

## 2 Attributes and relations

### 2.1 Introduction

In Chapter 1, we argued that social network analysis focuses on social relations. A network is a set of vertices and lines. Both vertices and lines have characteristics which we may want to include in our analysis, e.g., the gender of people and the strength of their relations. As we noted in the previous chapter, properties of relations are represented by line values in the network, e.g., first and second choices among girls in the dormitory. Now, we will add characteristics of the vertices to the analysis. How can we use information on the actors to make sense of the social network?

In this chapter, we present techniques which combine relational network data and non-relational attributes, such as psychological, social, economical, and geographical characteristics of the vertices in the network. The attributes enhance our interpretation of network structure and they enable us to study subsections of the network. In addition, we will briefly discuss how to use the network position of vertices in statistical analysis; social network analysis and statistics are two complementary sets of techniques. After having studied this chapter, you will understand the basic data used in network analysis and you will be able to combine relational and non-relational data.

### 2.2 Example: the world system

Social network analysis can be applied to large scale phenomena, as we will see now. In 1974, Immanuel Wallerstein introduced the concept of a capitalist world system, which came into existence in the sixteenth century. This system is characterized by a world-economy which is stratified into a core, a semiperiphery, and a periphery. Countries owe their wealth or poverty to their position in the world-economy. The core, Wallerstein argues, exists because it succeeds in exploiting the periphery and, to a lesser extent, the semiperiphery. The semiperiphery profits from being an intermediary between core and periphery.

The world system is based on a global division of labor. Countries in the core specialize in capital intensive and high tech production, whereas peripheral countries apply themselves to low-valued, labor intensive products or unprocessed, raw materials. Core countries import raw and less-processed products from the periphery and turn them into expensive high tech products which are exported to countries in the core, semiperiphery, and periphery. In consequence, there is much trade among core countries, but little trade between countries in the periphery. The core dominates the world trade in a double sense: core countries are more often involved in trade relations than peripheral countries

and the value of exports from core countries exceeds the value of imports because their products have higher added value. This is why core countries do very well economically.

Which countries belong to the core, semiperiphery, or periphery? In political economy, several attempts have been made to answer this question, some of which are based on social network analysis. The network analysts analyzed the structure of the trade relations and classified countries according to the pattern of their trade relations, for instance, the nations which trade with almost all other countries are classified as core countries in the world system. World trade statistics, which are widely available, offer the data required for this analysis.

In this chapter, we use statistics on world trade in 1994. We included all countries with entries in the paper version of the *Commodity Trade Statistics* published by the United Nations, but we had to add data on some countries for 1993 (Austria, Seychelles, Bangladesh, Croatia, and Barbados) or 1995 (South Africa and Ecuador) because they were not available for 1994. Countries which are not sovereign are excluded because additional economic data are not available: Faeroe Islands and Greenland, which belong to Denmark, and Macau (Portugal). In the end, the network contains 80 countries and most missing countries are located in central Africa and the Middle East, or belong to the former USSR.

The arcs in our network represent imports by one country from another. We restrict ourselves to one class of commodities rather than total imports and we picked miscellaneous manufactures of metal, which represents high technology products or heavy manufacture. We use the absolute value of imports (in 1,000 US\$) but we did not register imports with values less than 1% of the country's total imports on this commodity. The network data are stored in the file `Imports_manufactures.net`.

In addition, we use several attributes of the countries in our analysis, viz., their continent (`Continent.clu`), their world system position in 1980 (`World_system_1980.clu`), and their Gross Domestic Product per capita in US\$ in 1995 (`GDP_1995.vec`). The network and attributes are also available as a Pajek project file (see Section 2.3).

Note that we will not determine the world system position of the countries in this chapter. We will use their position in 1980 according to the results of previous research by Smith and White (see the reference in Section 2.10), who subdivided the semiperiphery into two parts: a strong and a weak semiperiphery. They applied an advanced structural technique called blockmodeling, which we will present in Chapter 12.

### 2.3 Partitions

A partition of a network is a classification or clustering of the network's vertices. Each vertex is assigned to exactly one class or cluster (we will use these words as synonyms), for example, one country is assigned to the core and another to the

semiperiphery. A partition may contain a special class which collects the vertices which we cannot classify because data are missing: the ‘missing values’. Usually, the classes are identified by integers, for instance, a core country receives code 1 in the partition, and a country in the semiperiphery gets a 2. In this format, a partition is simply a list of integers, one for each vertex in the network.

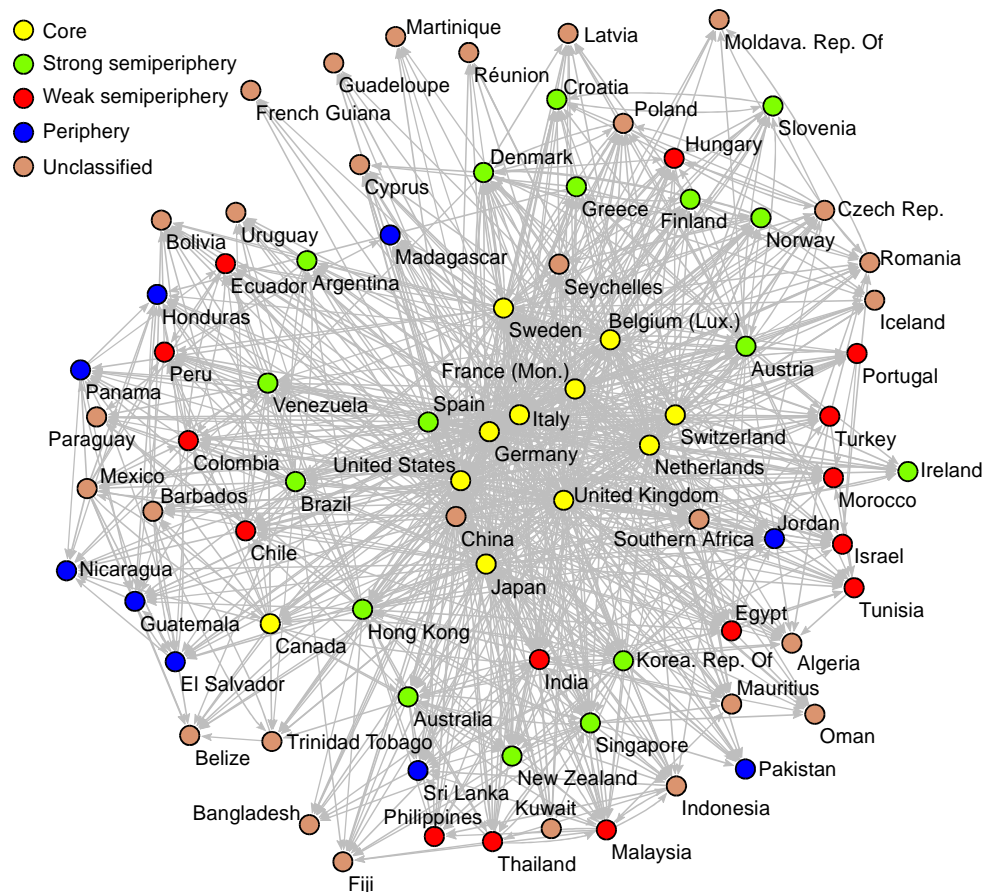
A **partition** of a network is a classification or clustering of the vertices in the network such that each vertex is assigned to exactly one class or cluster.

In network analysis, partitions store discrete characteristics of vertices. A property is **discrete** if it consists of a limited number of classes, for instance, we may code the continents of countries by digits such that class 1 contains the African countries, Asian countries constitute class 2, etc.; six classes suffice. A classification contains a limited number of classes and most classes contain several vertices because we want the classes to represent groups of actors rather than single actors or nothing at all. Partitions, therefore, are very useful for making selections from a network in order to reduce its size and complexity. We will discuss this in the next section.

In some cases, the order of class numbers in a partition is arbitrarily, e.g., in the partition of nations according to continents: there is no compelling reason why African countries should have a lower class code than Asian countries. In other instances, however, the order is meaningful, for instance, it would be foolish to assign the semiperiphery to class 1, the core to class 2, and the periphery to class 3, because this would not correctly reflect the obvious ranking of the three classes. Finally, the class codes may represent ‘real’ numbers, e.g., the number of lines incident with a vertex: all vertices in class 1 are incident with one line, vertices in class 2 are incident with two lines, etc., so make sure that you attach the right meaning to class numbers!

Partitions may specify a **structural property** such as world system position, which is a result of network analysis, or a characteristic measured independently of the network, e.g., the continent where a nation is located. We call the latter **attributes** of vertices.

