ArcSIE
User’s Guide

Xun Shi

xun.shi@dartmouth.edu
Last modified on June 18, 2008
Introduction

ArcSIE (SIE stands for Soil Inference Engine) is a toolbox for digital soil mapping. ArcSIE functions as an “Extension” of ArcMap.

ArcSIE generates soil maps based on the soil-environment model:

\[ S = f(E) \]

This model states that the information about soil (S) can be derived from the information about the soil formative environment (E), including topography, geology, climate, vegetation, etc.

The most critical component in this model is the relationship between the soil and its environment (f). ArcSIE supports a knowledge-based approach to establishing this relationship. ArcSIE provides tools for soil scientists to formalize the relationship based on their knowledge of local soils. ArcSIE works with two types of knowledge:

- **Rules** defined by environmental feature values; and
- **Cases** defined in geographical space. Cases can be represented by **point**, **line**, **polygon**, and **pixels**.

ArcSIE applies rules and/or cases to environmental data (stored as GIS data layers) to generate soil maps. The inference based on rules is called rule-based reasoning (RBR), and the inference based on cases is called case-based reasoning (CBR).

*Cases* can be applied to the entire mapping area (*global cases*) or a local region (*local cases*).
It is recommended that the user first uses RBR or global CBR (i.e., CBR using *global cases*) to create draft maps. The user can then create and use *local cases*, as needed, to *fine-tune* the draft map in *exceptional* areas. This procedure can achieve both high efficiency (through the RBR or global CBR) and high accuracy (through the local CBR).

ArcSIE performs soil inference under the fuzzy logic. The immediate output from the inference is a series of *fuzzy membership maps* in the raster format, one for each soil type under concern.

In addition to the soil inference function, ArcSIE also provides tools for result validation, terrain analysis, pre- and post-processing for raster data, vectorization, and data format conversion.
Launch ArcSIE

Launch ArcSIE in the same way as you launch other ArcMap Extensions.

1. In the Tools menu of ArcMap, select Extensions and check Soil Inference Engine.

2. In the View menu of ArcMap, select Toolbars and check Soil Inference Engine.

Then you should be able to see that the ArcSIE menu bar appears in ArcMap.
Chapter 1. Inference

1.1. Fuzzy Soil Mapping Using RBR and CBR

At the center of ArcSIE is an inference engine performing fuzzy soil mapping based on the concept of fuzzy soil classification. Fuzzy soil mapping does not label the soil at a given location with a single soil name, but assigns to the soil fuzzy membership values for all the soil types under concern. The fuzzy membership values represent the similarities of the soil to those soil types. Rule-based reasoning (RBR) and case-based reasoning (CBR) are the two inference methods implemented by ArcSIE for calculating fuzzy membership values.

RBR is based on rules like “if the elevation is 1000 ft, then the soil is typical for type A”. In this example, ArcSIE will use the GIS database to identify all the locations where elevations are 1000 ft, and assign full fuzzy membership to those locations for soil type A. The typical soil in this example is defined using only one environmental feature, elevation. More often, ArcSIE uses multivariate inference to determine the fuzzy membership based on more than one environmental feature. In ArcSIE, a rule is represented by a fuzzy membership function defining the relationship between the values of an environmental feature and the values of fuzzy membership for a given soil type.

While in RBR, the 1000 ft elevation in above example is explicitly specified by the soil scientists, in CBR this value is retrieved from the GIS database at the location(s) pinpointed or delineated by the soil scientist. The soil scientist identifies these case locations as places where the soil is typical for the given soil type. CBR is useful when explicit rules based on environmental feature values are not easy to state, but the soil scientist is able to visually identify the landscape units or locations for the given soil type using a topographic map, DEM, or orthophoto.
1.2. Launching the Inference Engine

ArcSIE provides an integrated interface for RBR and CBR. In the Soil Inference Engine menu, clicking *Inference* opens the Inference dialog box:
1.3. Preparing the Environmental Database

Before any inference can be performed, environmental data layers must be loaded to ArcSIE to build the environmental database. The environmental database:
- provides data for the environmental features used in the soil inference; and
- defines the spatial extent and resolution of the resulting soil maps.

1.3.1. Creating and Editing the Environmental Database

All environmental data layers must be raster data in the ArcInfo Grid format.

In the Inference dialog box, clicking launches the Environmental Data Layer Editor dialog box (see next page). On the dialog box:

- Use \(\text{select from the layers currently loaded to ArcMap and add the selected to the environmental database}.\)

- \(\text{browses disks to choose a data layer to add to the environmental database.}\)

- \(\text{ saves the feature names and corresponding Grid names currently in the environmental database into a list file (.lst) for future quickly loading.}\)

- \(\text{browses disks and opens a list file (.lst) to quickly build a new environmental database using the layers recorded in that list file. The new environmental database automatically replaces the environmental database currently being used.}\)
If a layer is successfully added to the *environmental database*, its name and data source will appear in the form at the middle of the dialog box.

If an error occurs when ArcSIE tries to load a data layer specified in the *list file*, e.g., the Grid cannot be found in the specified folder, ArcSIE will display the following dialog box:
If you choose to use another Grid, a dialog box will appear for you to browse disks and select the Grid you want.

If you choose to drop the environmental feature, this feature will not be included in the environmental database.

Once the data have been added, right-clicking on a layer opens a popup menu:
The *Load Data* item in the popup menu allows you to replace the current Grid with another one. This replacement does not change the name of the feature.

The *Remove* item removes the selected *environmental feature* from the *environmental database*.

### 1.3.2. Spatial Setting of the Environmental Database

You can load data layers with different spatial extents and different spatial resolutions to the *environmental database*. Once they are loaded, they will be automatically aligned based on the *spatial setting* of the *environmental database*. 
However, these data layers MUST be already in the same coordinate system, and this coordinate system should also be the system of the current mxd of ArcMap.

The *spatial setting* of the *environmental database* determines the spatial extent and resolution of the inference result, but do not affect the output from other tools in ArcSIE.

The default *spatial setting* is determined by the first loaded data layer. You can customize the *spatial setting* using the Spatial Setting dialog box. Click on the *Spatial Setting* button on the Environmental Data Layer Editor brings the Spatial Setting dialog box to the desktop:
On this dialog box, you can choose to use a data layer as the template to set spatial extents and cellsize, or you can set arbitrary values for these settings.

1.3.3. Naming an Environmental Data Layer

The default name of a layer is the name of the Grid for that layer.

In the Environmental Data Layer Editor dialog box, clicking on the name of a data layer makes the name editable, allowing you to rename the data layer.

When you create a new knowledgebase based on the current environmental database, the names of the environmental data layers in the current environmental database will be used as the names of the environmental features in the new knowledgebase.

When you load an existing knowledgebase, the names of environmental features in this knowledgebase must match the names of the environmental data layers in the current environmental database. ArcSIE automatically checks the names when you load an existing knowledgebase and presents a warning if the two sets of names do not exactly match.

1.3.4. Finishing Creation of the Environmental Database

Clicking the OK button closes the Environmental Data Layer Editor dialog box. This completes the creation of the environmental database, and returns you to the Inference dialog box.
The Environmental Data Layer Editor dialog box can also be used to edit the *environmental database* after it is created. Changes made to the *environmental database* will be reflected in the *feature* list in the Inference dialog box.
1.4. Specifying Knowledge Type

1.4.1. Types and Forms of Knowledge

ArcSIE supports two general types of knowledge: rule that is directly defined in the attribute (i.e. environmental feature) space and case that is initially defined in the geographical space. The case can be in point, line, polygon, and pixel forms.

*Attribute Rules* are used to perform rule-based reasoning (RBR). They can be used to represent the soil scientist’s explicit knowledge between soils and environmental features. *Attribute rules* are created in ArcSIE and all their values are directly specified by the user.

*Point Case* (also called *tacit points*) are used to perform point case-based reasoning (point CBR). They can be used to represent the soil scientist’s knowledge of the soils at specific locations.

*Line Case* can be used to represent the soil scientist’s knowledge about features that have linear shapes. For example, they can be used to represent typical locations of a soil that occur along ridgelines. In ArcSIE, a *line case* is treated as a series of independent point cases. The locations of these “points” are determined by the length and orientation of the line and the cells size of the environmental database. The parameter setting of a *line case* is applied to every “point” in the series.

*Polygon Case* can be useful if the soil scientist wants to delineate the typical locations of a soil as polygons rather than points. In ArcSIE, a *polygon case* is treated as an agglomeration of independent point cases. The locations of these “points” are determined by the coverage of the polygon and the cells size of the environmental database. The parameter setting of a *polygon case* is applied to every “point” in the series.
Pixel Cases are usually derived from a terrain analysis process. For example, streamlines derived from a DEM can be used as the typical locations for valley areas. In ArcSIE, one pixel case is treated as point case. The parameter setting for a raster casebase is applied to every pixel case in this casebase.

1.4.2. Specifying Knowledge Type and Form

Before creating or loading the knowledgebase and conducting the inference, you need to specify which type and form of knowledge you are using:

Attribute Rule for RBR is the default option when you first launch the Inference dialog box.
1.4.3. Structure of ArcSIE Knowledgebase

A knowledgebase for performing rule-based reasoning (RBR) is called a rulebase. A rulebase of ArcSIE has the following structure:

```
rulebase
  |
  soil type (instance list)
  |
  instance
  |
  rule
```

A rulebase can contain knowledge of one or more soil types. A soil type may have one or more instances, each is about a unique environmental configuration for the soil type. For example, a soil may occur on both south facing and north facing slopes, but the elevation and slope gradient conditions for the soil on the two slopes may be different. These two environmental configurations form two instances of the soil. An instance may have one or more rules, each characterizing the relationship between the fuzzy membership of the soil type and a specific environmental feature.
A knowledgebase for performing case-based reasoning (CBR) is called a *casebase*. A *casebase* containing *vector cases* (point, line, and polygon) has the following structure:

```
casebase
  | soil type (case list)
  |    case (either point, line, or polygon)
  |      rule and spatial setting
```

A *casebase* contains *raster (pixel) cases* has the following structure:

```
casebase
  | case (pixel)
  |      rule and spatial setting
```

In terms of content, a *case* differs from an *instance* in that a *case* has a *spatial setting*. A *spatial setting* contains the spatial information about a *case*, including its location and some parameter values for performing local CBR. In terms of knowledge acquisition, the fundamental difference between an *instance* and a *case* is on how their *central values* are determined: the *central values* of an *instance* are directly specified by the user, whereas the *central values* of a *case* are identified by the inference engine according to the location of the *case*. 

