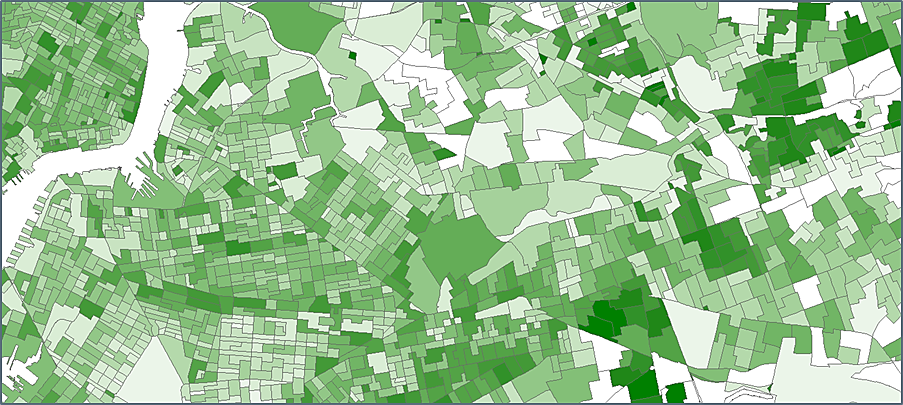
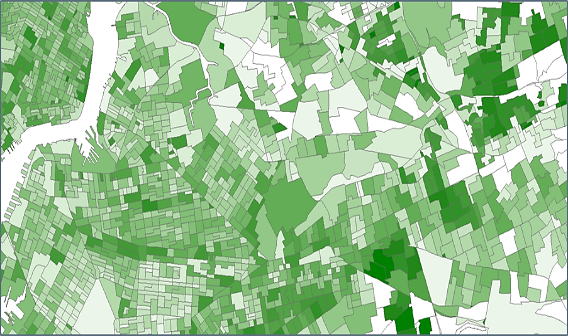
MCDA4ArcMap

version 1.1A



MCDA

MCDA

Table of Contents

[MCDA4ArcMap in a Nutshell 1](#_Toc444114904)

[Installation 2](#_Toc444114905)

[Walkthrough 4](#_Toc444114906)

[Normalization Strategies 8](#_Toc444114907)

[Data Manager 10](#_Toc444114908)

[MCDA Methods 11](#_Toc444114909)

[Visualization 19](#_Toc444114910)

[Configuration 21](#_Toc444114911)

[Important Hints 22](#_Toc444114912)

[Change Log 23](#_Toc444114913)

[Contact Information 24](#_Toc444114914)

# MCDA4ArcMap in a Nutshell

MCDA4ArcMap is an add–in that extends ArcMap with multi–criteria decision analysis capabilities. Moreover, it introduces a component for real–time visualization of the decision making process. This makes the analysis process highly interactive and the complete workflow is covered within a single piece of software.

## MCDA methods

The add‑in supports the following MCDA methods:

* Weighted Linear Combination (WLC)
* Ordered Weighted Averaging (OWA)
* Locally Weighted Linear Combination (LWLC)

They are described in detail in the MCDA Methods section.

## Limitations

* Only polygon geometry is supported
* Criterion values must be numeric
* Rook contiguity for large data sets (> 1000 polygons) is slow (several minutes processing time, depending on your hardware setup)
* Sessions are not persistent, MCDA results are lost when MCDA4ArcMap is closed. Results can be exported as comma separated values.

## License

The add–in is licensed under the Apache License 2.0 (Apache)[[1]](#footnote-2). Whereas the license grants a fair amount of rights to use the software, it is noteworthy that point 7 and 8 in the license limit the warranty and liability. It is the responsibility of the user to verify that everything is implemented as expected. Everyone has the chance to do so by reviewing the source code[[2]](#footnote-3).

## Looking Ahead

Feel free to suggest new features or report problems via the project website[[3]](#footnote-4). In addition, all kind of feedback is appreciated.

# Installation

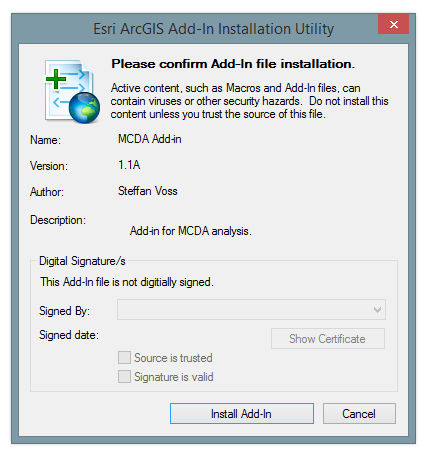
The add–in is shipped as a single file and can be installed with a simple double click. However, before you install the add‑in, make sure that your system meets the requirements. Two versions are provided. One targets ArcMap 10.1 and the other ArcMap 10.2. All later ArcMap versions should work with the add–in targeting ArcMap 10.2.

**Requirements for ArcMap 10.1: The version targeting ArcMap 10.1 is required. In addition, Microsoft’s .NET Framework 4.0[[4]](#footnote-5) is required.**

**Requirements for ArcMap 10.2: The version targeting ArcMap 10.2 is required. In addition, Microsoft’s .NET Framework 4.5[[5]](#footnote-6) is required.**

Go to the project website[[6]](#footnote-7) and download the latest release (MCDA4ArcMap 1.1A). Once you downloaded the add–in, double click on the MCDA4ArcMap 1.1A.esriAddIn file. A dialog as given in Figure 1 appears and you are asked to confirm the installation.

