

Appendix VI: Clustering and LOICZView

This appendix contains material adapted from the WLV tutorial and help material that can be found in complete form on the report CD-Rom and on the web-sites. It is intended to provide a basic understanding of the tool and how it is used.

VI-A: Clustering

Clustering involves grouping data points together according to some measure of similarity. One goal of clustering is to extract trends and information from raw datasets. An alternative goal is to develop a compact representation of a dataset by creating a set of models that represent it. The former is generally the goal in geographic information systems, the latter generally the goal of pattern recognition systems. Both fields use similar, or identical techniques for clustering datasets.

There are two general types of clustering that are used on geographic data: supervised and unsupervised clustering. Supervised clustering uses a set of example data to classify the rest of the dataset. For example, consider a set of colored balls (all colors) that you want to classify into three groups: red, green, and blue. A logical way to do this is to pick out one example of each class - a red ball, a green ball, and a blue ball - and set them each next to a bucket. Then go through the remaining balls, compare each ball to the three examples and put each ball in the bucket whose example it matches the best.

This example of supervised clustering is illustrative because there are two potential problems. First, the result you get is going to be dependent upon the balls you select as examples. If you were to select a red, an orange and a blue ball, then it might be difficult to classify a green ball. Second, unless you are careful about selecting examples, you may select examples that don't represent the distribution of data. For example, you might select red, green and blue balls, only to discover that most of the colored balls were cyan, purple and magenta (which are in between the other 3 colors). This shows the importance of selecting representative samples when you execute supervised clustering.

Unsupervised clustering, on the other hand, tries to discover the natural groupings inside a dataset without any input from a trainer. The main input a typical unsupervised clustering algorithm takes is the number of classes it should find. In the colored balls case, this would be like dumping them into an automatic sorting machine and telling it to create three piles. The goal of unsupervised clustering is to create three piles where the balls within each pile are very similar, but the piles are different from one another.

WLV implements both unsupervised and supervised clustering. The unsupervised clustering algorithm is the k-means algorithm, originally described by MacQueen (1965). It incorporates some modifications to improve its robustness to missing data and poorly-behaved datasets.

One of the most important characteristics of any supervised or unsupervised clustering process is how to measure the similarity of two data points. In the case of geographically indexed data, a data point is a geographic location. A single location will generally have multiple variables associated with it. So we can define a similarity measure between two data points based on the values of their variables.

In WLV, the clustering tab lets the user control all of the important parameters for supervised or unsupervised clustering. For supervised clustering, the only relevant parameter is the distance measure, or similarity measure. Typically, the scaled distance measure is the first one to try on geographic datasets.

Logic2View Username: Password:

Current Data Set: Image height:

Clustering Options

Number of Clusters: (MAX 100)

Maximum Number of Iterations:

Number of Clustering Runs:

Random Seed:

Distance Calculation: ☒ Unscaled ☒ Scaled ☒ Maximum

Segmentation Method: ☒ k-means ☒ Region Growing

For unsupervised clustering, all of the parameters and checkboxes affect the outcome of the algorithm. The most important box is the *Number of Clusters* parameter that specifies how many classes the algorithm will create from the data. If you are at a loss as to how many clusters there should be, try using the [minimum description length \[MDL\] tool](#). Often, the best thing to do is experiment. Start with ten and then go up or down depending upon how you like the results.

The remaining parameters on the left side of the tab control mostly internal aspects of the k-means clustering algorithm. The *Maximum Number of Iterations* parameter specifies how long the program waits for a single clustering run to finish. If you have a large or complex dataset, it is reasonable to make this number larger. There is almost no reason to make it smaller.

The *Number of Clustering Runs* parameter is kind of like the quality vs speed slider on a color printer dialog window. When you make this number smaller (e.g., 1 or 2), you will get results faster, but they may not be as good. When you really want a high quality result, boost the number to a value from 10-20. Using the value 5 for this parameter seems to be a good tradeoff of speed versus quality.

Finally, the *Random Seed* parameter controls the random numbers used by the clustering algorithm (which is a stochastic, or randomized process). If you want to duplicate a previous clustering, for example, set the random seed to the previous value. This will guarantee that you get identical results. Most of the time, however, you will not need to change this parameter.

The *K-means/Region Growing* checkboxes are currently not worth changing. Make sure K-means is checked as in the above example.