1.5. Preparing a Rulebase

1.5.1. Creating a New Rulebase

Note: To create a new knowledgebase, you must first create or load an *environmental database*. However, a loaded *environmental database* can be used for all the knowledge types or forms, so you do not have to reload it after selecting a different knowledge type or form, if you still want to use the same *environmental database*.

1.5.1.1. Creating a New Rulebase

Click ![image] to create a new *rulebase*. The new *rulebase* contains one default *soil type*, and the default *soil type* contains one default *incidence*. Default *rule(s)* will be created for that *instance* based on the *environmental features* contained in the current *environmental database*.

At any time, ArcSIE can host only one *rulebase*. Therefore if there is a *rulebase* currently being used, the current *rulebase* will be unloaded before the new one is created. You will be prompted to save the changes you made to the current *rulebase* before it is unloaded.

1.5.1.2. Creating a New Soil Type

Right-click the *rulebase* name and select *New Soil Type* from the drop-down menu. The new *soil type* will be appended to the end of the list of *soil types*. It will contain a default *incidence*.

1.5.1.3. Creating a New Instance
Right-click the soil type for which you want to create a new instance and select New Instance. The new instance is appended to the end of the list of instances in the current soil type and is assigned a unique default name. The new instance contains default rule(s) for the environmental features supported by the current environmental database.

1.5.2. Loading an Existing Rulebase

Click to load an existing rulebase. The file format of a rulebase is DBF (.dbf). If you try to open an existing rulebase before creating or loading the environmental database, a dialog box will appear to guide you:
Clicking Load environmental data opens the Environment Data Layer Editor. The environmental features required by the rulebase will be listed in the editor. You will need to right-click on a feature and select Load Data to load the data file for that feature.

If the knowledgebase you are opening contains environmental features that do not match the features in the current environmental database, a dialog box will appear to show options for resolving the mismatch:
If you choose to *Update factors using the current env db*, the original *environmental features* in the knowledgebase will be replaced with the *features* in the current *environmental database*. For any matched *features*, the settings in the knowledgebase will be applied to them.

At any time, ArcSIE can host only one *rulebase*. Therefore if there is a *rulebase* currently being used, the current *rulebase* will be unloaded before the new one is opened. You will be prompted to save the changes you made to the current *rulebase* before it is unloaded.
1.5.3. **Saving Rulebase**

Clicking ⌘ saves the current rulebase into a rulebase file (.dbf). All soil types, instances, and rules in the current rulebase will be saved into this file.

1.5.4. **Unloading Rulebase and Removing Soil Type and Instance**

1.5.4.1. **Unloading the Rulebase**

Right-click the rulebase name and select Remove to unload the current rulebase from ArcSIE. You will be prompted to save the changes you made to the current rulebase before it is unloaded.

1.5.4.2. **Removing a Soil Type**

Right-click the soil type you want to remove and select Remove to remove the soil type. Note: This will physically and permanently remove all the content of this soil type.

The last soil type in a rulebase cannot be removed.

1.5.4.3. **Removing an Instance**

Right-click the instance you want to remove and select Remove to remove the instance. Note: This will physically and permanently remove all the content of this instance from the rulebase.

The last instance in a soil type cannot be removed.
1.5.5. Changing Names

To change or edit the name of an existing rulebase, soil type, or instance, click on it and type in the new name.

In a rulebase, each soil type must have a unique name; and in a soil type, each instance must have a unique name. However, instances in different soil types can have identical names.

1.5.6. Editing Knowledge

The equation below describes how the knowledge of a given soil type will be used in an RBR process:

\[
    s_{ij,k} = T_k \left\{ \prod_{t=1}^{n} P_t \left[ E_{t,v} (z_{ij,v}) \right] \right\}
\]

The meanings of the symbols in the above equation are as follows:

\( s_{ij,k} \): the fuzzy membership value at location \((i, j)\) for soil \(k\).

\( m \): the number of environmental features used in the inference.

\( n \): the number of incidences for soil type \(k\).

\( z_{ij,v} \): the value of the \(v\)th environmental feature at location \((i, j)\).

\( E \): the function for evaluating the optimality value of the \(v\)th feature for soil \(k\).
$P$: the function for evaluating the fuzzy membership at the *instance* level.

$T$: the function for deriving the final fuzzy membership value for soil $k$ at site $(i, j)$ based on all the *instances* for soil $k$.

For the $T$ function, ArcSIE implements the *max* function for RBR, i.e., if more than one *instance* is used for a soil type and these *instances* give different fuzzy membership values for that soil type at a given location, the maximum value will be assigned to that location.

ArcSIE allows the user to adjust the $E$ and $P$ functions.

### 1.5.6.1. Adjusting the $E$ Function

#### 1.5.6.1.1. Selecting a Function Type

ArcSIE provides five choices for $E$ in the general inference equation: *continuous*, *cyclic*, *ordinal*, *nominal*, and *raw*, based on the nature of the *environmental feature*.

To assign a function to the current *environmental feature*, move the cursor into the graphic curve defining area of the Inference dialog box and right-click the mouse button. A pop-up menu will open and show a list of the $E$ functions implemented by ArcSIE. Choosing a function from the list will assign that function to the current *environmental feature*. After selecting the function type, you can adjust the parameter values for the function to precisely define it.

#### 1.5.6.1.1.1. Continuous Function
The continuous function is applicable to environmental features with interval or ratio values (e.g., temperature, elevation, and slope gradient). The mathematical representation of a continuous $E$ function is as follows:

$$
\begin{align*}
    s &= e^{[(z_{ij,v} - v_1) / w_1] \ln(c_1)} & \text{if } z_{ij,v} < v_1 \\
    s &= 1 & \text{if } z_{ij,v} = v_1 \\
    s &= e^{[(z_{ij,v} - v_2) / w_2] \ln(c_2)} & \text{if } z_{ij,v} > v_2
\end{align*}
$$

The continuous $E$ equation group defines two halves of a Gaussian-style function curve. The idea of this equation group is to use two sets of parameters to give the user the flexibility to define an asymmetric curve. Definitions of the symbols in the equation group and methods to adjust their values are explained below:

$s$: optimality value of environmental feature $v$ at location $(i, j)$ for soil type $k$. This is the output from the $E$ function.

$z_{ij,v}$: value of environmental feature $v$ at location $(i, j)$. This value is read from the environmental database.

$v_1$ and $v_2$: These are two user-specified central values that define the lower and upper limits of the most optimal range of environmental feature $v$ for soil $k$. In other words, if $z_{ij,v}$ falls between $v_1$ and $v_2$, then location $(i, j)$ will get the maximum optimality value for soil $k$ based on environmental feature $v$. Technically, $v_1$ and $v_2$ determine the width of the flat top of the function curve. These two values can
be adjusted by editing the values in $v_1$ and $v_2$. It is required that $v_1$ is less than or equal to $v_2$.

$w_1$ and $c_1$: The user can adjust the shape of the left half of the curve by specifying that if $z_{ij}^v$ is smaller than $v_1$ and the difference between $z_{ij}^v$ and $v_1$ is $w_1$, the output optimality value should be $c_1$. To simplify the adjusting operation, ArcSIE uses a fixed value, 0.5, for $c_1$. Therefore, $w_1$ is used to specify what value of the current environmental feature should correspond to an optimality value of 0.5. For example, if the environmental feature is elevation, and the soil scientist specifies $v_1 = 800$ ft and $w_1 = 20$ ft, then a location with elevation of 780 ft will get an optimality value of 0.5.

$w_2$ and $c_2$: The user can adjust the shape of the right half of the curve by specifying that if $z_{ij}^v$ is greater than $v_2$ and the difference between $z_{ij}^v$ and $v_2$ is $w_2$, the output optimality value should be $c_2$. To simplify the adjusting operation, ArcSIE uses a fixed value, 0.5, for $c_2$. Therefore, $w_2$ is used to specify what value of the current environmental feature should correspond to a fuzzy membership of 0.5. For example, if the environmental feature is elevation, and the soil scientist specifies $v_2 = 1000$ ft and $w_2 = 20$ ft, then a location with elevation of 1020 ft will get a fuzzy membership of 0.5.

$r_1$ and $r_2$: These two values control the flatness of the top parts of the curve that are beyond the $v_1$-to-$v_2$ range and the steepness of the side parts of the curve. The higher these two values, the flatter the tops and the steeper the sides.

ArcSIE provides three basic function curves based on which you can further fine-tune the curve shape:

- **bell-shape**: A symmetric shape. The optimality value decreases as the difference between the environmental feature value and the central values ($v_1$ and $v_2$) increases.
- **s-shape**: A the-higher-the-better shape. The optimality will always get the maximum value if the \textit{environmental feature} values are greater than \( v_2 \). For example, you may want to define that for a certain soil type, the higher the elevation, the better.

- **z-shape**: A the-lower-the-better shape. The optimality will always get the maximum value if the \textit{environmental feature} values are smaller than \( v_1 \). For example, you may want to define that for a certain soil type, the smaller the slope gradient (flatter), the better.

The diagrams on the following pages show examples of how the function curve changes according to the adjustments of different parameters.
The shape of the function curve changes against $w$. 

$v_1 = v_2$, $w_1 = w_2 = 0.3$, $c_1 = c_2 = 0.5$, $r_1 = r_2 = 2$

$v_1 = v_2$, $w_1 = w_2 = 0.5$, $c_1 = c_2 = 0.5$, $r_1 = r_2 = 2$

$v_1 = v_2$, $w_1 = w_2 = 0.8$, $c_1 = c_2 = 0.5$, $r_1 = r_2 = 2$
The shape of the function curve changes against $r$. 
1.5.6.1.1.2. **Cyclic Function**

The cyclic function is designed to handle a directional feature such as slope aspect. It is similar to the continuous function except that the maximum value actually goes back to the minimum value.

