *LISREL 7: A Guide to the Program and Applications*, SPSS Inc., Chicago, Illinois.

[445] Johnson, K., Mandal, A. & Ding, T. (2007), "Software for implementing the sequential elimination algorithm of level combination algorithm"; JSS 2007.

[446] Johnson, M. E. (1987), *Multivariate Statistical Simulation*, New York: John Wiley & Sons.

[447] Jonker, R. & Volgenant, A. (1983), "Transforming asymmetric into symmetric traveling salesman problems", *Operations Reseatrch Letters*, **2**, 161-163.

[448] Jonker, R. & Volgenant, A. (1987), "A shortest augmenting path algorithm for dense and sparse linear assignment problems", *Computing*, **38**, 325-340.

[449] Kahaner, D., Moler, C. & Nash, S. (1989), *Numerical Methods and Software*, Prentice Hall, Englewood Cliffs, NJ.

[450] Kantz, H. (1994), "A robust method to estimate the maximal Lyapunov exponent of a time series", *Phys. Lett.*, A 185, 77

[451] Kantz, H. & Schreiber, T. (2004), *Nonlinear Time Series Analysis*, 2nd edition, Cambridge: University Press.

[452] Kass, G.V. (1980), "An exploratory technique for investigating large quantities of categorical data", *Applied Statistics*, **29**, 119-127.

[453] Kaufman, L. & Rousseeuw, P.J. (1990), *Finding Groups in Data*, New York: John Wiley & Sons.

[454] Kay, S. M. & Marple, S. L. Jr. (1981), "Spectrum analysis - a modern perspective"; *Proceedings of the IEEE*, **69**, 1380-1419.

[455] Keerthi, S.S., Shevade, S.K., Bhattacharyya, C, & Murthy, K.R.K. (1999a), "A fast iterative nearest point algorithm for support vector machine classifier design", Technical Report TR-ISL-99-03, Department of CSA, Bangalore, India.

[456] Keerthi, S.S., Shevade, S.K., Bhattacharyya, C, & Murthy, K.R.K. (1999b), "Improvements of Platt's SMO algorithm for SVM classifier design", Technical Report CD-99-14, Dep. of Mechanical Production and Engineering, National University of Singapore.

[457] Kellerer, H., Pferschy, U., & Pisinger, D. (2004), *Knapsack Problems*, Berlin, Heidelberg: Springer Verlag.

[458] Kennedy, W.J. & Gentle, J,E. (1988), *Statistical Computing*, New York: Marcel Dekker.

[459] Kernighan, B.W., & Ritchie, D.M. (1978), *The C Programming Language*, Prentice Hall, Englewood Cliffs, NJ.

[460] Keuls, M. (1952), "The use of the studentized range in connection with an analysis of variance", *Euphytica*, **37**, 112-122.

[461] Kim, M. (1993), "Forecasting the Volatility of Financial Markets: ARCH/GARCH Models and the AUTOREG Procedure", in *Proceedings of the Eighteenth Annual SAS Users Group International Conference*, pp. 304-312, Cary: SAS Institute Inc.

[462] Kleinbaum, D.G. (1998), Kupper, L.L., Muller, K.E., & Nizam, A. (1998), *Applied Regression Analysis and Other Multivariate Methods*, North Scituate, MA: Duxbury Press.

[463] Klugkist, I., Laudy, O., & Hoijtink, H. (2005), "Inequality constrained analysis of variance, A Bayesian Approach", *Psychological Methods*, **10**, 477-493.

[464] Klugkist, I. & Hoijtink, H. (2007), "The Bayes factor for inequality and about equality constrained models", *Computational Statistics and Data Analysis*, **51**, 6367-6379.

[465] Knuth, D.E. (1986), *The TEXbook*, Seventh Printing, Addison-Wesley, Reading, MA.

[466] Knuth, D.E. (1973), *The Art of Computer Programming; Vol.1: Fundamental Algorithms*, *Vol.2: Seminumerical Algorithms*, *Vol.3: Sorting and Searching*, Addison-Wesley, Reading, MA.

[467] Kohonen, T. (2001), *Self-Organizing Maps*, Springer Verlag, Berlin.

[468] Kolda, T.G. & O'Leary, D.P. (1998), "A semidiscrete matrix decomposition for latent semantic indexing in information retrieval"; *ACM Trans. Inf. Syst.*, **16**, 322-346.

[469] Kolda, T.G. & O'Leary, D.P. (1999a), "Latent semantic indexing via a semi-discrete matrix decomposition", in *The Mathematics of Information Coding. Extraction and Distribution*, Vol 107 of the *IMA Volumes in Mathematics and Its Applications*, pp. 73-80, Springer Verlag.

[470] Kolda, T.G. & O'Leary, D.P. (1999b), *Computation and Uses of the Semidiscrete Matrix Decomposition*, Technical Report, Sandia National Laboratories, Livermore, CA.

[471] Kolmogorov, A. (1933), "Sulla determinazione empirica di una legge di distributione", *Giornale dell' Instituto Italiano degli Attuari*, **4**, 83-91.

[472] Krane, W.R. & McDonald, R.P. (1978), "Scale invariance and the factor analysis of correlation matrices", *British Journal of Mathematical and Statistical Psychology*, **31**, 218-228.

[473] Kreft, I. & de Leeuw, J. (1998), *Introducing Multilevel Modeling*, Beverly Hills, CA: SAGE Publications, Inc.

[474] Kruskal, J.B. (1964a), "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis", *Psychometrika*, **29**, 1-27.

[475] Kruskal, J.B. (1964b), "Nonmetric multidimensional scaling: A numerical method", *Psychometrika*, **29**, 115-129.

[476] Kruskal, J. B., Young, F. W. & Seery, J. B. (1978), "How to use KYST, a very flexible program to do multidimensional scaling and unfolding"; Technical Report, Murray Hill: Bell Laboratories.

[477] Kruskal, W.H. & Wallis, W.A. (1952), "Use of ranks in one-criterion variance analysis", JASA, **47**, 583-621.

[478] Kruskal, W.H. (1957), "Historical Note on the Wilcoxon unpaired two-sample test", JASA, **52**, 356-360.

[479] Kugiumtzis, D. (2002), "Surrogate Data Test for Nonlinearity using Statically Transformed Autoregressive Process", *Physical Review E*, **66**, 025201.

[480] Kugiumtzis, D. (2009), "Measures of analysis of time series toolkit (MATS)", paper submitted to JSS.

[481] Kuiper, R.M., Klugkist, I. & Hojtink, H. (2007), "A Fortran 90 Program for Confirmatory Analysis of Variance", *JSS* 422 .

[482] Kursa, M.B. & Rudnicki, W.R. (2010), "Feature Selecion with the Boruta Package"; *JSS*, 2010.

[483] Lambert, Z.V., Wildt, A.R. & Durand, R.M. (1990), "Assessing sampling variation relative to number-of-factors criteria"; *Educational and Psychological Measurement*, **50**, 253-257.

[484] Lambert, Z.V., Wildt, A.R. & Durand, R.M. (1991), "Approximating confidence intervals for factor loadings"; *Multivariate Behavioral Research*, **26**, 412-434.

[485] Lamport, L. (1986), *LaTeX - A Document Preparation System - User's Guide and Reference Manual*, Fifth Printing, Addison-Wesley, Reading, MA.

[486] "Using SAS GENMOD procedure to fit diagonal class models to square contingency tables having ordered categories"; *Proceedings of the Midwest SAS Users Group*, 149-160.

[487] Lawal, H.B. & Sundheim, R.A. (2002), "Generating factor variables for asymmetry, non-independence, and skew-symmetry models in square contingency tables using SAS", *JSS*, 2002.

[488] Lawless, J.F. (1982), *Statistical Models and Methods for Lifetime Data*, New York: John Wiley & Sons, Inc.

[489] Lawley, D.N. & Maxwell, A.E. (1971), *Factor Analysis as a Statistical Method*, New York: American Elsevier Publishing Company.

[490] Lawson, C.L. & Hanson, R.J. (1974), *Solving Least Squares Problems*, Englewood Cliffs, NJ: Prentice Hall.

[491] Lawson, C.L. & Hanson, R.J. (1995), *Solving Least Squares Problems*, Philadelphia, PA: SIAM.

[492] Lawson, C.L.,Hanson, R.J., Kincaid, D.R., & F.T. Krogh (1979), "Basic linear algebra subprograms for Fortran usage", *ACM Transactions on Mathematical Software*, **5**, pp. 308-323 and 324-325.

[493] L'Ecuyer, P. (1996a), "Combined Multiple Recursive Random Number Generators", *Operations Research*, **44**, 816-822.

[494] L'Ecuyer, P. (1996b), "Maximally equidistributed combined Tausworthe generators", *Math. of Comput.*, **65**, 203-213.

[495] L'Ecuyer, P. (1999), "Tables of maximally equidistributed combined LFSR generators", *Math. of Comput.*, **68**, 261-269.

[496] Ledoit, O. & Wolf, M. (2003a), "Improved estimation of the covariance matrix of stock returns with an application to portfolio selection"; *Journal of Empirical Finance*, **10**, 603-621.

[497] Ledoit, O. & Wolf, M. (2003b), "Honey I shrank the sample covariance matrix"; Technical Report.

[498] Lee, W. and Gentle, J.E. (1986), "The LAV Procedure", *SUGI Supplemental Library User's Guide*, Cary: SAS Institute, Chapter 21, pp. 257-260.