VI-B: Visualization

The visualization tab is where you look at the results of classification, or clustering runs. There are a variety of ways to look at the data; which one is best for your situation depends upon what you are trying to accomplish. The following chart gives a quick overview and identifies the strengths of each visualization option.

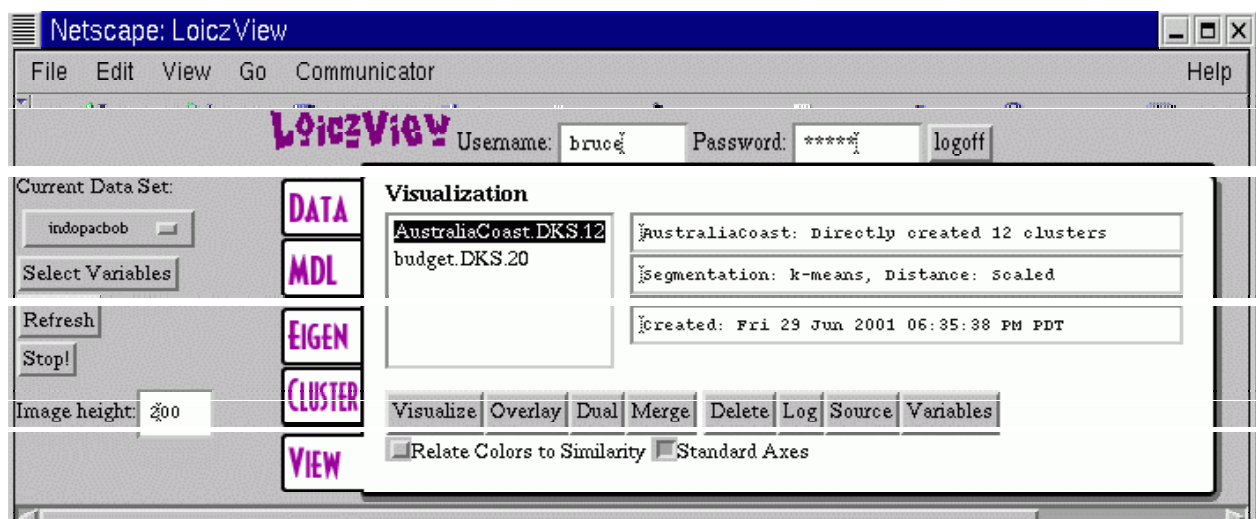
Visualization Method	Summary description	Strengths of this method
Visualize with Standard Axes	Plots a 2D map of the data points, colored by cluster, using longitude/latitude coordinates and a geographic projection.	This visualization is useful for seeing geographic relationships and making high-level comparisons with expert analyses.
Visualize with User-Selected Axes	Plots the data in 2D using the user-selected variables as coordinates. Each cluster is displayed in a unique color.	This visualization is useful for seeing how well the clusters map to natural breaks in the data space and discovering outliers and potentially bad data.

Overlay Visualization (standard axes)	Plots a standard visualization of the data points, and then overlays one selected variable on top of it. The overlaid variable is divided into classes, with each class receiving a different color.	If the variable was not used to generate the clusters, this is one way to examine how well the clusters predict a particular variable or expert classification. If the variable was used to generate the clusters, this provides a graphical representation of how the clusters track, or are influenced by the variable.
Dual Visualization (standard axes)	Plots two clusterings of the same dataset using latitude/longitude, one on top of the other.	This visualization is useful for comparing two clusterings created using different variables or variable weightings.

For all visualizations, if the *relate colors to similarity* box is not checked, the colors are selected for maximum differentiation between the clusters. If the box is checked, then the colors are selected so that similar clusters get similar colors.

Visualize with Standard Axes

To execute a standard visualization, select a file in the large list box in the View tab and then click on the *Visualize* button. For standard axes (latitude/longitude), make sure the *Standard Axes* checkbox is checked.



The visualization screen provides a large amount of information about the dataset and the clustering results (for examples, see the WLV tutorial on-line, or the CD-ROM version of this report). To begin with, when you click on a file to visualize inside the *View* tab the fields to the right of the file display information about the cluster file. The top line indicates the name of the dataset, and whether the clusters were created directly, or are the result of merging together a larger number of clusters. The second line indicates the method used to classify the data and the [distance measure](#). Finally, the third line provides a unique time stamp. This information lets you quickly identify exactly which clustering result you want to visualize.