ArcSIE interprets specifications for a cyclic feature in a clockwise way as shown in the figure below:

For a cyclic feature:

- $v_1$, $v_2$, $w_1$, and $w_2$ must be degree measurements (between 0 and 360).
- If $z_{ij}^v$ falls into the range between $v_1$ and $v_2$ (the range is defined along the clockwise direction), the fuzzy membership based on this environmental feature will get full value (100%). Otherwise, the fuzzy membership will be calculated using the continuous function.
- Because the membership at a location may be calculated from both sides, one location can get two values. In ArcSIE, the greater value will be used.
1.5.6.1.1.3. Ordinal Function

ArcSIE uses the following function for ordinal data:

\[
\begin{align*}
  s &= \left( \frac{z_{ij,v} - z_{\text{min}}}{v_1 - z_{\text{min}}} \right) \quad \text{if } z_{ij,v} < v_1 \\
  s &= 1 \quad \text{if } v_1 \leq z_{ij,v} \leq v_2 \\
  s &= \left( \frac{z_{\text{max}} - z_{ij,v}}{z_{\text{max}} - v_2} \right) \quad \text{if } z_{ij,v} > v_2
\end{align*}
\]

In this equation group, \(z_{\text{min}}\) is the minimum value in the \textit{environmental database} for the current \textit{environmental feature} and \(z_{\text{max}}\) is the maximum value. The idea here is to use a linear function to quantify the difference between ordinal values, which is convenient for converting ordinal values into continuous values. Based on the nature of ordinal data, the ordinal function assumes the data for this \textit{environmental feature} to be integers, and \(v_1\) and \(v_2\) should also be integers, but these are not required by ArcSIE. It is still required that \(v_1\) is less than or equal to \(v_2\), though. An example of the curve defined by this function is given below:
1.5.6.1.1.4. Nominal Function

ArcSIE uses the following function for nominal data (e.g., geological type data):

\[
\begin{cases}
  s = 1 & \text{if } v_1 \leq z_{ij,v} \leq v_2 \\
  s = 0 & \text{otherwise}
\end{cases}
\]

In the *nominal* function, it is required that \( v_1 \) is less than or equal to \( v_2 \). The curve defined by the *nominal* function is shown below:
1.5.6.1.1.5. **Default Function**

The default function used by ArcSIE is a *continuous bell-shape* function with parameter values as follows:
- \( v_1 = v_2 = \text{mean of the values in the environmental database for the current environmental feature} \)
- \( w_1 = w_2 = 0.2 \times \text{standard deviation of the values in the environmental database for the current environmental feature} \)
- \( r_1 = r_2 = 2 \)

1.5.6.1.1.6. **Use Raw Values**

If you choose to use the raw values of a data layer, the inference engine will do nothing but simply uses the original values as the optimality values. This option is useful when you want to post-process the result from a previous inference. For example, if you want to apply a fuzzy mask to an inference result to reduce the membership values in a certain region, you should apply the fuzzy mask to the raw values in that inference result.

1.5.6.1.2. **Three Ways to View the Function Curve**

The function curve of E can be viewed in three ways:

- **Data Range:** Shows the relative position and width of the current curve in the data range (from the minimum to the maximum) of the data for the current environmental feature.
- **Relative:** Shows a comparison of the two halves (the left and the right) of the curve. Under this option, you can always see the entire curve.
- **Standard Deviation**: Shows a comparison of the width of the curve and the standard deviation of the data for the current *environmental feature*. This option brings a scale bar to the x axis, with each interval representing the width of one standard deviation of the data.

You can switch among these three options using the dropdown menu.

1.5.6.1.3. **Values Displayed on the Function Curve**

Once you check on an *environmental feature*, a green curve appears in the $E$ function defining area on the Inference dialog box. This green curve is a graphic representation of the $E$ function for the current *environmental feature*. There are some small red handles attached to the curve. The two round handles on the top of the curve correspond to $v_1$ and $v_2$; the two round handles at the bottom of the curve correspond to the *environmental feature* values giving very small fuzzy membership values (0.0001); and the two square handles on the sides of the curve indicate the positions where $c_1$ and $c_2$ equal to 0.5, respectively.

Refer to the image below, on the image the values associated with the function curve are labeled by letters A – H and are explained as follows:
A: The maximum optimality the rule can generate. The default value is 1.

B: The middle value between the maximum and the minimum optimality values the rule can generate. The default minimum optimality is 0.

C: The minimum value in the data for the current environmental feature. This value only shows up under the “Data Range” option.

D: The value of the environmental feature at the left bottom red handle. This is the “left” (small) value of the environmental feature that gives a very small optimality value (set to be 0.0001).

E: The value of the environmental feature at the left middle red handle. This is the “left” (small) value of the environmental feature that gives the “middle” optimality value. \( E = v_i - w_i \).

F: The value of the environmental feature at the right middle red handle. This is the “right” (large) value of the environmental feature that gives the “middle” optimality value. \( F = v_2 + w_2 \).
G: The value of the *environmental feature* at the right bottom red handle. This is the “right” (large) value of the *environmental feature* that gives a very small optimality value (set to be 0.0001).

H: The maximum value in the data for the current *environmental feature*. This value only shows up under the “Data Range” option.

### 1.5.6.1.4. Adjusting the Function Curve

**Adjusting the Curve Using the Input Fields**

You can directly type in values for \(v_1, w_1, r_1, v_2, w_2, \text{ and } r_2\) in their corresponding input fields to adjust shape of the function curve. ArcSIE uses a fixed value, 0.5, for both \(c_1\) and \(c_2\), so there are no input fields for adjusting \(c_1\) and \(c_2\).

**Adjusting the Curve Using the Graphic Tool**

For the *continuous* and *cyclic* functions, you can use the mouse to adjust the shape of the curve in a *click-and-drag* manner. To adjust the curve, put the cursor close enough to either of the two red square handles on the sides of the green curve, click, hold, and drag. Dragging a handle also causes the value in the corresponding \(w\) field and the “middle” and “bottom” values on the curve graphic to change accordingly.

You cannot use the graphic tool to adjust the *ordinal* and *nominal* functions. To make changes to these functions, you need to explicitly type in values for \(v_1\) and \(v_2\) in their corresponding input fields.

### 1.5.6.1.5. Applying an E Function to all the Incidences of a Soil Type or Rulebase
Right-clicking an *environmental feature* in the *environmental feature* list window opens a pop-up menu. This menu allows you to apply the setting of the $E$ function for the selected *feature* of the current *instance* to the corresponding *features* in all the *instances* in the current *soil type* or even the entire *rulebase*.

When you apply the setting for a feature in the current *instance* to other *instances*, only the values of $w$ and $r$ in the corresponding *features* of other *incidences* will be modified. The original values of $v_1$ and $v_2$ of those *incidences* will not change.
1.5.6.2. Adjust the $P$ Function

The $P$ in the general inference equation integrates the fuzzy membership values calculated based on individual *environmental features*. The output of the $P$ function is the fuzzy membership value based on the whole *instance*. ArcSIE implements three methods for the $P$ function: Limiting-Factor, Weighted-Average, and Multiplication. You can choose a method by moving the cursor to the header of the first column in the *feature* list window and right-clicking. A pop-up menu will appear and you can select the method from this menu (see the image below).
1.5.6.2.1. Limiting-Factor

For the limiting factor method, ArcSIE uses the MIN operation in fuzzy logic to integrate the optimality values of individual environmental features. In other words, it chooses the minimum value among those values as the overall output value of the whole instance. The theoretical basis for this option is the limiting-factor principle in ecology.

If you choose to use the limiting-factor method, the appearance of the feature list window will change accordingly allowing you to specify if an environmental feature in the current feature list should or should not to be used in the inference. You do this by checking on or off an individual feature.

1.5.6.2.2. Weighted-Average

Under the Weighted-average method, ArcSIE calculates a linear weighted average of the optimality values of individual environmental features to get the overall output value of the whole instance.

If you choose to use this method, the interface of the feature list will change accordingly and allow you to assign weight to each feature.

ArcSIE provides two ways to specify weights for the environmental features:

- You can directly type in the weigh values in the feature list window.
- You can use the Analytic Hierarchy Process (AHP) tool to determine the weights in a more structured way.

When you choose to use the weighted average method, the “Weight Features” button on the Inference dialog box will become available. Clicking this button brings the AHP interface to the front:
The basic process of the AHP method contains two steps: first, the user conducts pair-wise comparisons on the features using a nine-score scale, and second, the computer performs matrix calculation using the score matrix to determine the weight for each feature. Refer to the image in the next page, for example, you can specify that “slope is strongly more important than elevation”. With this specification, you eventually specify that the score of slope against elevation is 5, and the score of elevation against slope to be 1/5. In the same way, you compare all the pairs that can be formed by the environmental features in the current environmental database. When you have all the pairs done, you get the weights of the environmental feature right away.

To conduct the comparison using the AHP tool in ArcSIE, click an environmental feature in the left list box to select it; select a score from the score list, referring to the natural language descriptions of the scores; select the other environmental feature from the right list box; and then click the “Apply” button. The result of the operation will appear in the list window. The design of the tool requires you to always select the “stronger” environmental feature in a pair from the left list box.

You can save the AHP matrix you create into an AHP file by clicking the “Save” button and “Load” it in another time.

When you are done, click the “OK” button and return to the Inference dialog box. The weights will appear in the feature list window of the Inference dialog box.
### Weighting Environmental Factors

<table>
<thead>
<tr>
<th></th>
<th>Slope</th>
<th>PlanformCurvature</th>
<th>meanWetness</th>
<th>Elevation</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slope</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>5</td>
<td>0.54</td>
</tr>
<tr>
<td>PlanformCurvature</td>
<td>1/4</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>0.21</td>
</tr>
<tr>
<td>meanWetness</td>
<td>1/3</td>
<td>1/2</td>
<td>1</td>
<td>4</td>
<td>0.18</td>
</tr>
<tr>
<td>Elevation</td>
<td>1/5</td>
<td>1/3</td>
<td>1/4</td>
<td>1</td>
<td>0.07</td>
</tr>
</tbody>
</table>

- **meanWetness**
  - (1) is equally important as
  - (2)
  - (3) is moderately more important than
  - (4)
  - (5) is strongly more important than
  - (6)
  - (7) is very strongly more important than
  - (8)
  - (9) is extremely more important than

- Elevation
  - (1)
  - (2)
  - (3)
  - (4)
  - (5)
  - (6)
  - (7)
  - (8)
  - (9)
1.5.6.2.3. **Multiplication**

The *Multiplication* method calculates the product of the values from the individual *environmental features* and uses it as the overall output value from the whole instance.

The *Multiplication* method is useful when you want to represent the interactions among *environmental features* or when you want to mask an inference result.