[499] Lee, D.D. & Seung, S. (1999), "Learning the parts of objects by non-negative matrix factorization"; *Nature*, **401**, 788-791.

[500] Lee, D.D. & Seung, S. (2001), "Algorithms for non-negative matrix factorization"; *Advances in Neural Information Processing Systems*, **13**, 556-562.

[501] Lee, J.Y., Soong, S.J., Shaw, H.M., Balch, C.M., McCarthy, W.H. & Urist, M.M. (1992), "Predicting survival and recurrence in localized melanoma: A multivariate approach"; *World Journal of Surgery*, **16**, 191-195.

[502] Lee, J., Yoshizawa, C., Wilkens, L., & Lee, H.P. (1992), "Covariance adjustment of survival curves based on Cox's proportional hazards regression model", *Comput. Appl. Biosc.*, **8**, 23-27.

[503] Lee, S.Y. (1985),"On Testing Functional Constraints in Structural Equation Models", *Biometrika*, **72**, 125-131.

[504] Lee, S.Y. & Jennrich, R.I. (1979), "A Study of Algorithms for Covariance Structure Analysis with Specific Comparisons Using Factor Analysis", *Psychometrika*, **44**, 99-113.

[505] Lee, Y., Lin, Y., & Wahba, G. (2003), *Multicategory Support Vector Machines, Theory and Application to the Classification of Microarray Data and Satellite Radiance Data*, Technical Report 1064, Dep. of Statistics, Univ. of Wisconsin, 2003.

[506] Lee, Y., Kim, Y., Lee, S. & Koo, J.Y. (2005), *Structured Multicategory Support Vector Machines with Analysis of Variance Decomposition*, Technical Report .

[507] Lee, Y.-Y. & Mangasarian, O. (1999), *SSVM: A smooth support vector machine*, Technical Report 99-03, Computer Sciences Dept., University of Wisconsin, Madison.

[508] Lee, Y.-Y. & Mangasarian, O. (2000), *RSVM: Reduced support vector machines*, Technical Report 00-07, Computer Sciences Dept., University of Wisconsin, Madison.

[509] Lehoucq, R.B., Sorensen, D.C., Yang, C. (1998), *ARPACK Users' Guide*, Philadelphia: SIAM.

[510] Leong, P.H.W., Zhang, G., Lee, D.U., Luk, W., & Villasenor, J.D. (2005), "A comment on the implementation of the ziggurat method"; JSS **12**.

[511] Lewis,J.G. (1977), "Algorithms for Sparse Matrix Eigenvalue Problems", Report STAN-CS-77-595, Computer Science Department, Stanford University, Stanford, California, March 1977.

[512] Li, K.C. (1991), "Sliced inverse regression for dimension reduction", *JASA*, **86**, 316-342.

[513] Li, K.C. (1992), "On principal Hessian directions for data visualization and dimension reduction: Another application of Stein's lemma", *JASA*, **87**, 1025-1034.

[514] Liaw, A. (2009), *randomForest* package in R.

[515] Liaw, A. & Wiener, M. (2002), "Classification and Regression by randomForest", *R News*, **2**, 3.

[516] Liang, K.Y. & Zeger, S.L. (1986) *Longitudinal data analysis using generalized linear models*, *Biometrika*, **73**, 13-22.

[517] Liebman, J.; Lasdon, L.; Schrage, L.; and Waren, A. (1986), *Modeling and Optimization with GINO*, California: The Scientific Press.

[518] Lin, C.-J. & Moré, J.J. (1999), "Newton's method for large bound constrained problems", *SIAM Journal of Optimization*, **9**, 1100-1127.

[519] Lin, S. (1965), "Computer solutions of the traveling-salesman problem", *Bell Systems Technology*, **44**, 2245-2269.

[520] Lin, S. & Kernighan, B. (1973), "An effective heuristic algorithm for the traveling-salesman problem", *Operations Research*

[521] Lin, Y., Lee, Y., & Wahba, G. (2000), *Support Vector Machines for Classification in Nonstandard Situations*, Technical Report 1016, Dep. of Statistics, Univ. of Wisconsin, 2000.

[522] Lindström, P. & Wedin, P.A. (1984), "A new linesearch algorithm for nonlinear least-squares problems", *Mathematical Programming*, **29**, 268-296.

[523] Liu, H & Wu, T. (2003), "Estimating the Area under a Receiver Operating Characteristic (ROC) Curve for Repeated Measures Design", *JSS*, 2003.

[524] Liu, J. W-H. (1985), "Modification of the minimum degree algorithm by multiple elimination", *ACM Trans. Math. Software*, **11**, 141-153.

[525] Liu, L., Hawkins, D.M., Ghosh, S., & Young, S.S. (2003), "Robust Singular Value Decomposition Analysis of Microarray Data"; *PNAS*, **100**, 13167-13172 (Nov. 11, 2003).

[526] Liu, W., Jamshidian, M., Zhang, Y., Bretz, F., & Han, X. (2007), "Some new methods for the comparison of two regression models", *Journal of Statistical Planning and Inference*, **137**, p. 57-67.

[527] Loevinger, J. (1948), "The technic of homogeneous tests compared with some aspects of scale analysis and factor analysis"; *Psychological Bulletin*, **45**, 507-529.

[528] Long, J.S. (1983), *Covariance Structure Models, an Introduction to LISREL*, Beverly Hills, CA: SAGE Publications, Inc.

[529] Long, J.S. (1997) *Regression Models for Categorical and Limited Dependent Variables*; Beverly Hills, CA: SAGE Publications, Inc.

[530] Lopuhaä, H.P. & Rousseeuw, P.J. (1991), "Breakdown Points of Affine Equivariant Estimators of Multivariate Location and Covariance Estimators", *The Annals of Statistics*, **19**, 229-248.

[531] Lord, F.M., & Novick, M.R. (1968), *Statistical theories of mental test scores*; Reading, MA: Addison-Wesley.

[532] Lucas, J. W. (2003), "Status processes and the institutionalization of women as leaders"; *American Sociological Review*, **68**, 464-480.

[533] Lüscher, M. (1994), "A portable high-quality random number generator for lattice field theory simulations"; *Computer Physics Communications*, **79**, 100-110.

[534] MacKinnon, D.P. (1992), "Statistical simulation in CALIS", *Proceedings of the 17th Annual SAS Users Group International Conference*, 1199-1203, Cary NC: SAS Institute Inc.

[535] Madsen, K. (1975), "Minimax Solution of Non-Linear Equations Without Calculating Derivatives", *Mathematical Programming Study*, **3**, 110-126.

[536] Madsen, K. & Nielsen, H.B. (1990), "Finite algorithms for robust linear regression", *BIT*, **30**, 682-699.

[537] Madsen, K. & Nielsen, H.B. (1993), "A finite smoothing algorithm for linear l1 estimation", *SIAM Journal on Optimization*, **3**, 223-235.

[538] Madsen, K., Nielsen, H.B., & Pinar, M.C. (1995), "A finite continuation algorithm for bound constrained quadratic programming", Technical Report IMM-REP-1995-22, Technical University of Denmark.

[539] Madsen, K., Nielsen, H.B., & Pinar, M.C. (1996), "A new finite continuation algorithm for linear programming", *SIAM Journal on Optimization*, **6**, 600-616.

[540] Madsen, K., Tingleff, O., Hansen, P.C., Owczarz, W. (1990), "Robust Subroutines for Non-Linear Optimization", Technical Report NI-90-06, Technical University of Denmark.

[541] Mair, P., de Leeuw, J., Borg, I., & Groenen, P.J.F. (2016), "Package smacof", Softwarepackage available in CRAN.

[542] Maiti, S. I. (2009), "Seasonal adjustment in SAS/BASE software - An alternative to PROC X11/X12"; JSS 442, 2009.

[543] Malinowski, E. R. (1991), *Factor Analysis in Chemistry*, New York: John Wiley & Sons.

[544] Malkiel, B. C. (1985), *A Random Walk Down Wall Street*, New York: Norton, Fourth Edition.

[545] Mandal, A., Wu, C.F.J., & Johnson, K. (2006), "SELC: Sequential elimination of level combinations by means of modified genetic algorithms"; *Technometrics*, **48(2)**, 273-283.

[546] Mandal, A., Johnson, K., Wu, C.F.J., & Bornemeier, D. (2007), "Identifying promising compounds in drug discovery: Genetic algorithms and some new statistical techniques"; *Journal of Chemical Information and Modeling*, **47(3)**, 981-988.

[547] Mangasarian, O.L. (1995), *Nonlinear Programming*, Philadelphia: SIAM.

[548] Mangasarian, O.L. & Musicant, D.R (1999), "Successive overrelaxion for support vector machines", *IEEE Transactions on Neural Networks*, **10**, 1032-1037.

[549] Mangasarian, O.L. & Musicant, D.R (2000a), "Active Support Vector Machine Classification", Technical Report 00-04, Data Mining Institute, University of Wisconsin, Madison, Wisconsin.

[550] Mangasarian, O.L. & Musicant, D.R (2000b), "Lagrangian Support Vector Machines", Technical Report 00-06, Data Mining Institute, University of Wisconsin, Madison, Wisconsin.

[551] Mangasarian, O.L. & Thompson, M.E. (2006), "Massive data classification via unconstrained support vector machines", *Journal of Optimization Theory and Applications.* Technical Report 06-07, Data Mining Institute, University of Wisconsin, Madison, Wisconsin.

[552] Mangasarian, O.L. & Thompson, M.E. (2006), "Chunking for massive nonlinear kernel classification", Technical Report 06-07, Data Mining Institute, University of Wisconsin, Madison, Wisconsin.