Figure 1 displays the classification of countries according to their world system position in 1980 and their trade relations in 1994. The network is energized and it is quite apparent to the eye that core countries (yellow) are located in the center of the network, although Canada is a bit eccentric. The core is surrounded by countries in the strong semiperiphery (green) and peripheral countries (dark blue) are indeed located in the margins of the network. In broad lines, world system positions of 1980 still appear to be valid in 1994. Unclassified countries (brown) were excluded by Smith & White because they were judged to be too small or data were missing. Most of them seem to belong to the periphery or weak semiperiphery, with the obvious exception of China and perhaps South Africa, which would probably be more central if data had been available on all countries in Africa.



**Figure 1** - World trade of manufactures and world system position.

### Application

*File>Partition>Read*

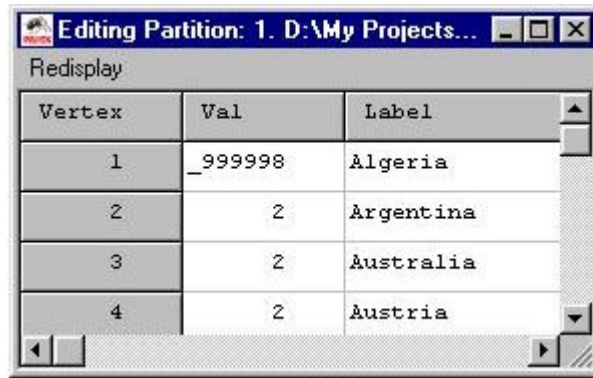
*File>Partition>Save*

*Partition drop list*

In Pajek, partitions are a data object on their own, so they are accessible from a drop list once they are read (*File>Partition>Read*). Partitions are saved in files with extension .clu (*File>Partition>Save*) and these files are just lists of integers preceded by a line which specifies the number of vertices. The first integer represents a property of the first vertex, e.g., world system position, the second integer belongs to the vertex with serial number two, et cetera. Note that unknown world system position is coded by the missing value 999998, e.g., Algeria in Figure 2. You should not change neither the sequence of vertices in a network nor the sequence of entries in a partition because this will destroy the compatibility of the partition and the network: vertices are no longer associated with the right clusters.

*File>Pajek Project File*

Although partitions can be stored separately, we can also save them with the network to which they belong. In Pajek, a project file may contain all networks, partitions, and other data objects which belong together. You may, for example, read the world trade network and associated partitions and vectors from the project file *World\_trade.paj* with the command *File>Pajek Project File>Read*. With the *Save* command from the same submenu, you can create your own project file, which contains all data (networks, partitions, and so on) and all results printed in the Report screen at the time you save the project.



**Figure 2** - Edit screen with partition according to world system position.

*File>Partition>Edit*

It is easy to see that a partition consists of a series of integers if you edit it. Choose the command *Edit* in the *File>Partition* submenu or simply click on the edit button at the left of the Partition drop list (with the writing hand) to open the Edit screen (see Figure 2). The Edit screen contains three columns, which display the vertex number (*Vertex*), the class code (*Val*), and the vertex label (*Label*). The first vertex in the world trade network represents Algeria, which has class number 999998: it was unclassified in the world system circa 1980. Click on the class code to change it. Note that the labels of the vertices are displayed only if the associated network is selected in the Network drop list.

**Table 1** - Tabular output of the command *Info>Partition*.

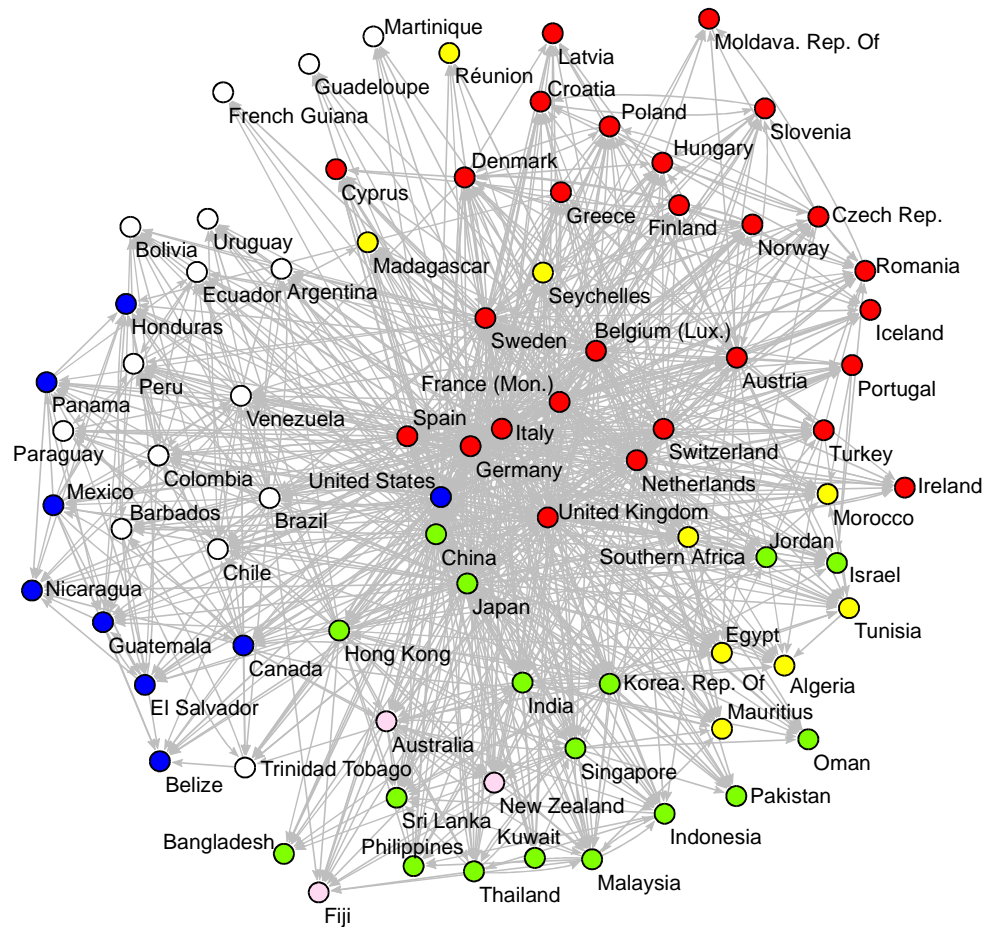
| Class   | Freq | Freq%    | Valid%   | CumFreq | CumFreq% | CumValid% | Representative |
|---------|------|----------|----------|---------|----------|-----------|----------------|
| 1       | 11   | 13.7500  | 21.1538  | 11      | 13.7500  | 21.1538   | Belgium (Lux.) |
| 2       | 17   | 21.2500  | 32.6923  | 28      | 35.0000  | 53.8462   | Argentina      |
| 3       | 15   | 18.7500  | 28.8462  | 43      | 53.7500  | 82.6923   | Chile          |
| 4       | 9    | 11.2500  | 17.3077  | 52      | 65.0000  | 100.0000  | El Salvador    |
| Sum     | 52   | 65.0000  | 100.0000 |         |          |           |                |
| Unknown | 28   | 35.0000  |          |         |          |           |                |
| Total   | 80   | 100.0000 |          |         |          |           |                |

*Info>Partition*

On execution, the command *Info>Partition* displays two dialog boxes. In the first box, which is similar to the dialog box associated with the *Info>Network>General* command (see Section 1.3.3), the user may request a listing of vertices with highest or lowest class numbers. Type a positive integer to list vertices with highest class numbers and type a negative integer to list vertices with lowest class numbers. The second dialog box allows for suppressing classes in the table which occur seldom, for instance, type 5 in this dialog box to exclude classes with four vertices or less from the frequency tabulation.

The *Info>Partition* command presents a table which lists the number of vertices in each class of the partition. In Table 1, we can see that 28 countries are not classified. Because they have special class code 999998, they are reported at the bottom of the table in a line with header *Unknown*. They are automatically excluded from calculations. The core (class 1) contains 11 countries (second column), which is 13.75 percent (third column) of all countries in the network but 21.2 percent of the countries with known world system position (fourth column). Belgium is one of the core countries. Together, the core and strong semiperiphery contain 28 out of 52 classified countries (column CumFreq), which is 53.8

percent (CumValid%). The number of vertices in a class is called a *frequency* (abbreviated to *Freq* in the table), hence the table is called a frequency distribution.



**Figure 3** - World trade and continents.

*Draw>Draw-Partition*

In network drawings, partitions are represented by the colors of the vertices. Each class is associated with a color, e.g., vertices in class number one are yellow. It is easy to obtain a colored sociogram. First, make sure the right network and partition are selected in the drop lists of the Main screen. To make Figure 3, we need the manufactures network (*Imports\_manufactures.net*) and a partition according to continent (*Continent.clu*). Next, execute the command *Draw-Partition* from the *Draw* menu (or press Ctrl-P). If the selected network and partition are compatible, that is, if the number of entries in the partition is equal to the number of vertices in the network, Pajek draws the network with vertex color determined by the partition. In Figure 3, we see that geography is important in the world system, since countries are clustered by continent. African countries (yellow) are not grouped, perhaps because countries in central Africa are missing.