Figure : The installation confirmation dialog.



The functionality provided by the add‑in is encapsulated into a toolbar. Open Customize-> Toolbars as in Figure 2 and click on MCDA and the toolbar will appear.

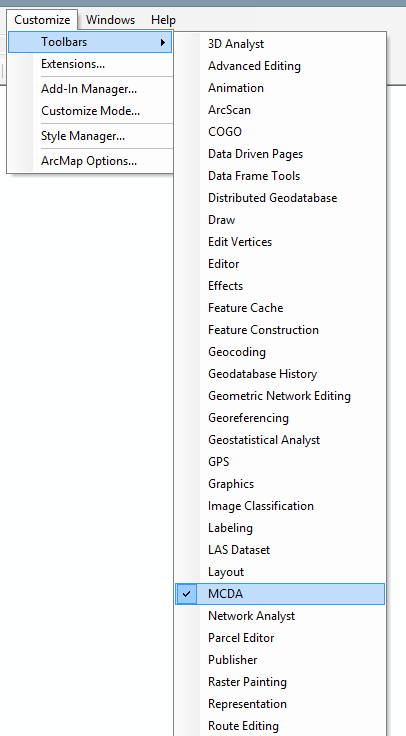


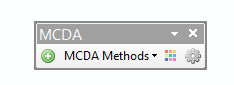
Figure 2: The add–in can be found next to the default toolbars.

# Walkthrough

This section serves as a quick introduction. A typical workflow is given, such that all components are covered.

The add–in basically consists of four components. The data manager, the MCDA methods, the visualization options and the configuration options. The arrangement is given in Figure 3.

Figure : The toolbar consists of four components.



Visualization

Configuration

MCDA methods

Data Manager

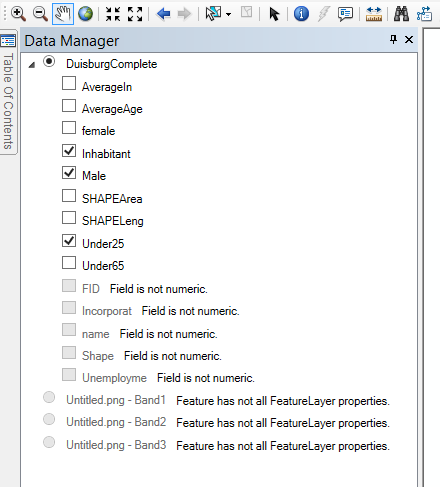
A typical workflow makes use of all four components with respect to the order from left to right.

## Data Manager

Suitable data for multi–criteria decision analysis has to fulfill certain conditions. First of all the add–in requires vector data and only polygon[[7]](#footnote-8) geometry is allowed. In addition, a field has to be numeric, it cannot contain NULL values and distinct values are required. The data manager provides feedback why certain data is not suitable or why certain fields are not suitable. The data manager keeps track of every item that is added or deleted. It is to some extent a copy of ArcMap’s Table of Contents with special capabilities required by the add–in. Only one feature can be active at once. An overview of a typical situation is given in Figure 4.

## MCDA method in General

Figure : The data manager with example data.



The selected data.

Pictures etc. are disabled as they do not meet the requirements.

Non numeric fields are disabled.

A selected field.

The functionality of the supported MCDA methods is encapsulated into dialogs. For each method a dialog is available — which is referred to as tool. Additional functions are available in the toolbar of these dialogs. Functions like the normalization are common to all tools, whereas functions like the neighborhood definition are exclusive for the LWLC tool.

A tool with annotations of the basic functions is given in Figure 5. The purpose of these buttons etc. is explained in the following.

The **Lock data** button is useful when the user wants to be independent from the selection in the data manager. Usually, the criteria selected in the data manager is available in every non–locked tool, i.e. the data in the tool is overridden. This is also true when a totally different feature is selected. As a consequence, removing the feature from ArcMap, removes it from the tool. Using the lock data function prevents such behavior. Internally, an in–memory representation of the selected feature is created, thus the tool is totally independent. Conversely, closing ArcMap will lead to the loss of the in–memory representation. As a final point, this feature makes it possible to have different tools with diverse data open at the same time.

The **distribution of weights** is a handy feature for equally distributing the weights between the chosen criteria. This is also the default setup when opening a tool the first time.

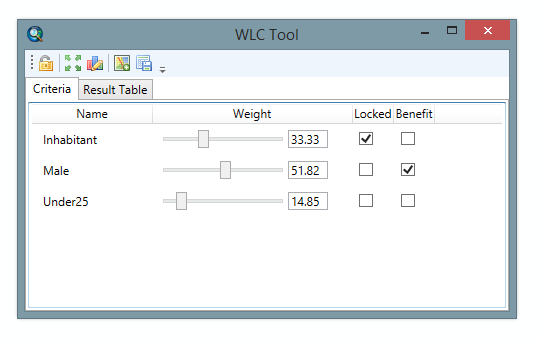
**Normalization** is crucial when it comes to data preparation. The topic is covered in detail in the chapter Normalization Strategies. For the sake of simplicity, it is possible to stick to the default setup.