[553] Mangasarian, O.L. & Wild, E.W. (2004), "Feature Selection in k-Median Clustering", Technical Report 04-01, Data Mining Institute, University of Wisconsin, Madison, Wisconsin.

[554] Mangasarian, O.L. & Wild, E.W. (2006), "Feature Selection for Nonlinear Kernel Support Vector Machines", Technical Report 06-03, Data Mining Institute, University of Wisconsin, Madison, Wisconsin.

[555] Mann, H. & Whitney, D. (1947), "On a test whether one of two random variables is stochastically larger than the other"; *Annals of Mathematical Statistics*, **18**, 50-60.

[556] Mardia, K. V. (1970), "Measures of multivariate skewness and kurtosis with applications", *Biometrika*, **57**, 519-530.

[557] Mardia, K. V. (1974), "Applications of some measures of multivariate skewness and kurtosis in testing normality and robustness studies", *Sankhya: The Indian Journal of Statistics*, *Series B*, **36**, 115-128.

[558] Mardia, K. V. & Foster, K. (1983), "Omnibus tests of multinormality based on skewness and kurtosis", *Communications in Statistics*, *Part A: Theory and Methods*, **12**, 207-221.

[559] Mardia, K. V., Kent, J.T., & Bibby, J.M. (1979), *Multivariate Analysis*,

[560] Markowitz, H. (1952), "Portfolio Selection", *Journal of Finance*, **7**, 99-91.

[561] Markowitz, H. (1959), *Portfolio Selection: Efficient Diversification of Investments*, New York: John Wiley & Sons, Inc.

[562] Markus, M.T. (1994), *Bootstrap Confidence Regions in Nonlinear Multivariate Analysis*, Leiden: DSWO Press.

[563] Marple, S. L. Jr. (1991), "A fast computational algorithm for the modified covariance method of linear prediction"; *Digital Signal Processing*, **1**, 124-133.

[564] Marsaglia, G., "The Marsaglia Random Number CDROM", with "The Diehard Battery of Tests of Randomness", `www.stat.fsu.edu/pub/diehard`

[565] Marsaglia, G. (1965), "Ratios of normal variables and ratios of sums of uniform variables"; *JASA*, **60**, 193-204.

[566] Marsaglia, G. (2003), "Random number generators", *Journal of Modern Applied Statistical Methods*, **3**, .

[567] Marsaglia, G. (2003), "Xorshift RNGs", *JSS*, 2003.

[568] Marsaglia, G. (2004), "Evaluating the normal distribution function", JSS, 2004.

[569] Marsaglia, G. (2004), "Fast generation of discrete random variables", JSS, 2004.

[570] Marsaglia, G. (2006), "Ratios of normal variables"; *JSS*, 2006.

[571] Marsaglia, G., & Tsang, W.W. (1998), "The Monte Python method for generating Gamma variables", JSS **3**, 1998.

[572] Marsaglia, G., & Tsang, W.W. (2000), "The Ziggurat method for generating random variables", JSS **5**, 2000.

[573] Marsaglia, G. & Tsang, W.W. (2002), "Some difficult to pass tests of randomness", *JSS*, 2002.

[574] Marsaglia, G., Tsang, W.W., & Wang, J. (2003), "Evaluating Kolmogorov's Distribution", JSS, **8**, 2003.

[575] Marsaglia, G., Tsang, W.W., & Wang, J. (2004), "Fast generation of discrete random variables"; JSS, 2004.

[576] Marsaglia, G. & Tsay, L.H. (1985), "Matrices and the structure of random number sequences", *Linear Algebra and its Applications*, **67**, 147-156.

[577] Marsh, H.W., Balla, J.R., & McDonald, R.P. (1988), "Goodness-of-fit indices in confirmatory factor analysis. The effect of sample size", *Psychological Bulletin*, **103**, 391-410.

[578] Martello, S. (1983), "Algorithm 595: An enumerative algorithm for finding Hamiltonian circuits in a directed graph", *ACM TOMS*, **9**, 131-138.

[579] Martello, S. & Toth, P. (1990), *Knapsack Problems: Algorithms and Computer Implementations*, New York: Wiley & Sons.

[580] Mashtare, T. L., Jr. & Hutson, A. D. (2010), "SAS Macro for estimating the standard error of area under the curve with an application to receiver operating curves"; paper submitted to *JSS*.

[581] Masters, G.N. (1982), "A Rasch model for partial credit scoring"; *Psychometrika*, **47**, 149-174.

[582] *MATLAB Reference Guide* (1992), The MathWorks, Inc., Natick MA.

[583] Matsumoto, M. & Nishimura, T. (1998), "Mersenne-Twister: A 623-dimensionally equidistributed uniform pseudo-random number generator"; *ACM Transactions on Modeling and Computer Simultaion*, **8**, 3-30.

[584] May, W.L. & Johnson, W.D. (1997), "A SAS macro for constructing simultaneous confidence intervals for multinomial proportions", *Computer Methods and Programs in Biomedicine*, **53**, 153-162.

[585] May, W.L. & Johnson, W.D. (2000), "Constructing two-sided simultaneous confidence intervals for multinomial proportions for small counts in a large number of cells", *JSS*, 2000.

[586] Maydeu-Olivares, A. (2001), "Multidimensional IRT Modeling of Binary Data: Large Sample Properties of NOHARM Estimates", *J. of Educational and Behavioral Statistics*, **26**, 49-69.

[587] Maydeu-Olivares, A. (2005), "Further empirical results on parametric vs. non-parametric IRT modeling of Likert-type personality data"; *Multivariate Behavioral Research*, **40**, 275-293.

[588] Maydeu-Olivares, A. (2006), “Limited information estimation and testing of discretized multivariate normal structural models”; *Psychometrika*, **71**, 57-77.

[589] Maydeu-Olivares, A., Coffman, D.L. & Hartmann, W. (2007), “Asymptotically distribution-free (ADF) interval estimation of coefficient alpha”, *Psychological Methods*, **12**, 157-176.

[590] Maydeu-Olivares, A., & Joe, H. (2005), “Limited and full information information estimation and goodness-of-fit testing in 2^n contingency tables: A unified framework”; *JASA*, **100**, 1009-1020.

[591] Maydeu-Olivares, A., & Joe, H. (2006), “Limited information goodness-of-fit testing in multidimensional contingency tables”; *Psychometrika*, **71**, 713-732.

[592] McArdle, J.J. (1980), ”Causal Modeling Applied to Psychonomic Systems Simulation”, *Behavior Research Methods & Instrumentation*, **12**, 193-209.

[593] McArdle, J.J. (1988), ”Dynamic but Structural Equation Modeling of Repeated Measures Data”, in *The Handbook of Multivariate Experimental Psychology*, eds. J.R. Nesselroade and R.B. Cattell, New York: Plenum Press.

[594] McArdle, J.J. & Boker, S.M. (1986), *RAMpath - Path Diagram Software*, Denver: DATA Transforms, Inc.

[595] McArdle, J.J. & McDonald, R.P. (1984), “Some Algebraic Properties of the Reticular Action Model”, *Br. J. math. statist. Psychol.*, **37**, 234-251.

[596] MacCallum, R. (1986), “Specification Searches in Covariance Structure Modeling”, *Psychological Bulletin*, **100**, 107-120.

[597] McBane, G.C. (2006), “Programs to compute distribution functions and critical values for extreme value ratios for outlier detection”; *JSS*, 2006.

[598] McKean, J.W. and Schrader, R.M. (1987), ”Least absolute errors analysis of variance”, In: Y. Dodge, ed. *Statistical Data Analysis - Based on L_1 Norm and Related Methods*, Amsterdam: North Holland, 297-305.

[599] McDonald, R.P. (1978), “A Simple Comprehensive Model for the Analysis of Covariance Structures”, *Br. J. math. statist. Psychol.*, **31**, 59-72.

[600] McDonald, R.P. (1980), “A Simple Comprehensive Model for the Analysis of Covariance Structures: Some Remarks on Applications”, *Br. J. math. statist. Psychol.*, **33**, 161-183.

[601] McDonald, R.P. (1984), “Confirmatory Models for Nonlinear Structural Analysis”, in *Data Analysis and Informatics*, III, eds. E. Diday et al., North Holland: Elsevier Publishers.

[602] McDonald, R.P. (1985), *Factor Analysis and Related Methods*, Hillsdale NJ and London: Lawrence Erlbaum Associates.

[603] McDonald, R.P. (1989), ”An Index of Goodness-of-Fit Based on Noncentrality”, *Journal of Classification*, **6**, 97-103.

[604] McDonald, R.P. (1997), “Normal ogive multidimensional model”, in W.J. van der Linden and R. K. Hambleton (Eds.): *Handbook of Modern Item Response Theory*, pp. 257-269, New York: Springer.

[605] McDonald, R.P., & Hartmann, W. (1992), “A Procedure for Obtaining Initial Values of Parameters in the RAM Model”, *Multivariate Behavioral Research*, **27**, 57-76.

[606] McDonald, R.P. & Marsh, H.W. (1988), “Choosing a Multivariate Model: Noncentrality and Goodness of Fit”, Distributed Paper.

[607] McDonald, R.P. & Mok, M.C. (1995), "Goodness of fit in item response models", *Multivariate Behavioral Research*, **54**, 483-495.

[608] McDonald, R.P., Parker, P.M., & Ishizuka, T. (1993), "A scale-invariant treatment of recursive path models", *Psychometrika*, **58**, 431-443.