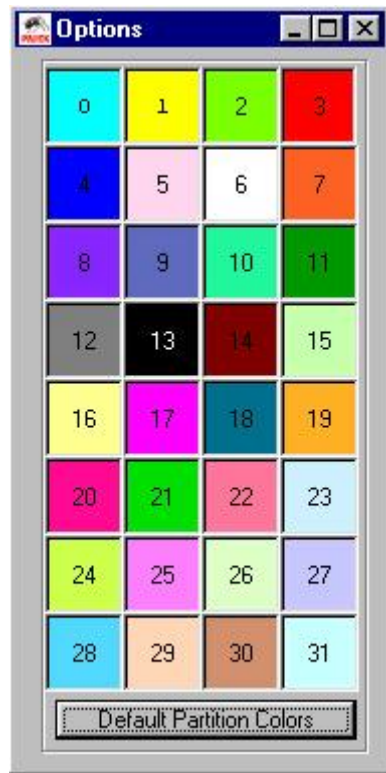
*Partition*  
*>Create Null Partition*

*Draw>Draw-SelectAll*

In Pajek, you can create a new partition which you can edit manually. In the *Partition* menu, the command *Create Null Partition* makes a new partition for the selected network. All vertices are placed in class zero. With the edit command, which we discussed previously (*File>Partition>Edit*), you can assign vertices to other classes: just change their class numbers in the list. You can obtain the same result with the command *Draw-SelectAll* from the *Draw* menu. This command



creates a new partition and displays it in the Draw screen. Here, you can raise the class number of a vertex by one in the following way: click on the vertex with the left mouse button while you press the Shift key on the keyboard. If the cursor is not on a vertex, all class numbers are raised. Clicking a vertex while the Alt-key is pressed subtracts one from the class number. In this way, you can easily create a partition which clusters a small number of vertices.



**Figure 4** - Screen with colors of partitions in Pajek.

*Options>Colors*  
*>Partition Colors*

By default, Pajek uses yellow to color vertices in class one, light green for vertices in class two, red for class three, et cetera, but it is easy to change colors of classes. Select the command *Partition Colors* from the *Options>Colors* submenu of the Draw screen and Pajek will display the 32 colors which it uses (Figure 4). In each colored quadrangle, an integer indicates the class currently associated with the color. If you want to change the color associated with a particular class, click on the quadrangle with the desired color and type the number of the class you want this color to be associated with in the dialog box that appears and Pajek will swap the colors. Press the button labeled 'Default Partition Colors' if you want to restore original colors of classes.

*Layout>Energy*  
*>Kamada-Kawai*  
*>Fix selected vertices*

In a sociogram with colored classes, it is very easy to move all vertices which belong to one class. Position the cursor *near* but *not on* a vertex, press the left mouse button and drag: all vertices in the class will move simultaneously. This is a very useful technique. In addition, the Kamada-Kawai energy procedure has a special option for energizing networks with class colors, e.g., you can fix the location of the vertices in class 1 (yellow) and optimize the position of all other vertices with the *Fix selected vertices* command.

menu *Partition*

menu *Partitions*

There are several ways to manipulate partitions, e.g., make a new partition or combine two partitions. We will encounter most of these techniques in later chapters. Suffice it to say here that these commands are located in the *Partition* and *Partitions* menu of the Main screen.

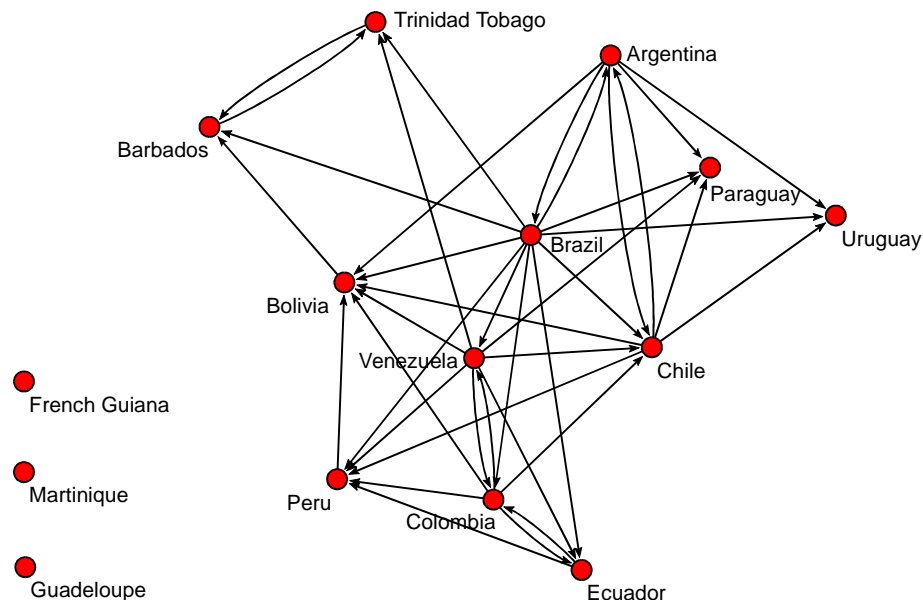
## 2.4 Reduction of a network

Partitions divide the vertices of a network into a number of mutually exclusive subsets. In other words, a partition splits a network into parts. Therefore, we can use partitions to reduce a network in three ways: extract one part (local view), shrink each class of vertices into one new vertex (global view), or select one part and shrink neighboring classes to focus on the internal structure and overall position of this class (contextual view). We will discuss the three types of reduction now.

### 2.4.1 Local view

The easiest way to reduce a network is to select one class of vertices. Of course, a mere selection of vertices is not very interesting; we must also select all lines between them - and their loops eventually - if we want to inspect the structure of relations in this part of the network. This operation is called subnetwork extraction and its result is called an **induced subnetwork** which offers a **local view**.

To **extract** a subnetwork from a network, select a subset of its vertices and all lines which are only incident with the selected vertices.



**Figure 5** - Trade relations within South America.

If we want to examine the trade relations within South America, we can extract the subnetwork of South American countries, which is depicted in Figure 5. From the energized drawing, it is clear that Brazil occupies a central position, whereas



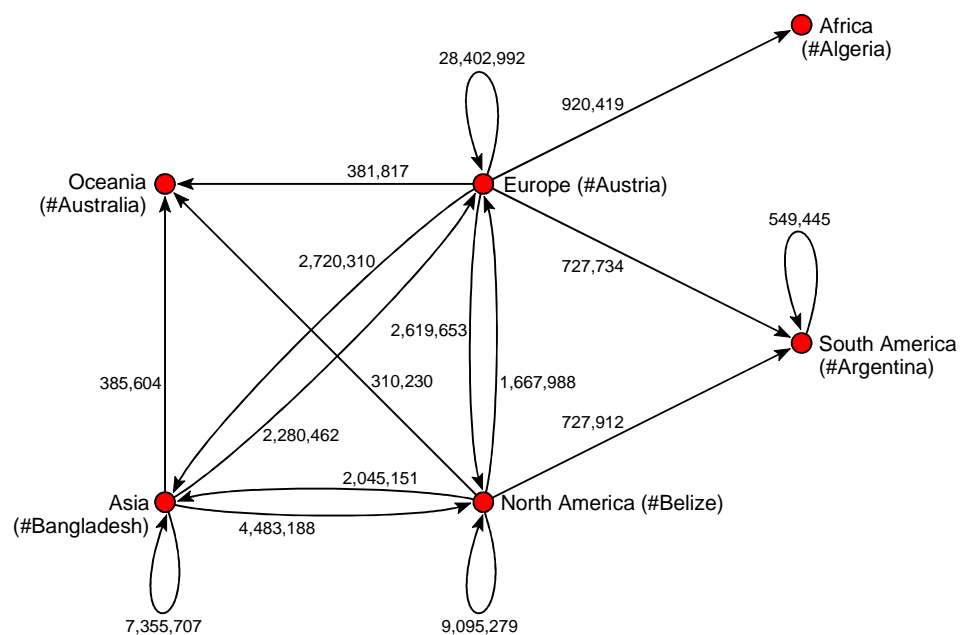
French Guiana, Martinique, and Guadeloupe are isolated. Apparently, these islands trade with countries in other parts of the world. Finally, geography appears to be important within the continent, since neighboring countries are close in the network.

#### 2.4.2 Global view

Instead of zooming in on a particular region of the network, we may also zoom out to obtain a global view. Now, we are no longer interested in each individual vertex but we want to study relations between classes, for instance, continents. Which continents have strong trade relations, which have not? In this example, a global view of the network covers the whole world but we should note that a global view may also pertain to relations between groups in a local setting.

To **shrink** a network, replace a subset of its vertices by one new vertex which is incident to all lines that were incident with the vertices of the subset in the original network.

