

## Pathfinder 9.0

Contents (Ctrl click opens section)

Pathfinder 9.0.....	1
What's New? .....	1
Quick Start .....	1
Introduction: Pathfinder APP.....	2
Pathfinder APP.....	3
Derive Networks .....	4
Merged Networks.....	7
Node Labels .....	7
Proximity Data File Formats.....	8
Matrices (including half matrices) and Lists .....	9
Coordinates, Feature, or Attributes.....	10
Menu Bar .....	11
Proximity Info.....	11
Network Info.....	12
Network Similarity.....	12
The Structure of Pathfinder Data in MATLAB .....	13
Sample Data Sets .....	14
Example Table .....	15

### *What's New?*

Version 9.0 of Pathfinder only runs under MATLAB 2020a or later. You must have MATLAB installed in order to use it, or you can install the standalone version which will download and install the appropriate MATLAB Runtime library.. The software should run on various platforms (PC, Mac, etc.).

The new features are:

- A new data format providing for the input of coordinates or feature values for each item (node) and a network layout option that uses some of the input coordinates to position the nodes in network drawings.
- New network generation techniques in addition to the standard Pathfinder methods. These include nearest neighbor networks, and threshold networks.
- Provision for selectively removing weaker links in network displays to reveal the relative strength of the links.
- Display of the links found in multiple networks.
- The documentation includes information about the MATLAB data file created by using the software.

### *Quick Start*

When you run Pathfinder for the first time on a computer, a folder “pfdir) will be created in your user folder. The pfdir folder is used to keep track of your project folders and to hold some sample data sets to illustrate the working of the APP. You can “Add Data” by clicking on the button and then selecting one or both of the sample proximity data files. Then “Derive Networks” by clicking that button, and the

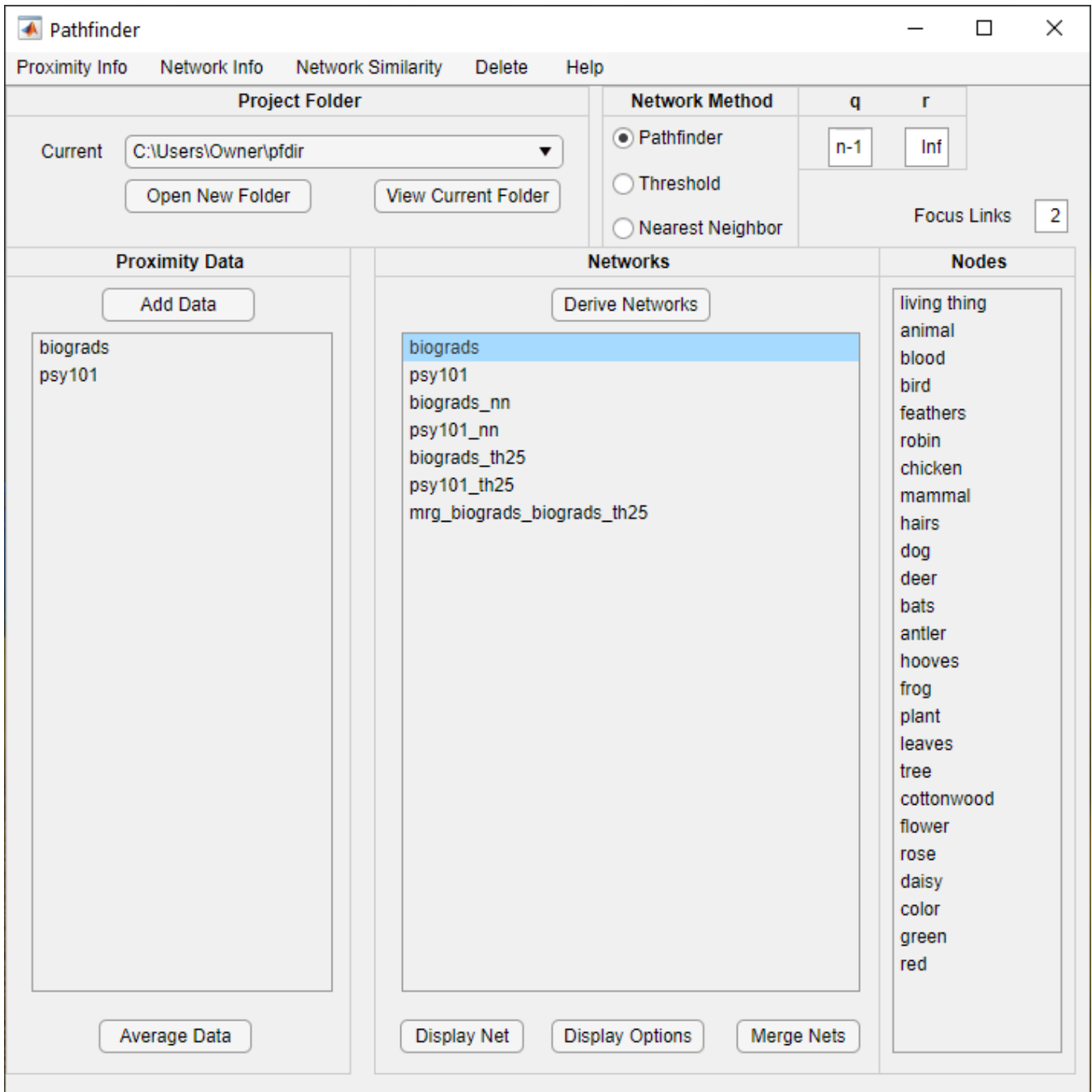
Pathfinder Networks ( $q = n-1$ , and  $r = \text{infinity}$ ) will be created and will appear in the Networks list. Select one of these and click “Display Net” to see the selected network. The “Help” menu opens documents like this one.