The *image height* field lets you control the height of the visualization image that will appear below the screen shown above. In this example it is 200, which permits all of the visualization information to fit on the screen at once. For printing, or visualizing larger datasets, you may want to make this value larger. Note that the bigger you make it, the longer it takes to upload the images to your computer.

In the visualization image that is created after the dataset is selected and the *Visualize* button is clicked, each cluster will have a unique color and a checkbox next to some highlighted text in the identical color. If you uncheck a box, the associated cluster will turn grey. Clicking on the *Select All* or the *Deselect All* buttons in the image has the effect of checking or unchecking all of the boxes. Being able to turn a cluster on and off makes it easier to see the extent of particular clusters in the image. If you click on the colored text, the program will send up a new window that has statistical information about that cluster, including the mean, standard deviation and max and min values for each variable.

If you click on the visualization image itself, you will get a large cross-hair, and the latitude and longitude of the closest data point will display on the two text fields. You can then get the actual data values for that data point by clicking on the *Data point info* button. Points that may be of particular interest to examine are the archetype points, which are shown at double size in the visualization. The archetype point in each cluster is the data point that is closest to that cluster's mean value. Thus, the archetype is a typical point for that cluster.

The *Cluster Summary* button is a way of saving all of the key information associated with a visualization in a compact form. You can choose whether the cluster summary is in *pdf* or *html* format (If you want to include the cluster summary information in another document, choose the *html* format.)

In some situations, you may wish to select a subset of the clusters to recluster in order to get a finer resolution in some areas. The *Create dataset from selected clusters* builds a new dataset from the currently selected clusters, just as it says. You can then work with this subset of the data as an independent dataset and recluster it as you wish.

Finally, the *View image as one Layer (for downloading)* button is also self-explanatory. If you want to include the visualization image in another document, use this button to get a useful version of the image. The standard visualization map is actually multiple layers to facilitate turning individual clusters on and off.

Visualize with User-Selected Axes

To execute a visualization with user-selected axes, first un-check the *Standard Axes* checkbox in the View tab. Then select a cluster file in the large list box and click on the *Visualize* button. WLV will then ask you to select which variables you wish to use as axes for plotting. After selecting the axes, click on the *Visualize* button in the lower frame to continue. The functionality of the resulting screen is similar to the standard visualization. The only major difference is the change in axes.

Overlay Visualization (standard axes)

To execute an overlay visualization, select a cluster file in the large list box in the View tab, then click on the *Overlay* button. Then select which variable to overlay from the options that appear in a menu box below the screen. If the variable is a continuous variable, enter how many classes to divide it into for the visualization. Then click *Select* to continue.

In the overlay case, the colored text markers below the image correspond to subdivisions of the overlaid variable. Probably the most important feature of this visualization is the *overlay statistics* button. This brings up a window with tables indicating the percentage of overlap between the variable classes and the computed clusters.

When executing an overlay visualization, you can use either a discrete or a continuous variable. The program will automatically determine whether a variable is discrete. If the variable appears to be continuous, it will divide the overlay variable into the specified number of classes. Currently, the program simply divides the overlay variable into equal magnitude divisions.

Dual Visualization

To execute a dual visualization, first select a cluster file in the large list box in the View tab, then click on the *Dual* button. Then select which cluster file to overlay and click on *Select* to continue.

This visualization is intended to permit visual comparison of two different clusterings of the same dataset. In addition, like the overlay visualization, it provides statistics on the overlap of the different clusters.