**Add as managed layer** adds the feature behind the tool to the current map. However, the important point is that it appears now in the visualization view where the result column is selectable. By doing so, the changes of the weights or other settings will be directly visualized as configured. This feature makes MCDA4ArcMap a highly interactive tool as the decision process can be evaluated by direct feedback.

Using the **export** function will result in a comma separated file including the result table and the most important settings.

The **Result Table** lists all features with the normalized value, the weight and the result. It is a read only table and offers detailed insights. It is the exact same data that will be exported when using the export functionality.

The **criteria tab** lists the chosen criteria and offers ways to define the weight for every single criterion. This is done by either using the slider or by typing the value directly into the text field. As required by the supported MCDA methods, the sum of the weights has to be 100. A newly entered weight is taken into account once enter is pressed or the field is left (on focus lost). The **lock** mechanism is useful when certain criteria need constant weights and are not subject of the interactive decision making process. To accomplish this behavior, the locked criteria will be excluded from further redistribution. Every MCDA tool is sophisticated in a way that it allows only weight changes that are tolerable within the defined boundaries.



Lock

Add as managed layer

Normalization

Distribute weights

Lock data

Result table

Export

Benefit

Figure : A typical tool with three criteria and assigned weights.

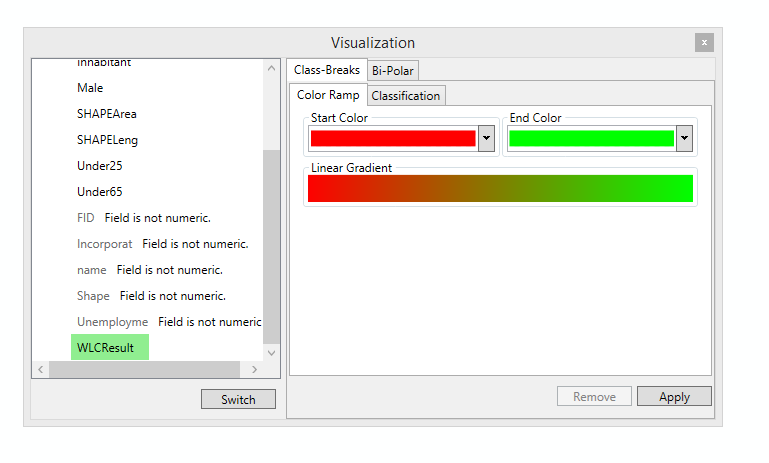
Sometimes are criterions with lower values desirable and sometimes with higher values. A typical benefit criterion might be the income and a typical non–benefit criterion might be the unemployment rate. I.e., higher values will result in higher scores. The default behavior (non–checked benefit checkbox) is that higher values will result in lower scores. The **Benefit** checkbox is a way to change this behavior for every single criterion.

## Visualize the result

The decision result can be visualized either by using the default ArcMap functions or the visualization component shipped with MCDA4ArcMap. The major feature of the newly added function is the possibility to get direct visual feedback. A typical list of possible fields for visualization is given in Figure 6.

The list on the left hand side of the Visualization window lists either all available fields or only the managed fields. The **Switch** button can be used to switch between these two lists. A field is managed when an active tool exists and the **add as managed layer** button is used. The managed field is highlighted in green.

Figure : The visualization view with a managed field (in green).



Switch between all and managed

The managed

criterion

Class-Breaks Renderer

Bi-Polar Renderer

Apply the defined renderer

Once a renderer is applied every change in the corresponding tool will be instantly visualized (especially when changing the weights or other settings). The capabilities of the Visualization component are explained in more detail in the Visualization chapter.

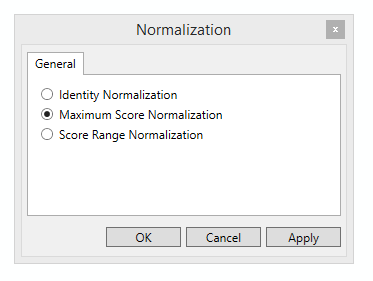
The visualization is also related to the configuration. More can be found in the Configuration chapter.

# Normalization Strategies

It is notable that the terms normalization and standardization are often used interchangeably[[8]](#footnote-9). In the context of this add–in, the term normalization refers to rescaling. Consider a problem with the two hypothetical criteria average income and average age. It would be inappropriate to use large scale values like 50000 for the income in comparison to small scale values like 40 for the age in the same analysis. Consequently, the criteria must be normalized into the same numeric range.

Every tool offers three different normalization strategies as given in Figure 7.

Figure : The normalization window as offered for every MCDA method.



## Score Range Normalization

The score range normalization does not preserve proportionality. The highest value is always 1 and the lowest value is always 0.

Equation (1) is used for benefit criteria.

|  |  |  |
| --- | --- | --- |
|  |  | (1) |

For cost criteria Equation (2) is given.

|  |  |  |
| --- | --- | --- |
|  |  | (2) |

One corner cases has to be considered. In case all values are equal, the current implementation will not transform the entire list of values. The first reason is to prevent a division by zero and the second reason is that such a list does not provide any value at all. Because of that, it is not even possible to select such a field for analysis. These fields are disabled in the data manager.