To prepare data files for Pathfinder analysis, you must follow the conventions described in the section on Proximity Data File Formats. As the data are read in, the node labels are obtained from a file described in the section on Node Labels.

## **Introduction: Pathfinder APP**

A Pathfinder network is derived from proximities for pairs of entities. Proximities can be obtained from similarities, correlations, distances, conditional probabilities, or any other measure of the relationships among entities. The entities are usually concepts of some sort, but they can be anything with a pattern of relationships. In the Pathfinder network, the entities correspond to the nodes of the generated network, and the links in the network are determined by the patterns of proximities. For example, if the proximities are similarities, links will connect nodes of high similarity. With distance data, the network will connect close nodes. The links in the network will be undirected (lines) if the proximities are symmetrical for every pair of entities. Symmetrical proximities mean that the order of the entities is not important, so the proximity of  $i$  and  $j$  is the same as the proximity of  $j$  and  $i$  for all pairs of  $i,j$ . If the proximities are not symmetrical for every pair, the links will be directed (arrows).

Pathfinder uses two parameters. (1) The  $q$ -parameter which constrains the number of indirect proximities examined in generating the network. The  $q$ - parameter is an integer value between 2 and  $n-1$ , inclusive where  $n$  is the number of nodes. (2) The  $r$ -parameter defines the metric used for computing the distance of paths (cf. the Minkowski  $r$ -metric). The  $r$ -parameter is a real number between 1 and infinity, inclusive. A network generated with particular values of  $q$  and  $r$  is called a PFnet( $q, r$ ). Both of the parameters have the effect of decreasing the number of links in the network as their values are increased. The network with the minimum number of links is obtained when  $q = n-1$  and  $r = \text{infinity}$ , i.e., PFnet( $n-1, \text{infinity}$ ). With ordinal data, the  $r$ -parameter should be infinity (inf). Other values of  $r$  require data measured on a ratio scale. This level of measurement is difficult to achieve, so usually  $r$  should be set to infinity. The  $q$ -parameter can be set to the value that yields the desired number of links in the network. As  $q$  decreases, links may be added to the network.



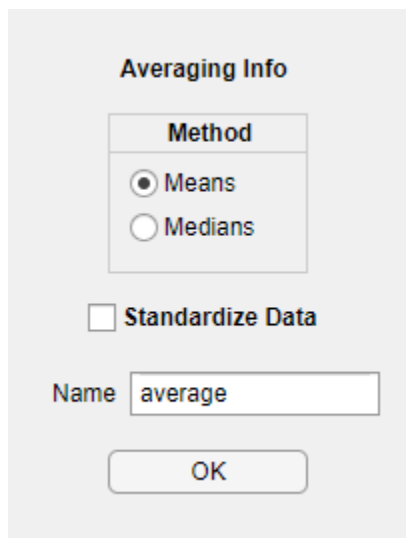
## Pathfinder APP

The Pathfinder APP allows you to initiate the functions accomplished by the software. Let's review the various controls and options. Each new window launched by the software will be described in detail in subsequent sections. The **Bold Items** below refer to sections of the interface and action you can take..

**Project Folder** is where the data you analyze and the results you create are stored. Select this folder as you begin a project. Different projects should be located in different folders. It's best to keep all of the data files in this same folder, but you can retrieve them from other folders as you work if you prefer. In any case, the results of your work will be stored in the Project Folder. This folder should also contain the terms.txt file or other term files as discussed below. You can change projects by selecting one in the

drop-down menu which remembers the previous projects you have worked on. Start a new project with **Open New Folder** which will allow you to select the project folder you wish to work on. You should select the folder containing your proximity data and terms file. The next time you start Pathfinder, the last project you worked on will load automatically. View Current Folder will just open an Explorer window with that folder allowing you work on files in that folder including opening files with results of Pathfinder analysis.

In the **Proximity Data** panel, there is a button to **Add Data** and a list box which will show the data sets that have been added to the project. Clicking **Add Data** will open a file selection window pointing to the **Current** folder. It is set to look for txt files with prx in the name, like bio.prx.txt, but you can change the files listed if the default is not appropriate for your data file names. Just select the data file or files you wish to add and click **Open**. The selected files will be read to create the appropriate data for further analysis.



Clicking the **Average Data** button after selecting the data sets to be averaged will open a dialog window for you to select some options for averaging the distances for each pair of items across the selected data sets. You can use either **Means** or **Medians** and can determine whether to **Standardize Data** or not. Medians are best if there are any missing data items or values outside the min and max data values because that leads to infinite distances. Standardizing is especially appropriate if the data sets to be averaged differ in scale or variance. Modify the **Name** box to a name appropriate for your data sets. Your name will have the number of data sets appended to it after the averaging is complete. The averaged data set will appear in the **Proximity Data** list after you click **OK**.