1.5.6.3. **Positive Instance vs. Negative Instance**

An instance can be *positive* and *negative*. A *positive* instance defines how the optimality value decreases as the *environmental feature* value deviates from the most optimal condition. A typical *positive* function curve is like below:

![Positive Function Curve](image)

A *negative* instance defines how the optimality value increases as the *environmental feature* value deviates from the least optimal condition. A typical negative function is like below:
By default, an instance is *positive*. You can specify an instance to be *positive* or *negative* by toggling between the two radio buttons.

A *Soil Type* can contain either *positive* or *negative* instances or both.

- If it contains only *positive* instances, the results from individual instances will be integrated (the $T$ function in 1.3.5.2) using the MAX operation.
- If it contains only *negative* instances, the results from individual instances will be integrated using the MIN operation.
- If it contains both, the inference engine will first separate the two types of instances and run inferences with them separately. The results from the two types of instances will then be integrated using the MIN operation.
1.5.6.4. Partial Membership Instance

By default, the maximum fuzzy membership defined by a *positive* instance for the most optimal condition is 1, the full membership; and the minimum fuzzy membership defined by a *negative* instance for the least optimal condition is 0. You can specify other values for the maximum/minimum membership for an instance in $\frac{0.8}{0.8}$. The value you specify must be between 0 and 1.

- The fuzzy membership value given by a *positive* instance will vary between 0 and the maximum value you specify.
- The fuzzy membership value given by a *negative* instance will vary between the minimum value you specify and 1.

The mark values on the vertical axis of the curve plot will change in accordance with the value you specify. For example, if you specify the maximum value for a *positive* instance is 0.8, the mark values will be as follows, where 0.4 is the middle value between 0 and 0.8:
If you specify 0.2 as the minimum value for a *negative* instance, the mark values will be as follows, where 0.6 is the middle value between 0.2 and 1.

![Graph showing mark values for P function](image)

1.5.6.5. **Turning on/off an Environmental Feature**

When the P function is *Limiting Factor* or *Multiplication*, you can turn on or turn off an *environmental feature* by checking the small box before the *feature* name in the *feature* list window.

If you turn off a *feature*, the *feature* will not be used in the inference and the function curve defining area will be grayed (disabled) for this *feature*.

If you turn off all the *features* of an *instance*, you eventually turn off the *instance*, i.e., the *instance* will not be used in the inference.

When you load a new *environmental database* or add new environmental data layers, the default state of the corresponding new *environmental features* is *off*.
1.6. Preparing a Vector (Point, Line, or Polygon) Casebase

1.6.1. Creating a New Vector Casebase

Note: To create a new knowledgebase, you must first create or load an environmental database. However, a loaded environmental database can be used for all the knowledge types or forms, so you do not have to reload it after you switch the knowledge type or form.

ArcSIE uses a Shapefile to create a new vector casebase.

To create a new vector casebase, click [button]. The Read Shapefile dialog box will appear requiring you to select the Shapefile. You also need to select the field in the attribute table of the Shapefile containing the soil type names. The data type of the specified attribute field must be “string”. The values in this field will be used to create soil types in the casebase. ArcSIE automatically checks the data types of all the fields in the attribute table and lists those fields whose types are “string” as candidates for you to select.
You CANNOT create new soil types or new cases for a casebase in ArcSIE. Cases and the soil types they belong to can only be created in a GIS, e.g., 3dMapper or ArcGIS.

At any time, ArcSIE can host only one point casebase, one line casebase, and one polygon casebase. Therefore, for example, if there is a point casebase currently loaded, it will be replaced by the new point casebase you are creating. You will be prompted to save the changes you made to the old casebase before it is unloaded.

In a vector casebase, all the cases for one soil type are organized into a soil type named after that soil type.

The default names for the soil type and cases are the soil names in the selected field from the Shapefile’s attribute table. The name of a case is created by adding an integer to the soil name to make the case’s name unique.
1.6.2. Loading an Existing Vector Casebase

Loading an existing vector casebase has the exactly same procedure of loading an existing rulebase (see 1.4.2), except the file format for casebase is Shapefile. A Shapefile of casebase contains some special attribute fields added by ArcSIE.
If the type of the specified Shapefile does not match the specified case form (point, line, or polygon), or the required attribute fields for a casebase are not found in the attribute table of the Shapefile, an error message will appear and the loading process will abort.

1.6.3.  Saving Vector Casebase

Selecting the casebase name and clicking saves the current vector casebase into a vector casebase file, which is in the Shapefile format. The information about the soil types and cases in this casebase are saved into the attribute table of this Shapefile.

1.6.4.  Unloading Vector Casebase

To unload the current vector casebase from ArcSIE, right-click on the casebase name and select Remove.

You CANNOT remove a soil type or a case from a vector casebase. You can only do this in a GIS, e.g., 3dMapper or ArcGIS.

1.6.5.  Editing the Rule Part of a Case

Editing the rule part of a vector case is exactly the same as editing an instance. See 1.5.6.

Note: For a point case, the feature values (v₁ and v₂) and coordinates (x and y) displayed in the dialog box are the values and coordinates of the location of the case.
For a line case, the $x$ and $y$ displayed in the dialog box are coordinates of one of the “points” in the series. The $v_1$ and $v_2$ fields are disabled under the Line Case option so that the user cannot type in arbitrary values. The displayed central value ($v_1$ always equals to $v_2$ under the Line Case option) can be different statistics of the pixels that are along the line case. The user can specify which statistic to display on the dialog box and to use in the inference.

For a polygon case, the $x$ and $y$ displayed in the dialog box are coordinates of one of the “points” in the agglomeration. The $v_1$ and $v_2$ fields are disabled under the Polygon Case option so that the user cannot type in arbitrary values. The displayed central value ($v_1$ always equals to $v_2$ under the Polygon Case option) can be different statistics of the pixels that are within the polygon case. The user can specify which statistic to display on the dialog box and to use in the inference.

1.6.6. Editing the Spatial Setting of a Case

The spatial setting defines the spatial aspect of a case. It includes the geographical location of a case, its influence region, and other specifications for adjusting the optimality values calculated based on environmental features.

1.6.6.1. Global Case vs. Local Case

Global CBR and local CBR are distinguished by whether the cases are applied to the entire mapping area or applied to only a limited region (Influence Region). Cases used for these two types of CBRs are called global cases and local cases, respectively.

A global case is considered exactly the same as an instance in RBR, except that the central values of the case are not directly specified by the user, but are found out by the inference engine from which
the environmental data layers according to the case’s location. A local case contains certain parameter values for the local CBR, which makes it somewhat different from an instance.

Using [Spatial Setting >>], you can toggle between using and not using a spatial setting, i.e., make the cases global or local. When the spatial setting panel of the Inference dialog box is showing, the values of the parameters displayed in the panel will be applied to the currently selected case and the case becomes a local case. If you close the spatial setting panel, the cases in the current casebase become global cases, meaning that they do not have influence regions associated with them and during the inference they will be applied to the whole mapping area.

ArcSIE does not allow a mixed use of global cases and local cases. Whether the spatial setting panel (see below) is open or closed when you start the inference determines whether the cases in the current casebase are used as global cases or local cases in the inference. If the panel is open, all the cases will be used as local cases; and if closed, all the cases will be used as global cases.
On the spatial setting panel, $x$ and $y$ fields show the coordinates of the location of the current case. The location of a case is specified by the soil scientist when creating the case and cannot be modified in ArcSIE.

For local cases, you need to specify three spatial settings: Influence Region, Distance-adjusted Fuzzy membership, and $T$ Function.

### 1.6.6.2. Influence Region

Influence Region defines the spatial extent affected by a case.
For a **point case**, the *Influence Region* is defined around the point of the case.

For a **line case**, the *Influence Region* is defined for each “point” in the *line case*, i.e. each pixel passed through by the line.

For a **polygon case**, the *Influence Region* is defined for each “point” in the *polygon case*, i.e., each pixel within the spatial extent of the polygon.

The *Influence Region* is defined by two parameters: *Search Distance* and *z Factor*.

- **Search Distance**

*Search Distance* is the distance for defining a buffer zone around a case location. However, in ArcSIE this distance is not a Euclidean distance between two locations but the distance along the terrain surface, as illustrated by the figure below:
Measuring the distance in this way may result in an irregular-shaped Influence Region.

The unit of the Search Distance is the unit used by the environmental database.

- **z Factor**

  The z factor allows you to adjust the importance of the vertical variation (i.e., change of elevation) in calculating the Search Distance. If you want to emphasize the importance of vertical variation, you can specify a value greater than 1 for the z Factor to exaggerate it. A z Value between 0 and 1 understates the vertical variation.

### 1.6.6.3. Distance-adjusted Fuzzy Membership

You can use distance information to adjust the optimality value calculated using the rule part of a case for a given location. The basic idea is that the geographically closer a location to a case, the more similar the location to the case. The function of this adjustment is shown below:

\[
 s'_{ij,t} = s_{ij,t} e^{ \left( d_{ij,t} / w_t \right)^{z} \ln(0.0001)} 
\]

The meanings of the symbols in the equation are as follows:

- \( s'_{ij,t} \): the distance-adjusted fuzzy membership at location \((i,j)\) calculated based on case \(t\).
- \( s_{ij,t} \): the fuzzy membership value calculated using the rule part of case \(t\) (i.e., the output from function \(P\) in the general inference equation described in 1.3.4.2).
\(d_{ij,t}\): the *surface distance* between the location \((i, j)\) and *case* \(t\).

\(w_t'\): the *Search Distance* of *case* \(t\);

\(r'\): performs the same function as that of \(r\) in the general inference equation described in 1.3.4.2.1.2.

When location \((i, j)\) is within the Inference Region of *case* \(t\), the value of the \(e\) part of the above equation varies between 0.0001 and 1 making continuous adjustment of \(s_{ij,t}\)

The equation above defines a function curve by specifying that if the distance between a location and *case* \(t\) is the *Search Distance*, that location has a very small similarity value to \(t\).

By checking the box, you specify that you want to perform this adjustment. If you want to perform this adjustment, you need to specify values for \(w_t'\) and \(r'\).

