Practical handout for the workshop “Introduction to the analysis of large-scale data on social connections”

Prepared for the workshop ‘Introduction to analysing social connections and occupational structure’, runby the ‘Social Networks and Occupational Structure’ project (www.camsis.stir.ac.uk/sonocs), Department of Sociology, University of Cambridge, 12 September 2012

Paul Lambert and Dave Griffiths, University of Stirling, September 2012

Version 1.2

Contents

Introduction

This handout accompanies the lab sessions for the workshp 'Introduction to analysing social connections and occupational structure' (12 September 2012, University of Cambridge).

For Stata and R, the step-by-step implementation instructions for each session are largely to be found within the specific 'syntax' files for the relevant sessions (.do and .R files). For Pajek, step-by-step instructions with screenshots are provided below.

A few sections of this handout are copied from a more extended handout on using data analysis packages for social science research, produced by Lambert for the DAMES Node workshop programme (see www.dames.org.uk) and for his course 'Introduction to multilevel models with applications' to the Essex Summer School in Social Science data analysis (www.staff.stir.ac.uk/paul.lambert/essex_summer_school).

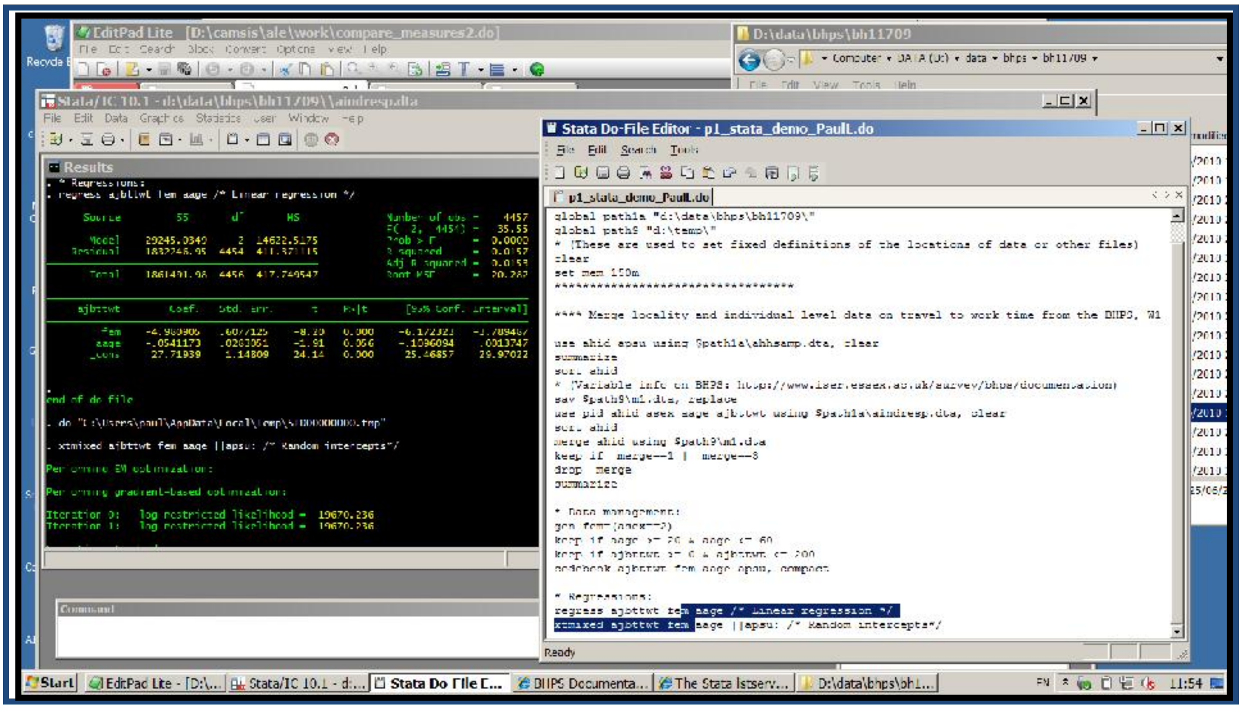
General arrangements for the practicals

Unless noted otherwise the data files used are for distribution for this lab session only and should not be transferred elsewhere. The sources of these files are ultimately avaiable online from international data providers such as IPUMS-I (https://international.ipums.org/international/), NAPP (http://www.nappdata.org/napp/) or the UK's ESDS (www.esds.ac.uk).

You will need to have the relevant packages installed to undertake the relevant exercises (though it should be possible to use only some of the packages if relevant). Introductory notes on the packages are included below under 'lab 1'. Some of the packages and lab exercises have online dependencies (e.g. they need to use a data file or programme extension which is available online). We have tried to note the details below, but information may not be comprehensive.

In general terms, the task in the labs is to open up the relevant syntax files, and work your way through them, digesting the examples shown (and potentially adding your own notes, adjustments or examples). You'll ordinarily need to have the analytical software open and the relevant tool for working with a syntax file (e.g. 'syntax window' or 'do file editor'). In addition it will typically also be helpful to have open some applications to remind you of where the data is stored, and perhaps a plain text editor allowing you to conveniently open up several other syntax files for purposes of comparison.

- When working with Stata a typical view of your desktop might be something like:



Description: The first two interfaces you can see in this screenshot are respectively the Stata do file editor (where I write commands and send them to be processed, such as by highlighting the relevant lines and clicking 'ctrl-d'); and the main Stata window (here Version 10) which includes the results page. Note that the syntax file open is a modified (personalised) version of the supplied illustrative syntax file – the name has been changed so that I can save the original file plus my own version of it after edits (e.g. with my own comments). Behind the scenes I've also got open an 'Editpadlite' session which I'm using to conveniently open up and compare some other sytnax files that I don't particularly want in my do file editor itself; I've also got a file manager software open showing the data I'm drawing upon (in what Stata will call 'path1a'); and I've got some Internet Exporer (IE) sessions open (I'm looking up the online BHPS documentation, and the Stata listserv, where information on Stata commands is available).

Materials referred to in the sessions will include:

- data files (copies of survey and other data used);
- sample command files (pre-prepared materials which include programming commands in the language of the relevant software)
- supplementary 'macros' or 'sub-files' (further pre-prepared materials featuring programming commands in relevant languages, usually invoked as a sub-routine within the main sample command files)

An important point to make is that some of the command files will need to draw upon other files (e.g. data files) in order to run. To do this, they need to be able to reference the location of the required files. In most applications, we do this be defining 'macros' which point to specific 'paths' on your computer (see also software sections below). For the labs to work successfully, it will be necessary to ensure that the command file you are trying to run is pointing to the right paths at the right time. In general, this only requires one specification to be made at the start of the session, for instance whereby in Stata we define 'macros' for the relevant paths. Sometimes however it can be necessary to edit the full path reference of a particular file in order to be able to access it.

