SOM Analyst v1 documentation »

next | modules | index

## SOM Analyst 1.0

Contents:

- Quick Start Guide
  - Install and Update
    - Direct Download
    - SVN Client
    - Standalone GUI
    - ArcGIS GUI
  - Updates
- Self-Organizing Maps
  - Introduction
    - Method
    - Overview of Parameters
      - Dimensions of the SOM
    - Characteristics of SOM Training
      - Run Length
      - Neighborhood Size
      - Learning Rate
    - Choosing Parameters
- SOM Analyst
  - System Requirements
  - Data Preprocessing
    - File Format Conversions
      - Conversion to Database File
        - ArcGIS Reference
        - Code Reference
      - Database File to SOM Data File
        - ArcGIS Reference
        - Code Reference
    - Data Management
      - Combine

- ArcGIS Reference
- Code Reference
- Select Data
  - ArcGIS Reference
  - Code Reference
- Value Transformations
  - Minimum-Maximum Normalization
    - ArcGIS Reference
    - Code Reference
  - Normalize by Variable
    - ArcGIS Reference
    - Code Reference
  - Z-Score Normalization
    - ArcGIS Reference
    - Code Reference
- SOM Computation
  - Create Initial SOM
    - ArcGIS Reference
    - Code Reference
  - Train SOM
    - ArcGIS Reference
    - Code Reference
  - Calculate Best Matching Units
    - ArcGIS Reference
    - Code Reference
  - Calculate U-Matrix
    - ArcGIS Reference
    - Code Reference
- SOM Visualization
  - SOM to Shapefile
    - ArcGIS Reference
    - Code Reference
  - Best Matching Unit to Shapefile
    - ArcGIS Reference

- Code Reference
- Group Shapes
  - ArcGIS Reference
  - Code Reference
- Utilities
  - Create Extent
    - ArcGIS Reference
    - Code Reference
  - Send Email
    - ArcGIS Reference
    - Code Reference
- Tutorial
  - System Requirements
  - Download
  - Adding the Toolbox
  - Convert Data Format
  - Normalize Data
  - Select Variables
  - Export Data
  - Create Initial SOM
  - Train SOM
  - Calculate U-Matrix
  - Project Data onto SOM
  - Create SOM Shapefile
  - Create Data Shapefile
  - Group Data Shapefile
  - Create Extent Shapefile
  - Visualization

## Indices and tables

- Index
- Search Page

SOM Analyst v1 documentation »

next | modules | index

SOM Analyst v1 documentation »

modules | index

## Python Module Index

-	

u uiBMUtoSHP **uiCODtoSHP** uiDBFmerge uiDBFtoDAT uiEmailMe uiExtent uiGroup uiMapInit uiNormMinMax uiNormVar uiNormZ uiSelect uiToXbase uiUmatrix uiVisual **uiVsom** 

SOM Analyst v1 documentation »

modules | index

SOM Analyst v1 documentation »

# Quick Start Guide

### Install and Update

There are two options for installing and updating SOM Analyst. The preferred method is to download the latest release from http://somanalyst.googlecode.com. The download is available on the download page, which can be reached by click on the download tab. The available downloads are compressed into ZIP format and must be unzipped before using, but require no additional setup. Updating SOM Analyst only requires downloading the latest release and unzipping it to the same folder as the older release, overwriting the old files.

The other option for downloading SOM Analyst is to use a SVN client to retrieve the repository from http://somanalyst.googlecode.com/svn/trunk/. Unlike the ZIP of SOM Analyst, the SOM Analyst SVN repository no longer includes the shapefile classes needed, which must also be downloaded and is available from http://shapefile.googlecode.com. Updating SOM Analyst can be preformed by using the SVN update command. Note the SVN repository is where revisions to SOM Analyst are immediately available, and as such might be unstable. If necessary revert to an earlier revision.

Python must be installed on the computer in order for SOM Analyst to work. If you do have Python download it from http://www.python.org and follow the instructions for installation.

#### Direct Download

This is the preferred method of installation.

- 1. Go to http://somanalyst.googlecode.com
- 2. Click on the Downloads tab

- 3. Select the desired release and download
- 4. Unzip the download to the desired location. (Usually short names without spaces works best for folders).
- 5. Updates can be performed by repeating this process.

### **SVN Client**

This method should only be chosen by developers and testers.

- 1. Create or choose a folder to use to checkout SOM Analyst. (Usually short names without spaces works best for folders).
- 2. Checkout the repository at http://somanalyst.googlecode.com/svn/trunk/ to the desired folder.
- 3. The folder now contains SOM Analyst, but requires an additional library.
- 4. The SOM Analyst folder contains a folder named lib, inside the lib folder create a folder called shp.
- 5. Checkout the repository at http://shapefile.googlecode.com/svn/trunk/ to the folder called shp.
- 6. Updates can be performed by calling the SVN update function on both the SOM Analyst folder and the shp folder.

#### Standalone GUI

The standalone GUI for SOM Analyst does not support all the features of the ArcGIS GUI and requires an additional library called WxPython, if you do not plan on using the standalone GUI you do not need it.

- 1. Go to http://www.wxpython.org/
- 2. Select the download link and choose the appropriate version for your system.

3. Follow the installation instructions for your system.

### ArcGIS GUI

The ArcGIS GUI was created with ArcGIS 9.3 and exported into additional formats for ArcGIS versions 9.0-9.2.

- 1. In ArcMap, from the Window menu select toolbox.
- 2. Right click the toolbox area and click on Add Toolbox.
- 3. Browse to the SOM Analyst folder and select the appropriate toolbox file (.tbx), then click open.
- 4. SOM Analyst Toolbox should now appear in the toolbox area. If desired right click the toolbox area and save the current setup.

### Updates

If you used the direct download installation method, repeat the installation process. If are using a SVN client see the instruction manual for your SVN client on how to perform a update. Remember when using SVN you must update both the SOM Analyst folder and the shp folder contained in its lib folder.

SOM Analyst v1 documentation »

SOM Analyst v1 documentation »

# Self-Organizing Maps

Contents:

- Introduction
  - Method
  - Overview of Parameters
    - Dimensions of the SOM
  - Characteristics of SOM Training
    - Run Length
    - Neighborhood Size
    - Learning Rate
  - Choosing Parameters

SOM Analyst v1 documentation »

SOM Analyst v1 documentation » Self-Organizing Maps » previous | next | modules | index

## Introduction

SOM Analyst is software written in the Python language and was created to make the use of self-organizing maps (SOMs) more accessible for users of geographic information systems (GIS). The broader impact of this work is to contribute to geographic information science, the science behind GIS, by incorporating a method that is usable with datasets that challenge current methods. These challenges include the sear volume of data as well as the number of variables that can cause some methods, or at least their implementation, to fail altogether or become unusable because of the amount of time need for computation. SOMs are by no means a cure all, they have several limitations and can be difficult to interpret. The following is a brief introduction to SOMs and a quick start guide to using SOM Analyst.

### Method

Self-organizing maps (SOMs) are artificial neural networks created using a method originally developed for signal processing. However, the use of this method is not limited to signal processing, it can be used for any dataset, especially those that contain many variables and/or instances. Essentially the method approximates an "optimal" spatial arrangement such that spatial distances are related to data similarity. As with all reductions of the number of independent variables distortions are produced. This means that spatial distances are consequently not strictly proportional to data similarity, however there is a tendency towards proportionality and as such can demonstrate qualities similar to geographic space. This is especially true since SOMs typically create two-dimensional spaces, although more dimensions and different shapes can be used. The space a SOM creates can have widely differing results from producing only clusters to producing only hierarchies and everything in between.

### **Overview of Parameters**

The results of a SOM depend on both the parameters specified by the user as well as qualities of the data. I will begin by defining each of the parameters of a SOM as well as some of the qualities of data that are important to keep in mind.