Examples:

|  |  |  |  |
| --- | --- | --- | --- |
| Input | Benefit | Cost | REmark |
| [-3, -2, 0] | [0, 0.33, 1] | [1, 0.66, 0] |  |
| [0, 0, 0] |  |  | Not allowed |
| [1, 1, 1] |  |  | Not allowed |
| [1, 2, 3] | [0, 0.5, 1] | [1, 0.5, 0] |  |

## Maximum Score Normalization

The maximum score normalization produces values between 0 and 1 (both inclusive). For benefit criteria, the result is anchored at 1 and for cost criteria at 0.

For benefit criteria Equation (3) is given.

|  |  |  |
| --- | --- | --- |
|  |  | (3) |

For cost criteria Equation (4) is used.

|  |  |  |
| --- | --- | --- |
|  |  | (4) |

Examples:

|  |  |  |  |
| --- | --- | --- | --- |
| Input | Benefit | Cost | REmark |
| [-3, -2, 0] |  |  | No normalization because of the division by zero |
| [0, 0, 0] |  |  | Not allowed |
| [1, 1, 1] |  |  | Not allowed |
| [1, 2, 3] | [0.33, 0.66, 1] | [0.66, 0.33, 0] |  |

## Identity Normalization

The identity normalization is in fact no normalization. It was introduced for data that is already normalized. Identity normalization does not alter the values at all and could be referred to as no normalization.

# Data Manager

The basic functionality of the data manager was already described in the Data Manager section. The most important fact is that only polygon geometry is supported. As previously mentioned, the fields have to satisfy further conditions. Basically, only numbers are allowed. ESRI has a constant[[9]](#footnote-10) that includes all data types. The add–in can work with the following data types:

* esriFieldTypeSmallInteger Short Integer.
* esriFieldTypeInteger Long Integer.
* esriFieldTypeSingle Single-precision floating-point number.
* esriFieldTypeDouble Double-precision floating-point number.

However, the correct data type does not guarantee that the field is eligible. Besides the data type three more criteria have to be fulfilled:

* a field cannot include NULL[[10]](#footnote-11) values (meaning no value(s) provided)
* a field cannot contain only non‑distinct values
* a field cannot be of type object id (oid)

With respect to the mentioned restrictions the following messages are displayed:

* Field is not numeric.
* Field contains NULL values.
* Field has no distinct values.
* Field is OID.

Even if more than one restriction is violated, only one message is displayed.

# MCDA Methods

The method descriptions refer to the implementation. An explanation is necessary as different varieties may exist in the literature. This is especially true for the Ordered Weighted Averaging (OWA) method.

## Weighted Linear Combination

Weighted Linear Combination (WLC) is possibly the simplest way of multi criteria decision analysis. The criteria is normalized, multiplied by an assigned weight and summed up for each decision alternative.

### Algorithm

WLC is defined and implemented as follows: For a given set of criteria, WLC is defined as a combination procedure that associates a set of criteria weights with the ‑th decision alternative (the location or polygon). Where the weights satisfy the following properties: . The weights are multiplied with the criterion values as in Equation (5). To make the criterion values more comparable, they are normalized with a value function. These value functions are described in the Normalization Strategies chapter.

|  |  |  |
| --- | --- | --- |
|  |  | (5) |

### Example

In Table 1 an example with 5 criterion values is given. The criterion weights are assigned and multiplied by the criterion values. The actual result (for a specific location) is 0.46. It is the sum over all products of the weights and values.

Table 1: An example how WLC is calculated.

|  |  |  |
| --- | --- | --- |
| Criterion Values | Criterion Weights |  |
| .3 | .3 | .09 |
| .8 | .1 | .08 |
| 1 | .2 | .2 |
| .3 | .3 | .09 |
| 0 | .1 | 0 |
|  |  | .46 |

In Figure 8 it is demonstrated how such a calculation looks like for the WLC Tool. The example consists of three criteria and nine locations. The numbers listed in the *FID* column refer to a specific polygon (they are the unique identifiers of the polygons). The three columns *AverageIn*, *AverageAge* and *Under25* are the selected criteria (selected in the data manager). The values are already normalized and multiplied by the weights, as defined by the user. The *WLCResult* column is the sum of the three weighted criteria in every row. The result with the highest value is the best possible location (the benefit checkbox has to be applied accordingly).

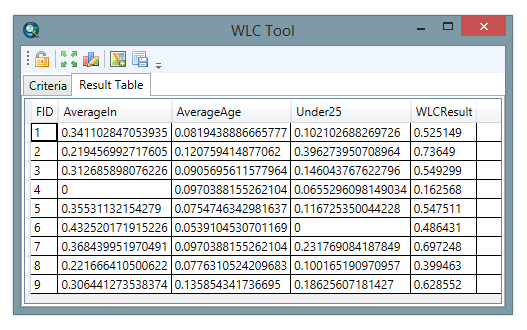


Figure 8: The result tab of the WLC tool with three criteria and nine locations.

More information is available in the introduction to LWLC in a publication authored by Malczewski, J. [[11]](#footnote-12)

## OWA

Ordered Weighted Averaging consists of basically three steps. Firstly, the user defined weights are applied in the same way as for WLC. Secondly, the result is reordered. Thirdly, the order weights are calculated and applied. In fact, the order weights are calculated only once due to the fact that they depend only on the chosen value and are independent from the selected criteria.