### *Derive Networks*

There are three different methods for deriving networks: **Pathfinder**, **Threshold**, and **Nearest Neighbor**. If you **select** data sets to analyze, networks for those sets will be derived. If you do not select data sets, all data sets will have networks derived from them. When you select a method, parameters for the method are made available for editing. When you **select Pathfinder Networks**, you can **edit** the values of the *q* parameter and the *r* parameter. The default values for the parameters are shown in the figure. Networks created with these parameters will have the same names as the data sets they are derived from. Any change in the parameters will lead to the creation of a new network, and the name will include the values of the changed parameters. For example if we analyzed the data, bio, with *q* set to 2, the name of the network would be bio\_q2. If we used the default values, the network name would be the same as the data set, bio.

The Threshold Method allows you to specify how many links you would like to see in the network by providing a multiplier for the number of nodes (1 yields the number of nodes). The resulting network will have at least that many links by linking all nodes whose proximity data value is within a cutoff

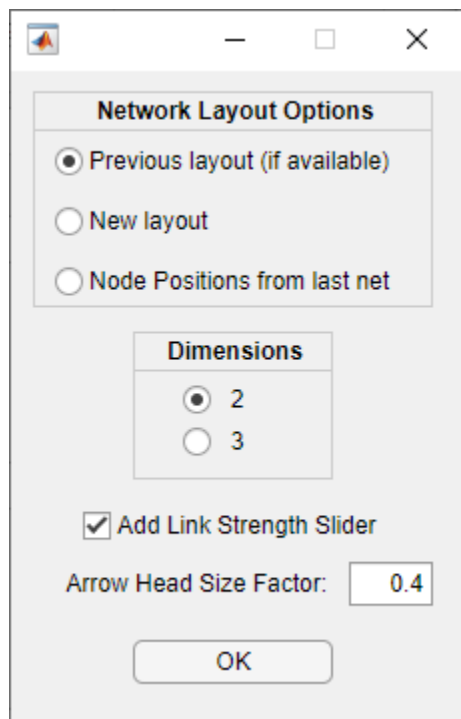
value yielding the desired number of links (smallest for dissimilarity or largest for similarity). There will be more links than requested when tied proximity values exist at the cutoff.

Nearest Neighbor Networks simply identify the closest (or most similar for similarity data) node to each node. Nearest Neighbor nets are directed networks with arrows pointing from a node to its nearest neighbor. If there are ties, a node may have more than one nearest neighbor. Nearest Neighbor Networks are usually disconnected. Nearest Neighbors have `_nn` appended to the data name.

Clicking **Derive Network** will initiate the network creation and the names of the nets will appear in the **Networks** list panel.

When a network is selected, the nodes for that network are shown in the **Nodes** panel. Clicking **Display Net** will create and display a picture of the network. Selecting a node will produce a picture of the part of the network focused on that node. All nodes within the **Focus Links** number of links from the selected node will be included in the focus picture. If multiple networks are selected (using Ctrl Click), then **Merge Nets** will produce a network which merges the links from all selected networks (the union of the links in the selected networks).