SOMs have the following variables:

- x-dimension: the resolution in the x direction
- y-dimension: the resolution in the y direction
- number of variables: the number of variables from the data
- topology: the arrangement of the units in the SOM
- neighborhood type: the relationship between units in the SOM

#### Dimensions of the SOM

The user has complete control of the x and y dimensions of the SOM and this greatly affects what kind of results will be achieved. It can be helpful to think of the size of the dataset and what you want the results to be. For example, if you are more interested in a classification you might choose to use 7 times as many units in the SOM as you have in your data. In this way, a perfectly even distribution would yield a single neuron for each of the input data with no adjacency. If you wanted to use the SOM more for clustering then you might only use 9 units in total. Once you have chosen the number of units, then the question of what ratio between the x and y dimensions is relevant. The easiest solution is to have both dimensions be the same, however varying their ratio changes the shape of the achievable topology. For example, if you were interested in a simple ordering you might choose one of the dimensions to only have a length of 1, which would result in a linear SOM and each unit would represent a place on a scale. Ultimately,

the decision must be based on the desired results and the compatibility of SOM topology with that of the data topology.

### **Characteristics of SOM Training**

SOM training has the variables: \* run length: the total number of times data is present to the SOM \* neighborhood size: the distance over which the training acts \* learning rate: the rate at which the training changes values

### Run Length

The run length of the training is the number of times data is presented to the SOM. For example, if your data had 10 vectors in it and you chose a run length of 100, each of the vectors would be presented 10 times. The vectors are presented in the order they appear in the data, so a run length of 102 would present the first two vectors more than the rest.

#### Neighborhood Size

The neighborhood size is the distance over which the neighborhood area is selected. This affects the resolution at which the data acts and is what directly defines area over which the lateral cohesion of the SOM forms. If the neighborhood size is too small, the SOM will result in little areas that are overspecialized surround by large areas that are not specialized. If the neighborhood size is too big then the SOM will be one large area that doesn't distinguish data into different classes. In many instances the SOM is first trained with a large neighborhood with smaller and trained a second time а neighborhood.

#### Learning Rate

The learning rate is the rate at which the SOM changes its values to correspond with the input data. A rate too high will produce areas

that are overspecialized. A rate too low will produce areas that are not specialized enough.

### **Choosing Parameters**

Determining the correct parameters for your data can be difficult to determine, however that are several guidelines that can be helpful. Small numbers of neurons produces clusters, while large numbers produce hierarchies. Small neighborhood sizes will yield spatially small clusters or hierarchies that can be widely dispersed, while large neighborhood sizes yield large clusters and little hierarchy. High learning rates yield a more specialized SOM, but this can also be overspecialized, while low learning rates yield a more generalized SOM that may have a clear structure.

SOM Analyst v1 documentation » Self-Organizing Maps » previous | next | modules | index

SOM Analyst v1 documentation »

## SOM Analyst

Contents:

- System Requirements
- Data Preprocessing
  - File Format Conversions
    - Conversion to Database File
      - ArcGIS Reference
      - Code Reference
    - Database File to SOM Data File
      - ArcGIS Reference
      - Code Reference
  - Data Management
    - Combine
      - ArcGIS Reference
      - Code Reference
    - Select Data
      - ArcGIS Reference
      - Code Reference
  - Value Transformations
    - Minimum-Maximum Normalization
      - ArcGIS Reference
      - Code Reference
    - Normalize by Variable
      - ArcGIS Reference
      - Code Reference
    - Z-Score Normalization
      - ArcGIS Reference
      - Code Reference
- SOM Computation
  - Create Initial SOM
    - ArcGIS Reference

- Code Reference
- Train SOM
  - ArcGIS Reference
  - Code Reference
- Calculate Best Matching Units
  - ArcGIS Reference
  - Code Reference
- Calculate U-Matrix
  - ArcGIS Reference
  - Code Reference
- SOM Visualization
  - SOM to Shapefile
    - ArcGIS Reference
    - Code Reference
  - Best Matching Unit to Shapefile
    - ArcGIS Reference
    - Code Reference
  - Group Shapes
    - ArcGIS Reference
    - Code Reference
- Utilities
  - Create Extent
    - ArcGIS Reference
    - Code Reference
  - Send Email
    - ArcGIS Reference
    - Code Reference

SOM Analyst v1 documentation »

SOM Analyst v1 documentation » SOM Analyst »

## System Requirements

- 1. Windows (any version)
- 2. ArcGIS 9.3 (legacy toolboxes for ArcGIS 9.0-9.2 are provided, but untested)
- 3. Python 2.5 (included in the default ArcGIS 9.3 installation)

SOM Analyst v1 documentation » SOM Analyst »

SOM Analyst v1 documentation » SOM Analyst »

## Data Preprocessing

Contents:

- File Format Conversions
  - Conversion to Database File
    - ArcGIS Reference
    - Code Reference
  - Database File to SOM Data File
    - ArcGIS Reference
    - Code Reference
- Data Management
  - Combine
    - ArcGIS Reference
    - Code Reference
  - Select Data
    - ArcGIS Reference
    - Code Reference
- Value Transformations
  - Minimum-Maximum Normalization
    - ArcGIS Reference
    - Code Reference
  - Normalize by Variable
    - ArcGIS Reference
    - Code Reference
  - Z-Score Normalization
    - ArcGIS Reference
    - Code Reference

SOM Analyst v1 documentation » SOM Analyst »

SOM Analyst v1 documentation » SOM Analyst » Data

previous | next | modules | index

Preprocessing »

## **File Format Conversions**

#### Contents:

- Conversion to Database File
  - ArcGIS Reference
  - Code Reference
- Database File to SOM Data File
  - ArcGIS Reference
  - Code Reference

SOM Analyst v1 documentation » SOM Analyst » Data previous | next | modules | index Preprocessing »

SOM Analyst v1 documentation » SOM Analyst » Data

previous | next | modules | index

Preprocessing » File Format Conversions »

## **Conversion to Database File**

Converts various data file formats to the database file format.







### **ArcGIS Reference**

💲 1. Data File to Database File	_ 🗆 🗵	
input data file		
C:\somanalyst\dat\census.csv	<b>2</b>	
input file format		
Comma Separated Values (CSV)	•	
output database		
C:\somanalyst\dat\census.dbf		
detect data types (optional)		
OK Cancel Environments Show H	Help >>	

#### Parameters

input data file

The input data file.

input file format

The input file format.

output database file

The output database file.

detect data types

An optional mode that automatically detects the data type in a column.

### Code Reference

uiToXbase. toXbase(inName, inType, outName, detectTypes) Creates a DBF file from the input file

	inName The input filename.
Arguments :	inType The input file type.mro outName The ouput filename. detectTypes An optional mode that detects and sets the data types for each column in the output file.

SOM Analyst v1 documentation » SOM Analyst » Data previous | next | modules | index Preprocessing » File Format Conversions »
previous | next | modules | index

Preprocessing » File Format Conversions »

## Database File to SOM Data File

Creates a SOM data file from a database file.



### **ArcGIS Reference**

\$ 2. Database to SOM_PAK Data	- 🗆 ×
	<u> </u>
C:\somanalyst\dat\demographics.dbf	<b></b>
output SOM data	
C:\somanalyst\dat\demographics.dat	<b>1</b>
label columns (optional)	
Region	<b>_</b>
Division	
State	
Abbrev	
I Vear	
☐ 15 64 ◀	- -
Select All Unselect All Add Fi	eld 🖵
OK Cancel Environments Show	Help >>

### Parameters

input database file

The input database data file.

output SOM data file

The output SOM data in the SOM\_PAK format.

label columns

An optional mode that treats selected columns as labels.

## Code Reference

uiDBFtoDAT. **DBFtoDAT**(*inName*, *labels*, *outName*) Creates a SOM data file from the input datbase file.

	inName The input filename.
Arguments :	labels The column headers for the label columns.
	outName The output filename.

SOM Analyst v1 documentation » SOM Analyst » Data previous | next | modules | index Preprocessing » File Format Conversions »

previous | next | modules | index

Preprocessing »

# Data Management

Contents:

- Combine
  - ArcGIS Reference
  - Code Reference
- Select Data
  - ArcGIS Reference
  - Code Reference

SOM Analyst v1 documentation » SOM Analyst » Data

previous | next | modules | index

Preprocessing »

previous | next | modules | index

Preprocessing » Data Management »

# Combine

Combines multiple database files into a single database file.