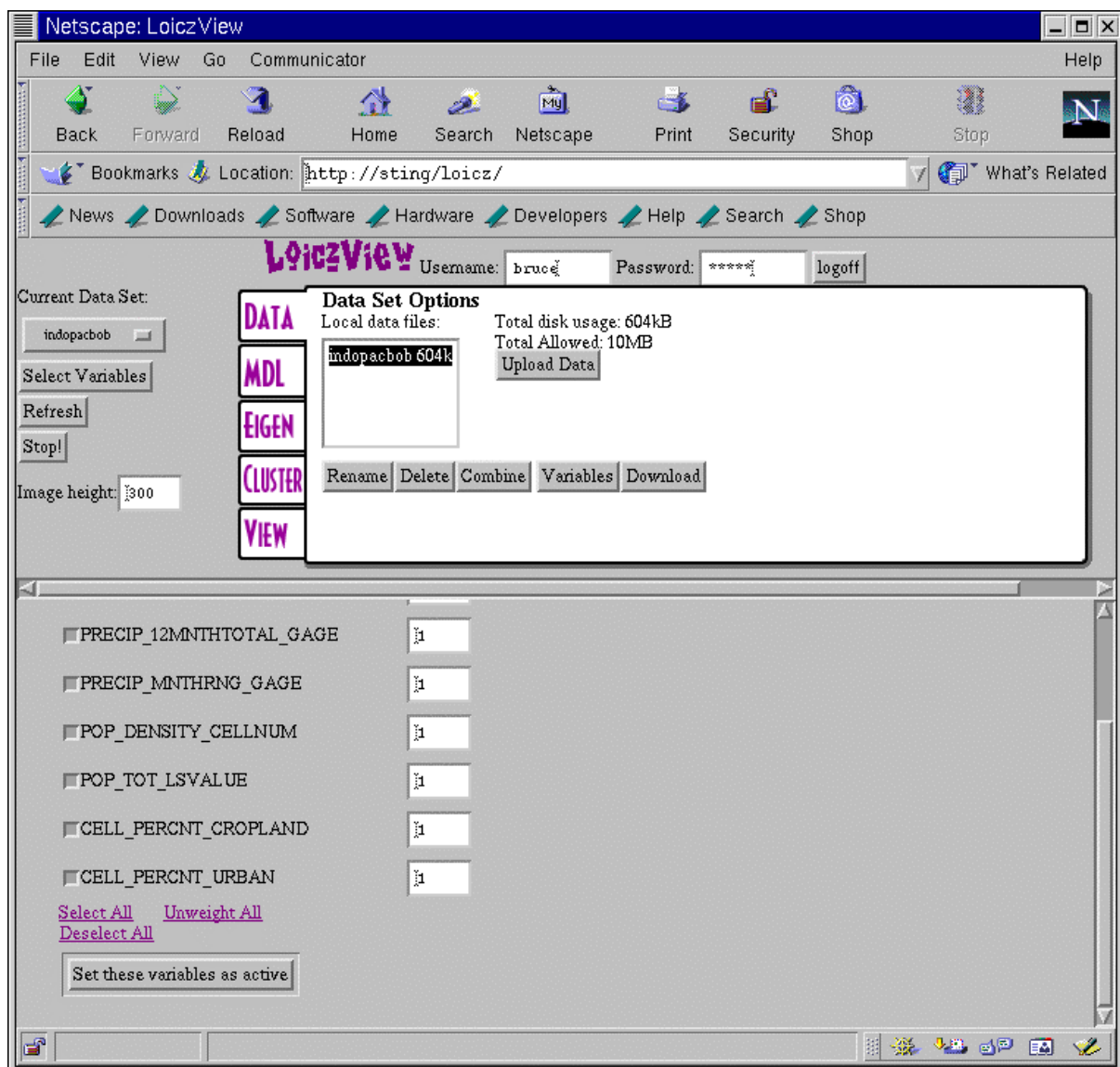
In the dual visualization, the checkboxes control the overlaid cluster result. The background image (what you get when all of the check boxes are unchecked) is the base visualization.

In addition to the visualization, you can also get the overlap statistics, and statistics on both clusterings. The highlighted text below the checkboxes gives access to this data.

VI-C: Variable Selection

One of the key decisions a user must make before executing the data analysis tools is to select the variables to use. It is not necessary to use all of the variables in a dataset, and often this is not desirable.

To select and weight the variables use the *select variables* button on the left of the WLV screen. You do not have to be in the Data tab to select variables. Clicking on the *select variables* button brings up a list of all of the variables in the dataset, as shown below.



Next to each variable is a checkbox and a text box. The checkbox selects whether a variable is to be used (active). The text box indicates the relative weight of the variable. If all of the weights are the same, then each variable is treated equivalently during the data analysis. Variables with higher weights will receive proportionately more importance in subsequent analyses. Weights can be any non-negative number, including decimals. It is good to try the first analysis of a dataset with uniformly weighted variables (the default values).