[609] McKean, J.W. & Schrader, R.M. (1987), "Least absolute errors analysis of variance", In: Y. Dodge, ed. *Statistical Data Analysis - Based on L_1 Norm and Related Methods*, Amsterdam: North Holland, 297-305.

[610] McLachlan, G.J. (1987), "On bootstrapping the likelihood ratio test statistic for the number of components in a normal mixture", *Applied Statistics*, **36**, 318-324.

[611] McLachlan, G.J. & Krishnan, T. (1997), *The EM Algorithm and Extensions*, New York: John Wiley & Sons.

[612] McMillan, G.P. & Hanson, T. (2005), "SAS Macro BDM for fitting the Dale (1984) regression model to bivariate mordinal response data"; JSS, 2005.

[613] McMillan, G.P., Hanson, T., Bedrick, E. & Lapham, S.C. (2005), "Application of the Bivariate Dale Model to Estimating Risk Factors for Alcohol Consumption Patterns", submitted.

[614] Meeker, W.Q. & Escobar, L.A. (1995), "Teaching about approximate confidence regions based on maximum likelihood estimation", *American Statistician*, **49**, 48-53.

[615] Mehta, C. R. & Patel, N. R. (1986), " Algorithm 643: FEXACT: A Fortran subroutine for Fisher's exact test on unordered $r * c$ contingency tables", *ACM Transactions on mathematical Software*, **12**, 154-161.

[616] Meulman, J. (1982) *Homogoneity Analysis of Incomplete Data*, Leiden: DSWO Press, 1982.

[617] Miha, A., Hayter, J., & Kuriki, S. (2003), "The evaluation of general non-centerd orthant probabilities", *British Journal of the Royal Statistical Society*, Ser. B, **65**, 223-234.

[618] Milan, L. & Whittaker, J. (1995), "Application of the parametric bootstrap to models that incorporate a singular value decomposition"; *Applied Statistics*, **44**, 31-49.

[619] Miles, R.E. (1959), "The complete amalgamation into blocks, by weighted means, of a finite set of real numbers", *Biometrika*, **46**, 317-327.

[620] Miller, A. (2002), *Subset Selection in Regression*, CRC Press, Chapman & Hall, 2nd Edition.

[621] Miller, A.J. (1990), *Subset Selection in Regression*, New York: Chapman and Hall.

[622] Mills T. C. (1990), *Time Series Techniques for Economists*, Cambridge: Cambridge University Press.

[623] *MKS Lex & Yacc*, (1993), Mortice Kern Systems Inc., Waterloo, Ontario CA.

[624] Mokken, R.J. (1971), *A Theory and Procedure of Scale Analysis*; DeGruyter.

[625] Mokken, R.J. (1997), "Nonparametric models for dichotomous responses"; in: W. J. van der Linden and R. K. Hambleton (eds): *Handbook of Modern Item Response Theory* New York: Springer.

[626] Molenaar, I.W. (1997), "Nonparametric models for polytomous responses"; in: W. J. van der Linden and R. K. Hambleton (eds): *Handbook of Modern Item Response Theory* New York: Springer.

[627] Molenberghs, G. & Lesaffre, E. (1994), "Marginal Modeling of Correlated Ordinal Data Using a Multivariate Plackett Distribution", *JASA*, **89**, 633-644.

[628] Momma, M. & Bennett, K.P. (2004), "Constructing orthogonal latent features for arbitrary loss"; Technical Report; Troy: Rensselaer Polytechnic Institute.

[629] Monahan, J.F. (2005), "Some algorithms for the conditional mean vector and covariance matrix"; JSS 2005.

[630] Moon, H., Lee, J.J., Ahn, H., & Nikolova, R.G. (2002), "Web-based simulator for sample size and power estimation in animal carcinogenicity studies", *JSS*, 2002.

[631] Moré, J.J. (1978), "The Levenberg-Marquardt Algorithm: Implementation and Theory", in *Lecture Notes in Mathematics 630*, ed. G.A. Watson, Springer Verlag, Berlin-Heidelberg-New York, 105-116.

[632] Moré, J.J., Garbow, B.S., & Hillstrom, K.E. (1980), *User Guide for MINPACK-1*, Argonne National Laboratory, Argonne, IL.

[633] Moré, J.J., Garbow, B.S., & Hillstrom, K.E. (1981), "Fortran Subroutines for Testing Unconstrained Optimization Software", *ACM Trans. Math. Software*, **7**, 17-41.

[634] Moré, J.J. & Sorensen, D.C. (1983), "Computing a Trust-Region Step", *SIAM Journal on Scientific and Statistical Computation*, **4**, 553-572.

[635] Moré, J.J. and Wright, S.J. (1993), *Optimization Software Guide*, Philadelphia: SIAM.

[636] Morgan, B.J.T. (1992), *Analysis of Quantal Response Data,* London: Chapman & Hall.

[637] Mosteller, F. & Tukey, J.W. (1977), *Data Analysis and Regression: A Second Course in Statistics*, Reading: Addison-Wesley.

[638] Möttönen, J. & Oja H. (1995), "Multivariate spatial sign and rank methods"; Journal of Noparametric Statistics, **5**, 201-213.

[639] Mudholkar, G. S., McDermott, M., & Srivastava, D. K. (1992), "A test of p-variate normality", *Biomerika*, **79**, 850-854.

[640] Mulaik, S.A. (1972) *The Foundations of Factor Analysis*, New York: McGraw Hill Comp.

[641] Mulaik, S.A., James, L.R., Van Alstine, J., Bennett, N., Lind, S., & Stilwell, C.D. (1989), "Evaluation of Goodness-of-Fit Indices for Structural Equation Models", *Psychological Bulletin*, **105**, 430-445.

[642] Murtagh, B. A. (1981), "Advanced Linear Programming: Computation and Practice", McGraw - Hill, New York.

[643] Murtagh, B.A. & Saunders, M.A. (1983), *MINOS 5.0 User's Guide*; Technical Report SOL 83-20, Stanford University.

[644] Muthèn, B.O. (1987), LISCOMP: *Analysis of Linear Structural Relations Using a Comprehensive Measurement Model*, Mooresville IN: Scientific Software, Inc.

[645] Nakaya, T. (1997), "Statistical inferences in bidimensional regression models", *Geographical Analysis*, **29**, 169-186.

[646] Narula, S.C. & Wellington, J.F. (1977), "Prediction, linear regression, and minimum sum of relative error", *Technometrics*, **19**, 185-190.

[647] Nazareth, J. L. (1987), "Computer Solutions of Linear Programs", Oxford University Press, New York - Oxford.

[648] Nelder, J.A. & Mead, R. (1965), “A Simplex Method for Function Minimization”, *Computer Journal*, **7**, 308-313.

[649] Nemhauser, G. L. & Wolsey, L. A. (1988), *Integer and Combinatorial Optimization*, New York: John Wiley & Sons

[650] Neter, J., Wasserman, W. & Kutner, M.H. (1990), *Applied Linear Statistical Models*, 3rd edition, Burr Ridge IL: R.D. Irwin.

[651] Nevalainen, J. & Oja, H. (2006), “SAS/IML macros for multivariate analysis of variance based on spatial signs”; *JSS*, 2004.

[652] Ng, E. & Peyton, B.W. (1993), “Block sparse Cholesky algorithms on advanced uniprocessor computers”, *SIAM J. Sci. Comput.*, **14**, 1034-1056.

[653] Nollau, V. & Hahnewald-Busch, A. (1975), *Statistische Analysen; Mathematische Methoden der Planung und Auswertung von Versuchen*, Leipzig: VEB Fachbuchverlag.

[654] Oberdiek, H. (1999), “PDF information and navigation elements with hyperref, pdfTeX, and thumbpdf”, Slides.

[655] Ogasarawa, H. (1998), “Standard errors for rotation matrices with an application to the promax solution”; *British Journal of Math. and Statist. Psychology*, **51**, 163-178.

[656] Ogasarawa, H. (1999), “Standard errors for Procrustes solutions”; *Japanese Psychological Research*, **41**, 121-130.

[657] Ogasarawa, H. (1999), “Stadard errors for the direct oblimin solution with Kaiser’s normalization”; *Japanese Journal of Psychology*, **70**, 333-338.

[658] Ogasarawa, H. (2000), *ROSEF: Version 2 User’s Guide*, Technical Report: Otaru University of Commerce, Otaru, Japan.

[659] Ogasarawa, H. (2000), “Standard errors for the principal component loadings for unstandardized and standardized variables”; *British Journal of Math. and Statist. Psychology*, **53**, 155-174.

[660] Ogasarawa, H. (2002), “Concise formulas for the standard errors of component loading estimates”; *Psychometrika*, **67**, 289-297.

[661] Osborne, M.R. (1985), *Finite Algorithms in Optimization and Data Analysis*, New York: John Wiley & Sons.

[662] Oja, H. & Randles, R.H. (2004), “Multivariate nonparametric tests”; *Statistical Science*, to appear.

[663] Osborne, M.R. (1985), *Finite Algorithms in Optimization and Data Analysis*, New York: J. Wiley.

[664] Outrata, J., Schramm, H., & Zowe, J. (1991), “Bundle Trust Region Methods: Fortran Codes for Nondifferentiable Optimization; User’s Guide”; Technical Report No. 269, Mathematisches Institut der Universität Bayreuth, 1991.

[665] Owen, “Tables for computing bivariate normal probabilities”, *Ann. Math. Statist.*, **27**, 1075-1090.