### **ArcGIS Reference**

### Parameters

input database files

The input database files.

combine type

The way in which to combine the database files.

output database file

The output database file that contains all the input data.

detect data types

An optional mode that automatically changes the data types in a column based on the values it contains. This mode is usueful if the data will be processed further using mathmatical operations.

## Code Reference

uiDBFmerge.dbfmerge(ifile1, ifile2, ofile)

Creates a single DBF file that contains both input DBF files.

	ifile1 The first input DBF file.
Arguments :	ifile2 The second input DBF file.
	ofile The output DBF file.

SOM Analyst v1 documentation » SOM Analyst » Data previous | next | modules | index Preprocessing » Data Management »

previous | next | modules | index

Preprocessing » Data Management »

# Select Data

Creates a database file with the selected values.





## **ArcGIS Reference**

ද් Select	
input database file	<u> </u>
C:\somanalyst\dat\norm01.dbf	- <i>2</i>
selection type	
columns	-
output database file	
C:\somanalyst\dat\demographics.dbf	- 🖻 -
columns (optional)	
<ul> <li>✓ Region</li> <li>✓ Division</li> <li>✓ State</li> <li>✓ Abbrev</li> <li>✓ Year</li> <li>✓ Female</li> <li>✓ Male</li> </ul>	<b>A</b>
Under_15	-
I 15 01	
Select All Unselect All Add	l Field
	0
index step (optional)	
	1
stop index (optional)	
	-1
detect data types (optional)	Y
OK Cancel Environments She	ow Help >>

### Parameters

input database file

selection type

The way in which data will be selected.

output database file

The output databse file.

columns

The columns to include in the data selection.

start index

The index from which to begin the selection.

index step

The increment from the start index for the next selection.

stop index

The index at which to stop selection. Negative indicies are equivalent of indexing from the end.

detect data types

An optional mode that automatically changes the data types in a column based on the values it contains. This mode is usueful if the data will be processed further using mathmatical operations.

## Code Reference

uiSelect.select(inName, selectionType, outName, columns, start, step, stop, detectTypes)

Creates a DBF file with the rows from a DBF file that meet the selection criteria.

	inName
	The input DBF filename.
	selectionType
	The select type (inclulsion or exclusion).
	outName
	The ouput DBF filename.
	columns
	The columns to keep. If no columns are
Arguments	specified, all columns are kept.
:	start
	The start index for rows.
	step
	The step between idecies for rows.
	stop
	The stop index for rows.
	detectTypes
	An optional mode that detects and sets the
	data types for each column in the output file.

previous | next | modules | index

Preprocessing »

# Value Transformations

### Contents:

- Minimum-Maximum Normalization
  - ArcGIS Reference
  - Code Reference
- Normalize by Variable
  - ArcGIS Reference
  - Code Reference
- Z-Score Normalization
  - ArcGIS Reference
  - Code Reference

SOM Analyst v1 documentation » SOM Analyst » Data previous | next | modules | index

Preprocessing »

previous | next | modules | index

Preprocessing » Value Transformations »

# Minimum-Maximum Normalization

Creates a databasefile with values normalized from a minimum value to a maximum value.

### **ArcGIS Reference**

S Min-Max Normalization	_ 🗆 🗵
input database file	<u> </u>
C:\somanalyst\dat\normVar.dbf	<b></b>
normalize by	
column	<b>•</b>
output database file	
C:\somanalyst\dat\norm01.dbf	<i>⊯</i>
division by zero value (max=min)	
scale minimum to	
	0
scale maximum to	
	1
decimal places to round to	
13 III III III III III III III III III I	
columns to normalize (optional)	
I Female	
Male Male	
Under_15	
✓ 15_64	
65_Over	
Mm_Indian	
Asian	
Variate	
Select All Unselect All Add Fi	eld
OK Cancel Environments Show	Help >>

#### Parameters

input database file

The input database file with the data to normalize.

normalize by

The direction in the table to perform the normailzation.

output database file

The output database file that will contain the normalized data.

division by zero value (max=min)

The value to assign when the resulting normalization requires a division by zero, as is the case when the minimum is also the maximum.

scale minimum to

The minimum value for the normalized data.

scale maximum to

The maximum value for the normalized data.

decimal places to round to

The number of places to round the normalized data to.

columns to normalize

The columns to normalize. If no columns are selected normalization will be performed on all columns that are numeric.

## Code Reference

uiNormMinMax. **normalize**(*inName*, *outName*, *start*, *end*, *direction*, *minEqMax*, *fieldNames*, *decimalPlaces*)

Creates a DBF with minimum-maxiumum normalized values from an existing DBF.

	inName
	The input DBF filename.
	outName
	The ouput DBF filename.
	start
	The minimum value to contain in the output range.
	end
	The maximum value to caontain in the output range.
Arguments	direction
:	The direction in which to determine minimum and maximum values in the input range.
	minEqMax
	The value to assign if the minimum is equal to the maxium. (division by zero)
	fieldNames
	The fields on which to perform the normalization.
	decimalPlace
	The number of decimal places to which numbers should be rounded.

SOM Analyst v1 documentation » SOM Analyst » Data Preprocessing » Value Transformations » previous | next | modules | index

previous | next | modules | index

Preprocessing » Value Transformations »

# Normalize by Variable

Creates a database file with values normalized by a column.

### **ArcGIS Reference**

S Normalize by Variable
input database file
C:\somanalyst\dat\census.dbf
normalize by column
Population 🗨
output database file
C:\somanalyst\dat\normVar.dbf
division by zero value
0
decimal places to round to
6 <u> </u>
0 12
columns to normalize (optional)
Female
₩ 15_64
Man Indian
White
Select All Linselect All Add Field
<u> </u>
OK Cancel Environments Show Help >>

#### Parameters

input database file

The database file to be normalized.

normalize by column

The columns that contain the values by which to normalize other data.

output database file

The ouput database file that will contain the normalized values.

division by zero value

The value to assign when the resulting normalization requires a division by zero, as is the case when the column contains zeros.

decimal places to round to

The number of decimal places to round the normaized values to.

columns to normalize

The columns to normalize. If no columns are selected normalization will be performed on all columns that are numeric.

## Code Reference

uiNormVar.**normalize**(*inName*, *outName*, *fieldNames*, *normBy*, *zeroDivision*, *decimalPlaces*)

Creates a DBF with values normalized by a column from within an existing DBF.

	inName
	The input DBF filename.
	outName
	The ouput DBF filename.
	normBy
Arguments	The column for normalizing values.
:	zeroDivision
	The value to assign if their is a division by
	zero.
	decimalPlace
	The number of decimal places to which numbers should be rounded.

SOM Analyst v1 documentation » SOM Analyst » Data	previous   next   modules   index
Preprocessing » Value Transformations »	

previous | next | modules | index

Preprocessing » Value Transformations »

## **Z-Score Normalization**

Creates a database file with Z-score values.

### **ArcGIS Reference**

S Z-score Normalization	
input database file	<u></u>
C:\somanalyst\dat\normalized.dbf	
normalize by	
column	<b>_</b>
output database file	
C:\somanalyst\dat\Zscore.dbf	🖻 🖻
division by zero value (std. dev = 0) (optional)	
	0
decimal places to round to (optional)	
6	
0	12
	-
Vunder 15	
15 64	
Ø 65 Over	
Am_Indian	
✓ Asian	
Black	
Select All Unselect All	Add Field
OK Cancel Environments	Show Help >>

#### **Parameters**

input database file

The input database file.

normalize by

The column by which to normalize data.

output database file

The output database file.

division by zero value (std. dev = 0)

The value to assign if there is a division by zero.

decimal places to round to

The number of decimal places to which to round numbers.

columns

The columns to normalize. If no columns are selected, all columns will be normalized.