For example, in the text below, we show some Stata (and SPSS) commands which in both cases define a macro (called ‘path3a’) which gives the directory location of the data file ‘aindresp.dta’ or ‘aindresp.sav’, so that subsequent commands calling it will go directly to that path:

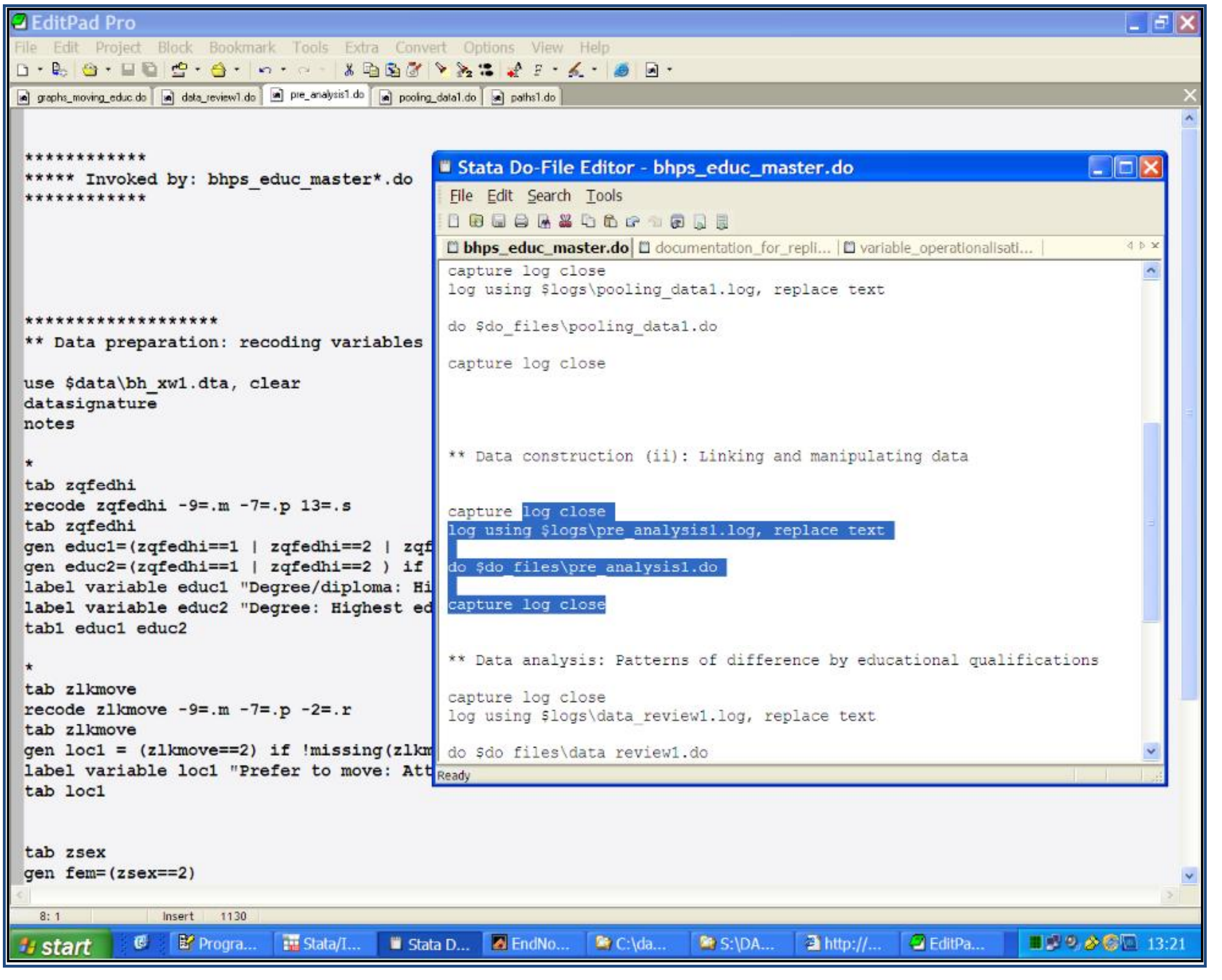
Stata example	***comparable SPSS example***
`global path3a “d:\data\bhps\”`	`define !path3a () “d:\data\bhps\” !enddefine.`
`use pid asex using $path3a\aindresp.dta, clear`	`get file=!path3a+”aindresp.sav”.`
`tab asex`	`fre var=asex.`

Relevant background: Thinking about workflows

There are very good expositions of the idea of workflows in the social science data analysis process in, amongst others, Long (2009); Treiman (2009), and Kohler and Kreuter (2008). A workflow in itself is a representation of a series of tasks which contribute to a project or activity. It can be a useful exercise to conceptualise a research project as a workflow (with components such as data collection, data processing, data analysis, report writing). However, when dealing with large scale data, a really useful contribution is to organise your data and command files that are associated with a project in a consistent style that recognises that relevant contributions to the workflow structure.