Once you have chosen the active variables and specified their weights, **you must click the button that says *set these variables as active* at the bottom of the variable list in order to save your settings.** After you have saved the active variables, WLV will print out the current set of active variables and their weights on the screen.

VI-D: Principal Components Analysis

Principal components analysis [PCA] is a tool for manipulating and visualizing a dataset, and for verifying and evaluating a particular clustering. It can be an extremely useful tool for understanding the relationships in a dataset, but you have to be careful how you interpret the results.

Overview

PCA is, at its essence, a rotation and scaling of a dataset. The rotation is selected so that the axes are aligned with the directions of greatest variation in the dataset. The scaling is selected so that distances along each axis are comparable in a statistical sense. Rotation and scaling are linear operations, so the PCA transformation maintains all linear relationships.

The rotation and scaling for PCA are given by the eigenvectors and eigenvalues of the covariance matrix. The covariance matrix contains the relationships (correlations) between the variables in the dataset.

One way to think about PCA is that it generates a set of directions, or vectors in the data space. The first vector shows you the direction of greatest variation in the dataset; the second vector shows the next direction of greatest variation, and so on. The amount of variation represented by each subsequent vector decreases monotonically.

In many datasets, the variables are related to one another (sea surface temperature and air temperature along a coastline, for example). What this means is that there are usually fewer directions (vectors) of useful variation than there are variables in the dataset. The directions of useful variation are sometimes called *factors*. The factors are weighted combinations of the variables, where the weights describe the influence of each variable on that factor. If there are fewer important factors than there are variables in the data, then we can express the dataset with fewer variables. Furthermore, the new variables are independent, which is a good property for clustering and analysis.

So once you've executed a PCA, what can you do? The following table gives an overview.

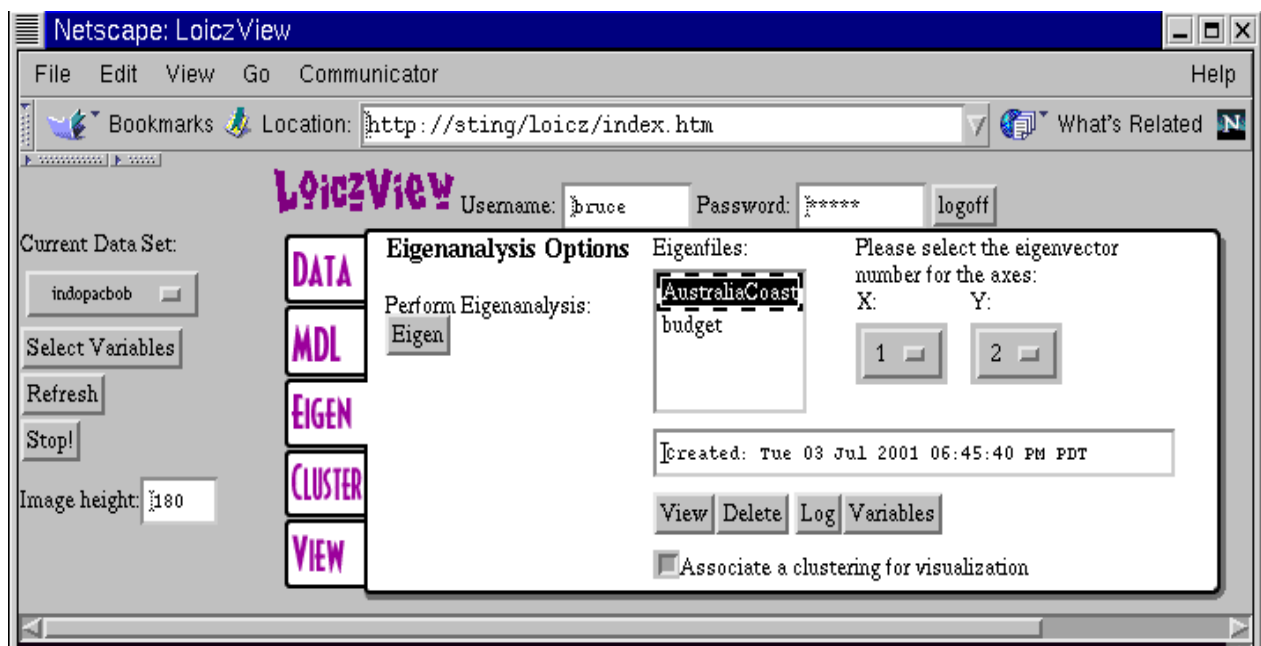
What can you do?	Summary	What's it good for?
Examine the eigenvalues associated with each principal component	Look at a plot of the eigenvalues from the first to the last principal component. The plot will generally fall off sharply from the first component and then level off.	This plot gives you an idea of how many independent <i>factors</i> there are in the dataset. When the plot levels off, the remaining principal components do not explain much about the dataset. Therefore, only the first N principal components really matter in terms of explaining the variation in the data.
Examine the principal components	Look at the numerical values associated with each variable for the first 2-3 principal components	This tells you what variables are the most important for each principal component. Variables with large magnitude weights in the principal component vector are more important. Variables with similar magnitudes are correlated.