## Code Reference

uiNormZ. **normalize**(*inName*, *outName*, *direction*, *zeroDivision*, *decimalPlaces*, *fieldNames*)

Creates a DBF with Z-score normalized values from an existing DBF.

	inName
	The input DBF filename.
	outName
	The ouput DBF filename.
	direction
	The direction in which to determine minimum and maximum values in the input range.
Arguments	zeroDivision
:	The value to assign if their is a division by zero.
	decimalPlace
	The number of decimal places to which numbers should be rounded.
	fieldNames
	The fields on which to perform the
	normalization.

SOM Analyst v1 documentation » SOM Analyst » Data	previous   next   modules   index
Preprocessing » Value Transformations »	

previous | next | modules | index

# **SOM Computation**

Contents:

- Create Initial SOM
  - ArcGIS Reference
  - Code Reference
- Train SOM
  - ArcGIS Reference
  - Code Reference
- Calculate Best Matching Units
  - ArcGIS Reference
  - Code Reference
- Calculate U-Matrix
  - ArcGIS Reference
  - Code Reference

SOM Analyst v1 documentation » SOM Analyst »

previous | next | modules | index
SOM Analyst v1 documentation » SOM Analyst » SOM

previous | next | modules | index

Computation »

## **Create Initial SOM**

Creates an initial SOM codebook file.

### **ArcGIS Reference**

🖇 1. Create Initial SOM	- 🗆 🗵
data for SOM	A
C:\somanalyst\dat\demographics.dat	<b>2</b>
topology type of map	
hexa	<b>T</b>
neighborhood type	
gaussian	•
x dimension	
	25
y dimension	
	25
initial SOM	
C:\somanalyst\dat\init.cod	
initilization type	
rand	-
random generator seed (optional)	
read buffer (optional)	
OK Cancel Environments Show	Help >>

#### **Parameters**

data for SOM

The input data for the SOM. Need to determine the number of dimensions, and if using linear (lin) initialization the appropriate seed values.

topology type of map

The topology (geometry) of the self-organizing map. Hexagonal (hexa) has been preselected because the center points are all equidistant. Rectangular (rect) has a horizontal and vertical bias because diagonals are farther apart.

neighborhood type

The type of neighborhood or relationship between units in the SOM. Gaussian is selected by default, because it is most typical.

x dimension

The number of units of the SOM along the X axis.

y dimension

The number of units in the SOM along the Y axis.

initial SOM

The ouput file for the intial SOM.

initilization type

The initialization type to use for the some. Random (rand) has been selected, because it is more consistent with the method of self-organization, however linear (lin) will produce results more quickly.

random generator seed

The seed to use for the random number generator, by default the current time is used.

read buffer

The number of data lines to read at a time.

### Code Reference

uiMapInit.mapinit(din, cout, topol, neigh, xdim, ydim, init, rand='#', buffer='#')

Creates a SOM codebook file using the specified parameters.

<b>Note:</b> This function makes system calls to SOM_PAK. SOM_PAK is limited to <i>non-commercial</i> use.		
Arguments :	din The training data. cout The codebook to create. topol The topology of the SOM. This can be hexagonal (hexa) or rectagular (rect). neigh The neighborhood type. This can be bubble or Gaussian. xdim The number of units in the X-axis of the SOM. ydim The number of units in the Y-axis of the SOM. init The initation type. This can be random or linear. rand (optional) The random seed. This is the current time by default. buffer (optional)	

The size of the read buffer.

#### Usage :

>>>

See also: The Self-Organizing Map Program Package Documentation for SOM\_PAK.

SOM Analyst v1 documentation » SOM Analyst » SOM previous | next | modules | index Computation » SOM Analyst v1 documentation » SOM Analyst » SOM

previous | next | modules | index

Computation »

## Train SOM

Trains a SOM.

### ArcGIS Reference

2. Train SOM			_ 🗆
initial SOM			
C:\somanalyst	dat\init.cod		
training data			
C:\somanalyst	dat\demograp	hics.dat	
length of training	9		
			4900
initial learning ra	te		
I			0.04
initial neighborho	ood radius		
Later d SOM			25
trained SOM	1.11.1		
C:\somanalyst	dat\stage1.co	bd	
distance metric (	(optional)		
Euclidean		P	
random generat	or seed (optio	nal)	
fixed points	(optional)		
use weights (op	tional)		
read buffer (opt	ional)		
alaba tuna (anti	Allena		
linear	orial)		<b>_</b>
snapshot file (or	tional)		
shanshot interva	al (ontional)		

#### Parameters

initial SOM

The initial SOM to train.

training data

The data with which to train the SOM.

length of training

The length of training as measured by the total number of

individual data row exposures.

initial learning rate

The initial learing rate for the SOM.

initial neighborhood radius

The initial distance over which a neighborhood is defined.

trained SOM

The ouput trained SOM.

distance metric

The distance metric to use when training the SOM. The Euclidean distance metric uses SOM\_PAK as released from the Helsinki University of Technology. The Cosine distance metric uses SOM\_PAK as modified by Fareed Qaddoura.

random generator seed

The seed for the random number generator, by default the current time is used.

fixed points

Use fixed point qualifiers in the training.

use weights

Use weighting qualifiers in the training.

read buffer

The number of lines to read at a time.

alpha type

The learning rate function type. Linear (linear) is defined as initial alpha \* [1 - (current training step/run length)]. Inverse-time (inverse\_t) is defined as initial alpha \* [(run length/100)/((run length/100)+current training step)].

snapshot file

A series of files to contain the intermediate SOMs. If the name contains %d the current step number will used in the file name.

shapshot interval

The interval at which to take the snapshots.

### Code Reference

uiVsom. **vsom**(*cin*, *din*, *cout*, *rlen*, *alpha*, *radius*, *rand='#'*, *fixed='#'*, *weights='#'*, *buffer='#'*, *alpha\_type='#'*, *snapfile='#'*, *snapinterval='#'*, *path='\\bin\\SOM\_PAK\\'*)

Trains a SOM using the specified parameters.

	cin		
	The input codebook file.		
	din		
	The training data.		
	cout		
	The output codebook file.		
	rlen		
	The length of training.		
	alpha		
	The initial rate of learning.		
	radius		
	The initial radius of neighborhoods.		
	rand optional		
Arguments	A seed for the random number generator.		
:	fixed optional		
	Use fixed points.		
	weights optional		
	Use weights for variables.		
	buffer optional		
	The read buffer size in number of lines.		
	alpha_type		
	The type of alpha decrease either linear		

(default) or inverse time.

snapfile

The snapshot filename.

snapinterval

The interval between snapshots.

SOM Analyst v1 documentation » SOM Analyst » SOM previous | next | modules | index

Computation »

SOM Analyst v1 documentation » SOM Analyst » SOM

previous | next | modules | index

Computation »

## Calculate Best Matching Units

Creates a best matching unit file by projecting data onto a SOM.

### **ArcGIS Reference**

3. Project Data onto SOM	<u> </u>
SOM	
C:\somanalyst\dat\stage2.cod	<b>2</b>
data to project	
C:\somanalyst\dat\demographics.dat	<b>2</b>
projected data	
C:\somanalyst\dat\demographics.bmu	🖻 🖻
distance metric (optional)	
Eudidean	
skip masked data vectors (optional)	
read buffer (ontional)	
OK Cancel Environments Show	Help >>
· / / / / /	

#### Parameters

SOM

The SOM on which to project data.

data to project

The data to project onto the SOM.

projected data

The ouput file to contain the projected data.

distance metric

The distance metric to use in projecting the data.

skip masked data vectors

Skip vectors that have been masked. Masking is done by begining vectors with a "x", but this is not reccomended since SOM Analyst in several instances does matching based on line number.

read buffer

The number of lines to read at a time.

### Code Reference

uiVisual.**visual**(*cin*, *din*, *dout*, *noskip='#'*, *buffer='#'*, *path='\\bin\\SOM\_PAK\\'*)

Creates a BMU file by projecting data onto a SOM.

	cin The input codebook file.
	din The input data file.
Arguments :	The output BMU file. noskip <i>optional</i> Do not skip data vectors that have all components masked off. buffer <i>optional</i> A read buffer in number of lines size.