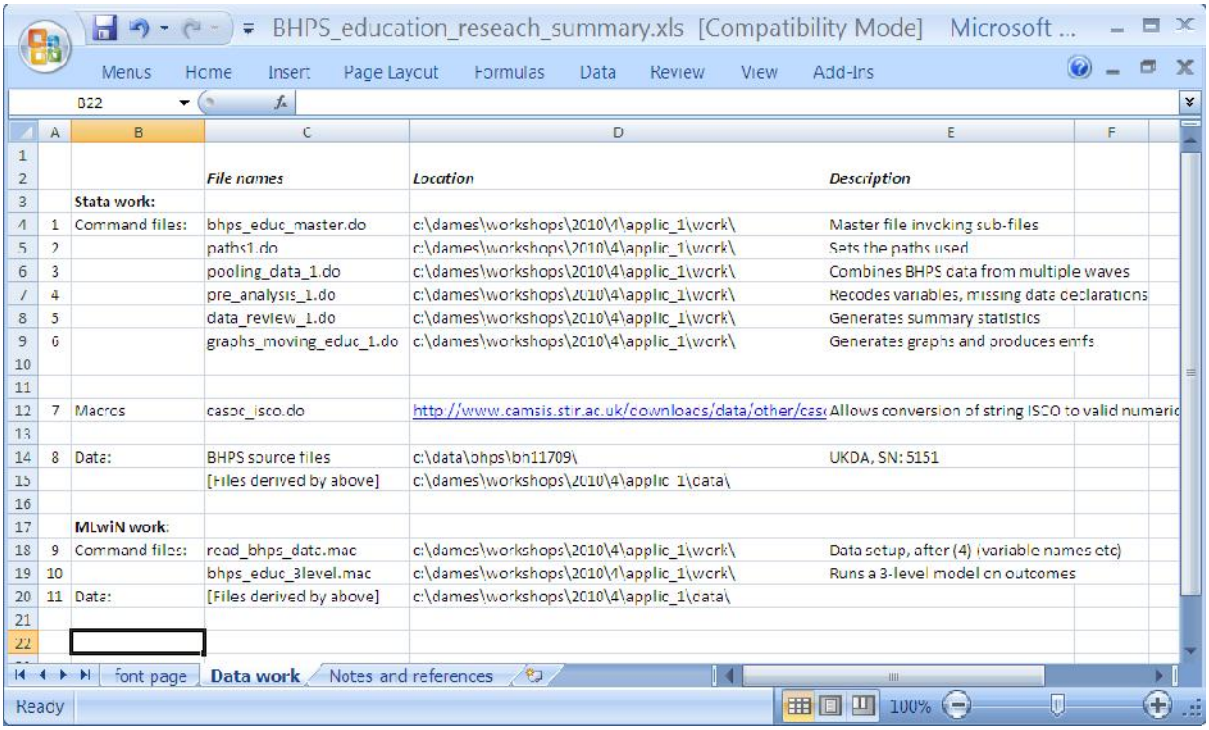
What does that involve? The issue is that we want to construct a replicable trail of our data oriented research, which allows us to go all the way from opening the initial data file, to producing the publication quality graph or statistical results which are our end products. We need the replicable trail in order to adjust our analysis on the basis of minor changes at any possible stage of the process (or to be able to transfer a record of our work on to others). However because when dealing with large-scale and complex data (e.g. on social connections) the trail is long and complex (and not entirely linear), we can only do this, realistically, if we break down our activities into multiple separate components.

There are different ways to organise files for these purposes, but a popular and highly effective approach is to design a ‘master’ syntax command file and a series of ‘sub-files’ which it draws upon. In this model, the sub-files cover different parts of the research analysis. Personally, my preference is to construct both the master and sub-files in the principle software package being used, though Long (2009) notes that creating a documentation master file in a different software (e.g. MS Excel) is an effective way to record a wider range of activities which span across different software. Here’s an example of a series of tasks being called upon via a Stata format ‘master’ file:



(This screenshot shows the Stata master file, and the sub-files which are mostly open within the EditPad editor - except for a few other files which I've opened in the do file editor. The Stata output file is not visible but is open behind the scenes).

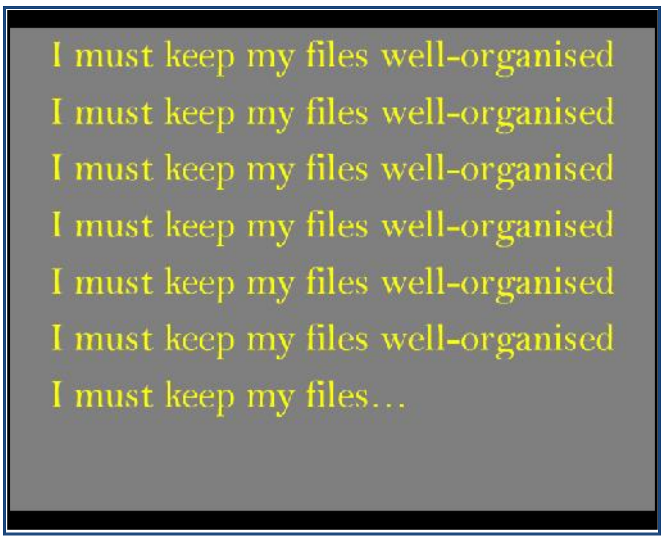
Here's an example of a project documentation file that might be constructed in Excel:



BHPS_education_reseach_summary.xls [Compatibility Mode] Microsoft ...

	A	B	C	D	E
1					
2			*File names*	*Location*	*Description*
3		**Stata work:**			
4	1	Command files:	bhps_educ_master.do	c:\dames\workshops\2010\4\applic_1\work\	Master file invoking sub-files
5	2		paths1.do	c:\dames\workshops\2010\4\applic_1\work\	Sets the paths used
6	3		pooling_data_1.do	c:\dames\workshops\2010\4\applic_1\work\	Combines BHPS data from multiple waves
7	4		pre_analysis_1.do	c:\dames\workshops\2010\4\applic_1\work\	Recodes variables, missing data declarations
8	5		data_review_1.do	c:\dames\workshops\2010\4\applic_1\work\	Generates summary statistics
9	6		graphs_moving_educ_1.do	c:\dames\workshops\2010\4\applic_1\work\	Generates graphs and produces emfs
10					
11					
12	7	Macros	casoc_isco.do	http://www.camsis.stir.ac.uk/downloads/data/other/cas	Allows conversion of string ISCO to valid numeric
13					
14	8	Data:	BHPS source files	c:\data\bhps\bn11709\	UKDA, SN: 5151
15			[Files derived by above]	c:\dames\workshops\2010\4\applic_1\data\	
16					
17		**MLwiN work:**			
18	9	Command files:	read_bhps_data.mac	c:\dames\workshops\2010\4\applic_1\work\	Data setup, after (4) (variable names etc)
19	10		bhps_educ_3level.mac	c:\dames\workshops\2010\4\applic_1\work\	Runs a 3-level model on outcomes
20	11	Data:	[Files derived by above]	c:\dames\workshops\2010\4\applic_1\data\	

Note that the other tabs in the Excel file can be used to show things like author details, context of the analysis, and last update time. The file also notes some (though not all) dependencies within the workflow – for instance step 9 requires step 4 to have been take (the macro reads in a plain text data file that was generated in Stata by do file pre_analysis1.do).



