kSpectra Scripting

application *n* [inh. application; see also Standard Suite] :

ELEMENTS
contains datavectors, datamatrices, plots.

PROPERTIES
matrices (text, r/o) : Return list of names of all matrices
vectors (text, r/o) : Return list of names of all vectors
information (text, r/o) : Return the version
licensed (boolean, r/o) : Returns YES for a licensed copy

RESPONDS TO
open, save, loadvector, plotvector, deletevector, savevector, loadmatrix, deletematrix, savematrix, ssa, plotSSA, mssa, plotMSSA, fft, plotFFT, mem, plotMEM, mtm, plotMTM, getdata, matrixncol, matrixnrow, ssarc, plotSSArc, ssamodes, mtmrc, plotMTMrc, mtmmodes, mtmrc1, mssarc, plotMSSArc, plotMSSArcCh.

---

datavector *n* [inh. item] : This is data vector

ELEMENTS
contained by application.

PROPERTIES
name (text) : Vector name
length (integer) : Length of vector
contents (text, r/o) : Contents of this dataobject
comments (text) : Comments of this dataobject
type (type, r/o) : Type of the dataobject

RESPONDS TO
remove, store.

---

datamatrix *n, pl* **datamatrices** [inh. item] : This is data matrix

ELEMENTS
contained by application.

PROPERTIES
name (text) : Matrix name
rows (integer, r/o) : Number of rows
columns (integer, r/o) : Number of columns
type (type, r/o) : Type of the dataobject
comments (text) : Comments to this dataobject
contents (text, r/o) : Contents of this dataobject

RESPONDS TO
remove, store.

---

plot *n* [inh. item] : This is a plot object

ELEMENTS

contained by [application](#).

title (text) : Title of the Plot
xlabel (text) : x-label
ylabel (text) : y-label
xmax (real) : maximum for x-axis
xmin (real) : minimum for x-axis
ymin (real) : minimum for y-axis
ymax (real) : maximum for y-axis
xmargin (real) : xmargin
ymargin (real) : ymargin
nticx (integer) : Number of tics for X-axis
nticy (integer) : Number of tics for Y-axis
x1 (real) : transformation factor in X=X1*X+X2
x2 (real) : transformation factor in X=X1*X+X2

RESPONDS TO
[print](#), [store](#), [remove](#).

---

ssa *v* : do SSA analysis (available in licensed copy only)

ssa
  vector text : Name of vector dataobject to analyze
  [**basis** text] : Type of EOFs basis: can be 'data' for data based EOFs, or 'ar1'
  [**mcsurr** integer] : Number of AR(1) realizations for Monte-Carlo test
  [**window** integer] : SSA window size
  [**test** text] : Type of signficance test: can be of those types only: 'mc' (Monte Carlo),
    'chi2' Chi-squared, or 'default'
  [**spectrum** text] : Results: name of matrix dataobject with SSA spectra
  [**level** integer] : Confidence level ("0" is for 99%, "1" is for 95%, "2" is for 90%)
  [**cov** text] : Covariance method: "Burg" for Burg, "VG" for Vautard-Ghil, and "BK" for
    Broomhead-King
  [**unit** real] : sampling units
  [**modes** integer] : Number of SSA modes to retain
  → text : Name of matrix dataobject with SSA spectra

---

plotSSA *v* : Plot SSA results

plotSSA [text] : Name of matrix dataobject with SSA spectra

---

mem *v* : do MEM analysis

mem
  vector text : Name of vector dataobject to analyze
  [**order** integer] : MEM order
  [**spectrum** text] : Results: name of matrix dataobject with MEM spectra
  [**unit** real] : Sampling units
  [**freqn** integer] : Number of frequencies in spectral estimate
  → text : Name of matrix dataobject with MEM spectra

---

plotMEM *v* : Plot MEM results

    plotMEM [text] : Name of matrix dataobject with MEM spectra

---

mtm *v* : do MTM analysis (available in licensed copy only)

    mtm
        vector text : Name of vector dataobject to analyze
        [**resol** integer] : MTM resolution
        [**test** text] : Confidence test, can be 'red', 'white' or 'locwhite'
        [**spectrum** text] : Results: Name of matrix dataobject with MTM spectra
        [**level** integer] : Confidence level ("0" for 99%, "1" for 95%, "2" for 90%)
        [**unit** real] : sampling units
        → text : Name of matrix dataobjects with MTM spectra