### Options

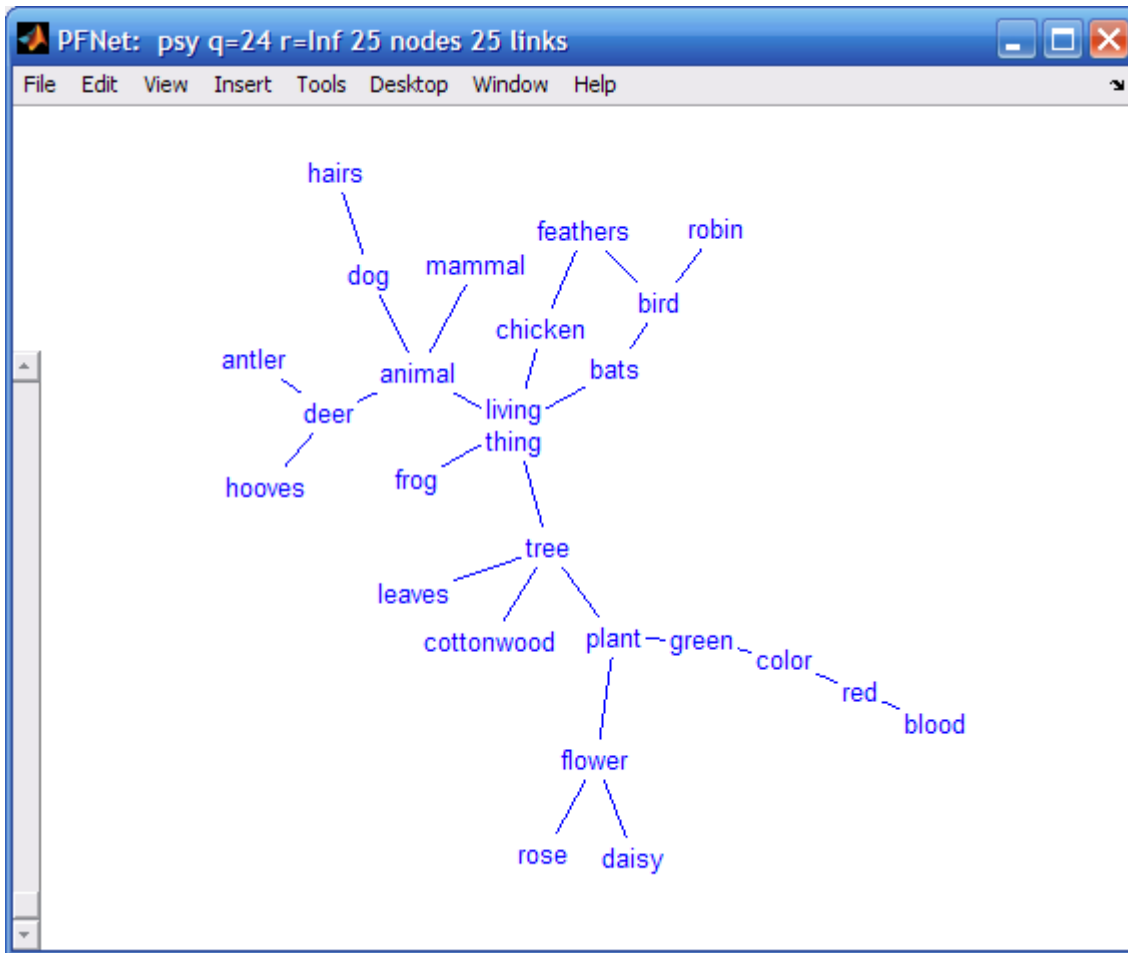


Checking the Options box will allow you to change options for drawing networks. You can **select** whether to provide a 2 or 3 dimensional display. Because you can rotate the pictures, 3D can often be useful. However, 3D is not available for directed networks. There are various options for determining the positions of the nodes in the picture. Select **Previous layout (if available)** if you want to recall the positions from an earlier display of the same network. If the positions are not available, it will compute new ones. **New layout** will discard the old positions and generate new ones. **Node Positions from last net** will keep the nodes in the same position as the last displayed network. This can facilitate comparing different networks by keeping the nodes in the same position. If you first generate a merged network with two network you want to compare, then layout the merged network, next Retain Node Positions while drawing each of the two networks in turn. **Add Link Strength Slider** adds a slider to the drawing of a network. The slider allows you to selectively remove the weakest links in a network in steps until all of the links are removed.

**Arrowhead Size Factor** allows you to control the size of arrowheads in layouts of directed networks – values less than one will decrease the size, values greater than one will increase the size.

### Network Layouts (Pictures)

Here is a picture of a displayed network. The title shows the name of the networks and the  $q$  and  $r$  parameters used to generate it. The tools allow you to move it, resize it, and rotate it. The changes you make to its appearance are saved so when you view it again with **Previous layout (if available)**, it will show up just as you left it. Rotations with 3-D layouts are especially impressive. The *File* menu has the usual options for printing and saving. You can save the figure using various formats to enable you to get the image into other applications.



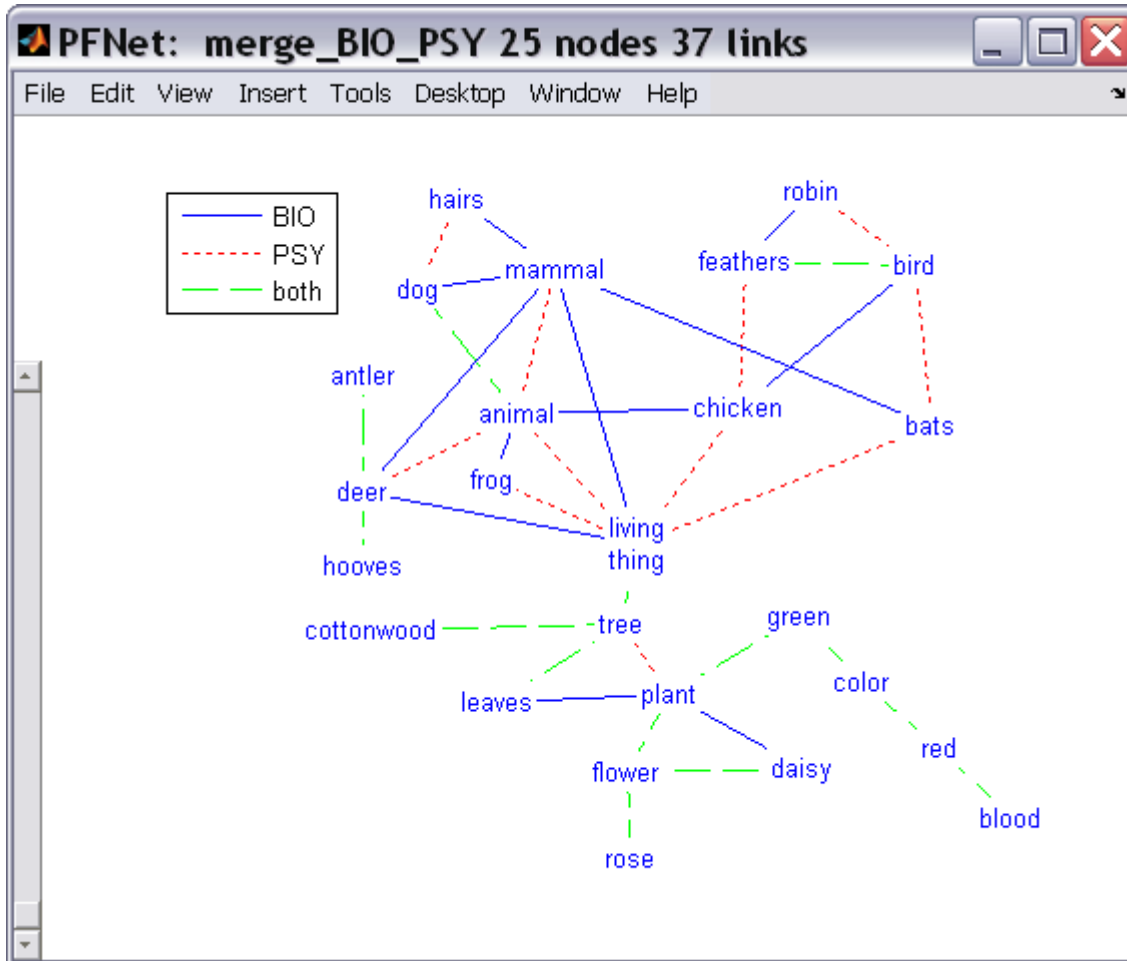
You can **click** on a node and **drag** it to move it to a new position (links follow). When you move the nodes to new positions, the new node positions are saved so future layouts of the same network will appear as you left it provided you select **Previous layout (if available)** in the **Options**. If you **double click** on a node, a network focused on the clicked node will appear.