In summary, we can't advise you strongly enough on the value of organising you data files around a workflow conceptualisation, such as through master and sub-files. Read the opening chapters of Long (2009), or the other references mentioned above, for more on this theme. *We'd encourage you to look at the workshop materials from the 'DAMES' research Node, at www.dames.org.uk, for more on this topic.*

Software alternatives

Many different software packages can be used effectively for applied research using complex data on social connections. Various packages support the estimation of a wide range of statistical models including association models, and there are numerous (mostly different) packages which feature techniques for social network analysis.

In this session we focus upon three packages which bring slightly different contributions to multilevel modelling.

- **Stata** is used because it is a popular general purpose package for data management and data analysis which also includes a substantial range of analysis options for dealing with data on social connections. Stata is attractive to applied researchers for many reasons, including its good facilities for storing and summarising estimation results; its support of a wide range of advanced analytical methods which complement a multilevel analysis (e.g. clustering estimators used in Economics); and its wide range of data management functions suited to complex data. Stata is proprietory and may be purchased from: www.stata.com.

- **R** is used because it is a popular freeware that supports many forms of statistical model estimation, social network analysis examples, and has various graphical and data construction capabilties. Many of its facilities are available via extension 'libraries' which are usually installed online. R is a difficult language for social scientists to work effectively with, however, because it brings with it very high 'overheads' in its programming requirements, especially for large and complex data. R is available to install as freeware from: http://www.r-project.org/

- **Pajek** is used because it is a freely available and popular package for social network analysis, featuring a wide range of graphical and statistical analysis possibilities. Pajek may be downloaded and installed as freeware from: http://pajek.imfm.si/doku.php

We should stress that many more packages can be used effectively for the analyses used below. In addition, an exciting software development in the area being led in the UK is the construction of a generic interface for specifying and estimating complex statistical models of 'arbitrary complexity'. These cover most forms of multilevel models, as well as many other statistical modelling devices. This project is called '**e-Stat**' and is expecting to generate it first publicly available resources over the period 2010-2012 (see http://www.cmm.bristol.ac.uk/research/NCESS-EStat/).

Lab 1: Introduction to the analysis of social connections data

This lab introduces a few examples of datasets on social connections, and provides illustrative analyses in Stata, R and Pajek.

The work of the Stata and R exercises is done by the corresponding command files, which should (hopefully) be self-explanatory. To run these exercises, open the relevant files in each package and work through them:

Lab1_stata.do
Lab1_R. R

We also introduce using Stata and R below.

The Pajek exercises are described through step-by-step instructions, provided below.

Background: Introducing Stata

Stata was first developed in 1984 and was originally used mainly in academic research in economics. From approximately the mid 1990's its functionalities for social survey data analysis began to filter through to other social science disciplines, and in the last decade it has displaced SPSS as the most popular intermediate-to-advanced level statistical analysis package in most academic disciplines which use social survey data (e.g. sociology, educational research, geography).

Stata is popular for many good reasons. The list of features of Stata that lead me personally to favour this package above others are:

- It supports explicit documentation of complex processes through a concise and 'human readable' syntax language
- It supports a wide range of data management functions including many routines useful in complex survey data which are not readily performed in other packages (e.g. 'egen', 'xtdes')
- It supports a very full range of statistical modelling options, including several advanced model specifications which are not widely available elsewhere
- It has excellent graphics capabilities, supporting the specification and export of publication quality graphs (in a syntactical, replicable manner)
- It features very convenient tools for storing the results from multiple models or analyses and compiling them in summary tables or files (e.g. 'est store', 'statsby')
- It can read online data files and run command files and macros from online locations
- It supports extended add-on programming capabilities, and benefits from a large, constructive community of user-contributed extensions (see e.g. http://www.stata.com/links/resources3.html)

In pragmatic terms, most users of Stata are reasonably confident programmers, and getting started with the package does need a little effort in learning about data manipulation and data analysis. This is one reason why Stata is not yet widely taught in introductory social science courses, though, in the UK for example, it is increasingly used in intermediate and advanced level teaching (e.g. MSc programmes or Undergraduate social science programmes with extended statistical components).

A common problem with working with Stata is that many institutions do not have site-level access to the software, and accordingly many individual researchers don't have access to the package - Stata is generally sold as an 'n-user' package, which means that an institution buys a specified number of copies at any one time. Recently however, access to Stata for academic researchers has probably be made easier by the Stata 'GradPlan', which allows purchase of personal copies of the package for students and faculty at fairly low price – see http://www.stata.com/order/new/edu/gradplan.html . Stata also comes in several different forms with different upper limits on the scale of data it may handle – 'Small Stata' is not normally adequate for working with advanced survey datasets; 'Intercooled' Stata (I/C) usually has more than enough capacity to support social survey research analysis (although, working with a large scale resources you may occasionally hit upper limits, such as on the number of variables or cases, it is usually possible to find an easy work-around such as by dropping unnecessary variables); Stata SE and MP offer greater capacity regarding the size of datasets and faster processing power, but they are more expensive to purchase. To my knowledge, most academic researchers use Intercooled Stata.

In summary, many users of Stata favour the package not because it offers one particular functionality which others don't, but because it offers an integrated set of advanced functionalities covering data management and data analysis which can't easily be matched by any other software. For other texts which explain the strengths and attractions of Stata, see for example Treiman (2009).