[666] Parlett, B.N. (1980), *The Symmetric Eigenvalue Problem*, Prentice Hall, Englewood Cliffs, NJ.

[667] Paige,C.C., & Saunders, M.A. (1975), “Solution of Sparse Indefinite Systems of Linear Equations”, *SIAM J. Numer. Anal.*, **12**, pp. 617-629.

[668] Paige,C.C., & Saunders, M.A. (1982), "LSQR: An algorithm for sparse linear equations and sparse least squares", *ACM Transactions on Mathematical Software* **8**, pp. 43-71.

[669] Paige,C.C., & Saunders, M.A. (1982), "Algorithm 583, LSQR: Sparse linear equations and least-squares problems", *ACM Transactions on Mathematical Software*, **8**, pp. 195-209.

[670] Pan, W. (2009), "A SAS/IML Macro for Computing Percentage Points of Pearson Distribution"; JSS 457, 2009.

[671] Pape, U. (1980), "Algorithm 562: Shortest Path Lengths", *ACM TOMS*, **6**, 450-455.

[672] Patefield, W.M. (1981), "Algorithm AS 159. An efficient method for generating $r * c$ tables with given row and columns totals", *Applied Statistics*, **30**, 91-97.

[673] Patefield, M. (2000), "Fast and Accurate Calculation of Owen's T Function", *JSS*, 2000.

[674] Pavlov, I. (2010), *Online Documentation for the 7-zip program*, `http://7-zip.org/7z.html`.

[675] *PCTEX32: User Manual*, Personal TEX, Inc., Mill Valley, CA.

[676] Pearson, E.S. & Hartley, H.O. (1972), *Biometrica Tables for Statisticians*, Vol II, New York: Cambridge University Press.

[677] Persson, P.O. & Strang, G. (2004), "A simple mesh generator in Matlab"; *SIAM Review*, **46**, 329-345.

[678] Piessens, R., de Doncker, E., Überhuber, C.W., & Kahane, D.K. (1983), *QUADPACK: A Subroutine Package for Automatic Integration*, Berlin: Springer Verlag.

[679] Pinar, M.C. & Elhedhli, S. (1998), "A penalty continuation method for the l_∞ solution of overdetermined linear systems"; BIT .

[680] Pinar, M.C. & Hartmann, W. (1999), "A Fast Huber Approximation Algorithm for Nonlinear ℓ_1 estimation", Sixth SIAM Conference on Optimization, Atlanta.

[681] Pinar, M.C. & Hartmann, W. (2006), "Huber Approximation Algorithm for Nonlinear ℓ_1 estimation", *European Journal of Operational Research*, **109**, 1096-1107.

[682] Pison, G., Rousseeuw, P.J., Filzmoser, P., & Croux, C. (2003), "Robust Factor Analysis"; *Journal of Multivariate Analysis*, **84**, 2003, 145-172.

[683] Platt, J. (1999), "Sequential Minimal Optimization: A fast algorithm for training support vector machines", in B. Schölkopf, C.J.C. Burges, and A.J. Smola (eds), *Advances in Kernel Methods: Support Vector Learning*, Cambridge: MIT Press.

[684] Pohlabeln, H., Wild, P., Schill, W., Ahrens, W., Jahn, I., Bohn-Audorff, U., Jöckel, K.H. (2002), "Asbestos fibreyears and lung cancer: A two-phase case-control study with expert exposure assessment", *Occupational and Environmental Medicine*, **59**, 410-414.

[685] Polak, E. (1971), *Computational Methods in Optimization*, New York, San Francisco, London: Academic Press, Inc.

[686] Powell, J.M.D. (1970) "A hybrid method for nonlinear equations", in *Numerical Methods for Nonlinear Algebraic Equations*, ed. by Rabinowitz; Gordon and Breach.

[687] Powell, J.M.D. (1977), "Restart Procedures for the Conjugate Gradient Method", *Mathematical Programming*, **12**, 241-254.

[688] Powell, J.M.D. (1978a), "A fast algorithm for nonlinearly constraint optimization calculations", in *Numerical Analysis, Dundee 1977, Lecture Notes in Mathematics 630*, ed. G.A. Watson, Springer Verlag, Berlin, 144-175.

[689] Powell, J.M.D. (1978b), "Algorithms for nonlinear constraints that use Lagrangian functions", *Mathematical Programming*, **14**, 224-248.

[690] Powell, M.J.D. (1982a), "Extensions to subroutine VF02AD", in *Systems Modeling and Optimization, Lecture Notes In Control and Information Sciences 38*, in: R.F. Drenick and F. Kozin (eds.), Springer Verlag, Berlin, 529-538.

[691] Powell, J.M.D. (1982b), "VMCWD: A Fortran subroutine for constrained optimization", *DAMTP 1982/NA4*, Cambridge, England.

[692] Powell, M.J.D. (1988), "A tolerant algorithm for linearly constrained optimization calculations", Report DAMTP/1988/NA17.

[693] Powell, J.M.D. (1992), "A Direct search optimization method that models the objective and constraint functions by linear interpolation", *DAMTP/NA5*, Cambridge, England.

[694] Powell, J.M.D. (2000), "UOBYQA: Unconstrained Optimization By Quadratic Approximation", Report DAMTP 2000/NA14, University of Cambridge.

[695] Powell, J.M.D. (2003), "On the use of quadratic models in unconstrained minimization without derivatives", Report DAMTP 2003/NA03, University of Cambridge.

[696] Powell, J.M.D. (2014), "On fast trust region methods for quadratic models with linear constraints", Report DAMTP 2014/NA02, University of Cambridge.

[697] Pregibon, D. (1981), "Logistic Regression Diagnostic", *Annals of Statistics*, **9**, 705-724.

[698] Prescott, P. (1975), "An approximate test for outliers in linear models", *Technometrics*, **17**, 129-132.

[699] Quinlan, J.R. (1993), *C4.5: Programs for Machine Learning*, Morgan Kaufman: San Mateo, CA.

[700] Rahtz, S. (1998), "Hypertext marks in LaTeX: the hyperref package", `http://www.tug.org`

[701] Ramirez, D. E. (2000), "The Generalized F Distribution", *JSS*, 2000.

[702] Ramirez, D.E. & Jensen, D.R. (1991), "Misspecified T^2 Tests. II Series Expansions", *Commun. Statist. - Simula.* **20**, 97-108.

[703] Ramsay, J. O. (1969), "Some statistical considerations in multidimensional scaling"; *Psychometrika*, **34**, 167-182.

[704] Ramsay, J. O. (1977), "Maximum likelihood estimation in multidimensional scaling"; *Psychometrika*, **42**, 241-266.

[705] Ramsay, J.O. (1978), "Confidence regions for multidimensional scaling analysis"; *Psychometrika*, **43**, 145-160.

[706] Ramsay, J.O. (1980), "Some small sample results for maximum likelihood estimation in multidimensional scaling"; *Psychometrika*, **45**, 139-144.

[707] Ramsay, J.O. (1982), "Some statistical approaches to multidimensional scaling"; *Journal of the Royal Statistical Society, Series A*, **145**, 285-312.

[708] Ränner, S., Lindgren, F. Geladi, P. & Wold, S. (1994), "A PLS algorithm for data sets with many variables and few objects", *Journal of Chemometrics*, **8**, 111-125.

[709] Rao, C.R. & Mitra, S.K. (1971), *Generalized Inverse of Matrices and Its Applications*, New York: John Wiley & Sons.

[710] Rasch, G. (1960), *Probabilistic models for some intelligence and attainment tests*; Danmarks Pedagogiske Institut, Copenhagen.

[711] Reed, B. C. (1989), "Linear least-squares fits with errors in both coordinates"; *American Journal of Physics*, **57**, 642-646.

[712] Reed, B. C. (1992), "Linear least-squares fits with errors in both coordinates II: Comments on parameter variances"; *American Journal of Physics*, **61**, 59-62.

[713] Reinelt, G. (1991), "TSPLIB - A Traveling Salesman Problem Library", *ORSA J. Comput.*, **3-4**, 376-385.

[714] Richardson, J.D. (2011), "Nonlinear pseudo random number generation using digit isolation and entropy buffers"; submitted to JSS, 2011.

[715] Rohlf, F. J. (1978), "A probabilistic minimum spanning tree algorithm", *Information Processing Letters*, **7**, 44-48.

[716] Rorabacher, D.B. (1991), "Statistical treatment for rejection of deviant values: Critical values of Dixon Q parameter and related subrange ratios at the 95 percent confidence level", *Analytical Chemistry*, **63**, 139-146.

[717] Rosenbrock, H.H. (1960), "An Automatic Method for Finding the Greatest or Least Value of a Function", *Computer Journal*, **3**, 175-184.

[718] Rosenkrantz, D. J., Stearns, R.E. & Lewis, P. M. (1977), "An analysis of several heuristics for the traveling salesman problem", *SIAM Journal on Computing*, **6**, 563-581.

[719] Rosenstein, M.T., Collins, J. J., De Luca, C. J. (1993), "A practical method for calculating largest Lyapunov exponents from small data sets", *Physica*, D 65, 117.

[720] Rosipal, R. & Trejo, L.J. (2001), "Kernel partial least squares regression in reproducing kernel Hilbert space", *Journal of Machine Learning*, **2**, 97-123.

[721] Rousseeuw, P.J. (1984), "Least Median of Squares Regression", *Journal of the American Statistical Association*, **79**, 871-880.