In network analysis, we obtain a **global view** by shrinking all vertices of a class to one new vertex. In our example, we shrink all countries within a continent to a new vertex which represents the entire continent (see Figure 6). Lines incident with shrunk vertices are replaced, for instance, all imports by South American countries from European countries are replaced by one new arc pointing from Europe to South America. Its line value is equal to the sum of all original line values. In the network of trade relations, line values indicate the value of imports in (thousand) US\$, so values of lines in the shrunk network represent the value of total imports. Lines between shrunk vertices within one class, e.g., trade relations within South America, are replaced by a loop.

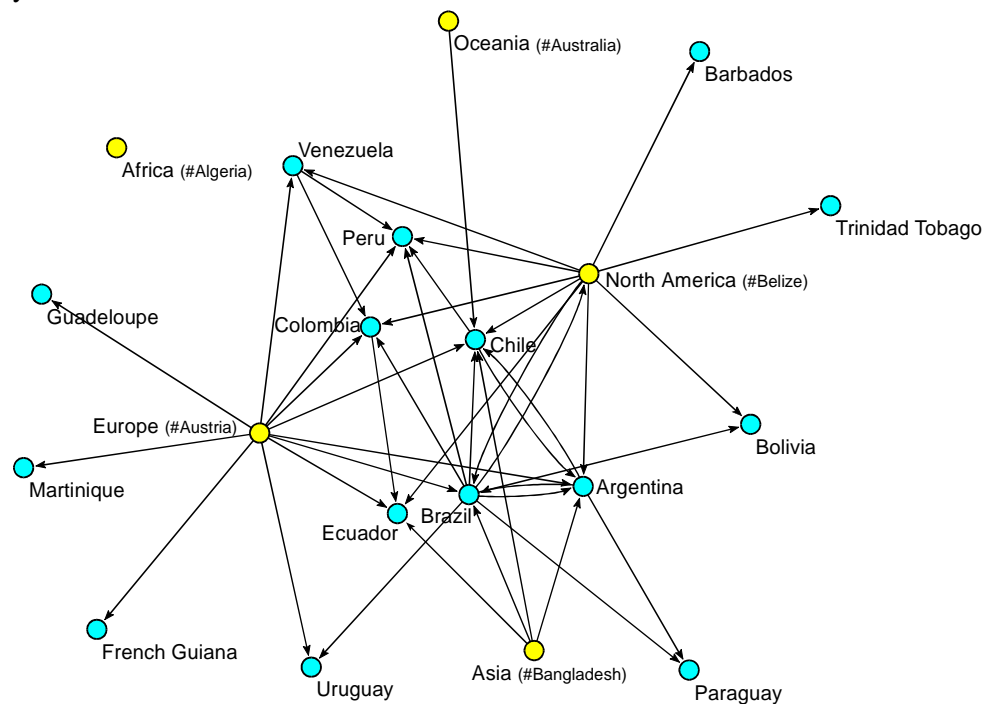


**Figure 6** - Trade relations among continents (imports in thousand US\$).

Figure 6 shows the world trade network which is shrunk according to continents. To obtain a clear picture, we deleted arcs (total imports) with values below 300 million US\$. We can see that South American countries import manufactures of metal with a total value of 727 million US\$ from Europe but they do not export a substantial amount to Europe, so there is an asymmetrical trade relation between South America and Europe. The countries in Oceania and Africa are also importing rather than exporting. Trade within South America (549 million US\$) is much higher than in Oceania and Africa but internal trade is at its highest in Europe.

#### 2.4.3 Contextual view

A global view shows the position of South America in the world system and a local view clarifies the central position of Brazil within the South American trade network. But how do countries in this regional network relate to the rest of the world? From which continents do the isolated islands import their manufactures of metal? If you want to focus on one class of vertices, e.g., countries on one continent, and take into consideration aggregated relations to the ‘outside world’, you need a contextual view.



**Figure 7** - Contextual view of trade in South America.

In a **contextual view**, all classes are shrunk except the one in which you are particularly interested. In Figure 7, all countries are shrunk to continents except the South American countries. We removed relations between continents because we already know them from Figure 6 and we discarded trade relations with values under 10 million US\$ to obtain an intelligible drawing. This sociogram shows the South American countries in the context of world trade. Clearly, Africa and Oceania are not important trading partners to South American countries. The larger countries on the continent import from Europe and North America, but

smaller countries are either connected to North America (Barbados, Trinidad and Tobago, Bolivia) or to Europe (Uruguay, Guadeloupe, Martinique, and French Guiana). Former colonial relations surely play a part here.

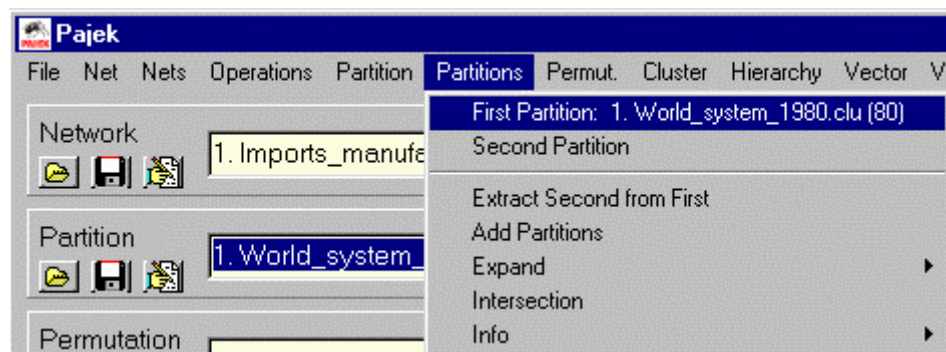
There are many ways to extract or shrink a network and extraction and shrinking can even be combined. Subnetwork extraction and shrinking are important techniques to dissect a network and to obtain partial views of a network when its structure is too complicated to understand at a first glance.

### Application

For both extraction and shrinking, a network is needed and a partition which assigns the vertices of the network to classes. Therefore, commands to perform extraction and shrinking are located in the *Operations* menu. In this section, we use the manufactures trade network (`Imports_manufactures.net`) and the continents partition (`Continent.clu`).

*Operations*  
*>Extract from Network*  
*>Partition*

To obtain a local view of the South American trade network, use the command *Partition* from the *Operations>Extract from Network* submenu. Because South America is the sixth class in the partition, select class 6 in the dialog boxes which appear. Pajek produces a new network, which contains 15 countries. Energize it to obtain a drawing similar to Figure 5.



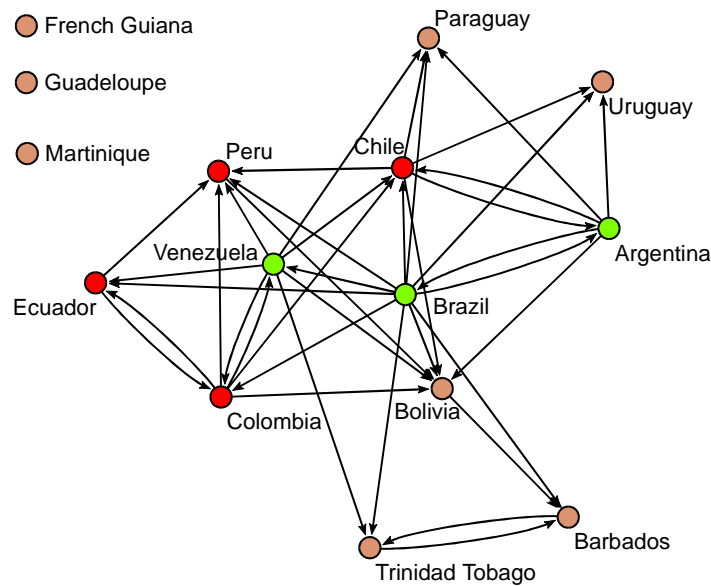
**Figure 8** - The *Partitions* menu.

It is very important to note that the original partitions do not fit the induced subnetwork. This is a pity, because we may want to know, for instance, the world system position of the countries in the South American trade network. However, we can extract a new partition which matches the induced subnetwork in a few steps.

*Partitions>First Partition,,*  
*Second Partition,*  
*Extract Second from First*

First, we must select the original partition which we want to translate to the induced subnetwork as the first partition in the *Partitions* menu. Make sure that the world system partition is selected in the *Partition* drop list. Then, select option *First* in the *Partitions* menu. Nothing seems to happen but open the *Partitions* menu again and you will see the name of the world system partition in it (Figure 8). Second, select the partition which you used to extract the network - in our case the continents partition - as the second partition in the *Partitions* menu. Finally, execute the command *Extract Second from First* from this menu. Dialog boxes will ask for the class numbers to be extracted and if you type 6 twice, Pajek

produces a new partition for the 15 South American countries. Figure 9 shows the South American trade network drawn with this partition.