Transform the dataset	Project each data point onto the first N principal components, where N is determined as noted above (by looking at the plot of the eigenvalues).	This reduces the size of the dataset and makes the variables independent, both of which generally make the clustering algorithms more effective.
Visualize the data projected onto the first 2-3 principal components	Take the dot product of each data point with the first 2 or 3 principal components. The resulting plot shows the data in the principal component space (2D or 3D).	This is a good space in which to view the data. You can project the clusters into this space and verify whether they form coherent groupings. It may also give you a sense as to how many natural groupings exist in the data space

There are lots of other sites devoted to PCA, factor analysis, and their applications. One that is a nice description with biological examples is <http://www.okstate.edu/artsci/botany/ordinate/PCA.htm>. If you are comfortable with statistics, you might try this high-level discussion of PCA and factor analysis: <http://www.statsoftinc.com/textbook/stfacan.html>.

Principal Components Analysis in WLV

The *Eigen* tab in WLV gives you access to the PCA tools. The screen shot below shows the control panel for the analysis. After selecting a dataset, the Perform Eigenanalysis button creates an eigen file which can be viewed.



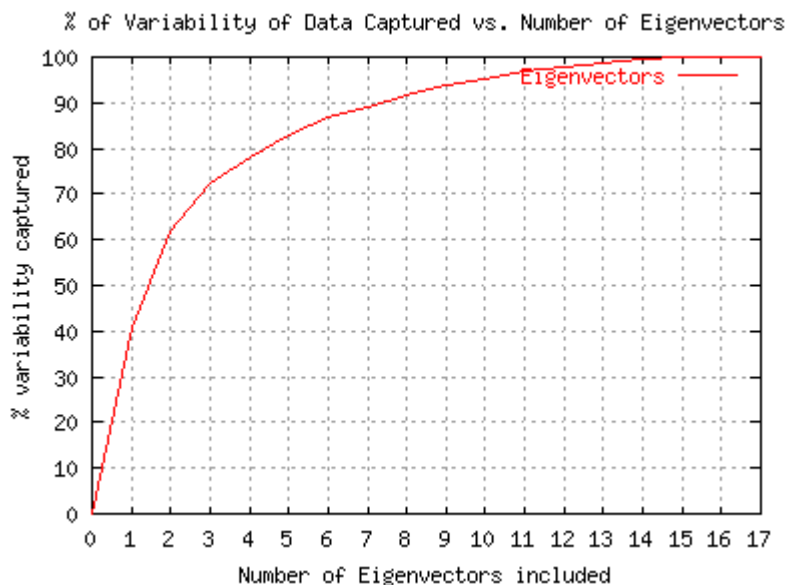
In order to get a plot of the eigenanalysis results, you have to execute the following steps:

1. Perform an eigenanalysis on the dataset selected by clicking the *Eigen* button on the *Eigen* tab. Once the computer has completed the eigenanalysis, the dataset's name will appear in the *eigenfiles* list box.
2. Select an eigenfile from the list box to view.
3. Select which principal components to use in the visualization. The default values are the first two principal components.

4. If you have executed a clustering on this dataset, you may want to associate a clustering with the visualization. To do this, click the *Associate a clustering for visualization* checkbox.
5. Click on the *View* button to generate the visualization. Like the standard visualization screen, if you have associated a clustering with the visualization, you can turn clusters on and off and view the cluster data. Likewise, you can generate a cluster summary in either PDF or HTML format.