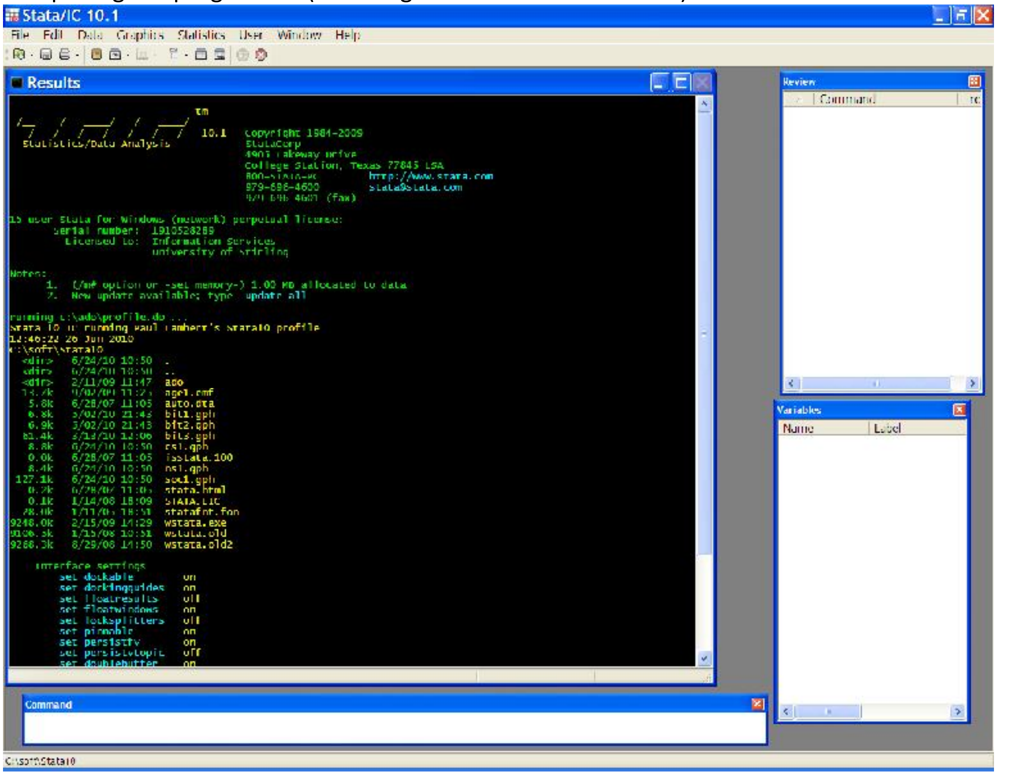
The steps below give you some relevant instructions on working with Stata for the purposes of the module (the examples are mostly from the Practical 1 Stata file). Many online resources on Stata are available, in particular we highlight:

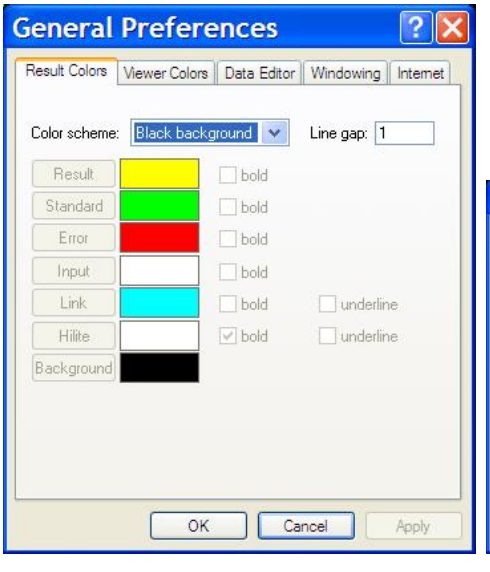
- UCLA's ATS pages: http://www.ats.ucla.edu/stat/stata/ (Features a wide range of materials including videos of using Stata and routines across the range of the package)
- The CMM's LEMMA online course: http://www.cmm.bristol.ac.uk/learning-training/course.shtml (includes detailed descriptions of running basic regression models and of specifying random effects multilevel models in Stata)
- In the first lab session we point you to an illustrative do file which serves as an introduction to Stata, available from www.longitudinal.stir.ac.uk

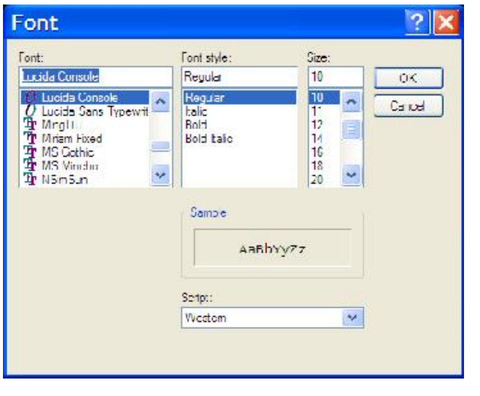
When you launch the package, you see the basic Stata window, here for version I/C 10.1.

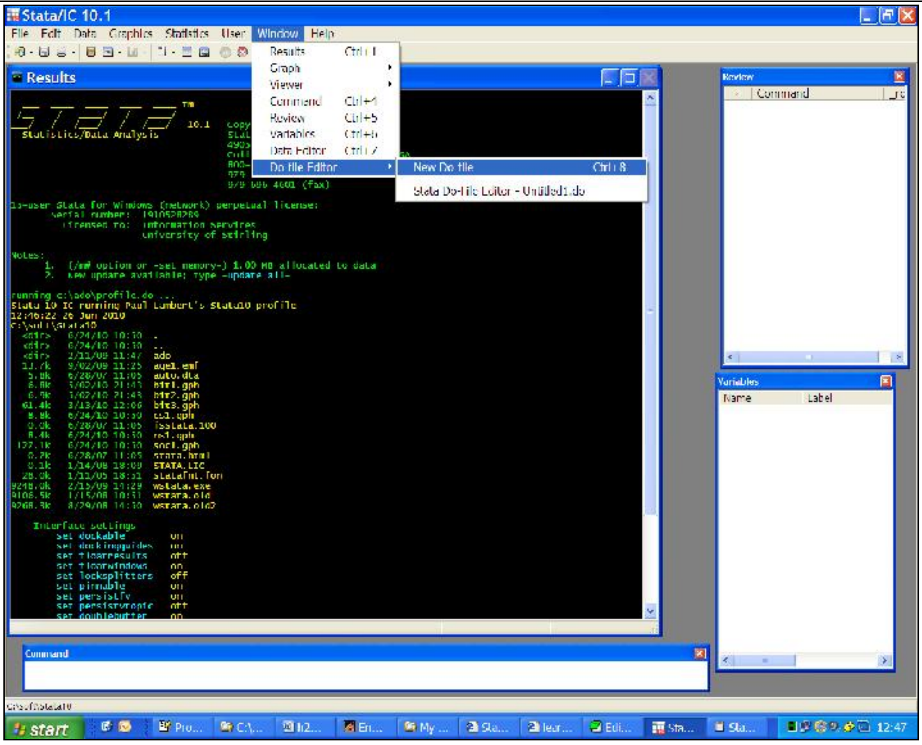
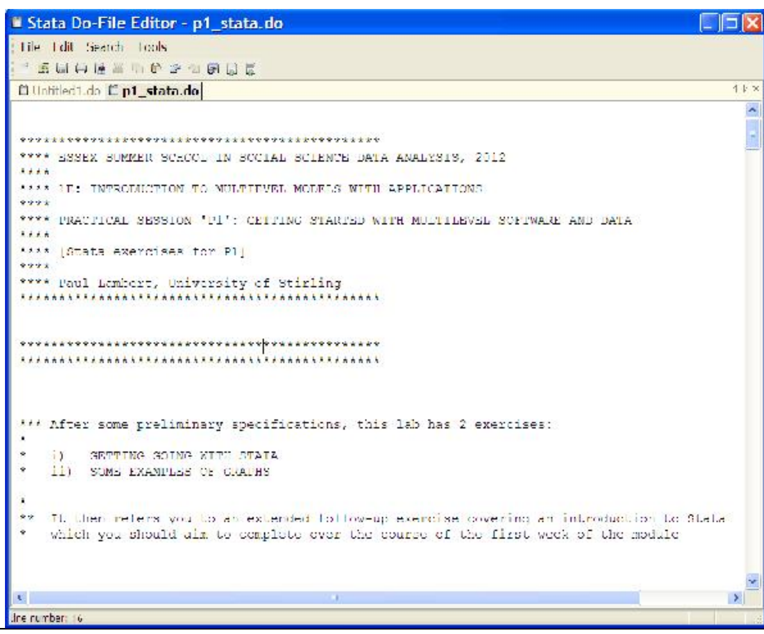
You can customise its appearance (e.g. right click on the results window) – the image on the right reflects how I've set up the windows on my machine, and will be slightly different to what you see by default on first launching the package in the lab at Essex.