SOM Analyst v1 documentation » SOM Analyst » SOM previous | next | modules | index Computation » SOM Analyst v1 documentation » SOM Analyst » SOM

previous | next | modules | index

Computation »

## Calculate U-Matrix

Creates a datbasefile with the U-matrix values for a SOM.

### ArcGIS Reference

🖇 Calculate U-matrix	_ 🗆 🗵
input SOM	<u> </u>
C:\somanalyst\dat\stage2.cod	<b></b>
output U-matrix database file	
C:\somanalyst\dat\umatrix.dbf	<b>2</b>
distance metric (optional)	_
Euclidean	-
	7
OK Cancel Environments Show	Help >>

#### Parameters

input SOM

The input SOM for which to calculate the U-matrix.

output U-matrix database file

The output database file that will contain the U-matrix values.

distance metric

The distance metric to use to calculate the U-matrix.

### Code Reference

uiUmatrix. uMatrix(*inName*, *outName*, *decimalPlaces*, *metric*) Calculates the U-matrix for a SOM.

	inName The input SOM filename.
Arguments :	outName The ouput U-matrix filename.
	decimalPlaces The number of decimals to round to.

SOM Analyst v1 documentation » SOM Analyst » SOM previous | next | modules | index Computation » SOM Analyst v1 documentation » SOM Analyst »

previous | next | modules | index

## SOM Visualization

Contents:

- SOM to Shapefile
  - ArcGIS Reference
  - Code Reference
- Best Matching Unit to Shapefile
  - ArcGIS Reference
  - Code Reference
- Group Shapes
  - ArcGIS Reference
  - Code Reference

SOM Analyst v1 documentation » SOM Analyst »

previous | next | modules | index

SOM Analyst v1 documentation » SOM Analyst » SOM

previous | next | modules | index

Visualization »

## SOM to Shapefile

Creates a shapefile from a SOM codebook file.

### **ArcGIS Reference**

\$ 1. SOM to Shapefile	- 🗆 🗵
SOM	<u> </u>
C:\somanalyst\dat\stage2.cod	<b>2</b>
shape type	
polygon	
SOM shapefile	
C:\somanalyst\dat\stage2.shp	🗃 🔁
SOM data for variable names (optional)	
C:\somanalyst\dat\demographics.dat	<b>2</b>
<ul> <li>Iabel SOM with data labels (optional)</li> <li>U-matrix (optional)</li> <li>C:\somanalyst\dat\umatrix.dbf</li> </ul>	<b>₽</b>
Cartesian quadrant	
1	<b>•</b>
spacing between center points	
	1
OK Cancel Environments Show	Help >>

#### **Parameters**

SOM

The input SOM codebook file from which to create a shapefile.

shape type

The type of shapes to use for the SOM.

SOM shapefile

The output shapefile.

SOM data for variable names

The data that contains the variable names.

label SOM with data labels

Label the SOM with the values from data that is the best match.

Cartesian quadrant

The Cartesian coordinatein which to place the SOM.

spacing between center points

The distance between the ceneter points of units in the SOM.

### Code Reference

uiCODtoSHP. **CODtoSHP**(*inName*, *outName*, *shapeType*, *labelData*, *labelNeurons*, *radius*, *quadrant*, *umatrix=False*)

Creates a shapfile from a codebook file.

	inName The input codebook filename.
	outName
	shaneTyne
	The type of shapes to create, either polygon or point.
Arguments :	labelData <i>optional</i> A data file that contains the labels for the column values.
	labelNeurons <i>optional</i> Whether or not neurons should be labeled by their best match to the data.
	radius <i>optional</i> The radius of the polygons to create.

SOM Analyst v1 documentation » SOM Analyst » SOM	previous   next   modules   index
Visualization »	

SOM Analyst v1 documentation » SOM Analyst » SOM

previous | next | modules | index

Visualization »

## Best Matching Unit to Shapefile

Creates a shapefile from projected data.

### ArcGIS Reference



#### Parameters

projected data

The input projected data file.

shape type

The type of shapes to use to represent the data.

projected data shapefile

The output shapefile to contain the projected data shapes.

label from SOM data

The unprojected SOM data file that contains the values and labels to use for the projected data.

Cartesian quadrant

The Cartesian quadrant in which to place the shapes.

spacing between center points

The spacing between the units in the SOM on which the projected data will be displayed.

placement

The placement location on each SOM unit to use for the projected data shapes.

maximum placement distance

The maximum distance from the center of the SOM units to place the projected data shapes.

### Code Reference

uiBMUtoSHP. **BMUtoSHP**(*bmufile*, *outfile*, *labels*, *quadrant*, *spacing*, *placement*, *distance*)

Creates a shapefile from a BMU file.

	bmufile	
	The input BMU filename.	
	outfile	
	The ouput shapefile name.	
	labels optional	
	A data file that contains the labels for the column values.	
Argument	ts quadrant optional	
:	The Cartesian quadrant to use.	
	spacing optional	
	The spacing between units in the SOM.	
	placement optional	
	The method for placement within a neuron.	
	distance optional	
	The maximum distance for the placement.	

SOM Analyst v1 documentation » SOM Analyst » SOM	previous   next   modules   index
Visualization »	

SOM Analyst v1 documentation » SOM Analyst » SOM

previous | next | modules | index

Visualization »

# **Group Shapes**

Groups the shapes in a shapefile.

### **ArcGIS Reference**

යි 3. Group Shapes	- 🗆 🗵
input shapefile	A
C:\somanalyst\dat\bmu.shp	<b></b>
group by column	
State	•
group type	
polyline	▼
value type	
maximum	▼
ouput shapefile	
C:\somanalyst\dat\trajectories.shp	🗃 🔁
sort by column (optional)	
Year	-
decimal places for averages	
6	
0 12	-
OK Cancel Environments Show	Help >>

#### Parameters

input shapefile

The input shapefile that contains the shapes to group.

group by column

The column in the data table contains the value on which to combine shapes.

group type

The new shape type to create.

value type

The method for selecting the value to use for each variable in the new data table.

ouput shapefile

The ouput shapefile that will contain grouped shapes.

sort by column
The column from the data table that contains the values on which to sort the shapes.

decimal places for averages

The number of decimal places to which to round numbers.

# Code Reference

uiGroup.group(inName, groupBy, groupType, valueType, outName, sortBy, decimalPlaces)

Groups shapes in a shapefile using the specified parameters.

	inName The input shapefile name.
	groupBy The data column to group on.
	groupType The type of grouping to create.
Arguments :	valueType The type of value to place in the final ouput.
	outName The output shapefile name.
	sortBy ( <i>optional</i> ) A column to sort the data by before grouping.

SOM Analyst v1 documentation » SOM Analyst » SOM	previous   next   modules   index
Visualization »	

SOM Analyst v1 documentation » SOM Analyst »

previous | next | modules | index

# Utilities

Contents:

- Create Extent
  - ArcGIS Reference
  - Code Reference
- Send Email
  - ArcGIS Reference
  - Code Reference

SOM Analyst v1 documentation » SOM Analyst »

previous | next | modules | index

SOM Analyst v1 documentation » SOM Analyst » Utilities » previous | next | modules | index

# **Create Extent**

Creates a rectangular bounding box around the extent of a shapefile.

# **ArcGIS Reference**

#### Parameters

input shapefile

The input shape from which to determine the extent.

extent type

The way to determine the extent.

ouput extent shapefile

The ouput shapefile.

# Code Reference

uiExtent.extent(ifile, ofile)

Creates a rectangular extent for a shapefile.

	ifile
	The input shapefile name.
Arguments :	ofile
	The output shapefile name.

SOM Analyst v1 documentation » SOM Analyst » Utilities » previous | next | modules | index

SOM Analyst v1 documentation » SOM Analyst » Utilities » previous | next | modules | index

# Send Email

Sends an email.