As you **move the slider** on the left-hand side, links will drop out of the network in the inverse order of strength. Weaker links will disappear first. At the top, no links will be present. As the slider is moved down. The strongest links will appear in order of strength.

Loops (links from a node to itself ) are shown with a border around the name of the node.

## Merged Networks

If two networks are selected, you will see which links are in each of the networks and which are in both networks. When more networks are selected, the links are identified by how many networks included the link. Here is an example with two networks:



## Node Labels

*Nodes* is a list of node labels for a selected network. It shows a list of the terms taken from a file, "terms.txt" or "terms," if it exists in the Project Folder. If different data sets have different terms, they should be in files called "<data>.trm" or "<data>.trm.txt" where <data> is the name of the corresponding data file, "<data>.prx" or "<data>.prx.txt". In this case, a list of these ".trm" files is shown in the terms window. The terms are the node labels for drawings of the networks. **It pays to keep the labels short so the networks look reasonable.** The terms file must follow a simple format. The label for each node is placed on a separate line in a text file. The first line is the label for the first node and so on. If you are using only one set of labels for one or more networks, use terms.txt as the name of the file. If you have different terms for different data sets in a single directory (folder) on your disk, use the name prxfile.trm.txt where prxfile is the name of the proximity data file. For example, if you have proximity data files called foo1.prx.txt and foo2.prx.txt, the corresponding terms files should be named foo1.trm.txt and foo2.trm.txt. These naming conventions are used by the Pathfinder software. If an appropriate terms file cannot be located for a given data set, the nodes will be numbered

consecutively. The number of lines in the terms file must match the number of nodes exactly. Blank lines at the end may prevent the file from being recognized as appropriate.

## Proximity Data File Formats

The data may be in the form of similarities, dissimilarities, probabilities, distances, coordinates, or features. With dissimilarities or distances, smaller numbers represent pairs of entities that are close or similar or related and larger numbers represent pairs of entities that are distant or dissimilar or unrelated. The opposite is true of similarities, probabilities, or relatedness i.e., smaller numbers represent entities that are distant or dissimilar or unrelated and larger numbers represent pairs of entities that are close or similar or related.

With distance measures, the distance between an entity and itself (the major diagonal entries in a data matrix) is usually 0 (zero). Pathfinder will handle non-zero entries on the diagonal, however. Such values will lead to "loops" (links from a node to itself) in the network, although they will not be displayed. Data derived from transition probabilities may lead to such non-zero entries for the diagonal. You must be certain that the diagonal in a matrix contains meaningful values. If all diagonal values are equal, they are taken to have 0 distance (or maximum similarity). All entries in the data must be positive or zero. Negative numbers are not allowed. Values outside the minimum – maximum range (see below) will never produce links in the networks generated.

A strictly formatted text file is required for proximity data. Here is a small example of such a file:

```
data
similarity
5 nodes
comment
10 minimum value
90 maximum value
lower triangular matrix
32
40 49
32 38 53
73 63 77 18
```

The required format of a data file is described below.

Data file format. / indicates alternatives:

- 
- Line 1:** Identification as data file = Data/DATA/data
  - Line 2:** Type of data = dissimilarity /distance /dis/similarity/sim/probability/prob/
  - Line 3:** Number of nodes = integer
  - Line 4:** comment : short description
  - Line 5:** Minimum data value = real number
  - Line 6:** Maximum data value = real number
  - Line 7:** Order of data values = matrix/upper/lower/list/coord/featur/attrib
  - Line 8:** Data
  - Line 9:** Data ...

.

.