**1.6.6.4. T Function**

Refer to the general inference equation, in CBR, the *T* function is used to integrate information from multiple *cases*. The *T* function is particularly important in local CBR, as a location can fall into the influence regions of multiple cases and the soil scientist may consider more complicated relationships among the cases than that in global CBR. The options implemented by ArcSIE for \(T\) are listed in the table below.
<table>
<thead>
<tr>
<th>Category</th>
<th>Algorithm</th>
<th>Description</th>
<th>Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>Using a single value</td>
<td><strong>Dominant</strong></td>
<td>Always use the value calculated based on this <em>case</em> no matter what values from other <em>cases</em> are. This indicates that the current <em>case</em> has an exclusive influence on the locations within the <em>case’s Influence Region</em>.</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td><strong>Maximum/Minimum</strong></td>
<td>Among the values calculated based on different <em>cases</em>, use the maximum/minimum one. This indicates that the most similar <em>case</em> has an exclusive influence. For <em>positive</em> cases, use maximum operation; and for <em>negative</em> cases, use minimum operation.</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td><strong>Nearest</strong></td>
<td>Use the value calculated based on this <em>case</em> if this <em>case</em>, among all the cases whose influence regions cover the location under concern, is geographically nearest to that location. This indicates that the nearest <em>case</em> has an exclusive effect.</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td><strong>Supplement</strong></td>
<td>Use the calculated value based on this <em>case</em> only when no other <em>cases</em> can provide non-zero values to the location under concern.</td>
<td>1.5</td>
</tr>
<tr>
<td>Using multiple values</td>
<td><strong>Similarity-weighted</strong></td>
<td>Use the weighted average of the values calculated based on different <em>cases</em>, in which the weights are the values themselves. This indicates that a more similar <em>case</em> should have more influence.</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td><strong>Inverse-distance</strong></td>
<td>Use the weighted average of the values calculated based on different <em>cases</em>, in which the weights are inverses of the <em>surface distances</em> between the location under concern and the <em>cases</em>. This indicates that a closer <em>case</em> should have more influence.</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td><strong>Average</strong></td>
<td>Use the simple average of the values calculated based on different <em>cases</em>.</td>
<td>6</td>
</tr>
</tbody>
</table>
Different functions have different priorities. If a location is within the Influence Region of multiple cases and these cases have been assigned different T functions, the case whose function has the highest priority will determine how to integrate the values calculated from different cases. The Supplement function will have effect only when the values from all the other cases are zero, therefore it can be considered to have a conditional highest priority. When the functions of two cases have the same priority, the case that is more similar to the given location has a higher priority.

Both Similarity-weighted and Inverse-distance functions calculate weighted average, thus the basic mathematical representation of both functions is:

\[
S_{ij,k} = \frac{\sum_{t=1}^{n}(d_{ij,t})^{-q}s_{ij,t}}{\sum_{t=1}^{n}(d_{ij,t})^{-q}}
\]

The meanings of the symbols in this equation are as follows:

- \(s_{ij,k}\): the final fuzzy membership at \((i, j)\) for soil \(k\) (i.e., the output from function \(T\)).
- \(s_{ij,t}\): the fuzzy membership value calculated based on case \(t\).
- \(d_{ij,t}\): For the Similarity-weighted function, this is \(s_{ij,t}\) again; for the Inverse-distance function, this is the surface distance between \((i, j)\) and \(t\).
- \(q\): the decay factor.
1.6.6.5. **Applying the Current Spatial Setting to all the Cases in a Soil Type or Casebase**

Right-clicking in the spatial-setting area opens a pop-up menu and allows you to apply the spatial setting of the current case to all the cases in the current soil type or even the entire casebase.

1.6.7. **Positive Case vs. Negative Case**

A case can be positive and negative. The way to define and work with positive and negative cases is the same as that of defining positive and negative instances (See 1.4.6.3.).

1.6.8. **Partial Membership Case**

If you pinpoint or delineate typical locations to create cases, those cases are usually assigned full fuzzy membership for the soil they represented. ArcSIE can also work with cases at non-typical locations, i.e., cases with partial memberships. The way to define and work with partial membership cases is similar to that of partial membership instances (See 1.4.6.4.).
1.7. Preparing a Raster Casebase

1.7.1. Components of a Raster Casebase

Different from a rulebase (a single DBF file) or a vector casebase (a single Shapefile), a raster casebase is composed of two parts:

- a DBF file containing rules and spatial setting;
- a raster layer containing pixels.

A raster casebase only contains one set of rules and one spatial setting. This one set of rules and one spatial setting will be applied to all the pixel cases in the raster layer.

1.7.2. Creating a Raster Casebase

ArcSIE uses an ArcInfo Grid to create a new raster casebase.

To create a new casebase, click Open. In the File Browsing dialog box specify the Grid you want to use.

The spatial extent and cellsize of the Grid must match the settings of the environmental database.

One Grid can only contain cases of one (soil/landform) type. The case pixels should have value 1 and the non-case pixels should have value 0.
At any time, ArcSIE can host only one raster casebase. Therefore, if there is a raster casebase currently loaded, it will be replaced by the new raster casebase. You will be prompted to save the changes you made to the old casebase before it is unloaded.

With a raster casebase you cannot add a second set of rules or remove the current set of rules.

1.7.3. Loading an Existing Raster Casebase

Click  ![Open](image) to load an existing raster casebase. You only need to specify the DBF file. ArcSIE will automatically load the corresponding raster layer.

If there is a raster casebase currently being used, the current one will be replaced by the one you are opening. You will be prompted to save the changes you made to the current raster casebase before it is unloaded.

If the DBF file does not contain the required information for a raster casebase, an error message will appear and the loading process will abort.

1.7.4. Editing the Rule Part of Pixel Cases

Editing the rule part of the pixel cases is exactly the same as editing an instance. See 1.5.6.

Note: The central values ($v_1$ and $v_2$) and coordinates ($x$ and $y$) displayed in the dialog box are the values and coordinates for one of the pixel cases in the raster casebase. These values only provide general information about the casebase. The actual values of each pixel case will be used in the inference.
1.7.5. Editing the Spatial Setting of Pixel Cases

Editing the *spatial setting* of the *pixel cases* is the same as editing the *spatial setting* of a *vector case*. See 1.6.6.

**Note:** The *Influence Region* is defined for each *pixel case* in the raster *casebase.*
1.8. Running Inference

The soil maps created by ArcSIE are fuzzy membership maps indicating how similar the soils at different locations are to the most typical soils of the specified soil types. The maps use the raster data model and are in ArcInfo Grid format.

1.8.1. Creating a Map for the Currently Selected Soil Type

Click and give a name to the output soil map. ArcSIE will run inference for the currently selected soil type or the soil type that contains the currently selected instance or case.

If the current selection is the knowledgebase name, ArcSIE will run inference for the first soil type in the knowledgebase.

1.8.2. Creating Maps for all the Soil Types

Click and specify a folder to host the created maps. ArcSIE will then create a set of fuzzy membership maps, one for each soil type in the current knowledgebase.

The maps will be named after their corresponding soil types.

Since all the pixel cases are of the same (soil/landform) type, the “Do Batch” function is not applicable to a raster casebase.

1.8.3. More Options on the Output
Clicking opens a dialog box in which you can specify:

- if you want to create a check file;
- if you want to use mask file in the inference;
- how you want to determine the central values for a line or polygon case; and
- if you want to resample the pixel cases before running the inference.

### 1.8.3.1. *Check File*

A check file is a map showing two pieces of information:

- which instance or case determines the fuzzy membership at a location; and
- which environmental feature determines or makes the greatest contribution to the fuzzy membership at the location.

A check file is an ArcInfo Grid and is named by adding “chk” at the end of the output soil map.

Each value in a check map has two parts: The integer part is the instance’s ID, indicating the determining instance at that location. The instance’s ID corresponds to the order of the instance in the soil type. If none of the existing instance has effect on a location, the integer part of the check file value at that location will be -1.

The decimal part is the environmental feature’s ID, indicating the determining environmental feature at that location. To find out the name of the feature, you need to check the look-up table created with the check file. The look-up table is a text file and has the same name as that of the check file, but has a suffix “lkt”.

65
For example, if a check file has value 3.04 for a location and its associated look-up table is as follows:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Elevation</td>
</tr>
<tr>
<td>1</td>
<td>Slope</td>
</tr>
<tr>
<td>2</td>
<td>Aspect</td>
</tr>
<tr>
<td>3</td>
<td>ProfileCurve</td>
</tr>
<tr>
<td>4</td>
<td>PlanformCurve</td>
</tr>
<tr>
<td>5</td>
<td>Wetness</td>
</tr>
</tbody>
</table>

The user will know that the fuzzy membership at this location is determined by the third instance or case and planform curvature is the environmental feature that determines or makes the greatest contribution to the fuzzy membership.

If Limiting Factor is used for the $P$ function, the environmental feature reported by the check file for a location is the one that has the smallest optimality value at that location.

If Weighted Average or Multiplication is used for the $P$ function, the environmental feature reported by the check file for a location is the one that makes the greatest contribution to the fuzzy membership. For Weighted Average, this environmental feature is the one that has the maximum (optimality value x weight); For Multiplication, this environmental feature is the one that has the maximum optimality value.

1.8.3.2. Mask File

A mask file is for masking out areas that you do not want to include in the inference. Those masked areas will be assigned Nodata in the created soil map.
If the value of a cell in the mask file equals to the specified Masking Value, the corresponding cell in the output soil map will be assigned Nodata.

This masking function is an “in or out” function, i.e., it simply classifies all the locations covered by the environmental database into two classes: locations that should be included in the inference and locations that should be excluded from the inference. This masking function is different from the masking function associated with an individual soil type, instance or case.

This masking function is mainly for defining the mapping area when you want to set a special boundary for the output soil map, e.g., when you only want to map the soils within a watershed.

1.8.3.3. Central value for line/poly case

When using a line/polygon case to perform inference, the line or polygon in vector form will first be resterized. That is, in the actual inference, a line will be treated as a series of pixels and a polygon will be treated as an agglomeration of pixels.

For a line/poly case, ArcSIE allows the user to specify what value to display in the $v_1$ and $v_2$ fields (for line/poly cases, $v_1$ always equals to $v_2$) and to use in the inference. ArcSIE provides four options:

(1) Use the local value at each pixel;

(2) Use the mean of all the pixels of this case;

(3) Use the median of all the pixels of this case;
(4) Use the mode of all the pixels of this case. "Mode" refers to the most frequently appearing value. When calculating the mode, all the values within the tolerance range of a value will be counted as if they are equal to that value i.e., if a value is between x - tolerance and x + tolerance it will be considered equal to x.

For options (1) and (2), the \( v_1 \) and \( v_2 \) fields will show the mean value; for (3) and (4), the fields will show the corresponding values.