Examining the eigenvalues and eigenvectors

As noted above, it is useful to have a plot of the eigenvalues. If you click on the *EigenInfo* button in the visualization frame it will bring up a window with a plot of the eigenvalues, and the numerical values for the eigenvectors. Below is an example plot of the eigenvalues for the AustraliaCoast dataset (all variables).



This example shows a typical eigenvalue plot. Depending upon your point of view, it would be possible to argue that anywhere from 4-10 of the principal components are important. The principal components from 10-17 account for less than 5% of the variability of the dataset.

3-D visualization

WLV also permits visualization of the dataset in 3D, projected onto the first three principal components. To assist in visualizing the 3D space, the 3D projection is animated (see example on website, or the CD-ROM of this report).

VI-E: Minimum Description Length

"Pluralitas non est ponenda sine neccesitate"

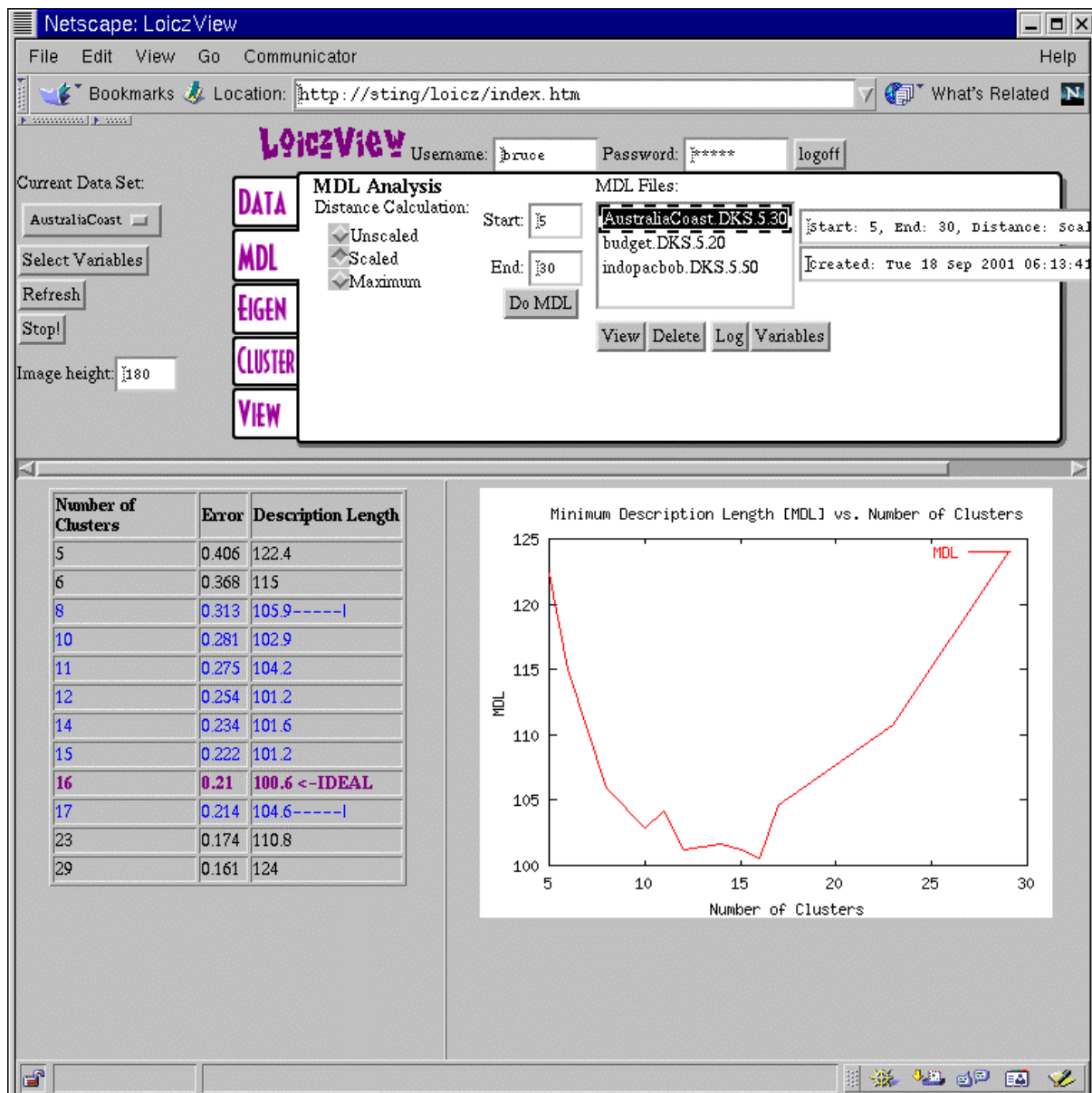
- Friar William of Occam

[Occam's razor](#) states that "*entities should not be multiplied unnecessarily.*" One of the major questions when trying to cluster data is how many clusters to create. Sometimes we have an *a priori* answer to this question based on knowledge of the dataset. Other times we have to fit the data into a certain pre-specified number of categories or management units. When exploring a dataset for the purpose of discovering relationships within it, however, it is important to avoid preconceived notions of the complexity of the data.

One way to explore the appropriate number of clusters is to simply try a number of different clusterings and see what provides the most interesting result. We can also use a concept like Occam's Razor to give us guidance. The main point of Occam's Razor applied to clustering is that, at some point, having more clusters for a given dataset is not worth the added information it may provide.

The Minimum Description Length [MDL] principle is a mathematical method for applying Occam's Razor to models for data - a set of clusters is a model for a given dataset. The MDL principle says that the model that takes the least number of bits to represent is the best model for a set of data. In the case of clusters, we can encode the number of cluster parameters and the representational error as the amount of information it takes to represent the data. When these two are balanced, then we have the optimal number of clusters.

The MDL tab in WLV allows the user to execute an MDL analysis of a particular dataset. It calculates many clusterings with different numbers of clusters and calculates the description length for each run. It then provides a plot of the description length values and suggests a range of values for the number of clusters to use. The screen shot below demonstrates a typical result for the AustraliaCoast dataset.