### Algorithm

The first step is already explained in the WLC section. Equation (6) is used to determine the order weights. Consider (used to determine the order weights) and 5 criterions. The order weights for the given setup are: , , …, . Note that results in exactly the same result as WLC with the given criteria and weights. In fact, WLC is a special case of OWA with . It is also notable that in the implemented OWA method the ordered weights are independent from the criterion weights defined by the user.

|  |  |  |
| --- | --- | --- |
|  |  | (6) |

### Example

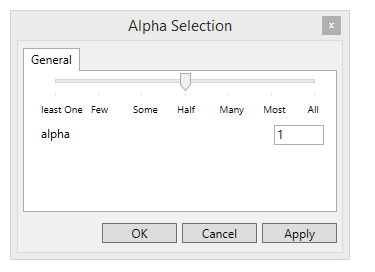
An example is given in Table 2. 5 criterion values and a user defined weight for every criterion is given. The criterion values are already normalized with one of the described normalization strategies (c.f. Normalization Strategies). The first step is analogous to WLC. Every criterion value is multiplied by the corresponding criterion weight. Afterwards the results are ranked or sorted. Subsequently the order weights are applied (For and Equation (6)).

Table 2: An example of how OWA is calculated with .

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Step 1 (WLC) | | | Step 2 | Step 3 | |
| Criterion Values | Criterion Weights |  | Ordered | Order Weights ; | Result |
| .1 | .07 | .007 | .198 | .2 | .0396 |
| 0 | .27 | 0 | .104 | .2 | .0208 |
| .6 | .33 | .198 | .06 | .2 | .012 |
| .8 | .13 | .104 | .007 | .2 | .0014 |
| .3 | .2 | .06 | 0 | .2 | 0 |
|  |  |  |  |  | .0738 |

Every OWA tool has a window for the value selection. As seen in Figure 9, it is possible to choose one of the 7 predefined linguistic quantifiers or to define a precise value. More about the linguistic quantifiers and the impact on the analysis can be found in a publication by Malczewski, J. [[12]](#footnote-13)

Figure : The alpha value selection for OWA.



A second example with is given in Table 3.

Table 3: An example of how OWA is calculated with .

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Step 1 (WLC) | | | Step 2 | Step 3 | |
| Criterion Values | Criterion Weights |  | Ordered | Order Weights ; | Result |
| .1 | .07 | .007 | .198 | 0.04 | 0.00792 |
| 0 | .27 | 0 | .104 | 0.12 | 0.01248 |
| .6 | .33 | .198 | 0.06 | 0.24 | 0.0144 |
| .8 | .13 | .104 | 0.007 | 0.4 | 0.0028 |
| .3 | .2 | .06 | 0 | 0.6 | 0 |
|  |  |  |  |  | 0.0376 |

## LWLC

The Local Weighted Linear Combination (LWLC) is related to the Weighted Linear Combination. The major difference is the usage of so–called local weights. These local weights reflect the potential spatial changes in the area of interest. To calculate local weights it is necessary to define a neighborhood. The LWLC tool provides 4 neighborhood definitions. The neighborhood definitions are explained in more detail in the algorithm section.

### Algorithm

Besides the calculation of the local weights, the algorithm is similar to WLC. Instead of the global weights, the local weights are applied within the cluster. The local weights are calculated as given in Equation (7).

|  |  |  |
| --- | --- | --- |
|  |  | (7) |

A local weight refers to the *i*‑th neighborhood (in the cluster) and the *k*‑th criterion. refers to the local range and refers to the global range. Both are defined in Equation (8) and Equation (9).

The local range is defined as the difference between the maximum and minimum for a specific criterion in the cluster.

|  |  |  |
| --- | --- | --- |
|  |  | (8) |

The global range is defined very similar, except for the phenomenon that the maximum and minimum of a criterion is not limited to the criteria of a cluster.

|  |  |  |
| --- | --- | --- |
|  |  | (9) |

It is noteworthy that the actual values have to be normalized within each cluster before the local weights are applied.

A major aspect of the LWLC method is the definition of the cluster or neighborhood. Figure 10 presents an example of each supported neighborhood definitions.

Both, the **Rook** and **Queen Contiguity** come from the possible movements of the name giving chess pieces. The rook is restricted to polygons that share a common border. Where border is defined as a polyline. That is the reason why the black polygon in the example is not part of the contiguity, despite the fact that they have a point in common. In contrast, the queen can move to polygons that have only a point in common.

Both contiguities are implemented in a way that they also accept intersections, i.e. two intersecting polygons are interpreted as polygons with a common border.

The **Distance** defintion is based on the distance between the centroids. Every polygon with its centroid in the given distance belongs to the neighborhood. The distance definition does not uses the shortest distance between two centroids , i.e. the euclidean distance is used. Further, independent from the selected distance not more than the default value of 20 neigborhood polygons are considered. The number of neighbors can be increased in the Neighborhood Definition window.

The **K Nearest Neighbors** definitionis related to the distance defintion. Instead of using all polygons in a certain distance, the closest *K* polygons are taken into account.

Under certain circumstance it is impossible to calculate the local weights because of a division by zero based on an unfortunate neighborhood arrangement. To avoid such situations one can chose the **Automatic** definition. Whenever the calculation of the local weights in a cluster might be impossible the definition will add a polygon. This is done by using the *K* Nearest Neighbor definition starting with *k=3* and increasing until *k=1000* (only if the number has been increased to 1000 in the Neighborhood Definition window)if necessary.