**Figure 9** - World system positions in South America.

*Operations*  
*>Shrink Network*  
*>Partition*

Shrinking a network to obtain a global or contextual view is just as simple. The command *Partition* in the *Operations>Shrink Network* submenu shrinks classes of vertices in a network according to the active partition. A dialog box asks for the minimum number of connections between clusters. This is the minimum number of lines which must exist between shrunk vertices in order to obtain a new line in the shrunk network. We recommend to choose 1, which is the default value. In a second dialog box, you may choose a class of vertices which must not be shrunk. Shrink all classes to obtain a global view, so type any class number that is not present in your partition, e.g., zero (default). Finally, Pajek shrinks all classes except the selected class and adds the shrunk network to the Network drop list.

Pajek does not know the meaning of classes, so it cannot assign meaningful new labels to shrunk classes. It chooses the label of the first vertex of a class which is shrunk and adds a #-sign to obtain a label for the shrunk class. For example, Argentina happens to be the first South American country in the network, so the vertex which represents this continent carries the label #Argentina in the shrunk network. We added the names of continents to Figure 6 and Figure 7 as well as the partition of countries versus continents manually.

In a shrunk network, a class of vertices is replaced by one new vertex, e.g., the Latin American countries are substituted by a new vertex representing this continent. Properties of the original vertices, such as their world system position, are lost: it is impossible to assign the entire continent to a particular world system position. Therefore, it is impossible to use the data from the original partition in a shrunk network.

*Options*  
*>Blockmodel - Shrink*

Note that the way in which a network is shrunk depends on the option selected in the *Options>Blockmodel - Shrink* submenu. By default, option 0 is

selected, which means that the number of links in the original network is used to decide whether a new line is added to the shrunk network. This option causes Pajek to display the dialog box mentioned above. We advise not to select another option from this submenu until we discuss blockmodels in Chapter 12.

A contextual view is obtained by partially shrinking the network, so we can use the *Shrink Network* command again. In the dialog box asking for the class which should not be shrunk, type the appropriate class number, e.g., class 6 for South America.

*Net>Transform>Remove*  
*>lines with value*  
*>lower than*

In Figure 7, we removed lines with values below 10 million US\$ and lines between continents to obtain a clear picture. Lines with low values can be removed automatically with the command *Remove>lines with value>lower than* in the *Net>Transform* submenu. We entered 10000 as the threshold for this operation, because import values are measured in thousand US\$.

*Operations>Transform*  
*>Remove Lines*  
*>Inside Clusters*

Lines between continents can be deleted automatically provided that you have a partition which identifies the continents in the shrunk network. You may create this partition manually with the commands described in the *Application* section of Section 2.3: create a null partition and assign the five shrunk continents to class one. Now, you can execute the command *Operations>Transform>Remove Lines>Inside Clusters* to remove the lines between vertices in the continents class (class one).

## 2.5 Vectors and coordinates

Partitions store discrete properties of vertices and vectors store continuous properties. In principle, a continuous property may take any value within a defined range, for instance, the surface of a country may take any value between zero square kilometers (or miles) and the total surface of the earth. When two countries have different sizes, it is always possible to imagine a country which is smaller than one and bigger than the other.

In practice, of course, we do not care about differences in sizes of countries smaller than a square kilometer, but the principle is important: continuous properties are not meant to group vertices into classes, so they cannot be used to reduce a network by extraction or shrinking. Continuous properties express a particular and often unique value of a vertex, for instance, the wealth of a country indicated by its Gross Domestic Product per capita. In practice, no two countries have exactly the same GDP per capita because it is equal to the quotient of the economic production of a country and the size of its population, both of which are usually figures that do not characterize any other country. Most quotients yield results with decimals, so a vector is not a list of integers but a list of real numbers.

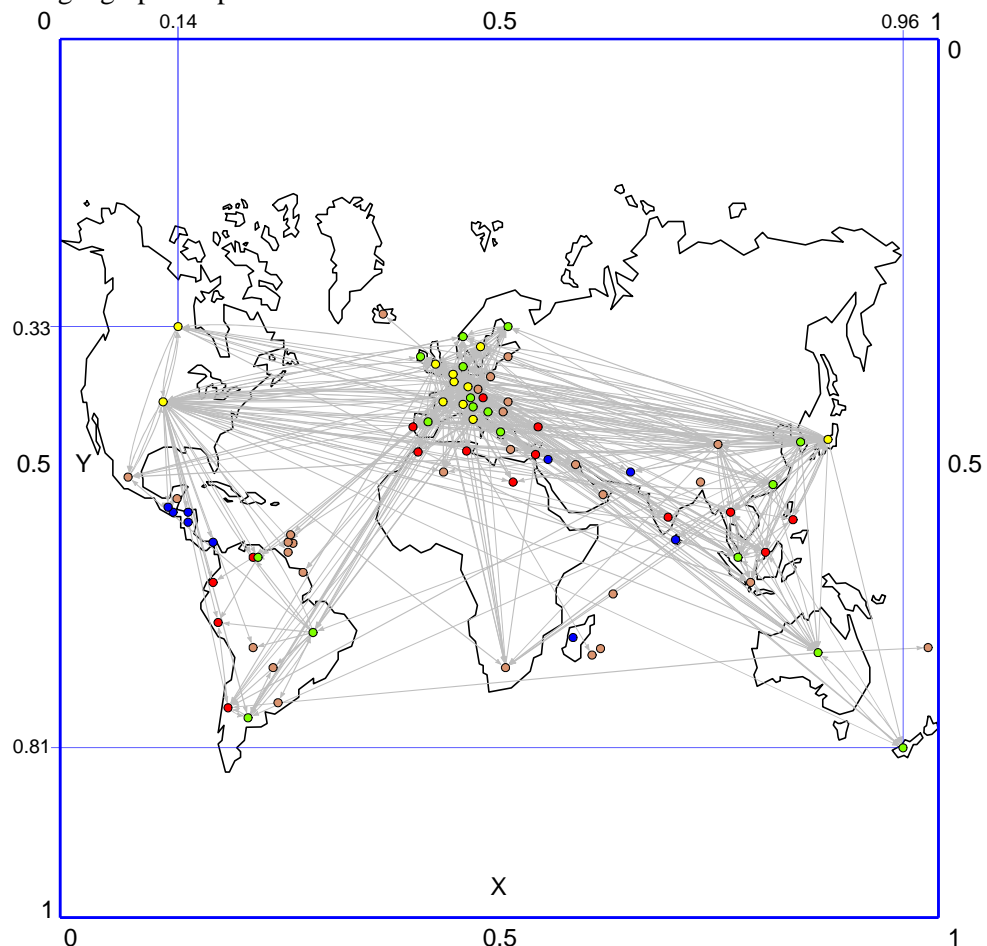
A **vector** assigns a numerical value to each vertex in a network.

In our discussion of partitions, we have distinguished between structural indices and attributes. This distinction applies to continuous properties or vectors as well. In later chapters, we will encounter several examples of vectors which represent

structural features of vertices, for instance, their centrality (Chapter 6). GDP per capita of a country is an example of a continuous attribute which is measured independently of the network and added to the graph as additional data: it is a continuous attribute.

A special continuous property of a vertex is its location. Location is expressed by coordinates, real numbers which correspond with positions on one or more axes. For example, the two-dimensional Draw screen has two axes, a horizontal X-axis and a vertical Y-axis. Both axes range from zero to one (see Figure 10). The position of a vertex in the Draw screen is determined by two coordinates, e.g., Canada is located at 0.14 on the X-axis and 0.33 on the Y-axis.

Since coordinates are real numbers, they can be stored as vectors: one representing the location of vertices on the X-axis, the other locations on the Y-axis. In Figure 10, we did something special: we made sure that locations in the Draw screen matched geographical locations on a map of the world projected behind the network. Now, vectors express geographical coordinates and we can see geographical patterns in a social network.



**Figure 10** - Geographical view of world trade.

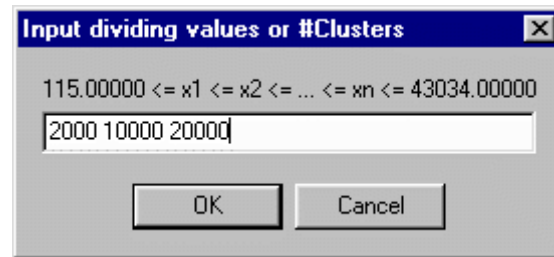
### Application

*File>Vector*  
*Info>Vector*

In Pajek, vectors have a drop list of their own. A vector can be opened, edited, and saved like a network or a partition. Basic information can be gathered from



the output of the *Info>Vector* command. When you execute this command, you can ask for a list of highest or lowest vector values. In a second dialog box, you can specify the number or boundaries of the classes of values which will be reported in the frequency distribution (Figure 11). Note that vector values are supposed to be continuous, so each value probably occurs only once. As a consequence, it is not informative to list each separate value as an entry in the frequencies table; values must be joined into classes. You may either list selected values that must be used as class boundaries (as in Figure 11) or specify the number of classes preceded by the # sign.