**Line ?:** Data

-----  
 The lines in the file must be organized as shown above. For the first six lines, the program reads only the first entry on the line and then goes to the next line. Anything can follow the first entry on the line; the program doesn't use it. Some descriptive information on the line can help to keep things straight, especially after some time has elapsed. Details on the required input are as follows:

- Line 1.** "Data," "DATA," or "data" is used to identify the file type
- Line 2.** "similarity," "dissimilarity," "probability," or "distance," (or "sim," "dis," or "prob,") are used to indicate the direction of the data. With similarity data, larger values represent greater similarity. With distance data, smaller numbers mean closer (or more similar).
- Line 3.** The number of nodes (or entities) to be analyzed. The word nodes is optional
- Line 4.** Comment: a short description of the data
- Line 5.** the minimum value in the data set. Words are optional.
- Line 6.** the maximum data value. Words are optional.  
 The minimum and maximum values are used as cutoffs in handling the data. Any value in the data outside the minimum - maximum range will never become a link in networks. In other words, two nodes with a proximity value outside the range can never be linked in any network generated by the program. Missing data can be handled by using values outside the range, or by using the "list" format for your data.
- Line 7.** "matrix" or "upper" or "lower" or "list" or "coord" or "featur." This line specifies the nature of the data following this line. Various ways of supplying proximities are possible based on a full matrix, an upper triangle, a lower triangle, a list, or vectors of features, attributes, or coordinates. The upper and lower (half-matrix) methods do not include the major diagonal (the proximity of an item with itself). Such values can lead to "loops" in the networks. If loops are appropriate, either a matrix or a list format must be used. The lines of data in the file do not have to have these shapes, but the data must be in the same order as they would if the lines did have those shapes when we characterize order as reading across each line in turn. The following examples may be of help.

***Matrices (including half matrices) and Lists***

matrix:	lower:	upper:	list:
0 1 3 2 3	1	1 3 2 3	10 pairs
1 0 1 4 6	3 1	1 4 6	symmetric
3 1 0 5 5	2 4 5	5 5	2 1 1
2 4 5 0 4	3 6 5 4	4	3 1 3
3 6 5 4 0			3 2 1
			4 1 2
			4 2 4
			4 3 5
			5 1 3
			5 2 6
			5 3 5
			5 4 4

These four sets of data are all the same. Of course, if your data are asymmetric, they must be input as a matrix or a list with "nonsymmetric" or "asymmetric" specified. If the data are symmetric, any of the four shapes is acceptable. With the list format, the number of pairs in the list must be specified on the line following "list." The next line specifies whether the pairs define symmetric or nonsymmetric data. With the list format, missing pairs will never be linked.

Following the header lines, the data must occur as discussed above. With  $n$  nodes (or entities), a matrix must contain  $n^2$  data elements, upper or lower triangles must contain  $n(n-1)/2$  data elements. The list data must contain 3 numbers for each pair listed, the source entity, the destination entity, and the proximity.

### ***Coordinates, Feature, or Attributes***

When data are in the form of coordinates or attributes, the appropriate format starting at Line 7 is as follows:

**Line 7.** "coord" or "feature" or "attrib" indicating that the data to follow are to be interpreted as vectors of numbers, one vector for each item or node.

**Line 8.** Integer = The number of dimensions, attributes, or features in the vector for each item

**Line 9.** "euclidean" or "city block" or "dominance" or "hamming" or "cosine" plus optional: "standardize"

**Line 10:** Vector for item 1

**Line 11:** Vector for item 2

Etc. (one vector for each item)

Here is a small example for 5 nodes in 2 dimensions:

coord:

```

2 dimensions
Euclidean Standardized Distances
9 1
8 4
2 6
5 4
1 8
    
```

In this case, each of 5 items (nodes) has values on each of two dimensions as in representing points in 2-dimensional space. The number of items or nodes is specified in Line 3 of the data file. The number of dimensions or features must be given following the "coord" or "feature" or "attribute" line. There can be any number of dimensions. With coordinates, we often think of the data as coming from the spatial layout of items. With features or attributes, the data may be the ratings of each item on several different Likert-type scales so the scales can be thought of as features. Features could also identify the presence or absence of features using 1's and 0's. The line following the number of dimensions or features must have "Euclidean," "City Block," "Dominance," "Hamming," or "Cosine." This determines how distance data are computed from the coordinates or features. Distances are computed for all pairs of items. With **Euclidean**, we find the straight line distance between the items (nodes) in multidimensional space. With **City Block**, the distances are determined by summing the distance between items (nodes) on each dimension or feature. *City Block is usually the most appropriate method for Likert-type rating scale*

*data*. With **Dominance**, the distance between items is the maximum difference for the items across all the dimensions or features. **Hamming** distances are the number of features on which items (nodes) differ. **Cosine** bases the distances on the cosine of the angle between the vectors which varies from 1 to -1. Because cosine is a measure of similarity rather than distance, a distance value is obtained by: 1 – cosine. This varies from 0 to 2 with 0 meaning vectors pointing in the same direction, and 2 meaning vectors pointing in opposite directions. You can also determine whether to **standardize** the vectors before computing distances. When “standard” is included on the line, the vectors are normalized to have a standard length 1. This is often appropriate when the vectors vary greatly in magnitude. For example, with vectors from Latent Semantic Analysis (LSA), computing Euclidean distances with standardized vectors provides a measure which is very similar to the cosine measure usually used in LSA work.