Figure : The implemented neighborhood definitions by example whereas the green polygon is the seed and the blue polygons belong to the cluster.

K (=2) Nearest Neighbors

Distance

Rook Contiguity

Queen Contiguity

Sdfsdfsdf

An overview of the neighborhood defintions is given in Figure 11.

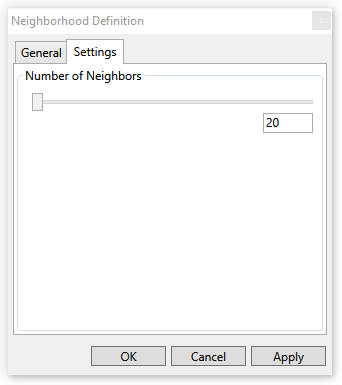
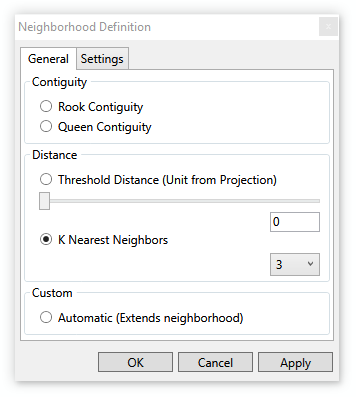


Figure 11: The neighborhood definitions available for LWLC.

### Example

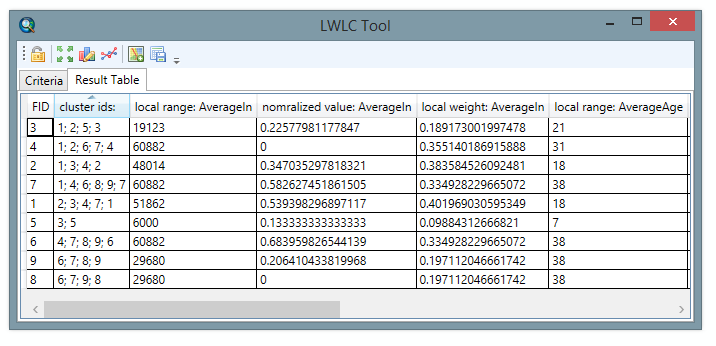
The example consists of three locations (A, B, C) and the three criterions (X, Y, Z). The locations are arranged in three clusters based on two locations each.

Table 4: LWLC and the local weight calculation for three locations and three criterions in three cluster.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Criteria | | | Cluster | Ranges | | | | Weights | |
| X | Y | Z |  | Local X | Local Y | Local Z | Global | Global | Local (Cluster) |
| Locations | A | 32 | 38000 | 4 | (A,B) | 9 | 4000 | 3 | X = 19 | = .3 | =.276;=.342; = .291 |
| B | 23 | 42000 | 7 | (B,C) | 19 | 5000 | 4 | Y = 9000 | = .5 | =.385;=.357; = . 257 |
| C | 42 | 47000 | 3 | (C,A) | 10 | 9000 | 1 | Z = 4 | = .2 | =.223;=.706; =.070 |

Using Equation (7) as given in the algorithm section a local weight for each criterion and cluster can be calculated.

Besides the final result, the intermediate steps like local range, normalized criterion and the local weight are available in the result table of each LWLC Tool. An example is given in Figure 12.



Cluster

Local Range

Normalized Criterion

Local Weights

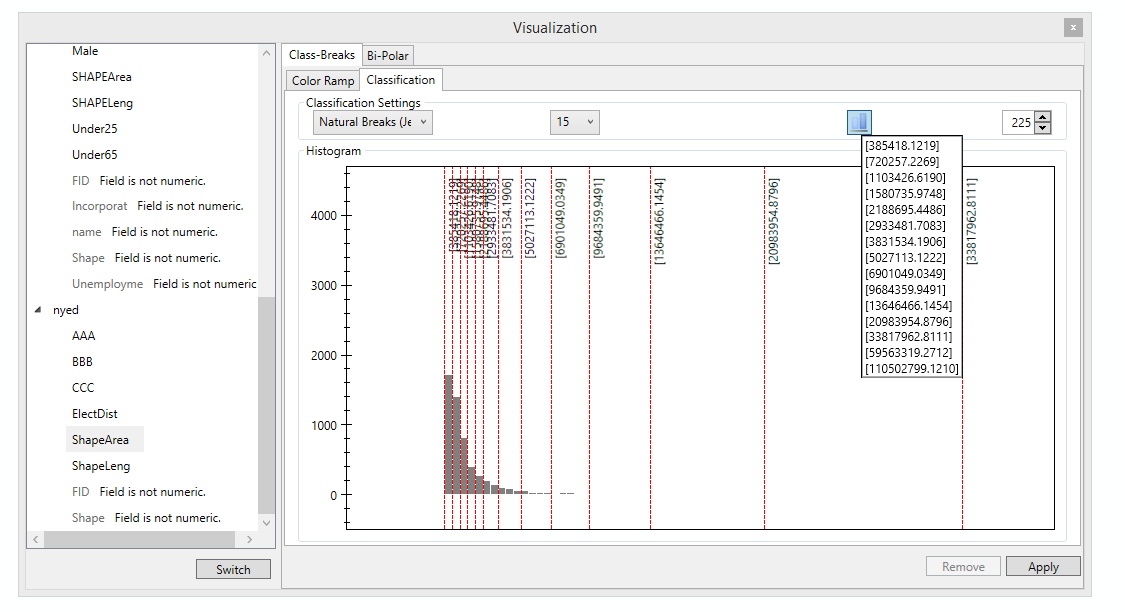
Figure 12: The result table of the LWLC Tool with a cluster column, the local ranges, normalized criterion values and local weights.