# **ArcGIS Reference**

#### Parameters

from

The from field for the email.

to

The to field for the email.

subject

The subject for the email.

text

The text for the email.

attachment

An optional attachment for the email.

email password

The password for the from email address.

smtp

The smtp server.

port

The port number of the smtp server.

# Code Reference

uiEmailMe.mail(email\_user, to, subject, text, attach, email\_pwd, smtp, port)

Sends an email.

	email_user
	The from field of the email.
	to
	The to field of the email.
	subject
	The subject of the email.
	attach optional
Arguments :	An attachment for the email.
	email_pwd
	The password for the email account.
	smtp
	The smtp server.
	port
	The smtp port.

SOM Analyst v1 documentation » SOM Analyst » Utilities » previous | next | modules | index

SOM Analyst v1 documentation »

previous | modules | index

# **Tutorial**

This tutorial contains step-by-step instructions on how to use the provided example dataset with SOM Analyst in ArcMap. The source data set for this tutorial is provided with SOM Analyst and is located in its sub-folder named **dat**. The file named **census.csv** contains gender, age, race, and housing data for each U.S. population census between the years 1900 and 1990.

First, the data is converted from the comma separated file format (.csv) to the database file format (.dbf) so that normalizations can be performed. Second, the raw count data are normalized by state population counts. Third, every variable is normalized into a 0 to 1 range and the preprocessed data are then exported to the SOM input format. Using those input data, a SOM is trained in two stages. The input data are then projected onto the finished SOM. Finally, a number of visualizations are produced.

# System Requirements

- 1. Windows (any version)
- 2. ArcGIS 9.3 (legacy toolboxes for ArcGIS 9.0-9.2 are provided, but untested)
- 3. Python 2.5 (included in the default ArcGIS 9.3 installation)

# Download

SOM Analyst is available for download from http://somanalyst.googlecode.com

# Adding the Toolbox

Add the SOM Analyst Toolbox to ArcGIS.

1. Open the ArcToolbox panel by clicking on the **Window** menu and select **ArcToolbox**. Alternatively, click on the toolbox icon on the menu bar.



2. Right click in the ArcToolbox panel and select Add Toolbox....



3. Browse to the location of SOM Analyst and select **guiArcGIS93.tbx** and click **Open**.

**Note:** Depending on your computer setup, it may be necessary to first "connect" to the folder that contains SOM Analyst. In that case, click in the dialog box on the icon of a folder with an arrow pointing to a globe.

Add Toolbox							×
Look in: 🧰	somanalyst	-	e	<b>3 3 1</b>	8-8- 8-8- 8-8-	<b>#</b> 88	
bin     dat     doc     lib     alphatools     guiArcGIS9     guiArcGIS9     guiArcGIS9	0						
Name:	guiArcGIS93					Open	
Show of type:	Toolboxes			•		Cancel	

The SOM Analyst toolbox is now accessible through the ArcToolbox panel.



Browse through the toolbox to familiarize yourself with the tools.



# **Convert Data Format**

Convert the data to a database file format.

1. Run the **Data File to Database File** tool by double clicking on it in the **File Format Conversions** toolbox of the **Data Preprocessing** toolbox.

💲 1. Data File to Database File	- 🗆 ×
input data file	
C:\somanalyst\dat\census.csv	<b>2</b>
input file format	
Comma Separated Values (CSV)	-
output database	
C:\somanalyst\dat\census.dbf	🖻 👘
detect data types (optional)	
OK Cancel Environments Show	Help >>

- 2. Select **census.csv** as the *input data file*.
- 3. Set Comma Separated Values (CSV) as the input file format.
- 4. Change the *output database file* to **census.dbf**.
- 5. Click **OK** to run the conversion.

In the table properties the data type for each column is text.

Primary Display	Field:	Region						
Name	lds will be visible. (	lick in the alias	column to	edit the alias	for any field	d. Number Format		
	OID	Object ID	4	0	0	Hambor Format		
Region	Region	Text	1	0	0			
Division	Division	Text	1	0	0			
✓ State	State	Text	20	0	0			
Abbrev	Abbrev	Text	2	0	0			
Year	Year	Text	4	0	0			
Female	Female	Text	8	0	0			
Male Male	Male	Text	8	0	0			
Under_15	Under_15	Text	7	0	0			
✓ 15_64	15_64	Text	8	0	0		-	
Select All	Clear All							

The values in the table are left justified indicating that they are text.

I	III Attributes of census									
Г		OID	Region	Division	State	Abbrev	Year	Female	Male	
		0	3	6	Alabama	AL	1900	911933	916764	
	E	1	4	8	Arizona	AZ	1900	51136	71795	
		2	3	7	Arkansas	AR	1900	636252	675312	
Г		3	4	9	California	CA	1900	664522	820531	
Г		4	4	8	Colorado	CO	1900	244368	295332	T II
		5	1	1	Connecticut	CT	1900	454126	454294	
Ŀ										
	Record: II I I I Show: All Selected Records (0 out of 490									

### Normalize Data

Normalize values in the database file.

1. Run the **Normalize by Variable** tool by double clicking on it in the **Value Transformations** toolbox of the **Data Preprocessing** toolbox.

Normalize by Variable	
input database file	<u></u>
C:\somanalyst\dat\census.dbf	
normalize by column	
Population	•
output database file	
C:\somanalyst\dat\normVar.dbf	🖻 🖆
division by zero value	
	0
decimal places to round to	
6	
0	12
columns to normalize (optional)	
Female	
Male	
✓ Under_15	
<b>№</b> 15_64	
✓ 65_Over	
Am_Indian	
	-
•	
	A LOCAL A
	Heid
	-
	Chau Hala >>
Cancel Environments	snow Help >>

- 2. Select **census.dbf** as the *input database file*.
- 3. Select **Population** as the *normalize by column*.
- 4. Change the *output database file* to **census.dbf**.

- 5. Select the columns male, female, Under\_15, 15\_64, 65\_Over, Am\_Indian, Asian, Black, and White in the *columns to normalize* field.
- 6. Click **OK** to run the normalization.

The resulting table contains population ratios.

▦	Attributes of normVar								
	OID	Region	Division	State	Abbrev	Year	Female	Male	
Þ	0	3	6	Alabama	AL	1900	0.498679	0.501321	
	1	4	8	Arizona	AZ	1900	0.415973	0.584027	
	2	3	7	Arkansas	AR	1900	0.485109	0.514891	
	3	4	9	California	CA	1900	0.447474	0.552526	
	4	4	8	Colorado	CO	1900	0.452785	0.547215	
	5	1	1	Connecticut	СТ	1900	0.499908	0.500092	-
◄					1	1		Þ	
	Record: II I I I Show: All Selected Records (0 out of 490								

7. Run the **Min-Max Normalization** tool by double clicking on it in the **Value Transformations** toolbox of the **Data Preprocessing** toolbox.

S Min-Max Normalization	<u>- 🗆 ×</u>
input database file	<u> </u>
C:\somanalyst\dat\normVar.dbf	<b>1</b>
normalize by	
column	
output database file	
C:\somanalyst\dat\norm01.dbf	i i i i i i i i i i i i i i i i i i i
division by zero value (max=min)	-
scale minimum to	
	0
scale maximum to	
	1
decimal places to round to	
6	
0 12	2
Villader 15	
▼ 15 64	
I 65 Over	
Am Indian	
Asian	
Black	
White	
Select All Unselect All Add F	ield 🗸
OK Cancel Environments Show	/Help >>

- 8. Select **normVar.dbf** as the *input database file*.
- 9. Select **column** as the *normalize by* field.
- 10. Change the *output database file* to **norm01.dbf**.
- 11. Select the columns male, female, Under\_15, 15\_64, 65\_Over, Am\_Indian, Asian, Black, and White in the *columns to normalize* field.
- 12. Click **OK** to run the normalization.

The resulting table contains normalized values.