In the inference, each pixel in the line/poly case is treated as a point case. Under option (1), the central values of such a point case are read from the pixel’s location; other than that, these point cases are exactly the same in terms of their \( E, P \) and \( T \) function settings, as they inherit these settings from the line/poly case they belong to. Under options (2), (3), and (4), all the point cases are exactly the same (including the central values), except the spatial locations.

1.8.3.4. Resampling pixel cases

When the spatial resolution of the raster casebase is high, the number of the pixel cases can be big, which results in a slow inference. You can resample the pixel cases to reduce the number of cases used in the inference.

The Resample Ratio should be an integer \( \geq 1 \). When this ratio \( \leq 1 \), no resampling will be performed. When it is 2, one from two nearby pixel cases will be used in the inference; when it is 3, one from three nearby pixel cases will be used in the inference; and so on.

1.8.4. Masking Function for a Soil Type or Instance/Case
You can associate a mask file to a soil type, instance or case to limit its effective area. To specify the settings for this function, right-click the name of the soil type, instance or case and select Mask from the popup menu. A dialog box will appear as follows:

Checking on the Mask checkbox enables the masking function for this soil type or instance and also activates the other options on this dialog box. The values in the mask file should vary between 0 and 1. In the inference, the inferred values for the soil type or from the instance or case will be multiplied by the corresponding values in the mask file. The multiplication operation allows the effective area to have a fuzzy boundary. If you check on Inverse values, calculation (1 – value) will be performed on the mask file before it is applied to the inferred values.

Checking off the Mask checkbox disables the masking function for this soil type, instance or case, even if you have associated a mask file to the soil type, instance or case.
1.8.5. Two Types of Polygon-based CBR

ArcSIE implements two ways to use polygon cases:
- Running point CBR using each location within the polygon as a typical point case
- Making fuzzy boundary of the polygon

1.8.5.1. Running point CBR using each location within the polygon as a typical point case

This inference assumes that the polygon defines a typical region for the given soil and every location within the polygon is a typical location for the soil. The inference engine will first rasterize the polygon and will use each pixel within the polygon as a point case to conduct point CBR.

All the “point cases” from the same polygon case will share the setting of the polygon case, except \( v_1 \) and \( v_2 \), which will be the actual value at each specific point. These two values for each “point case” will be equal and cannot be modified. The \( v_1 \) and \( v_2 \) fields are disabled under the Polygon CBR option.

1.8.5.2. Making fuzzy boundary of the polygon

If you consider that the polygon only roughly define the geographic region of a soil, and you can assume that the closer to the central part of the polygon, the more typical condition for the soil, you can use the Making Fuzzy Boundary function, available under the Polygon CBR option.

ArcSIE provides ways for you to specify how fuzzy and how broad the fuzzy boundary should be. The process to make a fuzzy boundary of a polygon is as follows:

a. Choose to conduct “Polygon Case” inference.
b. Load the polygon case Shapefile.
c. Specify positive or negative for each case, if you have not done that before.

d. Specify the “Membership Value” for each polygon case. Different from a normal CBR, this value will not be used as the maximum/minimum value for the cases, but will be used as the fuzzy membership values at the boundary of the polygons. For example, if you specify the value to be 0.4, then the inference engine will assign 0.4 as the fuzzy membership to the pixels right on the boundary of the polygon and will base other inference computing on this setting. Under the fuzzy boundary operation, the maximum fuzzy membership value for a positive case is always 1 and the minimum value for a negative case is always 0.

e. Load the environmental features you want to use and adjust the function curves, just like what you will do for all other types of inferences. However, for making fuzzy boundary, you should only use bell-shape or Default function curves.

f. Check [Make Fuzzy Bnd].

g. The Spatial Setting panel will be automatically open, because making fuzzy boundary operation can only be done spatially. The “Spatial Setting” button is disabled to avoid accidentally closing the Spatial Setting panel. Adjust the parameter values on the spatial setting panel as needed.

h. Click the appropriate one of the two “Do” button to run the inference.
Chapter 2. Validation

2.1. Generate Error Matrix

Error Matrix is a tool for evaluating accuracy of nominal data. It can be used to compare the soil type names inferred by ArcSIE with the type names assigned by the soil scientist based on field work.

Below is an example of error matrix generated by ArcSIE:

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>15</td>
<td>1</td>
<td>6</td>
<td>68</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>18</td>
<td>3</td>
<td>72</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>6</td>
<td>22</td>
<td>73</td>
</tr>
<tr>
<td>Percent</td>
<td>71</td>
<td>72</td>
<td>71</td>
<td>71</td>
</tr>
</tbody>
</table>

The first column contains the soil type IDs (1, 2, and 3) in the inference result, and the first row contains the soil type IDs (1, 2, and 3) in the soil scientist's decision. The numbers in the 2nd row of the matrix indicate that at 15 samples sites, both the inference engine and the soil scientist name the soils as type 1; at 1 site the inference engine names the soil as type 1, but the soil scientist names the soil as type 2; at 6 sites the inference engine names the soils as type 1, but the soil scientist names them as type 3; and overall among all the 22 (15+1+6) sample sites whose soils have been named by the inference engine as type 1, 68% (15/22) of them match the soil scientist's decisions. On the other hand, the last number in the 2nd column tells that among all the 21 (15+4+2) samples that have been named by the soil scientist as type 1, the inference engine have got 71% (15/21) of them right. The overall accuracy of the inference result is 71% (the number in the last row and the last column).
2.1.1. Generate with Map

This tool generates an error matrix with a hardened map (the inference result). This tool requires three inputs:

- A Point Shapefile of the sample sites
- A Field in the Attribute Table of the Point Shapefile: This field should contain the IDs of the soil types assigned by the soil scientist. This field must be an integer field.
- A Raster Layer of the hardened map.

2.1.2. Generate with Points

This tool generates an error matrix using two sets of corresponding values (paired values). The two sets of values should be stored in a "Value Table File" in the DBF format (e.g., the Attribute Table of a Point Shapefile).

2.1.3. Output

Both tools generate two outputs:

- Error Matrix File: An ASCII file containing the error matrix.
- Soil List File: An ASCII file containing a complete list of all the soil types (represented by their IDs) appearing in either the inference results or the soil scientist's decisions, or both.
2.2. Generate Property Map

This tool generates property map based on type maps. The equation used by this tool is as follows:

\[ D = \frac{\sum_{k=1}^{n} s_k d_k}{\sum_{k=1}^{n} s_k} \]

where \( D \) is the property value at a given location, \( s_k \) is the fuzzy membership value of soil type \( k \) at that location, \( d_k \) is the typical property value of soil \( k \), and \( n \) is the total number of soil types prescribed in the soil-landscape model used by the inference engine. The basic idea here is to use the fuzzy membership values of different soil types at a location as the weights to calculate the weighted average of the property values of different soil types.

You are going to use the dialogbox of "Creating Property Map" as below to create a list of input fuzzy membership maps, and assign the "typical" property value for each map (representing a soil type).
You can use the dropdown list to add fuzzy membership maps from the current ArcMap project, or use the browse button to load them from the disk.

The default typical property value is 0. You need to change it to the correct value.

When you finish creating the property-layer list, you can save the to a look-up table file (.lkt) using the save button.

You can load a look-up table from the disk using the load button.
The output is a raster layer containing the property value at each location.

2.3. Determining Sampling Sites

This tool provides four different ways to determine sampling sites.

2.3.1. Global Random

This method randomly picks locations within a rectangular region. The user needs to specify:

- the “Total Number of Samples”
- the coordinates for defining the “Sampling Region”.

2.3.2. Stratified

This method first divides the entire rectangular sampling region into regular rectangular “Sub-regions”, and then randomly picks locations within each sub-region. The user needs to specify:

- the “Total Number of Samples”
- one of the two ways to determine the number of samples in each sub-region:
  - sets a fixed number of samples within each sub-region. The program will then automatically calculate the number of sub-regions and their sizes accordingly. If the remaining of total
samples/sub-region samples is not zero, more sample sites than the specified total number will be generated to ensure every sub-region has equal number of samples.

- sets a fixed size for sub-region. The program will then automatically calculate the number of samples in each sub-region. If the specified total number of samples cannot be equally allocated to the sub-regions, more sample sites than the specified total number will be generated to ensure every sub-region has equal number of samples.

- the coordinates for defining the “Sampling Region”.

2.3.3. Systematic

This method allocates samples following a regular grid. The user needs to specify:

- the “Total Number of Samples”

- the coordinates for defining the “Sampling Region”.

2.3.4. Regional Random

This method picks random locations within polygons in a Shapefile specified by the user. The user also needs to specify a field in the attribute table of the Shapefile that contains the number of samples the user wants to pick in each polygon.
Chapter 3. Terrain Analysis

3.1. Filling Pits

This tool removes unwanted pits from DEM, which is a necessary preprocess in many terrain analysis operations.

Filling pits is mainly for avoiding unwanted stops when modeling flow paths using DEM. This should be done before calculating flow direction, flow accumulation, catchment, and wetness index.

The user needs to specify the maximum depth of pits that should be filled. All basin areas whose depths are smaller than the specified depth will be filled. A basin area is defined as an area whose water finally accumulates at the bottom of this area rather than flowing out through an outlet. The depth of a basin area is defined as the difference between the lowest point in the interior (i.e., the bottom) and the lowest point on the edge of the area (i.e., the outlet). After filling, all the interior pixels will be assigned the elevation value of the lowest point on the edge, i.e., a flat region will be created in the original basin area.

If you want to fill all internal catchments so that all the water flows to certain outlets at the edges of the DEM, you need to specify a large value for the maximum depth.

3.2. Shaving Spikes

Shaving spikes may remove many artifacts in high-resolution DEM and may also reduce the number of unreasonable isolated ridge pixels when creating a ridge layer from a DEM.
The process contains two general steps: identifying spikes and removing them.

The tool provides two methods for identifying spikes: automatic and user-specifying. The user can create a point Shapefile to specify the locations that he/she thinks to be spikes. For each point, the user can specify the size of the spike (stored in a "size" field in the attribute table), so that the program knows how big the spike it is dealing with. If the user does not provide the point Shapefile, the program will identify spikes using an automatic algorithm and apply a constant size (Default Size of Spike) to every spike it finds. If the user provides the point file but no "size" field, the constant user-specified default size will be applied to every user-specified spike.