[722] Rousseeuw, P.J. (1985), "Multivariate Estimation with High Breakdown Point", in *Mathematical Statistics and Applications*, Dordrecht: Reidel Publishing Company, pp. 283-297.

[723] Rousseeuw, P.J. & Croux, C. (1993), "Alternatives to the Median Absolute Deviation", *Journal of the American Statistical Association*, **88**, 1273-1283.

[724] Rousseeuw, P.J. & Hubert, M. (1997), "Recent developments in PROGRESS", in: Y. Dodge (ed.): *L_1-Statistical Procedures and Related Topics*, Institute of Mathematical Statistics, Lecture Notes, Vol. 31.

[725] Rousseeuw, P.J. & Leroy, A.M. (1987), *Robust Regression and Outlier Detection*, New York: John Wiley & Sons.

[726] Rousseeuw, P.R. & Van Driessen, K. (1997), "A fast Algorithm for the Minimum Covariance Determinant Estimator", paper presented at the Third International Conference on the L_1 Norm and Related Methods, Neuchâtel, Switzerland.

[727] Rousseeuw, P.J. & Van Zomeren, B.C. (1990), "Unmasking Multivariate Outliers and Leverage Points", *Journal of the American Statistical Association*, **85**, 633-639.

[728] Royston, J. P. (1983), "Some techniques for assessing multivariate normality based on the Shapiro-Wilk *W*", *Applied Statistics*, **32**, 121-133.

[729] Royston, J. P. (1992), "Approximating the Shapiro-Wilk W test for non-normality", *Statistics and Computing*, **2**, 117-119.

[730] Rubin, H. & Johnson, B.C. (), "Efficient generation of exponential and normal deviates"; *Journal of Statistical Computation and Simulation*, **76**, 509-518.

[731] Rudd, A. and Rosenberg, B. (1979), "Realistic Portfolio Optimization", *TIMS Studies in Management Sciences*, **11**, 21-46.

[732] Saad, Y. (1996), *Iterative Metods for Sparse Linear Systems*, Boston: PWS Publishing Company.

[733] Sachs, L. (1974), *Angewandte Statistik*, Berlin, Heidelberg, New York: Springer Verlag.

[734] Samejima, F. (1969), "Calibration of latent ability using a response pattern of graded scores"; *Psychometrika Monograph Supplement*, No. **17**.

[735] Sarle, W. S. (1983), "Cubic Cluster Criterion"; *SAS Technical Report A-108*, Cary NC: SAS Institute Inc.

[736] Sarle, W. S. (1994), "TREEDISC Macro", Cary NC: SAS Institute Inc.

[737] Sarle, W. S. (1995), *The STDIZE SAS Macro*, Cary NC: SAS Institute Inc.

[738] Sarle, W.S. (2000), *The Jackboot Macro*, Cary NC: SAS Institute Inc.

[739] *SAS/IML®Software*, (1989), Version 6, First Ed., SAS Institute Inc., Cary, NC.

[740] *SAS/STAT®User's Guide*, (1990), Version 6, Second Printing, SAS Institute Inc., Cary, NC.

[741] *The SAS®System* (2000), Version 8, SAS Institute Inc., Cary, NC.

[742] SAS Enterprise Miner documentations for PROC ASSOC, SEQUENCE, RULEGEN, etc. version 9.1.3 and 4.3: `http://support.sas.com/documentation/onlinedoc/miner/emtmsas913/TW10113.pdf`
`http://support.sas.com/documentation/onlinedoc/miner/em43/assoc.pdf`
`http://support.sas.com/documentation/onlinedoc/miner/em43/rulegen.pdf`
`http://support.sas.com/documentation/onlinedoc/miner/em43/sequence.pdf`
`http://support.sas.com/documentation/onlinedoc/miner/em43/dmvq.pdf`
`http://support.sas.com/documentation/onlinedoc/miner/em43/neural.pdf`
`http://support.sas.com/documentation/onlinedoc/miner/em43/split.pdf`

[743] SAS Institute Inc. (1990), *SAS Language: Reference, Version 6, First Edition*, Cary, NC: SAS Institute Inc.

[744] *SAS Procedures Guide*, (1990), Version 6, Third Ed., SAS Institute Inc., Cary, NC.

[745] SAS Institute Inc. (1995), SAS Technical Report P-250, *SAS/IML®Software: Changes and Enhancements, through Release 6.11;* SAS Institute Inc., Cary, N.C.

[746] Sasieni, M., Yaspan, A., & Friedman, L. (1968), *Methoden und Probleme der Unternehmensforschung*, Berlin: Verlag Die Wirtschaft

[747] Satorra, A. & Bentler, P.M. (1994), “Corrections to test statistics and standard errors in covariance structure analysis”, in: *Latent Variables Analysis*, A. von Eye & C. C. Clogg (ed.), Thousand Oaks: Sage Publications.

[748] Satterthwaite, F. W. (1946), “An approximate distribution of estimates of variance components”; *Biometrics Bulletin*, **2**, 110-114,

[749] Schill, W., Enders, D., & Drescher, K. (2013), “sas-twophase-package: A SAS Package for logistic two-phase studies”, paper and software submitted to JSS. Software can be downloaded from: `www.bips.uni-bremen.de/sastwophase`

[750] Schittkowski, K. (1978), “An adaptive precision method for the numerical solution of constrained optimization problems applied to a time-optimal heating process”, in *Proceedings of the 8th IFIP Conference on Optimization Techniques*, Springer Verlag, Heidelberg, New York.

[751] Schill, W., Wild, P., & Pigeot, I. (2007), “A planning tool for two-phase case-control studies”, *Computer Programs and Methods in Biomedicine*, **88**, 175-181.

[752] Schittkowski, K. (1987), *More Test Examples for Nonlinear Programming Codes*, *Lecture Notes in Economics and Mathematical Systems 282*, Springer Verlag, Berlin-Heidelberg-New York.

[753] Schittkowski, K. & Stoer, J. (1979), “A Factorization Method for the Solution of Constrained Linear Least Squares Problems Allowing Subsequent Data Changes”, *Numer. Math.*, **31**, 431-463.

[754] Schmid, J. & Leiman, J.M. (1957), ”The development of hierarchical factor solutions”, *Psychometrika*, **22**, 53-61.

[755] Schnabel, R.B. & Eskow, E.A. (1990), *SIAM Journal on Scientific and Statistical Computation*, **11**, 1136-1158.

[756] Schölkopf, B. (2000), “Statistical Learning and Kernel Methods”, Technical Report MSR-TR-2000-23, Microsoft Research Limited, Cambridge UK.

[757] Schoenemann, P. H. (1972), “An algebraic solution for a class of subjective metrics models”; *Psychometrika*, **37**, 441-451.

[758] Schrage, L. (1979), “A more portable random number generator”, ACM TOMS **5** 132-138.

[759] Schreiber, T. and Schmitz, A. (1996), “Improved Surrogate Data for Nonlinearity Tests”, *Physical Review Letters*, **77**, 635-638.

[760] Schrepp, M. (1999), “On the empirical construction of implications on bi-valued test items”; *Mathematical Social Sciences*, **38**, 361-375.

[761] Schrepp, M. (2003), “A method for the analysis of hierarchical dependencies between items of a questionnaire”, *Methods of Psychological Research*, **19**, 43-79.

[762] Schrepp, M. (2006), *ITA 2.0: A program for classical and inductive item tree analysis*, JSS 2006.

[763] Searle, S.R. (1971), *Linear Models*, New York: John Wiley & Sons.

[764] Serneels, S., Filzmoser, P., Croux, C., & Van Espen, P.J. (2004), “Robust continuum regression”; Technical Report.

[765] Shaffer, J.P. (1995), “Multiple hypothesis testing”, *Annual Review of Psychology*, **46**, 561-576.

[766] Shanaz, F., Berry, M.W., Pauca, V.P. & Plemmons, R.J. (2004), “Document clustering using nonnegative matrix factorization”; *J. of Information Processing and Management*, to appear.

[767] Shao, J. & Tu, D. (1995), *The Jackknife and Bootstrap*, New York: Springer Verlag.

[768] Shapiro, S. S. & Wilk, M. B. (1965), "An analysis of variance test for normality (Complete Samples)", *Biometrika*, **52**, 591-611.

[769] Sharpe, W.F. (1987), "An algorithm for portfolio improvement", in: K.D. Lawrence, J.B. Guerard Jr, and G.R. Reeves (ed.), *Advances in Mathematical Programming and Financial Planning*, London: JAI Press Inc., 155-169.

[770] Shechter, G. (2004), *Matlab package kDtree*

[771] Shepard, R. N. (1962), "Analysis of proximities: Multidimensional scaling with an unknown distance function"; *Psychometrika*, **27**, 125-140, 219-246.

[772] Sheppard, K. (2009), *Oxford MFE Toolbox*, Oxford. `kevin.sheppard@economics.ox.ac.uk`

[773] Silvapulle, M. J. & Sen, P. K. (2005): *Constrained Statistical Inference*, New Jersey: Wiley.

[774] Simard, R. & L'Ecuyer, P. (2010), "Computing the Two-Sided Kolmogorov-Smirnov Distribution", Journal of Statistical Software.

[775] Simonetti, N. (1998), "Subroutines for dynamic program for the Traveling Salesman Problem", `www.contrib.andrew.cmu.edu/ neils/tsp/index.html`

[776] Sison, C.P. & Glaz, J. (1995), "Simultaneous confidence intervals and sample size determination for multinomial proportions", *JASA*, **90**, 366-369.