---

plotMTM *v* : Plot MTM results

    plotMTM [text] : Names of matrix dataobject with MTM spectra

---

fft *v* : do Blackman-Tukey FFT analysis

    fft
        vector text : Name of vector dataobject to analyze
        [**window** integer] : Size of window
        [**shape** text] : window shape, can be only 'hamming', 'hanning' or 'bartlett'
        [**test** text] : Confidence test, can be 'none', 'ar1' or 'self'
        [**spectrum** text] : Results: name of matrix with Blackman-Tukey FFT spectra
        [**unit** real] : Sampling units
        [**freqn** integer] : number of frequencies in spectral estimate
        → text : Names of matrix dataobject with FFT spectra

---

plotFFT *v* : plot FFT results

    plotFFT [text] : Name of matrix dataobject with BT-FFT spectra

---

mssa *v* : do MSSA analysis (available in licensed copy only)

    mssa
        [**basis** text] : Type of EOFs basis: can be 'data' for data based EOFs, or 'ar1'
        [**pca** boolean] : YES - do pre-processing with PCA, NO - do not pre-processing with PCA
        [**mcsurr** integer] : Number of AR(1) realizations for Monte-Carlo test
        [**window** integer] : Size of window
        [**test** text] : Type of signficance test: can be 'mc' (Monte carlo), 'chi2' Chi-squared test, or 'none'
        matrix text : Name of matrix dataobject to analyze
        [**spectrum** text] : Results: name of matrix with MSSA spectra
        [**cov** text] : Type of covariance: "BK" for Broomhead-King, "VG" for Vautard-Ghil, and "Fast" for 'Reduced', see Help for more details
        [**level** integer] : Confidence level: (0 for 99%, 1 for 95%, 2 for 90%)
        [**unit** real] : sampling units

[**seofs** integer] : Number of Spatial EOFs to be retained after PCA pre-analysis (if applied)

→ text : Name of matrix dataobject with MSSA spectra

---

plotMSSA *v* : plot MSSA results

plotMSSA [text] : Name of matrix dataobject with MSSA spectra

---

open *v* : open a project file (.tkt)

open text : path to a file to open

---

save *v* : Save to a project file (available for a licensed copy only)

save text : path to a file where to save

---

loadvector *v* : load a vector

loadvector text : file to load
→ text

---

plotvector *v* : Plot vector

plotvector text : name of vector to plot

---

deletevector *v* : Delete vector

deletevector text : name of a vector to delete

---

savevector *v* : Save vector object to a file (available for a licensed copy only)

savevector text : name of the vector object to save
file text : path to the file to save
→ text

---

loadmatrix *v* : load a matrix

loadmatrix text : file to load
→ text

---

savematrix *v* : Save matrix object to a file (available for a licensed copy only)

savematrix text : name of the matrix object to save
file text : path to the file to save
→ text

---

deletematrix *v* : Delete a matrix

deletematrix text : name of a matrix to delete

---

print *v* : print a selected plot

    print [plot] : plot to print

---

store *v* : Save an object.

    store any : the object to save
        in text : Path to a file in which to save the object (plot, datavector or datamatrix)
        as text : Format to save: EPS or PDF for plots, ASCII for dataobjects

---

remove *v* : remove an object

    remove any

---

getdata *v* : get vector (or matrix) from another matrix

    getdata
        matrix text : Name of target matrix
        name text : Name of new vector (if 'col' is a number) or matrix (if 'col' is a list of numbers)
        col text : column(s) to get

---

matrixnrow *v* : get number of rows in a given matrix

    matrixnrow text : Name of matrix
        → integer

---

matrixncol *v* : get number of columns in a given matrix

    matrixncol text : Name of matrix
        → integer

---

ssarc *v* : Perform SSA reconstruction (available in licensed copy only)

    ssarc
        [**rcvec** text] : Results: Name of vector dataobject with reconstruction
        spectrum text : Input: name of matrix dataobject with SSA spectra
        rcs integer : List with components to reconstruct
        [**lead** integer] : Lead time to forecast
        [**order** integer] : Order of AR for forecasting
        → text : Name of vector dataobject with SSA reconstruction