More details about the LWCL method can be found in a publication by Malczewski, J. [[13]](#footnote-14) and Carter, B. & Rinner, C.[[14]](#footnote-15)

# Visualization

Basically two renderers are available, the class‑breaks renderer and the newly introduced bi‑polar renderer. ArcMap ships with comprehensive functionality when it comes to rendering. However, two reasons exist to extend the existing functions. Firstly, full technical control is required to enable real time rendering during the interactive decision process. Secondly, the so‑called bi‑polar renderer is new and cannot be found in ArcMap.

The **class–breaks renderer** works very similar to the ArcMap implementation. An overview of the class‑breaks renderer settings is given in Figure 13. Two colors have to be selected (Color Ramp Tab). The algorithm to calculate the color ramp between these two colors is referred to as CIE Lab[[15]](#footnote-16). The result is a smooth color ramp. The **Classification** methods (Natural Breaks, Quantile…) are well known[[16]](#footnote-17) and the add–in implementation reuses the ESRI implementation.



Classes

Bins

Classification

Breaks

Figure : The histogram which is part of the class–breaks renderer.

Further, it is possible to define the number of **Classes**. The (resulting number of) class breaks can be found as vertical red lines in the **histogram**. The **Breaks** list presents the exact values of the breaks. It is possible to select a handful of **Bins** such that multiple breaks fit between two bins. If this is the case they are written one behind the other. The default number of bins is calculated by the square‑root choice[[17]](#footnote-18). It is noteworthy that the histogram supports panning and zooming. These features can be used by pressing the right mouse button (hold pressed) or using the mouse wheel. In addition, holding the left mouse button pressed will display the value of the bar and the number values that are represented by that bar.

The **Bi‑polar renderer** requires in addition to the two colors that make the color ramp a third color that is known as the neutral color and also part of the color ramp. An example of the bi‑polar renderer is given in Figure 14. The position of the neutral color can be changed with the help of a slider next to the preview gradient. The algorithm for calculating the color ramp is the same as for the class‑breaks renderer (CIE Lab).

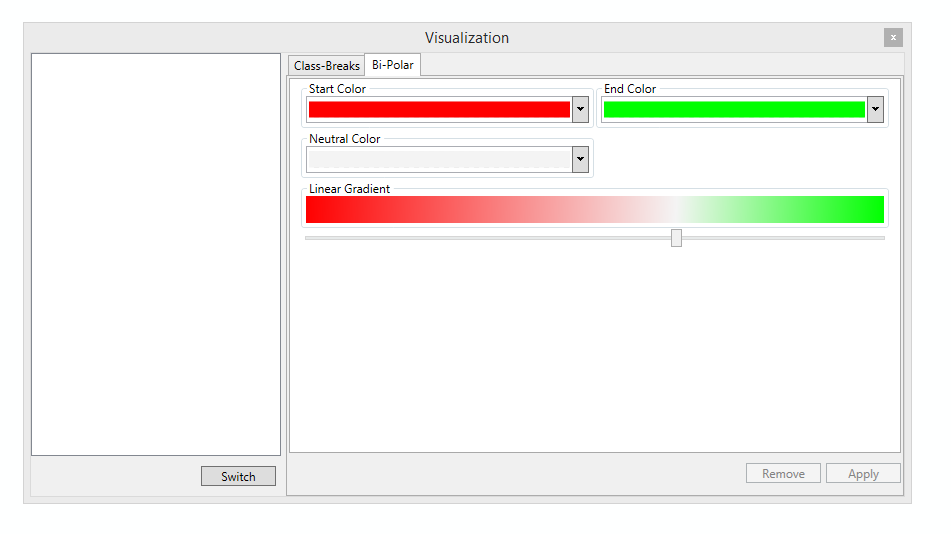


Figure 14: The bi‑polar renderer with white as neutral color and a shift to the end color.

In dependency of the position of the neutral color the focus may be one of the other two colors. The bi‑polar renderer might be an option in scenarios where for example of the map has to be marked as not suitable and the other as suitable. The setup for such a scenario might look like in Figure 14.

# Configuration

So far, the offered options concentrate on the rendering of the map and table of contents. Once a tool is chosen and a renderer is selected, the rendering frequency depends on the rendering options. For example a large data set may be too computation intensive for real time rendering. Thus, the after slider drag option may be favored over the real time option. The selected rendering option is valid for all tools.

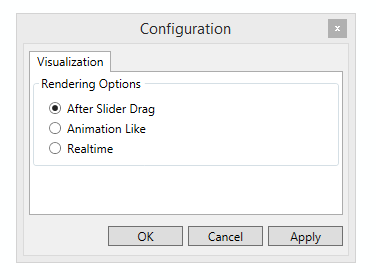


Figure 15: The three available rendering options.