Here is a complete sample data file with the coordinate/feature format:

<pre>data distances 5 nodes 0 decimal places 1 minimum data value 6 maximum data value features: 4 features City Block Metric 2 1 3 2 1 5 1 4 3 1 3 5 2 4 1 2 3 6 5 4</pre>	<p>For an example of how distances are created from the data, the distance between items 1 and 2 (rows 1 and 2) is computed as follows:</p> $ 2-1  +  1-5  +  3-1  +  2-4  = 9$ <p>the distance between items 2 and 4 is:</p> $ 1-2  +  5-4  +  1-1  +  4-2  = 4$ <p>Distance is computed for all pairs of items in this way.</p>
---	---

Following the header lines, the data must occur as discussed above. The coord data must have a value on each dimension or feature for each of the nodes. The data elements must be separated by one or more spaces and/or line breaks.

## Menu Bar

The Menu Bar gives access to Information about **Proximity** Data, **Networks**, and Network **Similarity** as well as a means to **Delete** various data sets and networks from the project and to get added **Help**. Delete and Help are self explanatory. The following sections provide a discussion of the information provided by the various selections. The information is presented to you in tables which you can view and save as .csv files. Such files can be opened by spreadsheet programs which displays the table in a spreadsheet.

### *Proximity Info*

#### **Proximity Details**

Provides information about the proximity data in the project including a measure of the coherence of the data. Coherence reflects the consistency of the data. The coherence of a set of proximity data is based on the assumption that relatedness between a pair of items can be predicted by the relations of the items to other items in the set. First, for each pair of items, a measure of relatedness (the indirect measure) is determined by correlating the proximities between the items and all other items. Then, coherence is computed by correlating the original proximity data with the indirect measures. The higher this

correlation, the more consistent are the original proximities with the relatedness inferred from the indirect relationships of the items. The coherence measure often correlates with expertise (or degree of learning). Very low coherence values (less than 0.20 or so) may indicate that raters did not (or could not) take the rating task seriously. Very low coherence may indicate an error in entering the proximity data so that it is scrambled in some way. For example the proximities may be in the wrong order for the format specified so low coherence is likely to result.

### **Proximity Correlations**

The data correlations are simply the Pearson product-moment correlations of the data sets. These correlations provide a measure of the similarity of data sets.

### **Proximity Distance Matrix**

Shows a table with the distances between all pairs of nodes.

## ***Network Info***

### **Net Details**

Provides information about the networks.

### **Net Properties**

Shows graph theoretic properties of a selected network. This includes the number of links to each node (Degree) of a node, the node or nodes with maximum degree, the Eccentricity (the maximum number of links between the node and all other nodes), the Center (the node or nodes with minimum Eccentricity), the Average Minimum Distance between each node and all other nodes, and the Median (the node or nodes with minimum Average Minimum Distance).

### **Net Link List**

A table showing the links in a network with the distance associated with each link and the type of link.

### **Net Links Between Nodes**

A matrix with the minimum number of links between nodes in a network.

## ***Network Similarity***

The similarity between two networks is determined by the correspondence of links in the two networks. The **Net Similarity** is the **Number of Common Links** divided by the total number of unique links in the two networks. Two identical networks will yield a similarity of 1 and two networks that share no links will yield similarity of 0. The measure is the proportion of all the links in the two networks that are in both networks. Also some statistical information is computed about the similarity. The **Net Similarity Above Chance** is the similarity minus the chance similarity. The **Probability Sim or Greater** is the probability that the two networks would share the given number of links or more by chance. It can be used as a statistical test of the similarity of two networks. These values reflect how much more (or less for negative values) similar two networks are than would be expected by chance. The similarity information is presented in a table which can be saved as a file. If one or more networks are selected, the table shows the selected nets compared to all others. If no network is selected, the table shows every net compared to every net.

Similarity measures and their applications are discussed in: Goldsmith & Davenport (1990), "Assessing structural similarity of graphs," and Goldsmith & Johnson (1990), "A structural assessment of classroom learning." Both in R. Schvaneveldt (Ed.), *Pathfinder Associative Networks: Studies in Knowledge Organization*. Norwood, NJ: Ablex.

## The Structure of Pathfinder Data in MATLAB

For users of MATLAB, it may be useful to know about the data created by the Pathfinder software. All of the data read into Pathfinder and all of the results generated by the use of the software are found in a single MATLAB variable: **pf**. This variable is saved in the MATLAB file, **pf.mat**. You can bring the variable into the MATLAB workspace by the command: "load pf". The global variable **pf** which is available in the MATLAB workspace is a MATLAB structure containing the following fields:

```
pf
    prx: [1x1 struct]
    net: [1x1 struct]
```

**pf.prx** is a structure containing fields which are structures containing proximity data and information about the data. **pf.net** is a structure containing fields which are structures containing networks and information about the networks.