---

plotSSArc *v* : Plot SSA reconstruction

    plotSSArc [text] : Name of vector dataobject with SSA reconstruction

---

ssamodes *v* : Get significant SSA modes (available in licensed copy only)

    ssamodes text : Name of matrix dataobject with SSA spectra
        → number : List of integers identifying significant SSA modes

mtmrc *v* : Perform MTM reconstruction (available in licensed copy only)

> **mtmrc**
>> [**rcvec** text] : Results: Name of vector dataobject with reconstruction
>> **spectrum** text : Input: name of matrix dataobject with MTM spectra
>> **sign** real : level of significance above which to reconstruct
>> → text : Name of vector dataobject with MTM reconstruction

---

plotMTMrc *v* : Plot MTM reconstruction

> **plotMTMrc** [text] : Name of vector dataobject with MTM reconstruction

---

mtmmodes *v* : Get significant MTM modes (available in licensed copy only)

> **mtmmodes**
>> **spectrum** text : Input: name of matrix dataobject with MTM spectra
>> **sign** real : level of significance above which to reconstruct
>> → integer : List of integers identifying significant MTM modes

---

mtmrc1 *v* : Perform MTM reconstruction (available in licensed copy only)

> **mtmrc1**
>> [**rcvec** text] : Results: Name of vector dataobject with reconstruction
>> **spectrum** text : Input: name of matrix dataobject with MTM spectra
>> **rcs** integer : List of integers indicating which MTM to reconstruct
>> → text : Name of vector dataobject with MTM reconstruction

---

mssarc *v* : Perform MSSA reconstruction (available in licensed copy only)

> **mssarc**
>> [**rcmat** text] : Results: Name of matrix dataobject with reconstructed components
>> **spectrum** text : Input: name of matrix dataobject with MSSA spectra
>> **rcs** integer : List with components to reconstruct
>> [**space** text] : Specifies the space where to reconstruct: "PCA" or "Grid".
>> **channel** integer : List of integers for channels to reconstruct
>> → text : Names of vector dataobjects for channels of reconstructed components

---

plotMSSArc *v* : Plot MSSA reconstruction

> **plotMSSArc**
>> **channel** integer : channel to plot against original data
>> **rcmat** text : name of matrix dataobject with MSSA reconstruction

---

plotMSSArcCh *v* : Plot channel of MSSA reconstruction against original data

> **plotMSSArcCh** text : Name of the vector with reconstruction

Standard Suite

Common classes and commands for most applications.

item *n* : A scriptable object.

PROPERTIES
class (type, r/o) : The class of the object.
properties (record) : All of the object's properties.

RESPONDS TO
count, delete, duplicate, exists, get, move, set.

---

application *n* [see also kSpectra Scripting] : An application's top level scripting object.

ELEMENTS
contains documents, windows.

PROPERTIES
name (text, r/o) : The name of the application.
frontmost (boolean, r/o) : Is this the frontmost (active) application?
version (text, r/o) : The version of the application.

RESPONDS TO
open, print, quit.

---

color *n* [see also Text Suite] : A color.

---

document *n* : A document.

ELEMENTS
contained by application.

PROPERTIES
path (text) : The document's path.
modified (boolean, r/o) : Has the document been modified since the last save?
name (text) : The document's name.

RESPONDS TO
close, print, save.

---

window *n* : A window.

ELEMENTS
contained by application.

PROPERTIES
name (text) : The full title of the window.
id (number, r/o) : The unique identifier of the window.
bounds (rectangle) : The bounding rectangle of the window.
document (document, r/o) : The document whose contents are being displayed in the window.
closeable (boolean, r/o) : Whether the window has a close box.
titled (boolean, r/o) : Whether the window has a title bar.
index (number) : The index of the window in the back-to-front window ordering.

floating (boolean, r/o) : Whether the window floats.
miniaturizable (boolean, r/o) : Whether the window can be miniaturized.
miniaturized (boolean) : Whether the window is currently miniaturized.
modal (boolean, r/o) : Whether the window is the application's current modal window.
resizable (boolean, r/o) : Whether the window can be resized.
visible (boolean) : Whether the window is currently visible.
zoomable (boolean, r/o) : Whether the window can be zoomed.
zoomed (boolean) : Whether the window is currently zoomed.