The three supported rendering options which are shown in Figure 15work as follows. The **After Slider Drag** option refreshes the map and table of contents after the weight slider is dragged. The **Animation Like** option calculates a subset of the intermediate steps and replays them after slider drag. Only the map but not the table of contents is updated. The **Realtime** option updates the map at every single intermedia step that is made by using the slider to define a weight. This requires usually the most computation power.

NOTE: It is necessary to choose real time to refresh the map when defining a weight by the help of the text box next to the slider.

# Important Hints

## The unique identifier (FID/OID)

The add–in creates a copy of the complete feature for every newly opened tool that is later added as managed layer. Consider data where the unique identifier starts with zero. For whatever reason ESRI decided that it is useful to renumber[[18]](#footnote-19) the unique identifier starting with one when coping the feature. As a consequence, the unique identifier in the result table refers to the unique identifier in the original feature.

*“It is important to be aware of these behaviors when selecting fields for joining or relating tables. If a shapefile were to be joined to a feature class using the FID and OBJECTID fields, the shapefile record with FID = 0 would not be matched to a record in the feature class, as there is no record in a feature class with OBJECTID = 0.”*

# Change Log

* [24 Feb 2016] When calculating the neighborhood for lager datasets (~>3000 polygons) for LWLC ArcMap crashed. Fixed on trunk with revision #36981 as follows: Only the closest 20 polygons are considered as default from now on. I.e. for the Threshold Distance the max distance is the max distance from one of the 20 neighbors over all polygons. That implies that by selecting the max distance for most polygons not more than 20 neighbors are considered, even if the distance may include more neighbors. Further, the Automatic neighborhood definition cannot include more as 20 neighbors for the polygon of interest. Nevertheless, it is possible to increase the number of neighbors up to 1000 in the Neighborhood Definition window.
* [10 Nov 2015] Issue #4585: Current layers not mirrored in data manager: Fixed on trunk with revision #36701.
* [10 Nov 2015] Change Log introduced.

# Contact Information

Steffan Voß [steffan.voss@outlook.com](mailto:steffan.voss@outlook.com)

Dr. Claus Rinner [crinner@ryerson.ca](mailto:crinner@ryerson.ca) <http://www.ryerson.ca/~crinner/>

<https://mcda4arcmap.codeplex.com/>

The names of companies and products herein are trademarks or registered trademarks of their respective trademark owners.

1. MCDA4ArcMap license: https://mcda4arcmap.codeplex.com/license [↑](#footnote-ref-2)
2. MCDA4ArcMap source code: https://mcda4arcmap.codeplex.com/SourceControl/latest [↑](#footnote-ref-3)
3. MCDA4ArcMap issue tracker: https://mcda4arcmap.codeplex.com/workitem/list/basic [↑](#footnote-ref-4)
4. The .NET Framework 4.0 is NOT part of Windows 7, for Windows 7 you may download it separately. [↑](#footnote-ref-5)
5. The .NET Framework 4.5 is part of Windows 8, for Windows 7 you may download it separately. [↑](#footnote-ref-6)
6. MCDA4ArcMap project website: https://mcda4arcmap.codeplex.com/ [↑](#footnote-ref-7)
7. To be exact: we need data that implements ESRI.ArcGIS.Carto.IFeatureLayer2, with shape type ESRI.ArcGIS.Geometry.esriGeometryType.esriGeometryPolygon and the interface ITopologicalOperator.

   Actually only the LWLC Tool requires a polygon geometry. It is planned to make the WLC and OWA Tool capable to work with polylines or points. [↑](#footnote-ref-8)
8. Stackexchange: http://stats.stackexchange.com/questions/10289/whats-the-difference-between-normalization-and-standardization [↑](#footnote-ref-9)
9. esriFieldType Constants: http://resources.arcgis.com/en/help/arcobjects-net/componenthelp/index.html#//00250000004r000000 [↑](#footnote-ref-10)
10. Database NULL: http://en.wikipedia.org/wiki/Null\_%28SQL%29 [↑](#footnote-ref-11)
11. Malczewski, J. (2011). Local Weighted Linear Combination. Transactions in GIS, 15(4), 439–455. [↑](#footnote-ref-12)
12. Malczewski, J. (2006). Ordered weighted averaging with fuzzy quantifiers: GIS-based

    multicriteria evaluation for land-use suitability analysis. International Journal of Applied Earth Observation

    and Geoinformation, 8(4), 270–277. [↑](#footnote-ref-13)
13. Malczewski, J. (2011). Local Weighted Linear Combination. Transactions in GIS, 15(4), 439–455. [↑](#footnote-ref-14)
14. Carter, B. Rinner, C. (2014) Locally Weighted Linear Combination in a Vector Geographic Information System. Journal of Geographical Systems 16(3): 343-361 [↑](#footnote-ref-15)
15. CIE Lab description: http://support.esri.com/fr/knowledgebase/techarticles/detail/17221 [↑](#footnote-ref-16)
16. Classification methods: http://resources.arcgis.com/en/help/main/10.1/index.html#//00s50000001r000000 [↑](#footnote-ref-17)
17. Histogram/ Bin number calculation: http://en.wikipedia.org/w/index.php?title=Histogram&oldid=641376957 [↑](#footnote-ref-18)
18. ESRI Knowledge Base for unique identifiers: http://support.esri.com/es/knowledgebase/techarticles/detail/37480 [↑](#footnote-ref-19)