Each of the fields of **pf.prx** is named corresponding to a data file that has been read into Pathfinder. For example two data files, **bio.prx.txt** and **psy.prx.txt**, were read to create the following contents of **pf.prx**:

```
pf.prx
    biograds: [1x1 struct]
    psy101:   [1x1 struct]
```

The fields of **pf.prx.psy** are shown below. If you wanted the data matrix, you can refer to it with **pf.prx.psy.dis**). The missing elements or elements outside the **minprx** to **maxprx** range are treated as infinite distance which are never linked. Comments on selected fields are shown following the % in the line:

```
pf.prx.psy
    contents: 'prx'
             file: 'C:\pfdir\pfddata\psy101.prx.txt'
             name: 'psy101'
    distances: 1           % true for dissimilarity or distance data else false
    nnodes: 25            % Number of nodes
    nodes: {25x1 cell}    % Names for nodes if available else node numbers
    decimals: 1           % not used
    minprx: 0             % Minimum allowable data value
    maxprx: 9.9000        % Maximum allowable data value
    shape: 'upper'        % the form of the original data file
    loops: 0              % false if all diagonal values are equal
    directed: 0           % false if data are symmetric else true
    dis: [25x25 double]  % matrix of data in distance direction
    coh: 0.7129           % coherence of the data set
    stddev: 1.9280        % standard deviation of the distances (scale)
```

The structure **pf.net** contains fields containing Pathfinder networks and information about the networks. The names of the **pf.net** fields refer to the data source and the method used to create the net. For Pathfinder nets, this means the **q** and **r** values used to create the **pfnets**. Default values of **q** (**n-1**) and **r** (infinity) were used when the data names are not modified. Other values of **q** and **r** are shown in the name. For other methods of deriving networks, the field name includes a code for the method and parameters. For example **bio\_nnc1** is a nearest neighbor + connections network with 1 nearest neighbor. Here are some example contents of **pf.net** with **pfnets** derived from the data shown above:

## Pathfinder 9.0

```
pf.net
  biograds: [1x1 struct] % pfnet with default q = n-1 and r = infinity
  psy101: [1x1 struct] % pfnet with default q = n-1 and r = infinity
  biograds_q2: [1x1 struct] % pfnet with q = 2 and r = infinity
  psy101_q2: [1x1 struct] % pfnet with q = 2 and r = infinity
  biograds_nn: [1x1 struct] % nearest neighbor net
  psy101_nn: [1x1 struct] % nearest neighbor net
```

The fields of **pf.net.psy101** are shown below. Many elements of the data structure are copied into the net structure so that information is available here too. You could retrieve the links with **pf.net.psy101.links** Comments on selected lines are show after the %.

### **pf.net.psy101**

```
  contents: 'net'
    name: 'psy101'
  distances: 1
    nnodes: 25
    nodes: {25x1 cell}
    minprx: 0
    maxprx: 8
    loops: 0 % true if loops are present else false
  directed: 0 % true for a directed links else false
    dis: [25x25 double] % distance matrix underlying the network
    q: 24 % the value of q used to derive the pfnet
    r: Inf % the value of r used to derive the pfnet
    type: 'pfn' % the type of net, here a pathfinder net
    adj: [25x25 boolean] % a matrix with true for links
    linkwts: [1x25 double] % a vector with distances for the links
    linktypes: [25x25 double] % a matrix with a type number for links
  linktypenames: {'pfn'} % a cell array of link type names
  ncomponents: 1 % the number of connected components in the net
  components: {[1x25 double]} % cell array with the nodes in each component
    nlinks: 25 % the number of links in the net
    nloops: 0 % the number of loops in the net
    links: [25x3 double] % links and link weights in a list
  connected: 1 % true if the network is connected
    view: [1x1 struct] % data for the network layout
  pfname: 'psy101' % the name of the net
```

## ***Sample Data Sets***

To get the feel of using the software, try running an analysis of the sample data files, psy.prx.txt and bio.prx.txt along with a terms file, terms.txt are included for you to experiment with.

Further information on Pathfinder can be found in: *Pathfinder Associative Networks: Studies in Knowledge Organization* Edited by: Roger W. Schvaneveldt Publication Date: 1990 Published by: Ablex Publishing Corp. 355 Chestnut Street Norwood, NJ 07648

## Example Table

Here is an example of a table showing a link list in the bio network.

26 Links in Undirected Network bio						
Link	node1	node2	Node1	Node2	Distance	Type
1	1	8	living thing	mammal	27	pfn
2	1	11	living thing	deer	27	pfn
3	1	18	living thing	tree	28	pfn
4	2	7	animal	chicken	26	pfn
5	2	10	animal	dog	26	pfn
6	2	15	animal	frog	31	pfn
7	3	25	blood	red	30	pfn
8	4	5	bird	feathers	18	pfn
9	4	7	bird	chicken	23	pfn
10	5	6	feathers	robin	17	pfn
11	8	9	mammal	hairs	30	pfn
12	8	10	mammal	dog	27	pfn
13	8	11	mammal	deer	23	pfn
14	8	12	mammal	bats	25	pfn
15	11	13	deer	antler	19	pfn
16	11	14	deer	hooves	23	pfn
17	16	17	plant	leaves	27	pfn
18	16	20	plant	flower	22	pfn
19	16	22	plant	daisy	22	pfn
20	16	24	plant	green	27	pfn
21	17	18	leaves	tree	24	pfn
22	18	19	tree	cottonwood	17	pfn
23	20	21	flower	rose	18	pfn
24	20	22	flower	daisy	21	pfn
25	23	24	color	green	26	pfn
26	23	25	color	red	24	pfn

Save to .csv File