On opening the programme (this image shows Stata version 10):



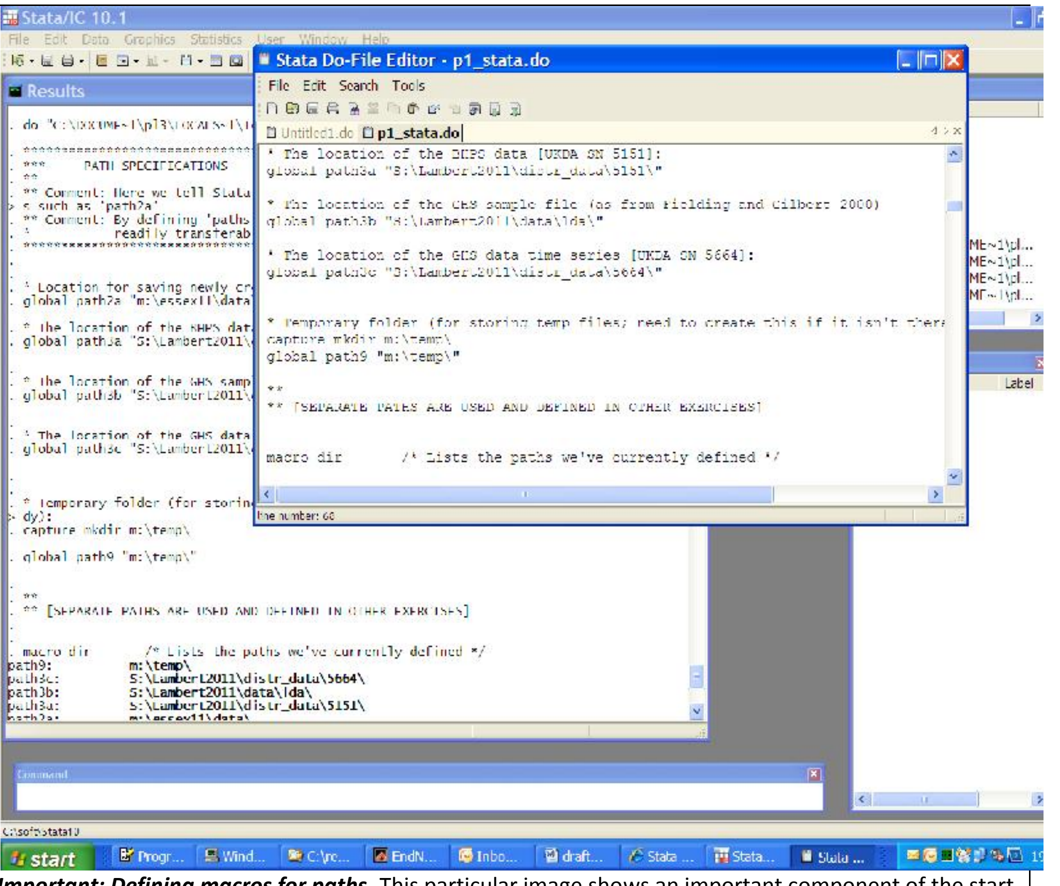




The very first thing you should do at the start of every session it so ask explicitly to open the ‘do file’ editor with ‘**ctrl+8**’ or via the GUI. Note below that we can have several do files open at once. Not shown below, but from Stata 11 onwards, it is possible to permit various formatting options in the do file editor (e.g. colour coding). It is also possible to set up Stata to run directly from a plain text editor if you wish to (search online for how to do this).	  

Once you've opened a 'do file' you can begin running commands by highlighting the segments of the relevant command lines and clicking 'ctrl+R'.

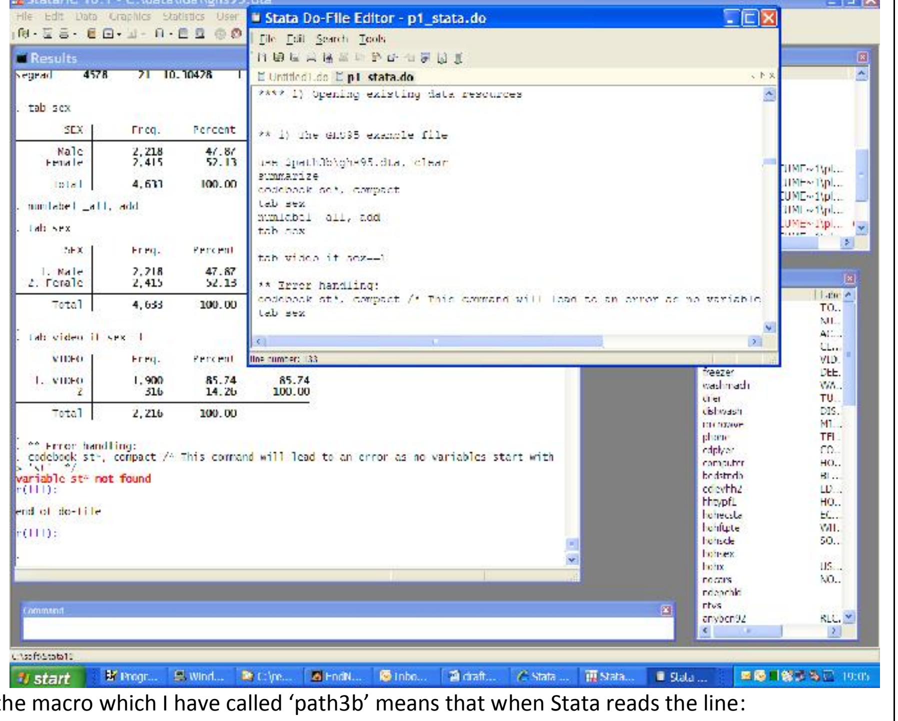
(Note that I've changed the windows display to 'white' background to make it easier to read the outputs)



Important: Defining macros for paths. This particular image shows an important component of the start of every session in the module lab exercises. The lines beginning with 'global' are ways of defining permanent 'macros' for the session. The macros serve to define locations on my machine where my files (e.g. data files) are actually stored. Doing this means that in later commands (e.g. the image below) I can call on files via their macro folder locations rather than their full path – this aids transferability of work between machines/locations.