[777] Skibicki, M. & Wywial, J. (2000), "On optimal sample allocation in strata"; Technical Report, Dept. of Statistics, University of Economics, Katowice.

[778] Small, N (1980), "Marginal skewness and kurtosis in testing multivariate normality", *Applied Statistics*, **29**, 85-87.

[779] Small, N (1985), "Multivariate normality, testing for", in Kotz,, S., Johnson, N.L., & Read, C.B. (eds.), *Encyclopedia of Stastistical Sciences*, **6**, Amsterdam: North Holland.

[780] Smith, B.T., Boyle, J.M., Dongarra, J.J., Garbow, B.S., Ikebe, Y., Klema, V.C. & Moler, C.B. (1976), *Matrix Eigensystem Routines - EISPACK Guide*, *Lecture Notes in Computer Science*, Vol. 6, 2nd ed., Springer Verlag, Berlin.

[781] Sobel, R.A. (1982), "Asymptotic confidence intervals for indirect effects in structural equations", in: *Sociological Methodology*, S. Leinhardt (ed.), Washington, DC: American Sociological Association.

[782] Sobel, R.A. (1986), "Some new results on indirect effects and their standard errors in covariance structure models", in: *Sociological Methodology*, N.B. Tuma (ed.), Washington, DC: American Sociological Association.

[783] Somerville, P.N. (1997), "Multiple testing and simultaneous confidence intervals: calculation of constants", *Computational Statistics and Data Analysis*, **25**, 217-233.

[784] Somerville, P.N. (1998), "Numerical computation of multivariate normal and multivariate-t probabilities over convex regions", *Journal of Computational and Graphical Statistics*.

[785] Somerville, P.N. & Bretz, F. (2001), "FORTRAN 90 and SAS-IML programs for computation of critical values for multiple testing and simultaneous confidence intervals", *Journal of Statistical Software*.

[786] Spaeth, H. (1987), *Mathematische Software zur Linear Regression*, München: R. Oldenbourg Verlag GmbH.

[787] Srivastava, J. (1975), “Designs for searching non-negligible effects”, in: *A Survey of Statistical Design and Linear Models*, 507-519. Amsterdam: Elsevier, North Holland.

[788] Stadlober, E. (1989), “Ratio of uniforms as a convenient method for sampling from classical discrete distribution”, *Proceedings of the 21st conference on Winter simulation.*

[789] Stadlober, E. & Zechner, H. (1999), “The patchwork rejection technique for sampling from unimodal distributions”, *ACM Transactions on Modeling and Computer Simulation*, **9**, 59-80.

[790] Stefanski, L.A. & Cook, J. R. (1994), “Simulation-Extrapolation: The Measurement Error Jackknife”, JASA **90**, 1247-1256.

[791] Spelucci, P., & Hartmann, W. (1999), “A QR decomposition for matrix pencils”, *BIT*, **40**, 183-189.

[792] Steiger, J.H. & Lind, J.C. (1980), ”Statistically Based Tests for the Number of Common Factors”, paper presented at the annual meeting of the Psychometric Society, Iowa City, IA.

[793] Stiefel, E.L. (1963), *An Introduction to Numerical Mathematics*, New York: Academic Press.

[794] Stine, R. (1989), “An introduction to the bootstrap methods: Examples and ideas”, *Sociological Methods and Research*, **18**, 243-291.

[795] Stoppiglia, H., Dreyfus, G. Dubois, R.& Oussar, Y. (2003), “Ranking a random feature for variable and feature selection”, *Journal of Machine Learning Research*, **3**, 1399-1414.

[796] Stromberg, A.J. (1993), “Computation of high breakdown nonlinear regression parameters”, *JASA*, **88**, 237-244.

[797] Suykens, J.A.K. & Vandevalle, J. (1999), “Least squares support vector classifiers”, *Neural Processing Letters*, **9**, 293-300.

[798] Suykens, J.A.K., Lukas, L., Van Dooren, P., De Moor, B., & Vandewalle J. (1999), “Least squares support vector machine classifiers : a large scale algorithm”, in *Proc. of the European Conference on Circuit Theory and Design (ECCTD’99)*, Stresa, Italy; 839-842. Technical Report, Kath. University of Leeuven.

[799] Svetnik, V., Liaw, A., Tong, C. & Wang, T. (2004), “Application of Breiman’s random forest to modeling structure-activity relationships of pharmaceutical molecules”; in F. Roll, J. Kittler, & T. Windeatt (eds.): *MCS 2004*, LNCS 3077, 334-343.

[800] Swaminathan, H. (1974), ”A General Factor Model for the Description of Change”, Report LR-74-9, Laboratory of Psychometric and Evaluative Research, University of Massachusetts.

[801] Swarztrauber, P.N. (1982), “Vectorizing the FFT’s”, in Rodriguez (ed.) *Parallel Computations*, 51-83, New York: Academic Press.

[802] Szekely, G. J. & Rizzo, M. (2005), “A new test for multivariate normality,” *Journal of Multivariate Analysis*, **93**, 58-80.

[803] Tagliasacchi, A. (2008), *Matlab package kDtree*

[804] Takane, Y. (1977), “On the relations among four methods of multidimensional scaling”; *Behaviometrika*, **4**,29-42.

[805] Takane, Y., Young, F. W. & de Leeuw, J. (1977), “Nonmetric individual differences multidimensional scaling: An alternating least squares method with optimal scaling features”; *Psychometrika*, **42**, 7-67.

[806] Takane, Y., Young, F. W. & de Leeuw, J. (1980), “An individual differences additive model: An alternating least squares method with optimal scaling features”; *Psychometrika*, **45**, 183-209.

[807] Talebi, H. & Esmailzadeh, N. (2011a), "Using Kullback-Leibler distance for performance evaluation of search designs", *Bulletin of the Iranian Mathematical Society*, **37**, 269-279.

[808] Talebi, H. & Esmailzadeh, N. (2011b), "Weighted searching probability for classes of equivalent search design comparison", *Communication in Statistics: Theory and Methods*, **40**, 635-647.

[809] Tarjan, R.E. (1972), "Depth first search and linear graph algorithms", *SIAM Journal on Computing*, **1**, 146-160.

[810] Thayananthan, A. (2005) "Template based pose estimation and tracking of 3D hand motion", PhD Thesis, Dept. of Engineering, University of Cambridge.

[811] Theiler, J., Eubank, S., Longtin, A., Galdrikian, B. (1992a), "Testing for Nonlinearity in Time Series: the Method of Surrogate Data", *Physica D*, **58**, 77-94.

[812] Theiler, J. et al (1992b), "Using Surrogate Data to Detect Nonlinearity in Time Series"; in "Nonlinear Modeling and Forecasting", edt. Casdagli, M. & Eubank, S., Addison-Wesley, Reading, MA, 163-188.

[813] Therneau, T.M. & Grambsch, P.M. (2000), *Modeling Survival Data, Extending the Cox Model*, New York, Springer.

[814] Thissen, D. & Steinberg, L. (1986), "A taxonomy of item response models"; *Psychometrika*, **51**, 567-577.

[815] Thompson, R. (1985), "A note on restricted maximum likelihood estimation with an alternative outlier model"; *Journal of the Royal Statistical Society*, Ser. B, **47**, 53-55.

[816] Thurstone, L.L. (1931), "Multiple factor analysis"; *Psych. Rev.*, **38**, 406-427.

[817] Tibshirani, R. (1996), "Regression shrinkage and selection via the Lasso", *J. Royal Stat. Soc., Ser. B*, **58**, 267-288.

[818] Tipping, M. E. (2001), "Sparse Bayesian learning and the relevance vector machine", *The Journal of Machine Learning Research*, **1**, 211-244.

[819] Tobler, W.R. (1965), "Computation of the corresponding of geographical patterns", *Papers of the Regional Science Association*, **15**, 131-139.

[820] Tobler, W. R. (1966), "Medieval distortions: Projections of ancient maps", *Annals of the Association of American Geographers*, **56**, 351-360.

[821] Tobler, W. R. (1994), "Bidimensional regression", *Geographical Analysis*, **26**,187-212.

[822] Tomizawa, S. (1987), "Decomposition for 2-ratio-parameter symmetry model in square contingency tables with ordered categories"; *Biom. Journal*, **1**, 45-55.

[823] Torgerson, W. S. (1958), *Theory and Methods of Scaling*, New York: Wiley.

[824] Trindade, A. A. (2003), "Implementing modified Burg algorithms in multivariate subset autoregressive modeling", *JSS*, 2003.

[825] Trujillo-Ortiz, A., Hernandez-Walls, R., Barba-Rojo, K., & Castro-Perez, A. (2007a), "AnDartest: Anderson-Darling test for assessing normality of a sample data", `http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectID=14807`.

[826] Trujillo-Ortiz, A., Hernandez-Walls, R., Barba-Rojo, K., & Cupul-Magana, L. (2007b), "HXmvntest: Henze-Zirkler's multivariate normality test", `http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectID=17931`.

[827] Tucker, L.R. & Lewis, C. (1973), "A reliability coefficient for maximum likelihood factor analysis", *Psychometrika*, **38**, 1-10.

[828] Tukey, J.W. (1977a), *Data Analysis and Regression*, Reading: Addison-Wesley.

[829] Tukey, J.W. (1977b), *Exploratory Data Analysis*, Reading: Addison-Wesley.

[830] Tyler, D.E. (1987), "A distribution-free M-estimator of multivariate scatter", *The Annals of Statistics*, **15**, 234-251.