**Figure 11** - *Info>Vector* dialog box.

In the Report screen, the *Info>Vector* command outputs some statistics on the vector values, for instance, it tells us that GDP per capita ranges from 115 to 43,034 US\$, with an average of 10,249.9 US\$ and standard deviation 10,834.7. Then it prints the requested frequencies table, which is shown in Table 2. The first entry of this table contains the 22 countries with GDP per capita ranging from the lowest value up to and including 2,000 US\$. The second entry contains the 27 countries with GDP per capita over 2000 US\$ up to and including 10,000 (thousand) US\$, etc. In the first column, the round bracket means ‘from’ and the square bracket means ‘up to and including’.

**Table 2** - Distribution of GNP per capita in classes.

| Vector Values      | Freq | Freq%    | CumFreq | CumFreq% |
|--------------------|------|----------|---------|----------|
| ( ... 2000]        | 22   | 27.5000  | 22      | 27.5000  |
| ( 2000 ... 10000]  | 27   | 33.7500  | 49      | 61.2500  |
| ( 10000 ... 20000] | 15   | 18.7500  | 64      | 80.0000  |
| ( 20000 ... 43043] | 16   | 20.0000  | 80      | 100.0000 |
| Total              | 80   | 100.0000 |         |          |

#### *Partition>Make Vector*

As we discussed before, partitions and vectors have different applications: partitions serve to select subsets of vertices from a network whereas vectors specify numerical properties which can be used in calculations. Sometimes, however, you may want to change partitions into vectors or vice versa. It is very easy to convert a partition to a vector: just use the *Make Vector* command in the *Partition* menu. Note, however, that conversion is only meaningful if classes in a partition express a quantity - if they are numbers that can meaningfully be added, subtracted, et cetera.

#### *Vector>Make Partition* *>by Truncating (Abs)*

The translation of a vector into a partition is more complicated because you have to change real numbers into integers, which can be done in several ways in the *Vector>Make Partition* submenu. Truncation is the easiest way, which means

that you drop the decimals from the real numbers in the vector to obtain integers which can be stored in a partition. A GDP per capita of 115.0 (thousand) US\$ up to but not including 116 (thousand) US\$ is changed to class 115 in a new partition. In the command name, *Abs* stands for absolute, which means that negative vector values are transformed into positive class numbers because partitions cannot hold negative integers.

*Vector>Make Partition*  
*>by Intervals*  
*>First Threshold and Step*

Another way to convert a vector into a partition is to recode vector values into classes of fixed width, for instance, recode GDP per capita into classes of 10,000 US\$. Use the command *by Intervals>First Threshold and Step* in the *Vector>Make Partition* submenu and specify the upper bound of the lowest class as the first threshold, e.g., 10000 for a class including minimum value up to and including 10,000, and class width as the step, e.g., fill in 10000 again. Note that Pajek does not accept a comma separating thousands from hundreds.

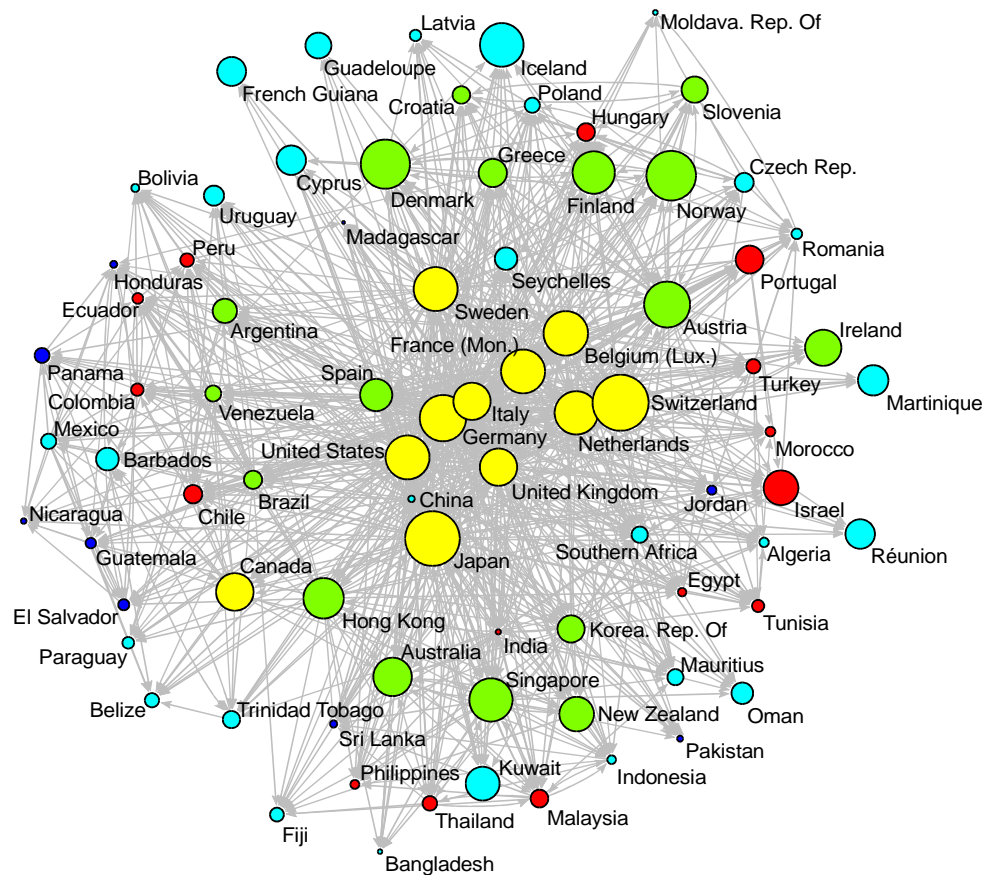
*Vector>Make Partition*  
*>by Intervals*  
*>Selected Thresholds*

In the case of GDP per capita, a conversion of values to classes with fixed width is not very useful. GDP per capita is unevenly distributed, so many countries are lumped together in the lowest classes, whereas higher classes contain few countries. In this example, it is better to create classes of unequal width, namely narrow classes for low values and wider classes for higher values, with the command *Selected Thresholds* in the *Make Partitions by Intervals* submenu. In the dialog box, specify the boundaries between classes, e.g., 2000, 10000, and 20000. Just type the numbers separated by spaces and do not use ‘inf’ or ‘=<’ as specified in the top of the dialog box. The word ‘inf’ (for infinity) merely indicates that the lowest class starts with the lowest value encountered and the highest class continues until the highest value encountered. The sign ‘=<’ means that all classes include the upper boundary.

*Draw>Draw-Vector*  
*Draw>Draw-Partition-Vector*

In a sociogram, vector values are represented by the size of vertices if you use *Draw-Vector* or *Draw-Partition-Vector* from the *Draw* menu. The area of a vertex is proportional to its vector value. In Figure 12, countries with higher GDP per capita have larger vertices. We can see that the wealthiest countries are part of the core and strong semiperiphery, but note that geographical location is also important, e.g., Scandinavian countries have similar (high) GDP per capita, whether they belong to the core (Sweden) or to the strong semiperiphery (Norway, Denmark, Finland). Countries in the weak semiperiphery (red) have smaller GDP per capita, except Israel and Portugal, and peripheral countries (dark blue) are clearly poorest. Probably, Israel and Portugal have experienced a rapid economic development since 1980, so their position in the weak semiperiphery is outdated.

Representation of vector values by size of vertices is not useful for negative numbers, since negative size is meaningless. Pajek ignores negative signs of vector values when it computes the size of a vertex in a drawing. Vertices with large negative value are drawn as big as vertices with large positive value, which is quite misleading. Always check whether negative vector values exist.



**Figure 12** - Trade, position in the world system, and GDP per capita.

[Draw] Options>Size  
>of Vertices

Note that GDP per capita ranges from 115 to 43,034 US\$, but vertices usually have sizes between 2 to 4 in the Draw screen. If you draw a network with vector sizes ranging from 115 to 43034, vertices are so large that they do not fit in the screen, so you have to adjust vector values to get reasonable sizes in a drawing. The *Options>Size>of Vertices* command of the Draw screen offers the simplest way to achieve this. In the dialog box issued by this command, enter 0 to activate the *AutoSize* utility of Pajek, which computes optimal vertex sizes automatically.

Options  
>Mark Vertices Using  
>Vector Values

Vector values may also be displayed as vertex labels in the Draw screen: select the option *Vector Values* in the *Options>Mark Vertices Using* submenu. In most networks, the result is visually not very attractive but it is a good way to check the exact vector values of vertices in a drawing.