Then the program replaces the values in the spike area with the values derived from its surrounding area. The higher the Distance Decay Factor, the less smoother the resulting new surface. The Number of Iterations specifies the times the removing process repeats. This option is only applied to the automatic process.

You can specify to apply the removing process only to the areas that are below a certain elevation.

### 3.3. Surface Derivatives

This is a toolbox for calculating some most commonly used terrain attributes, including slope gradient, aspect, and various types of curvatures. Mathematically these attributes can be considered as the first- or second-order derivatives of the continuous terrain surface.

#### 3.2.1 Terrain Attributes
Gradiente: Slope gradient, measuring the steepness at a location. The output is in percentage (45 degree = 100%).

Aspect: Slope aspect, giving the direction the slope facet is facing. The output is in degree, starting from north and increasing in the clockwise direction (east = 90, south = 180, west = 270).

Profile curvature: Measures the shape of the slope surface in the vertical direction. Negative values indicate convex shapes and positive values indicate concave shapes.

Planform curvature: Measures the shape of the slope surface in the horizontal direction. Negative values indicate convex shapes and positive values indicate concave shapes.

Tangent curvature: Measures the shape of the slope surface in the direction that is perpendicular to the surface at the given location. Negative values indicate convex shapes and positive values indicate concave shapes.

Min curvature:

Max curvature:

Curvature: Measures the overall shape at a location. Negative values indicate convex shapes and positive values indicate concave shapes.

Note: For all the curvatures, the sign (indicating convex or concave) in the result from ArcSIE is reversed to that in the results from ArcToolbox.
3.2.2. Algorithms

**Evans-Young:** Based on the idea that the terrain surface can be modeled by a quadratic polynomial.

**Horn:** Similar to Evans-Young method, but in the specific (finite-element) calculation gives the cells in the cardinal directions heavier weights.

**Zevenbergen-Thorne:** Based on the idea that the terrain surface can be modeled by a Lagrange polynomial.

**Shi:** A modified Zevenbergen-Thorne method. It first applies the Zevenbergen-Thorne method to the four cardinal pixels and four diagonal pixels separately, and then uses the average of the results from the two operations as the final result.

**Note:** For all the curvatures, the “Horn” option is equivalent to the “Evans-Young” option, and the “Shi” option is equivalent to the “Zevenbergen-Thorne” option.

3.2.3. Neighborhood size

Traditionally, the neighborhood size is determined by the pixel size. ArcSIE allows the user to define arbitrary size for the neighborhood. The size is measured from point to point. For example, with a 30m DEM, the size of a traditional 3x3 neighborhood is 20m, because the size is measured between the centers of pixels. The values of those elevation values on the neighborhood edge (needed for calculating the terrain attributes) are calculated through bilinear interpolation.

3.2.4. Neighborhood shape
The traditional neighborhood for calculating the terrain attributes is determined by 3x3 contagious pixels and therefore is a square, which leads to directional bias. ArcSIE implements a circular neighborhood to mitigate that problem. The two kinds of neighborhoods are illustrated as follows:

References:


3.4. Flow Direction

Uses D8 (assumes that water only flows into the neighboring cell that is in the steepest direction) to calculate flow direction. The code representing the direction at each cell is illustrated as follows:

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>64</td>
<td>128</td>
<td>10</td>
</tr>
<tr>
<td>32</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>16</td>
<td>8</td>
<td>4</td>
</tr>
</tbody>
</table>

3.5. Flow Accumulation

This tool sums up the areas of the pixels whose water flows into the pixel under concern, and assigns that total area (i.e., upstream drainage area) to the pixel under concern.

The area unit of the output layer is the same as that of the input layer (e.g., square meter). This is different from ArcGIS' flowaccumulation function, whose output values are number of pixels.

ArcSIE implements both uni-path and multipath algorithms.

**Uni-path algorithm**: This algorithm takes flow direction layer as the input and assumes water in a pixel only flows into one of its neighboring pixels. This is equivalent to the Flowaccumulation function of ArcGIS.

**Multipath algorithm**: This algorithm assumes that water in a pixel flows into all lower neighboring pixels, and the distribution of water among lower pixels is determined by the slope gradients between
those pixels and the center pixel. The distribution of water is represented by the proportions of the area of the center pixel allocated to its neighboring lower pixels, and the upstream drainage area of each pixel is calculated accordingly. This tool takes DEM as the input.

3.6. **Wetness Index**

This index is also called compound topographic index (CTI).

Wetness index is calculated as

\[ w = \ln(\text{Flow Accumulation/\text{Slope Gradient}}) \]

ArcSIE implements both uni-path and multipath algorithms.

**Uni-path algorithm:** The flow accumulation is calculated using the uni-path algorithm. This tool takes uni-path flow accumulation layer and slope gradient layer as inputs.

**Multipath algorithm:** The flow accumulation is calculated using the multipath algorithm. This tool takes DEM as the input and calculates multipath flow accumulation and slope gradient on the fly.

ArcSIE also implements a method to **smooth the multipath wetness index**. The objective of this method is to reduce the sensitivity of the wetness index to slight local variation. This method can be used to solve the “dry valley” problem that is obvious in the raw wetness index layer (no matter uni-path or multipath results). There are 3 parameters to adjust this sensitivity:

- **Vertical Smoothing Range**: If the difference of elevation between two pixels is within this range, the two pixels are considered to have the same elevation.
• *Horizontal Smoothing Range:* If the distance between two pixels is greater than this value, their elevations will not be compared.

• *Lag in calculating slope:* The horizontal distance that is actually used in slope calculation (i.e., the distance is not necessarily determined by the resolution of the DEM).

The two inputs of the wetness index smoothing method include the calculated wetness index layer (multipath result is preferred) and the DEM. The DEM is used to characterize the local variation, so is not necessary to have been filled.

### 3.7. Catchment

This tool identifies and labels all the basin areas and catchments that have outlets at the edge of the image (DEM). Each such catchment will be labeled with a unique id. The catchments are identified using the D8 flow direction algorithm. All the pixels whose water flows into the same basin bottom pixel or outlet pixel will be labeled by the same catchment id.

### 3.8. Terrain Partition

This tool partitions the entire study area into parts based on pixels' flow destinations. Pixels whose water flows into the same stream sections are labeled as belonging to the same “partition”. The flow path is determined using the D8 flow direction algorithm.

This tool takes coded streamline layer and the uni-path flow direction layer as the inputs. Preferably the streamlines is coded using Shi's method.
The user can specify to merge adjacent slope partitions that have similar slope aspects into a single partition. If the difference in aspect between two adjacent partitions is smaller than the user-specified angle, the two partitions will be merged.

3.9. Derive Ridgelines

This tool identifies and labels pixels that are on ridgelines.

3.8.1 Algorithms

*Peuker and Douglas*: This algorithm consists of the following two steps: first, create a binary image with the same dimension and resolution as the DEM and set each pixel in this image to be 1; and second, use a 3 by 3 moving window to scan the DEM: in this window, find the pixel with the lowest elevation value and change the value of the corresponding pixel in the binary image to 0. After one sweep with the window on the DEM, the pixels in the binary image still with value 1 mark the ridgelines.

*O'Callaghan and Mark*: This algorithm calculates the upslope accumulation area for each pixel. Those pixels whose upslope accumulation area values are zero represent the ridgelines.

*Skidmore*: In a 3 by 3 moving window, a pixel will be labeled as a ridge pixel if it has two opposite neighbors with lower elevations, and at least one the other two neighbors (orthogonal) is lower.

*Shi*: This algorithm labels terrain partition boundaries as the ridge. The user can specify to ignore partitions whose elevations do not vary enough. All the partitions whose highest elevation - lowest
elevation < user-specified “TP Elevation Diff” will be ignored. The user can also force the ridge pixels to be on the top part of a partition by specifying “In top proportion”.

3.8.2. Ancillary Rules

The user can apply rules to specify that ridge pixels must meet certain elevation and slope gradient criteria.

3.10. Broad and Narrow Ridgelines

This tool classifies identified (using Ridgeline tool) ridgeline cells into "broad" and "narrow" ridgelines.

The program uses slope gradient values to determine if the cells around those ridgeline cells are "flat". If the slope gradient value of a cell is smaller than the specified threshold, the cell will be considered "flat". Based on this checking, the programs identifies a contiguous "flat area", if any, around the ridgeline cell under concern.

If the area of the "flat area" is smaller than the specified threshold, the ridgeline cell under concern will be classified as "narrow ridge".

Otherwise, the program tests the width of the "flat area" at the ridgeline cell under concern. It tests width in 4 directions: W-E, N-S, NW-SE, and NE-SW. If the width is smaller than the specified "lower limit", the ridgeline cell is classified as "narrow ridge", or if the width is between the "lower limit" and "upper limit", the cell is classified as "broad ridge". If the width is above the "upper limit", the ridge cell is considered to be too "broad" even for a broad ridge.
3.11. Derive Streamlines

This tool derives streamlines from DEM.

3.10.1. Algorithms

*O'Callaghan and Mark*: This algorithm first calculates the upslope accumulation area for each pixel, and then marks those pixels whose upslope accumulation area values are greater than a specified threshold to be streamline pixels. This algorithm can produce connected and very realistic drainage network, and also provides the flexibility that the user can decide the detail level of resultant network by changing the threshold.

*Skidmore*: This algorithm uses a 3 by 3 window to scan the DEM. A pixel can be defined as a gully if it has two opposite neighbors with higher elevations and the other two neighbors (orthogonal) have a lower elevation and a higher elevation, respectively. If one of the neighbors has the same elevation as that of the center pixel, the same testing will be outwardly iterated on that neighboring pixel, until a pixel with different elevation is found.

*Peucker and Douglas*: This algorithm consists of the following two steps: first, create a binary image with the same dimension and resolution as the DEM and set each pixel in this image to be 1; and second, use a 3 by 3 moving window to scan the DEM: in this window, find the pixel with the highest elevation value and change the value of the corresponding pixel in the binary image to 0. After one sweep with the window on the DEM, the pixels in the binary image still with value 1 mark the stream channels.
3.10.2. Coding stream segments

The stream network derived using the *O'Callaghan and Mark* method can be coded so that different stream segments can be identified.

- **Simple Mark:** All the stream pixels will be labeled as 1, and all the other pixels are 0.
- **Horton's order:** Label the segments according to their orders in the network. The main stream has the smallest order number.
- **Strahler's order:** Label the segments according to their orders in the network. The main stream has the largest order number.
- **Shi's order:** Label segments with close-to unique ids. Generally the larger the number, the closer to the main stream.