Ⅲ	Attributes of norm01										
	OID	Region	Division	State	Abbrev	Year	Female	Male			
F	0	3	6	Alabama	AL	1900	0.784825	0.215175			
	1	4	8	Arizona	AZ	1900	0.32291	0.67709			
	2	3	7	Arkansas	AR	1900	0.709037	0.290963			
	3	4	9	California	CA	1900	0.498844	0.501156			
	4	4	8	Colorado	CO	1900	0.528506	0.471494			
	5	1	1	Connecticut	CT	1900	0.791689	0.208311	-		
◄					1			Þ	Г		
	Record: II I I I Show: All Selected Records (0 out of 490										

# **Select Variables**

Select the relevant variables from the database file.

1. Run the **Select** tool by double clicking on it in the **Data Management** toolbox of the **Data Preprocessing** toolbox.

\$ Select			<u>_   ×</u>
input database	file		<u> </u>
C:\somanalyst	\dat\norm01.db	f	🖻
selection type			
columns			•
output databas	e file		
C:\somanalyst	\dat\demograp	nics.dbf	🖻 🖻
columns (option	al)		
<ul> <li>Region</li> <li>Division</li> <li>State</li> <li>Abbrev</li> <li>Year</li> <li>Year</li> <li>Female</li> <li>Male</li> <li>Under_15</li> <li>15 64</li> <li>Image description</li> </ul>			
Select All start index (opt	Unselect All ional)		Add Field
index step (opti	ional)		0
			1
stop index (opt	ional)		
			-1
✓ detect data	types (optional)		×
ОК	Cancel	Environments	Show Help >>

- 2. Select **norm01.dbf** as the *input database file*.
- 3. Set **columns** as the *selection type*.
- 4. Change the *output database file* to **demographics.dbf**.
- 5. Select all columns except Population, Owner, Renter, and

Households in the *columns* field.

- 6. Enable detect data types.
- 7. Click **OK** to run the selection.

In table properties the value types for the columns has changed where appropriate.

Primary Display	Field:	State					
hoose which fie	elds will be visible.	Click in the alias	column to	edit the alias	for any fiel	d.	
	OID	Object ID	4	0	0		
Region	Region	Short	1	1	0	Numeric	
✓ Division	Division	Short	1	1	0	Numeric	
✓ State	State	Text	20	0	0	_	
Abbrev	Abbrev	Text	2	0	0		
Year	Year	Short	4	4	0	Numeric	
<ul> <li>Female</li> </ul>	Female	Float	8	7	7	Numeric	
Male Male	Male	Float	8	7	7	Numeric	
Under_15	Under_15	Float	8	7	7	Numeric	
✓ 15_64	15_64	Float	8	7	7	Numeric	
Calaat All	Class All						

The numeric values in the table are right justified indicating that they are numbers.

<b>=</b>	Attribu	ites of dem	nographics							×
	OID	Region	Division		State	Abbrev	Year	Female	Male	
Þ	0	3	6	Alabama		AL	1900	0.784825	0.215175	
	1	4	8	Arizona		AZ	1900	0.32291	0.67709	
	2	3	7	Arkansas		AR	1900	0.709037	0.290963	
	3	4	9	California		CA	1900	0.498844	0.501156	
	4	4	8	Colorado		CO	1900	0.528506	0.471494	T
┫									Þ	
	Red	cord: 🚺 🖣	1	<b>→ →</b> 1	Show: All Se	lected	Record	s (0 out of	490	•

**Note:** Detecting data types for columns requires checking the

data type of each value and can be time consuming for large datasets. This step is only necessary if performing normalizations or other calculations before using the data with a SOM.

### **Export Data**

Export the database file to the SOM data format.

1. Run the **Database File to SOM\_PAK Data** tool by double clicking on it in the **File Format Conversions** toolbox of the **Data Preprocessing** toolbox.

\$ 2. Database to SOM_PAK Data
A
C:\somanalyst\dat\demographics.dbf
output SOM data
C:\somanalyst\dat\demographics.dat
label columns (optional)
Region 🔺
Division
State
Abbrev
Year
Select All Unselect All Add Field
OK Cancel Environments Show Help >>

- 2. Select **demographics.dbf** as the *input database file*.
- 3. Change the *output SOM data file* to **demographics.dat**.
- 4. Select **Region**, **Division**, **State**, and **Year** in the *label columns* field.
- 5. Click **OK** to run the export.

# **Create Initial SOM**

Creating the initial SOM.

1. Run the **Create Initial SOM** tool by double clicking on it in the **SOM Computation** toolbox.

5 1. Create Initial SOM
data for SOM
C:\somanalyst\dat\demographics.dat
topology type of map
hexa 💌
neighborhood type
gaussian
x dimension
25
y dimension
25
initilization type
rand 💌
random generator seed (optional)
and huffer (antional)
OK Cancel Environments Show Help >>

- 2. Select demographics.dat as the data for SOM.
- 3. Select **hexa** as the topology of map.
- 4. Set **25** as the *x* dimension.
- 5. Set **25** as the *y* dimension.
- 6. Set **init.cod** as the *initial SOM*.
- 7. Click **OK** to run the creation of the initial SOM.

A window will open that indicates the progress of the process.

Cast C	:\Wi	ndov	ws\syst	em32\cmd.exe	- 🗆 ×
30	3/	61	sec.	<u>-</u>	<b></b>
					-

# Train SOM

Training the SOM.

**Note:** The SOM will be trained in two steps. The first training will create the overall structure in the SOM. The second training will create the finer specialization.

1. Run the **Train SOM** tool by double clicking on it in the **SOM Computation** toolbox.

🖇 2. Train SOM	_ 🗆 🗵
initial SOM	1
C:\somanalyst\dat\init.cod	
training data	
C:\somanalyst\dat\demographics.dat	🖻
length of training	
	4900
initial learning rate	
tottel antick ask and an disc	0.04
initial neighborhood radius	25
trained SOM	25
C:\somanalvst\dat\stage1.cod	
distance metric (optional)	
Euclidean	•
random generator seed (optional)	
fixed points (optional)	
use weights (optional)	
read buffer (optional)	
alpha type (optional)	
snanshot file (ontional)	
shanshot interval (ontional)	
OK Cancel Environments	Show Help >>

- 2. Select **init.cod** as the *initial som*.
- 3. Select **demographics.dat** as the *training data*.
- 4. Set **4900** as the *length of training*.
- 5. Set **0.04** as the *initial learning rate*.
- 6. Set **25** as the *initial neighborhood radius*.
- 7. Change the *trained SOM* to **stage1.cod**.
- 8. Click **OK** to run the training of the SOM.

A window will open that indicates the progress of the process as it did with the creation of the initial SOM.

9. Run the Train SOM tool.

2. Train SOM	_ 🗆 ×
initial SOM	· · · · · · · · · · · · · · · · · · ·
C:\somanalyst\dat\stage1.cod	😂
training data	
C:\somanalyst\dat\demographics.dat	🖻
length of training	
	49000
initial learning rate	
	0.03
initial neighborhood radius	
trained SOM	5
C:\somanalyst\dat\stage2.cod	
distance metric (optional)	
random generator seed (optional)	
fixed points (optional)	
use weights (optional)	
read buffer (optional)	
alpha type (optional)	
linear	•
snapshot file (optional)	
	🖻 🔁
shapshot interval (optional)	
	Shaw Hala >>
	Show Help >>

- 10. Select **stage1.cod** as the *initial som*.
- 11. Select **demographics.dat** as the *training data*.
- 12. Set **49000** as the *length of training*.
- 13. Set **0.03** as the *initial learning rate*.
- 14. Set **5** as the *initial neighborhood radius*.
- 15. Change the *trained SOM* to **stage2.cod**.
- 16. Click **OK** to run the training of the SOM.

### Calculate U-Matrix

Calculate the U-matrix of a SOM.