The results are shown in the results window. Error messages are by default shown in red text and lead to the termination of the sequence of commands at that point

(unlike in SPSS, which carries on, disregarding the error).



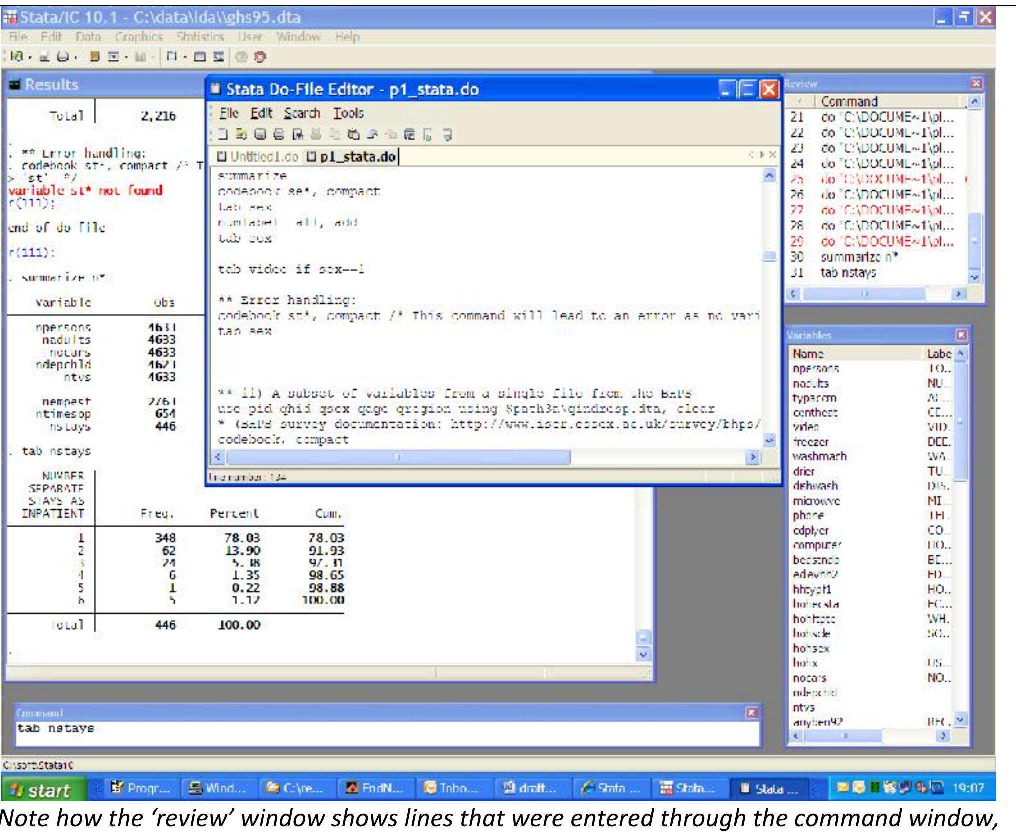
In the above, the macro which I have called 'path3b' means that when Stata reads the line:

use $path3b\ghs95.dta, clear

what it reads 'behind the scenes' is

use c:\data\lda\ghs95.dta, clear

You can also submit commands line by line through the command interface (e.g. if you don't want to log them in the do file).



Note how the 'review' window shows lines that were entered through the command window, but it just shows some programming code for commands run through the do file editor.

Note some of the features of the Stata syntax language:

You need to 'clear' the dataspace to read in a new file, e.g.

```
use $path3b\ghs95.dta, clear
```

You can't create a new variable with the same name as an existing one – if it's there already you need to delete it first, e.g.

```
drop fem
gen fem=(sex==2)
```

The 'capture' command suppresses potential error messages so is a useful way to make commands generic

```
capture drop fem
gen fem=(sex==2)
```

Using 'by:' or 'if..' within a command can usefully restrict the specifications:

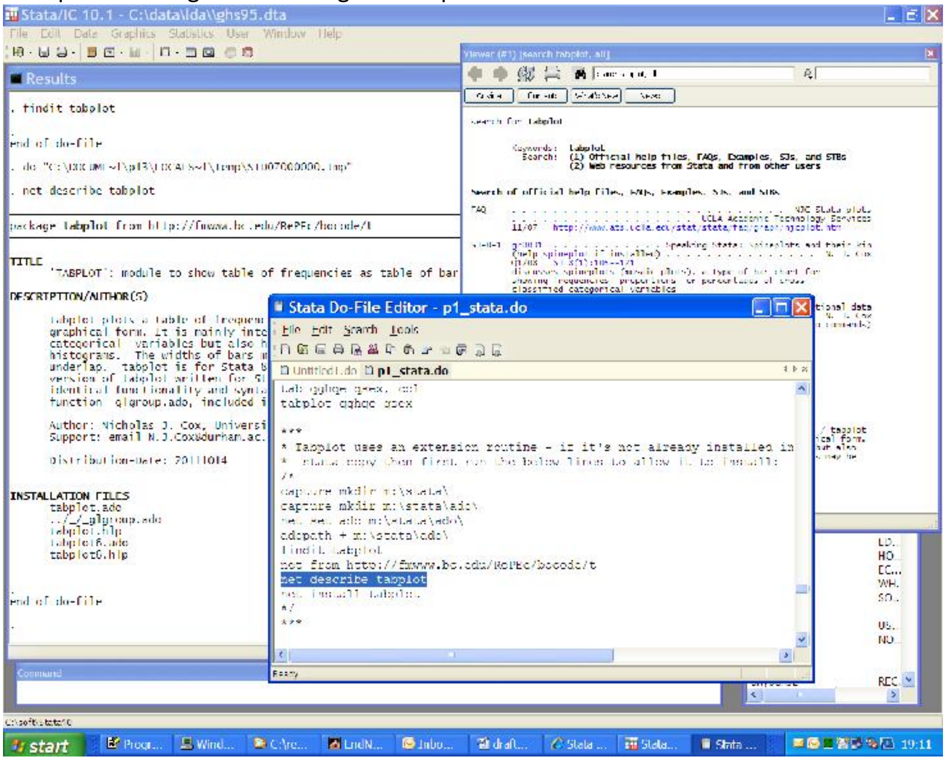
```
tab nstays if sex==1
bysort sex: tab nstays if age >= 20 & age <= 30
```