[831] Van der Voet, H. (1994), "Comparing the predictive accuracy of models using a simple randomization test", *Chemometrics and Intelligent Laboratory Systems*, **25**, 313-323.

[832] van der Vorst, H. (2000), *Iterative Methods for Large Linear Systems*, Utrecht: Utrecht University.

[833] Van Gestel T., Suykens J., De Brabanter J., De Moor B., & Vandewalle J., (2001), "Kernel Canonical Correlation Analysis and Least Squares Support Vector Machines", in *Proc. of the International Conference on Artificial Neureal Networks (ICANN 2001)*, Vienna, Austria; 381-386.

[834] Van Gestel, T., Suykens, J., Lanckriet, G., Lambrechts, A., De Moor, B., & Vandewalle, J., (2002), "Multiclass LS-SVMs : Moderated outputs and coding-decoding schemes", *Neural Processing Letters*, **15**, 45-48. Technical Report, Kath. University of Leeuven.

[835] Van Huffel, S. & Vandewalle, J. (1991), *The Total Least Squares Problem*, SIAM Philadelphia, PA.

[836] van Leeuwe, J. F. J. (1974), "Item Tree analysis", *Nederlands Tijdschrift voor de Psycholoie*, **29**, 475-484.

[837] Vansina F. & De Greve, J.P. (1982), "Close binary systems before and after mass transfer", Astrophys. Space Sci., **87**, 377-401.

[838] Vapnik, V.N. (1995), *The Nature of Statistical Learning*, New York: Springer.

[839] Venables, W. N., & Ripley, B. D. (1994), "Modern Applied Statistics with S-Plus"; New York: Springer.

[840] Venzon, D.J. & Moolgavkar, S.H. (1988), "A Method for Computing Profile-Likelihood-Based Confidence Intervals", *Applied Statistics*, **37**, 87-94.

[841] Vesanto, J., Himberg, J., Alhoniemi, E., & Parhankangas, J. (2000), "SOM Toolbox for Matlab 5"; Technical Report A57, Helsinki University of Technology,.

[842] Volgenant, A. & van den Hout, W. B. (1990), "TSP1 and TSP2 - Symmetric traveling salesman problem for personal computers", Technical Report with Borland Pascal implementation.

[843] Wahba, G. (1990), *Spline Models for Observational Data*, Series in Applied Mathematics, Vol. 59, SIAM Philadelphia.

[844] Wang, C.C. (2011), "TMVN: A Matlab package for multivariate normality test", JSS 2011.

[845] Wang, C.C. & Hwang, Y.T. (2011), "A new functional statistic for multivariate normality", *Statistics and Computing*, **21**, 501-509.

[846] Weber, E. (1972), *Grundriss der biologischen Statistik*, Jena: VEB Gustav Fischer Verlag, 1972.

[847] Weber, E. (1974), *Einführung in die Faktorenanalyse*, Jena: VEB Gustav Fischer Verlag, 1974.

[848] Wedin, P.A. & Lindström, P. (1987), *Methods and Software for Nonlinear Least Squares Problems*, University of Umea, Report No. UMINF 133.87.

[849] Wehrens, R. & Buydens, L. M. C. (2007), “Self and super-organizing maps in R: The Kohonen package”; *JSS*, **21**, 2007.

[850] Weisberg, S. (1980), *Applied Linear Regression*, New York: John Wiley & Sons.

[851] Weisberg, S. (2002), “Dimension reduction regression with R”, *JSS*, **7**, 2002.

[852] Weiss, A.A. (1984), ”ARMA Models with ARCH Errors,” *Journal of Time Series Analysis*, **5**, 129-143.

[853] Welch, B. L. (1947), “The generalization of ”Student’s” problem when several different population variances are involved”; *Biometrika*, **34**, 28-35

[854] Welch, B. L. (1951), “On the comparison of several mean values: an alternative approach”; *Biometrika*, **38**, 330-336.

[855] Welch, P. D. (1967), “The use of fast Fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms”; *IEEE Transactions on Audio Electroacoustics*, **AU-15(6)**, 70-73.

[856] Weston, J., Mukherje, S., Chapelle, O., Pontil, M., Poggio, T., & Vapnik, V. (2000), “Feature Selection for SVMs”, *Neural Information Processing Systems*, **13**, 668-674.

[857] Wheaton, B., Muthèn, B., Alwin, D.F., & Summers, G.F. (1977), ”Assessing Reliability and Stability in Panel Models”, in *Sociological Methodology*, ed. D.R. Heise, San Francisco: Jossey Bass.

[858] White, H. (2000), “A reality check for data snooping”; *Econometrica*, **68**, 1097-1126.

[859] Wiley, D.E. (1973), ”The Identification Problem for Structural Equation Models with Unmeasured Variables”, in *Structural Equation Models in the Social Sciences*, eds. A.S. Goldberger and O.D. Duncan, New York: Academic Press.

[860] Wilkinson, J.H. (1963), *Rounding Errors in Algebraic Processes*, Prentice Hall, Englewood Cliffs, NJ.

[861] Wilkinson, J.H. (1965), *The Algebraic Eigenvalue Problem*, Oxford University Press, Oxford.

[862] Wilkinson, J.H. & Reinsch, C. (1971), *Handbook for Automatic Computation*, Springer Verlag, Heidelberg.

[863] Willems, G., Pison, G., Rousseeuw, P.J. & Van Aelst, S. (2001), “A robust Hotelling test”, Technical Report, University of Antwerp, `http://win-www.uia.ac.be/u/statis`.

[864] Williams, E. (2014), “Aviation Formulary V1.46”, `http://williams.best.vwh.net/avform.htm`

[865] Wilson, E.B. & Hilferty, M.M. (1931), ”The Distribution of Chi-square”, *Proc. Nat. Acad. Sci.*, **17**, 694.

[866] Wold, S. (1994), “PLS for multivariate linear modeling”, in: *QSAR: Chemometric Methods in Molecular Design. Methods and Principles in Medicinal Chemistry*, ed. H. van de Waterbeemd, Weinheim: Verlag Chemie.

[867] Wold, S. (1996), “Estimation of principal components and related models by iterative least squares”, in: *Multivariate Analysis*, ed. P.R. Krisjhnaiah, New York: Academic Press, 391-420.

[868] Wright, S. J. (1997), *Primal-Dual Interior Point Methods*, SIAM, Philadelphia.

[869] Wright, S. P. (1992), “Adjusted p-values for simultaneous inference”, *Biometrics*, **48**, 1005-1013.

[870] Xie, X-J., Pendergast, J., & Clarke, W. (2008), “Increasing the power: A practical approach to goodness-of-fit test for logistic regression models with continuous predictors”; *Computational Statistics and Data Analysis*, **52**, 2703-2713.

[871] Xie, X-J. & Bian, A (2009), "A SAS Package for evaluating logistic and proportional odds model fit"; Submission to *JSS*.

[872] Yang, J. & Honavar, V. (1997), "Feature selection using a genetic algorithm", Technical Report, Iowa State University.

[873] Yates, F. (1960), *Sampling Methods Censuses and Surveys*, London: Griffin & Company Lth.

[874] Young, F. W. (1970), "Nonmetric multidimensional scaling: Recovery of metric information"; *Psychometrika*, **35**, 455-473.

[875] Young, F. W. (1975), "Methods of describing ordinal data with cardinal models", *Journal of Mathematical Psychology*, **12**, 416-436.

[876] Young, F. W. (1987), "Multidimensional scaling: History, Theory, and Applications"; Hillsdale NJ: Lawrence Erlbaum.

[877] Yuan, K.-H., Guarnaccia, C.A. & Hayslip Jr., B. (2003), "A study of the distribution of sample coefficient alpha with the Hopkins symptom checklist: Bootstrap versus asymptotics"; *Educational and Psychological Measurement*, **63**, 5-23.

[878] Zamar, D., Graham, J., & McNency, B. (2007), "elrm: Software implementing exact-like inference for logistic regression models"; *JSS*, **21**.

[879] Zamar, D., Graham, J., & McNency, B. (2013), "Package `elrm`", in CRAN.

[880] Zhang, H.H. (2006), "Variable selection for support vector machines via smoothing spline ANOVA," *Statistica Sinica*, **16(2)**, 659-674.

[881] Zhang, H.H., Ahn, J., Lin, X., & Park, C. (2006), "Gene selection using support vector machines with nonconvex penalty," *bioinformatics*, **22**, pp. 88-95.

[882] Zhang, X., Lobeiza, F.R., Zhang, M.J. & Klein, J.P. (2007), "A SAS Macro for estimation of direct adjusted survival curves based on a stratified Cox regression model"; *Computer Methods and Programme in Biomedicine*, **88**, 95-101.

[883] Zhu, C., Byrd, R.H., Lu, P., & Nocedal, J. (1994) "L-BFGS-B: FORTRAN Subroutines for Large Scale Bound Constrained Optimization", Tech. Report, NAM-11, EECS Department, Northwestern University, 1994.

[884] Zhu, J., Rosset, S., Hastie, T., & Tibshirani, R. (2003), "1-norm support vector machines," *Neural Inform. Processing Systems*, **16**, pp. 49-56.

[885] Ziff, R. M. (1998), "Four-tap shift-register-sequence random-number generators", *Computers on Physics*, **12**, 385-392.

[886] Zou, H., Hastie, T., & Tibshirani, R. (2004), "Sparse principal component analysis"; Technical Report, Stanford University.

Chapter 7

Index

Index