Vector  
>Extract Subvector

In Section 2.4, we learned how to select a part of a network, e.g., the trade relations among countries of one continent, and how to reduce a partition to fit the extracted network. In a similar way, we can adjust vectors to induced subnetworks. Recall that we need a partition to extract a network, e.g., a partition of countries according to their continent is needed for extracting countries of one continent from the network. We can use the same partition for extracting the vector values, for instance, GDP per capita, of the selected countries. First, make sure that the partition which was used to reduce the network is selected in the Partition drop list. Next, execute the command *Extract Subvector* in the *Vector* menu. Just like the command for extracting a partition, the *Extract Subvector* command asks the user to choose a class of vertices or range of classes which

must be extracted, e.g., enter 6 to select South American countries. Finally, Pajek adds the reduced vector to the Vector drop list.

As we have noted before, coordinates of vertices can be thought of as vectors. In Chapter 1, we have learned that coordinates of vertices are saved in the network data file, so we do not need vectors to store the location of vertices. However, vectors of coordinates are useful, for instance, if you want to use coordinates from one network in another network with the same vertices, for instance, you have a network of trade in another product and you want to use the geographical coordinates from the manufactures trade network.

*Net>Vector*  
*>Get coordinate*

*Operations>Vector*  
*>Put coordinate*

You can save the present coordinates of the vertices as vectors with the *Get coordinate* command in the *Net>Vector* submenu. You must create a vector of coordinates for each axis separately. The *Put coordinate* command in the *Operations>Vector* submenu loads coordinates from vectors: one vector for each axis. In our example, geographical location of countries can be read from two vectors: `x coordinates.vec` and `y coordinates.vec`. Note that the save and load commands are listed in different menus. For saving coordinates, you only need a network, but for loading coordinates from a vector you need a vector to load from and a network to add coordinates to. Therefore, the load command is situated in the *Operations* menu.

*Options>Transform*  
*>Fit Area*

When you load coordinates from a file which was not created by Pajek, the Draw screen may be blank. This will happen if the coordinates are not in the range between zero and one. In this case, use the *Options>Transform>Fit Area* command in the Draw screen and Pajek will fit the sociogram to the size of the screen.

## 2.6 Network analysis and statistics

In this chapter, we use attributes of vertices in social network analysis, for instance, we compare GDP per capita of countries to their visual positions in the trade network. We find that countries in the core of the world trade system have higher GDP per capita than countries in the periphery. Thus, social network analysis handles relational data as well as attributes of vertices.

Attributes like GDP per capita are usually analyzed with statistical techniques. GDP per capita is compared, for instance, to population growth and level of education to find out which properties of countries are associated. GDP per capita is found to be higher in countries with a lower population growth and a higher level of education. Statistics offers a wide range of techniques to describe attributes and investigate the association between attributes, but it cannot handle relational data directly. If we are able to express structural properties of vertices as attributes or properties of actors, however, they can be included in statistical analysis. Partitions and vectors, which store attributes, are the bridge between network analysis and statistics.

It is quite common to use structural indices in statistical analysis. The position of a country in the world system, for instance, may be calculated from

the world trade network. Actually, the partition of world system positions which we use here is the result of network analysis performed by Smith and White. Subsequently, Smith and White used statistical analysis to test whether the world system positions of countries predict their economic situation as measured by GDP per capita.

In this book, we will give some examples of statistical analysis, but we will restrict ourselves to basic statistical techniques which are incorporated in Pajek. After all, this is not a course in statistics. Nevertheless, the link between statistics and social network analysis should be well understood, because the two techniques make up a powerful combination.

### Application

One of the basic statistical techniques implemented in Pajek is the cross-tabulation of two partitions and some measures of association between the classifications represented by two partitions. Let us have a look at the world trade example to illustrate this. From Figure 12 we concluded that the core countries are much richer than the countries in the semiperiphery and periphery. We can be more specific about this relation between position in the world trade network captured by the world system classification by Smith and White (*World\_system\_1980.clu*) and economic prosperity as measured by GDP per capita (*GDP\_1995.vec*).

*Vector>Make Partition*  
*>by Intervals*  
*>Selected Thresholds*

First, we have to translate the GDP vector into a partition. We suggest to create four classes according to GDP per capita: 0 to 2000, 2000 to 10,000, 10,000 to 20,000, and 20,000 and over, which you have learned to make with the *Make Partition>by Intervals>Selected Thresholds* command from the *Vector* menu in Section 2.5.

*Partitions>First Partition*  
*Second Partition*

Second, we must select the two partitions which we want to compare. In Section 2.3, we learned how to do this. Select the newly created GDP per capita partition in the *Partition* drop list and execute the command *First Partition* from the *Partitions* menu. Next, select the world system positions partition in the drop list and execute the command *Second Partition* in the same menu. When you open the *Partitions* menu now, you will see the names of the selected partitions in the menu. To obtain a cross-tabulation and measures of association, the selected partitions must be compatible. It is meaningless to compare partitions which do not refer to exactly the same vertices.

*Partitions>Info*  
*>Cramer's V, Rajske*

Finally, select the *Info>Cramer's V, Rajske* command from the *Partitions* menu. Pajek will produce a new screen containing a cross-tabulation and some measures of association. If you want to add this output to the general Report screen, select the command *Append to Report* from the *File* menu of the screen containing the cross-tabulation.

*File>Append to Report*

Table 3 contains the results of the *Info* command. In the cross-tabulation, the rows contain the four classes according to GDP per capita. The columns contain the four world system position: core countries in class one, strong semiperiphery in class two, etc. Countries with unknown world system position are automatically

omitted by Pajek. As a result, the first row represents 13 of the 22 countries with GDP per capita up to and including 2,000 US\$: we do not know the world system position of nine countries in this class. Note that rows and columns are swapped when the first and second partition are exchanged in the *Partitions* menu.

**Table 3** - Output of the *Info* command.

|       | 1  | 2  | 3  | 4 | Total |
|-------|----|----|----|---|-------|
| 1     | 0  | 0  | 5  | 8 | 13    |
| 2     | 0  | 5  | 8  | 1 | 14    |
| 3     | 3  | 5  | 2  | 0 | 10    |
| 4     | 8  | 7  | 0  | 0 | 15    |
| Total | 11 | 17 | 15 | 9 | 52    |

Warning: 28 vertices with missing values excluded from crosstabulation!

Warning: 16 cells (100.00%) have expected frequencies less than 5!

The minimum expected cell frequency is 1.73!

-----  
Chi-Square: 48.2837

Cramer's V: 0.5563  
-----

Rajski(C1 <-> C2): 0.2559

Rajski(C1 -> C2): 0.4104

Rajski(C1 <- C2): 0.4046  
-----

From this table, we may conclude that the thirteen poorest countries (row one) are situated in the weak semiperiphery (column three) or in the periphery (column four). In contrast, the richest countries (row four) are situated in the core (eight countries in column one) or in the strong semiperiphery (column two). As we move from poor to rich (down the rows), we encounter less peripheral and more core countries in Table 3. Position in the world trade network is clearly related to the economic prosperity of a nation.

Statistical indices of association tell us how strong the association is. Indices range from 0 to 1 and as a rule of thumb we may say that values between 0 and .05 mean that there is no association, values between .05 and .25 indicate weak association, from .25 to .60 we say there is moderate association, and values over .60 indicate strong association.

In Pajek, two types of association indices are computed: Cramer's V and Rajski's information index. Cramer's V measures the statistical dependence between two classifications. It is not very reliable if the cross-tabulation contains many cells which are (nearly) empty, so Pajek issues a warning if this is the case. Rajski's indices measure the degree to which the information in one classification is preserved in the other classification. It has three variants: a symmetrical version, represented by (C1<->C2) in the output of Pajek, and two asymmetrical versions, which indicate the extent to which the first classification can be predicted by the second (C1<-C2) or the second classification can be predicted by the first (C1->C2).



In our example, Cramer's  $V$  is very not reliable because many cells have low expected frequencies. Rajske's information indices suggest that world system position and GDP per capita are moderately associated.

## 2.7 Summary

In this chapter, we use properties of vertices to find and interpret patterns of relations in a network. These properties are stored in partitions or vectors. Both partitions and vectors are lists of numbers, one number for each vertex in a network, but numbers in partitions refer to discrete classes whereas vector values express continuous properties of vertices. Classes in partitions are represented by integers, e.g., countries on the African continent have code 1, and negative class numbers are not allowed. Vectors contain real numbers, which can be negative.