3.12. Headwater

This tool finds typical locations of headwater. It assumes a typical location of headwater to be 1) on a streamline; 2) close to the junction of two streamlines; and 3) very convergent in morphology.

It uses a streamline layer to locate streamlines and junctions of streamlines. A junction is called a *node* in this tool.

It allows the user to determine if headwaters are close to all *nodes* or only *jump nodes*. *Jump nodes* are heads of streams or where two equal-order tributaries merge into a stream of another order. *Nodes* and *Jump nodes* are illustrated by the below figure:
The user needs to specify for what order of streamlines the headwaters are to identify.

To identify *jump nodes*, and to determine the *order* of a streamline, the input stream layer must be generated using O'Callaghan and Mark's algorithm and coded with either Horton's method or Strahler's method. Note that the *orders* from two coding methods are reversed to each other.

The convergence is calculated based on profile curvature and profile curvature (the planform curvature is optional). A weighted average of normalized profile and planform curvatures is calculated for each streamline pixel around a *node* (either every *node* or just *jump node*) within the range defined by “Up-search range” and “Down-search range”. For “up search”, the program searches both tributies the form the node (the program will not search the other tributies if there are more than two tributies meet at the *node*). The streamline pixel that has the highest weighted average curvature value will be labeled as the typical headwater location around the *node* under concern.
This method locates the heads of streams solely based on upper drainage area. Without considering local morphological factors, sometimes the heads of streams located by our method are not appropriate for locating headwaters. You can use the Prune tool to remove some heads that you do not want to in the headwater identification.

**References:**


Chapter 4. Tools

4.1. File Converters

These tools convert data between formats of ArcGIS (Grid) and 3dMapper (3dm and 3dr).

When converting a 3dr file to an ArcInfo Grid, you can specify to attach the spatial reference (i.e., coordinate system) of the current ArcMap mxd to the new Grid, but it is your responsibility to ensure that the 3dr file is indeed in that coordinate system.

4.2. Make Hardened Map

This tool integrates the fuzzy membership maps (for individual soil types) into a single raster map. The result can be used to create a traditional vector map.

The “hardening” process picks the soil type with the highest fuzzy membership value at a location as the representative soil type at that location. In a hardened map, a pixel is only labeled with its representative soil type.

You need to specify all the fuzzy membership maps that should be included in the hardening process. For each map, you also need to specify an integer number as the label of the soil type represented by the map. The label will be the value representing that soil type in the final hardened map. The interface as below facilitates this process.
For loading individual fuzzy membership maps one by one, use the dropdown list or the browsing button, and then edit the values in the Label column to give proper label value to each layer.

Using \[\text{browsing}\] to save the list of maps and their corresponding labels into a \textit{fuzzy membership map list} file (fst) for future use.

Using \[\text{browsing}\] to load the maps and their labels in a previously saved fst file.
This tool outputs three layers. Besides the hardened map itself, which will have the name as what you specify in the Output Layer box, there are two other layers created for representing the uncertainty caused by the hardening process. One of them, whose name is formed by attaching "ent" to the user-specified output name, contains entropy values calculated as follows:

\[-\sum_{k=1}^{n} \frac{S_k}{\sum_{k=1}^{n} S_k} \ln\left(\frac{S_k}{\sum_{k=1}^{n} S_k}\right)\]

\[\frac{\sum_{k=1}^{n} S_k}{\ln(n)}\]

In above equation, \(s_k\) is the fuzzy membership value of soil type \(k\) at a given location, and \(n\) is the total number of soil types.

The other uncertainty layer, whose name is formed by attaching "exg" contains exaggeration values calculated as follows:

\[1 - \max(s1, s2, ..., sk)\]

Note: Since an ArcInfo Grid can only have a name shorter than 12 letters, if you specify a long output name, the names of the two uncertainty layers may be truncated.

**Reference:**

4.3. Remove Slivers

This tool removes small patches (slivers) in a raster layer. It is usually used to “clean” the hardened map.

This tool removes slivers by progressively shrinking the sliver through allocating its cells to their adjacent patches.

The units of the specified thresholds are the same as the unit of the input raster layer (e.g., meter).

**Threshold for interior slivers:** If the total size of a group of connected cells is smaller than this threshold, and none of the cells is at the edge of the raster layer, the patch formed by this group of cells will be considered as a *sliver*, and will be removed.

**Threshold for border slivers:** If the total size of a group of connected cells is smaller than this threshold, and at least one of the cells is at the edge of the raster layer, the patch formed by this group of cells will be considered as a *sliver*, and will be removed.

**Number of steps:** If the program directly uses the specified thresholds to determine slivers, some removals may be unwanted, because a patch whose original size is smaller than the threshold may be enlarged after an adjacent smaller patch is removed, and thus becomes bigger than the threshold and should not be removed. Therefore it is recommended to start with a small threshold and progressively reach the wanted final thresholds. You specify the “Number of steps” that should be taken to reach the final thresholds. This number of iteration of removing process will be performed.
**Increment:** The increment of thresholds between two consecutive steps. Hence, the starting thresholds can be derived as the specified thresholds - Number of steps * Increment.

**Nodata:** The value specified by this parameter will not be processed. It also provides the “background” for the “Preserve isolated patches” option.

**4-connected vs. 8-connected:** This option specifies that when checking if two adjacent cells are connected (for determining if they belong to the same patch), the program should only checks the 4 cells in the cardinal directions or all the 8 neighboring cells.

**Preserve isolated patches:** If this box is checked, a patch surrounded by nodata will be preserved, no matter what its size is.

**Work on specific value:** If this box is checked, the program will only check the cells with the specified value. In other words, this option allows you to only remove slivers of a certain soil type.

### 4.4. Diversity

This tool compares a vector map and a raster (hardened) map. It reports how much information has been dropped through the sliver-removing and vectorizing processes.

The output of this tool is a table in text format.

The first column of the table contains IDs of the polygons. You need to specify which field in the attribute table of the polygon Shapefile contains the IDs to be used.
The following columns are in pairs. The first column in a pair contains the ID of a soil type and the second is the area percentage of this soil type in the polygon. The pairs are in a descending order based on the percentages; That is, the majority soil type goes first. You need to specify the number of soil types that need to be reported. If a polygon contains more soil types than the specified number, the areas of the other soil types will be summed up and reported in the “other” column pair.

4.5. **Prune Branches**

This tool prunes streamlines for creating the headwater layer. The streamlines must form a connected network, so it is recommended to use a stream layer created using O'Callaghan and Mark's algorithm.

4.6. **Stretch**

This tool changes the value range of a raster layer to a user-specified range.

The original values will be stretched linearly.

You can specify a particular range in the original values to stretch.

4.7. **Filter**

This tool calculates the mean of the specified neighborhood of a cell and replaces the original value of the cell with the mean. The output is a smoothed layer of the original one.

You can specify the size of the filter. The larger the filter, the smoother the result.
4.8. Frequency

This tool generates a table reporting the frequency of each unique value in a raster layer.
Appendix A Glossary

case: A case is a representation of the soil scientist's knowledge of the relationship between a soil type and its environment at a specific location. Conceptually, a case is a knowledge composition made up of the information from three spaces: the geographic space, the parametrical space (defined by environmental features), and the solution space (taxonomic space). In geographic space, a case corresponds to a location on the ground; in parametrical space, it corresponds to a combination of certain environmental feature values; in solution space, it corresponds to a specific soil type (or, in terms of fuzzy logic, the similarity to this soil type). When creating cases, the soil scientist does not have to specify values in parametrical space, but only needs to pinpoint locations in geographic space. The correspondence between the values in parametrical space and solution space will automatically be built by the inference engine through the correspondence between geographic space and solution space. Technically, in ArcSIE, a case is eventually an instance plus a spatial setting. The spatial setting includes the location information (coordinates) and values of some parameters for performing spatial inference.

casebase: A casebase is a collection of caselists. It is created for a certain mapping area and contains the caselists of the soil types for which case-based reasoning are to be performed.

instance: An instance is a representation of the soil scientist's knowledge of the relationship between a soil type and its environment characterized by topography, geology, climate, vegetation, and other environmental features. It is not explicitly location-specific and is usually applicable to a large portion of the mapping area or even the entire mapping area, thus is usually considered to be the general knowledge. Technically, an instance contains a set of rules regarding a set of environmental features.
**line case:** Case in the form of line. Suitable for soil types or geomorphic features of which the distributions show linear pattern. Technically, ArcSIE treats a line case as a series of point cases.

**point case:** Case in the form of point, also called tacit point. Each case corresponds to an exact location on the ground.

**polygon case:** Case in the form of polygon. Technically, ArcSIE treats a polygon case as an agglomeration of point cases.

**rule:** A rule is a fuzzy membership function defining the relationship between the values of an environmental feature and the optimality values for a soil type.

**soil type:** A soil type corresponds to a soil type and contains rule(s) or case(s)

**rulebase:** A rulebase is a collection of soil types. It is supposed to be created for a certain mapping area and contains the soil types for all the soil types in the mapping area.

**soil type:** A caselist corresponds to a soil type and contains case(s).

**tacit point:** See point case.
Appendix B  File Suffixes

.3dm: 3dMapper’s data format for orthophoto-DEM pair. Used by 3dMapper for visualization and terrain attributes calculations.

ArcInfo Grid: 3dMapper’s raster data format. Used by ArcSIE as the native raster data format. All the environmental data layers input to ArcSIE must be in this data format. All the maps produced by ArcSIE are in this format.

.asc: The ASCII format for Arc/Info Grid. Used to exchange raster data between 3dMapper/ArcSIE and ArcGIS.

.chk: Check file created by ArcSIE. See 1.3.5.3.1.

.chk.lkt: Look-up table file created by ArcSIE. See 1.3.5.3.1. This is a text file.

dbf: File format of dBASE table. ArcSIE uses dbf tables for storing rulebases and raster casebases.

.shp: Shapefile of ArcView. ArcSIE uses Shapefile as the data format for vector (point, line, and polygon) casebase. ArcSIE creates a new casebase using a Shapefile created in a GIS (e.g., 3dMapper or ArcView). The created casebase is still saved as a Shapefile of the same type as that of its source data, but the new Shapefile will contain new fields required by the casebase in its attribute table.