1. Run the **Calculate U-matrix** tool by double clicking on it in the **SOM Computation** toolbox.

nput SOM			
C:\somana	yst\dat\stage2.co	d	🖻 🖻
output U-ma	trix database file		
C:\somana	yst\dat\umatrix.db	of	🖻 🖻
distance met	ric (optional)		
Euclidean			-

- 2. Select **stage2.cod** as the *input SOM*.
- 3. Change the *output U-matrix database file* to **Umatrix.dbf**.
- 4. Click **OK** to calculate the U-matrix

▦	Attribu	ates of Un	atrix 💶 🛛 🗙
	OID	Uvalue	<b>A</b>
Þ	0	0.04474	
	1	0.04808	
	2	0.06109	
	3	0.06853	
	4	0.06678	
	5	0.06078	<b>_</b>
	Re	cord: 🛯 🖣	1 <b>) ) v</b>

# Project Data onto SOM

Project the data onto the SOM.

1. Run the **Project Data onto SOM** tool by double clicking on it in the **SOM Computation** toolbox.

3. Project Data onto SOM	
SOM	
C:\somanalyst\dat\stage2.cod	- E
data to project	
C:\somanalyst\dat\demographics.dat	
projected data	
C:\somanalyst\dat\demographics.bmu	🖻 🖻
distance metric (optional)	
skip masked data vectors (optional)	
OK Cancel Environments S	how Help >>

- 2. Select stage2.cod as the SOM.
- 3. Select **demographics.dat** as the *data to project*.
- 4. Change the *projected data* to **demographics.bmu**.
- 5. Click **OK** to project the data onto the SOM.

A window will open that indicates the progress of the process as it did with the creation of the initial SOM.
## Create SOM Shapefile

Creating the SOM shapefile.

1. Run the **SOM to Shapefile** tool by double clicking on it in the **SOM Visualization** toolbox.

1. SOM to Shapefile	_ 🗆
SOM	
C:\somanalyst\dat\stage2.cod	2
shape type	
polygon	-
SOM shapefile	
C:\somanalyst\dat\stage2.shp	<b>2</b>
SOM data for variable names (optional)	
C:\somanalyst\dat\demographics.dat	2
U-matrix (optional)	~
C:\somanalyst\cat\umatrix.dbr	
	-
spacing between center points	
	1

- 2. Select **stage2.cod** as the SOM.
- 3. Select **polygon** as the *shape type*.
- 4. Change the SOM shapefile to stage2.shp.
- 5. Set **demographics.dat** as the SOM data for variable names.
- 6. Enable label SOM with data labels
- 7. Set **Umatrix.dbf** as the *U-matrix*.
- 8. Click **OK** to create the SOM shapefile.

# Create Data Shapefile

Creating the data shapefile.

1. Run the **Projected Data to Shapefile** tool by double clicking on it in the **SOM Visualization** toolbox.

2. Projected Data to Shapefile	_ 🗆
projected data	
C:\somanalyst\dat\demographics.bmu	<b>2</b>
shape type	
point	-
projected data shapefile	
C:\somanalyst\dat\bmu.shp	<b>2</b>
label from SOM data (optional)	
C:\somanalyst\dat\demographics.dat	<b>2</b>
Cartesian quadrant	
1	-
spacing between center points	
	1
placement	
random around center	-
maximum placement distance	
	0.3
OK Cancel Environments Show	Help >>

- 2. Select demographics.bmu as the projected data.
- 3. Select **point** as the *shape type*.
- 4. Change the projected data shapefile to bmu.shp.
- 5. Select **demographics.dat** as the *label from SOM data*.
- 6. Select random around center as the *placement*.
- 7. Click **OK** to create the data shapefile.

# Group Data Shapefile

Grouping the shapes in the data shapefile.

1. Run the **Group Shapes** tool by double clicking on it in the **SOM Visualization** toolbox.

නී 3. Group Shapes	_ 🗆 🗵
input shapefile	E
C:\somanalyst\dat\bmu.shp	<b>2</b>
group by column	
State	-
group type	
polyline	-
value type	
maximum	▼
ouput shapefile	
C:\somanalyst\dat\trajectories.shp	<b>2</b>
sort by column (optional)	
Year	-
decimal places for averages	
6	
0 12	
OK Cancel Environments Show	Help >>

- 2. Select **bmu.shp** as the *input shapefile*.
- 3. Select State as the group by column
- 4. Select **polyline** as the *group type*.
- 5. Select **maximum** as the *value type*.
- 6. Change the *output shapefile* to **trajectories.shp**.
- 7. Select **Year** as the sort by column.
- 8. Click **OK** to create the trajectories.

# **Create Extent Shapefile**

Creating the extent shapefile.

1. Run the **Create Extent Shapefile** tool by double clicking on it in the **Utilities** toolbox.

input shapefile C:\somanalyst\dat\stage2.shp extent type rectangular from bounding box ouput extent shapefile	<u>**</u> F
C:\somanalyst\dat\stage2.shp extent type rectangular from bounding box ouput extent shapefile	ž
extent type rectangular from bounding box ouput extent shapefile	
rectangular from bounding box ouput extent shapefile	
ouput extent shapefile	-
C:\somanalyst\dat\extent.shp	¥ i
	<b>_</b>
OK Cancel Environments Show Help	

- 2. Select **stage2.shp** as the *input shapefile*.
- 3. Change the *output shapefile* to **extent.shp**.
- 4. Click **OK** to create the extent shapefile.

# Visualization

Visualizing the SOM and projected data.

#### 1. Open tutorial.mxd.

**Note:** Your map will not be identical, but should be very similar. The frames may appear rotated due to the initial random numbers used.



The large map shows the trajectory of each state across the SOM over time with a base of the U-matrix, a measure of distortion. The trajectories are color coded by census division, which are shown in the lower right. The other frames contain the component planes,

each showing the neuron weights for one variable across the entire SOM.

When examining the demographic trajectories of each state note that each shift in the trajectory corresponds to a census year and that at the end of the trajectory is an arrow that represents the year 1990. Parallel trajectories indicate a similar change in demographics over time. Parallel trajectories are particularly evident within the South Division (West South Central Region, East South Central Region, and South Atlantic Region) and Northeast Division (Middle Atlantic Region and New England Region). This demonstrates spatial autocorrelation and is consistent with the demographic changes over the last century. In the Northeast Division, the parallel trajectories split 40 years ago mainly into coastal and land locked areas with New York and New Jersey similar to each other, but dissimilar to the other coastal states.

When examining component planes you are seeing how the SOM allocates location based on that variable. In this map, darker color means high values and lighter color means low values. You can see that the female component plane is very dark in one corner and light in the opposite corner with a gradual change between the two. Conversely the male component plane is very dark in the opposite corner and has a similar pattern of gradual change. When comparing component planes to each other you can see how the SOM weights the variables in the same location and thus derive a relationship between them. You can see that that female and male have an inversely proportional relationship in the SOM that corresponds with reality, that is that a high number of females inherently means a low number of males and vice versa.

SOM Analyst v1 documentation »

modules | index

# Index

### **B|C|D|E|G|M|N|S|T|U|V**

В

BMUtoSHP() (in module uiBMUtoSHP)

codebook create

С

CODtoSHP() (in module uiCODtoSHP)

D

dbfmerge() (in module uiDBFmerge) DBFtoDAT() (in module uiDBFtoDAT) Е

extent() (in module uiExtent)

G

group() (in module uiGroup)

Μ

mail() (in module uiEmailMe)

mapinit() (in module uiMapInit)

## Ν

normalize() (in module uiNormMinMax) (in module uiNormVar) (in module uiNormZ) S

select() (in module uiSelect)

SOM create toXbase() (in module uiToXbase)

## U

uiBMUtoSHP (module) uiCODtoSHP (module) uiDBFmerge (module) uiDBFtoDAT (module) uiEmailMe (module) uiExtent (module) uiGroup (module) uiMapInit (module) uiNormMinMax (module) uiNormVar (module) uiNormZ (module) uiSelect (module) uiToXbase (module) uiUmatrix (module) uiVisual (module) uiVsom (module) uMatrix() (in module uiUmatrix) V

visual() (in module uiVisual)

vsom() (in module uiVsom)

SOM Analyst v1 documentation »

modules | index