RESPONDS TO
close, print, save.

---

open *v* : Open an object.

  open file : The file(s) to be opened.

---

print *v* : Print an object.

  print file : The file(s) or document(s) to be printed.

---

quit *v* : Quit an application.

  quit
    [**saving** yes/no/ask] : Specifies whether changes should be saved before quitting.

---

close *v* : Close an object.

  close specifier : the object to close
    [**saving** yes/no/ask] : Specifies whether changes should be saved before closing.
    [**saving in** file] : The file in which to save the object.

---

count *v* : Return the number of elements of a particular class within an object.

  count specifier : the object whose elements are to be counted
    [**each** type] : The class of objects to be counted.
    → integer : the number of elements

---

delete *v* : Delete an object.

  delete specifier : the object to delete

---

duplicate *v* : Copy object(s) and put the copies at a new location.

  duplicate specifier : the object(s) to duplicate
    to location specifier : The location for the new object(s).
    [**with properties** record] : Properties to be set in the new duplicated object(s).

---

exists *v* : Verify if an object exists.

  exists specifier : the object in question
    → boolean : true if it exists, false if not

get *v* : Get the data for an object.

> **get** specifier
> → any

---

make *v* : Make a new object.

> **make**
>> **new** type : The class of the new object.
>> [**at** location specifier] : The location at which to insert the object.
>> [**with data** any] : The initial data for the object.
>> [**with properties** record] : The initial values for properties of the object.
>> → specifier : to the new object

---

move *v* : Move object(s) to a new location.

> **move** specifier : the object(s) to move
>> **to** location specifier : The new location for the object(s).

---

save *v* : Save an object.

> **save** specifier : the object to save, usually a document or window
>> [**in** file] : The file in which to save the object.
>> [**as** text] : The file type in which to save the data.

---

set *v* : Set an object's data.

> **set** specifier
>> **to** any : The new value.

Text Suite

A set of basic classes for text processing.

---

rich text *n, pl* **rich text** : Rich (styled) text

> ELEMENTS
> contains characters, paragraphs, words, attribute runs, attachments.

> PROPERTIES
> **color** (color) : The color of the first character.
> **font** (text) : The name of the font of the first character.
> **size** (integer) : The size in points of the first character.

---

character *n* : This subdivides the text into characters.

> ELEMENTS
> contains characters, paragraphs, words, attribute runs, attachments; contained by rich text, characters, paragraphs, words, attribute runs.

color ([color](#)) : The color of the first character.
font (text) : The name of the font of the first character.
size (integer) : The size in points of the first character.

---

paragraph *n* : This subdivides the text into paragraphs.

ELEMENTS
contains [characters](#), [paragraphs](#), [words](#), [attribute runs](#), [attachments](#); contained by [rich text](#), [characters](#), [paragraphs](#), [words](#), [attribute runs](#).

PROPERTIES
color ([color](#)) : The color of the first character.
font (text) : The name of the font of the first character.
size (integer) : The size in points of the first character.

---

word *n* : This subdivides the text into words.

ELEMENTS
contains [characters](#), [paragraphs](#), [words](#), [attribute runs](#), [attachments](#); contained by [rich text](#), [characters](#), [paragraphs](#), [words](#), [attribute runs](#).

PROPERTIES
color ([color](#)) : The color of the first character.
font (text) : The name of the font of the first character.
size (integer) : The size in points of the first character.

---

attribute run *n* : This subdivides the text into chunks that all have the same attributes.

ELEMENTS
contains [characters](#), [paragraphs](#), [words](#), [attribute runs](#), [attachments](#); contained by [rich text](#), [characters](#), [paragraphs](#), [words](#), [attribute runs](#).

PROPERTIES
color ([color](#)) : The color of the first character.
font (text) : The name of the font of the first character.
size (integer) : The size in points of the first character.

---

attachment *n* [inh. [rich text](#)] : Represents an inline text attachment. This class is used mainly for make commands.

ELEMENTS
contained by [rich text](#), [characters](#), [paragraphs](#), [words](#), [attribute runs](#).

PROPERTIES
file name (text) : The path to the file for the attachment