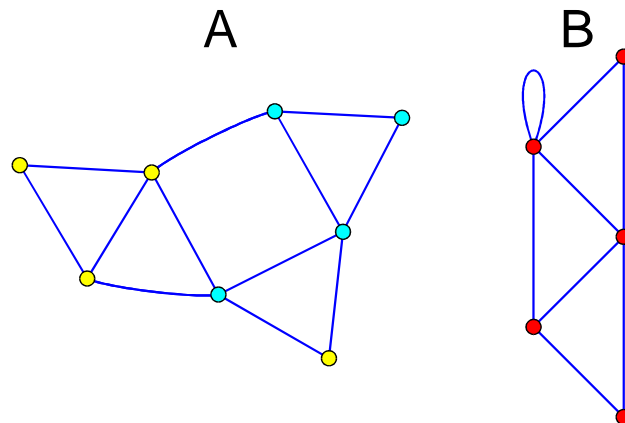
Social networks are often large and complicated. To understand network structure, it helps to study reductions of the network first. Partitions can be used to reduce a network in two ways: by extraction and by shrinking. Extraction is the selection of a subset of vertices from a network as well as the lines among these vertices. You can now concentrate on the structure of a part of the network, which is called a *local view*. In contrast, shrinking a network means that you lump together sets of vertices and lines incident with these vertices, e.g., you replace all African countries by one vertex representing the African continent. This yields a *global view* of the network if all classes are shrunk, and a *contextual view* if one class of vertices is not shrunk. In all cases, classes of a partition define the subsets of vertices which are extracted or shrunk.

Properties of social actors which are not based on their structural position in a network are called *attributes*. They are added to a network to enhance the analysis and interpretation of its structure. Partitions and vectors, however, can also contain structural indices of vertices, which result from network analysis, e.g., the world system position of countries which is inferred from the trade network. These properties may be included in statistical analysis, so partitions and vectors are the nexus between social network analysis and statistics. In the next chapter, we will discuss our first techniques to compute structural properties of vertices in a network.

## 2.8 Exercises

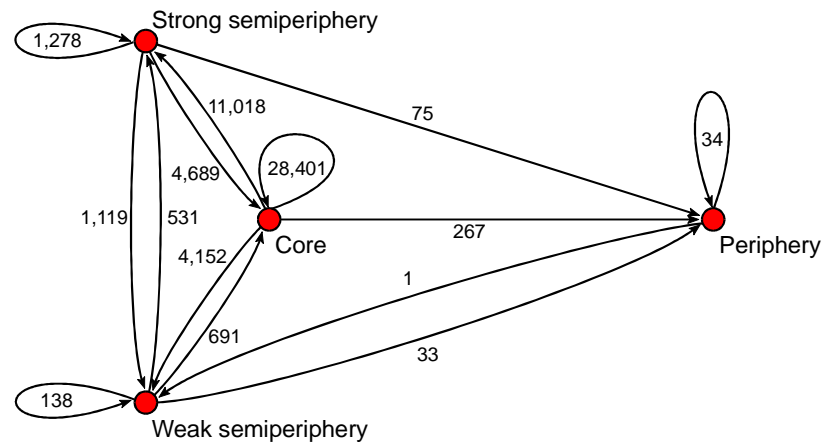
- 1 Which of the following statements is correct?
  - a Each property of a vertex or a line is an attribute.
  - b Each property of a vertex is an attribute.
  - c Each non-relational property of a vertex is an attribute.
  - d Each non-relational property of a line is an attribute.

- 2 There are two lists of numbers. Numbers in list A range from -1 to 1 and list B contains numbers between 1 and 10. Which statements can be correct?
- A and B are partitions.
  - A is a partition and B is a vector.
  - A is a vector and B is a partition.
  - A and B are vectors.
- 3 Suppose that the average population growth rate between 1990 and 1995 is available for the countries in the trade network, coded as 0 = 'negative growth', 1 = '0 to 1.0 percent', 2 = '1.0 to 3.0 percent', and 3 = '3.0 percent and over'. Would you use a partition or a vector for these data?
- A partition because we cannot add or subtract the codes meaningfully.
  - A partition because percentages are recoded to integers.
  - A vector because negative population growth is possible.
  - A vector because percentages are 'real' numbers.
- 4 What happens to the sociogram of a network if the order of numbers in the accompanying partition is changed?
- Nothing changes in the sociogram.
  - Pajek does not draw the partition any longer.
  - Pajek uses other colors to draw the vertices.
  - Vertices are drawn in the wrong color.
- 5 Which of the following statements is correct about the networks below?



- A is extracted from B.
  - B is extracted from A.
  - A is shrunk from B.
  - B is shrunk from A.
- 6 Which of the following statements is correct about the networks of exercise 5?
- B is a contextual view of A.
  - A is a global view of B.
  - B is a local view of A.
  - A is a local view of B.

- 7 Explain the structural differences between core, semiperipheries, and periphery in the world system from trade relations between world system positions depicted below. Line values represent the value of trade (miscellaneous manufactures of metal) in million US dollars.



## 2.9 Assignment

Analyze the manufactures trade network of Asian countries (local view) and their position with regard to other continents (contextual view). Does the structure of trade relations match the world system positions of countries, their prosperity (indicated by GDP per capita), and their geographical locations? Try to apply the theory of the world system, which was outlined in Section 2.2.

## 2.10 Further reading

- In *The Modern World System. Capitalist Agriculture and the Origins of the European World-Economy in the Sixteenth Century* (New York: Academic Press, 1974), Immanuel Wallerstein introduced the concept of a world system. *The Capitalist World-Economy* (Cambridge, Paris: Cambridge University Press & Editions de la Maison des Sciences de l'Homme, 1979) is a collection of essays which introduces the reader to the theory of a world-economy. Bornschier and Trezzini ('Social stratification and mobility in the world system - Different approaches and recent research'. In: *International Sociology*, vol. 12 (1997), p. 429-455) offer a summary of research traditions which use the concept of a world system or world-economy.
- See Snyder and Kick ('Structural position in the world system and economic growth 1955-70'. In: *American Journal of Sociology*, vol. 84 (1979), p. 1096-1126) for the first (published) application of social network analysis to world systems theory and see Smith and White ('Structure and Dynamics of the Global Economy - Network Analysis of International-Trade 1965-1980'. In: *Social Forces*, vol. 70 (1992), p. 857-893) for the research design followed here.
- Our data on imports are taken from the *Statistical Papers. Commodity Trade Statistics* (Series D Vol. XLIV, No. 1-34 (1994)) published by the United

Nations and we gathered additional economic and demographic data from the *Statistical Yearbook of the United Nations* (Ed. 43, IVATION Datasystems Inc.). Imports of miscellaneous manufactures of metal are listed as SITC code 69 in the commodity trade statistics.

### 2.11 Answers

- 1 Statement c is correct. Attributes, e.g., the continent of a country, are properties of vertices, viz., countries, and not lines, therefore statements a and d are incorrect. Recall that properties of lines are called line values. Properties which express the structural position of a vertex in a network, e.g., centrality, are distinguished from social, economic, psychological, and other characteristics which do not measure network position. We restrict the concept of an *attribute* to the latter, non-relational kind of property.
- 2 Statements c and d can be correct. List A contains negative numbers, so it must be a vector. List B contains positive numbers, so it is either a partition (if it contains integers) or a vector.
- 3 Statement a is the right choice. Aggregation of the percentages into four classes of unequal width implies that we cannot make meaningful calculations on the class numbers, for instance, add and divide class numbers of two countries. We cannot say that two countries have average population growth of 2.5. Classes of equal width, such as rounding to the nearest integer, would yield 'real' numbers which may safely be used in calculations. But this is not the case here (statement b). The properties of the original percentages do not matter any longer after they have been aggregated in classes, so statements c and d are incorrect.
- 4 Answer d is correct. Changing the order of numbers in a partition does not change the total number of entries or the class numbers. The number of entries of a partition must match the number of vertices in the network, otherwise Pajek does not draw the partition. Answer b is incorrect. Class numbers determine the colors used. Because they do not change, the same colors are used, (answer c is incorrect). The order of entries in a partition determines which class is assigned to which vertex: the class number in the first entry of a partition is assigned to the first vertex in the network, et cetera. Changing the order of entries in a partition changes the links between vertices and entries with class numbers, so at least some vertices will be drawn in other colors (answer d).
- 5 Statement d is correct. Network B contains less vertices than network A, so B is extracted or shrunk from A and not the other way around (statements a and c). B is shrunk from A (statement d), because we find a loop in B but no loops in A, and because there is no subset of five vertices in network A which are connected like B: three triangles which share a line with at least one other triangle. In fact, the yellow vertices in network A are shrunk to one new vertex, namely the vertex with the loop in network B.

- 6 Answer a is correct. B is shrunk from A, but not all vertices are shrunk: the light blue vertices are not shrunk (see answer to Exercise 5). A partially shrunk network offers a contextual view.
- 7 The shrunk network in exercise 7 shows several characteristics of the world system. First, the values associated with the loops reveal that trade between countries within one position of the world system decreases from core to periphery. There is very little trade among countries in the periphery. Second, values of arcs between world system positions show asymmetries in world trade. Core countries export goods at a much higher value to the semiperiphery than vice versa and the semiperiphery exports more to the periphery than the other way around. The values of arcs indicate that the larger the difference in world system position, the larger the difference between exports and imports.