In our experience with the MDL tool, the number of suggested clusters tends to be higher than experts have found useful. However, the low end of the suggested MDL range tends to be within an acceptable range. If you consider the graph of MDL values, it is clear that from 10 to 16 clusters the descriptions lengths all fall within a similar range. For this dataset, experts have found 10-12 clusters to be a useful number. Below that, important features in the coastline get merged together and lost. Above 16, the value of additional clusters to human analysis is unclear.

To execute an MDL analysis, first select the dataset to analyze. Then enter a starting and ending number of clusters to examine. Note that as the number of clusters gets higher, the clustering takes longer. Keep the *End* number of clusters as small as is reasonable for faster analysis. Once these fields are set, then click on *Do MDL*. When the analysis is complete, you can click on the MDL File in the list box and then click *View* to see the chart and plot of the MDL results. Clicking on the *Variables* button will show you the active variables for that particular MDL analysis. Note that you will usually get different MDL results for different variables.

In our experience with the MDL tool, the number of suggested clusters tends to be higher than experts have found useful. However, the low end of the suggested MDL range tends to be within an acceptable range. If you consider the graph of MDL values, it is clear that from 10 to 16 clusters the descriptions lengths all fall within a similar range. For this dataset, experts have found 10-12 clusters to be a useful number. Below that, important features in the coastline get merged together and lost. Above 16, the value of additional clusters to human analysis is unclear.

To execute an MDL analysis, first select the dataset to analyze. Then enter a starting and ending number of clusters to examine. Note that as the number of clusters gets higher, the clustering takes longer. Keep the *End* number of clusters as small as is reasonable for faster analysis. Once these fields are set, then click on *Do MDL*. When the analysis is complete, you can click on the MDL File in the list box and then click *View* to see the chart and plot of the MDL results. Clicking on the *Variables* button will show you the active variables for that particular MDL analysis. Note that you will usually get different MDL results for different variables.

Use the MDL tool as a guide in your exploration. As with all tools, use your judgement as to whether the results make sense. If they do not make sense, figuring out why can often lead to new insights about the dataset.

We have done our best to ensure the correctness of the underlying tools, but we can make no guarantees about the correctness or utility of the results. What you get out of WLV is largely dependent upon what you put into it. Furthermore, WLV is still under development as we continue to add features and tools. The help files on the web-site and in the CD-ROM are intended to help users understand how to effectively use WLV for their own tasks.

If you either A) have a suggestion about how to improve WLV, or B) believe you've found an error in one of the tools, please email maxwell@swarthmore.edu. New users may use one of the numbered user accounts (user##) or ask for a username and password by emailing the address above.