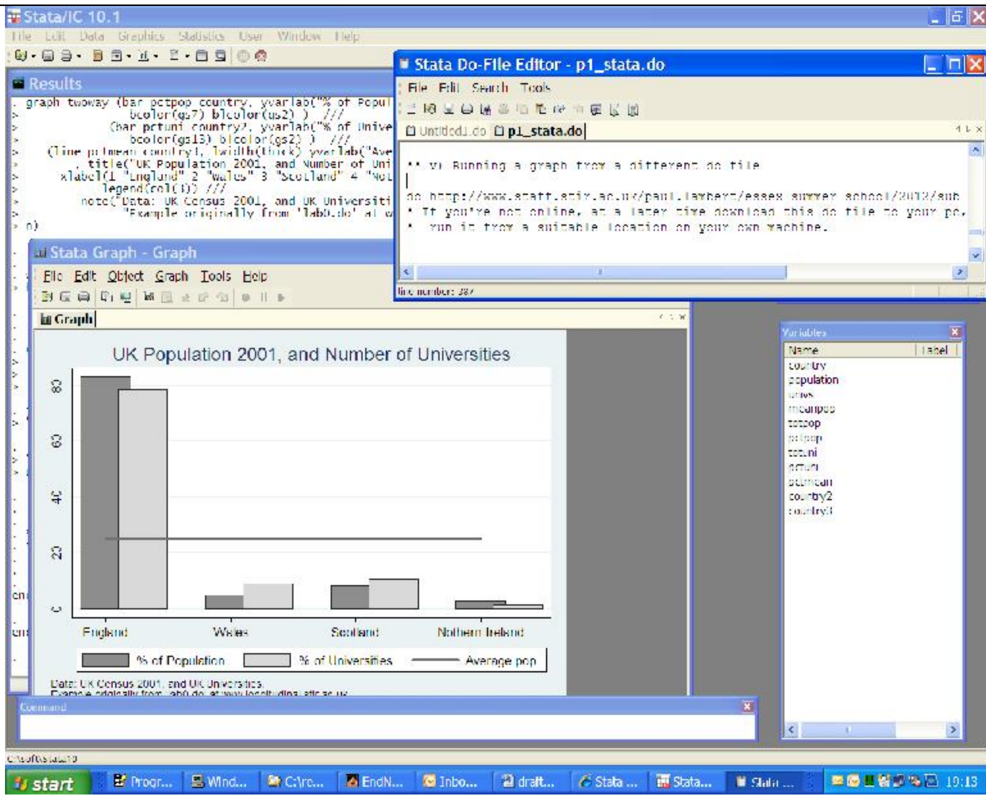
The 'numlabel' command is a useful way to show both numeric values and categorical labels compared to the default (labels only), e.g.;

```
tab ecstaa
numlabel _all, add
tab ecstaa
```

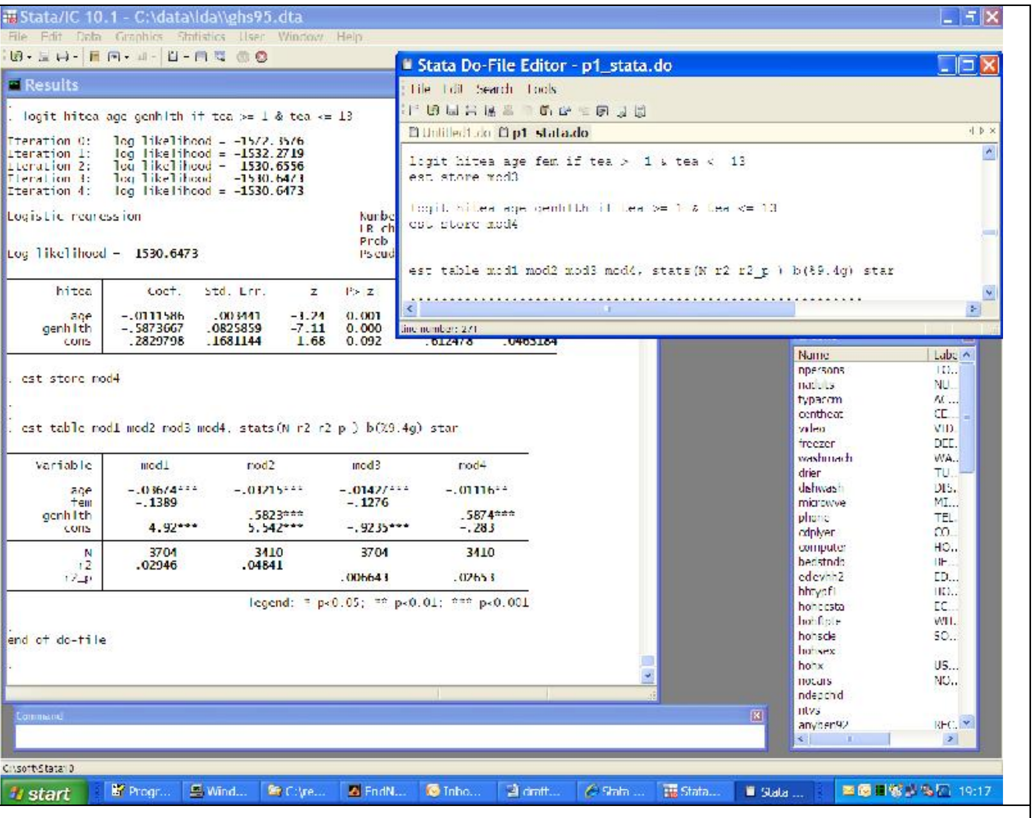
There's no requirement for full stops at the end of lines, but a carriage return serves as the default delimiter, and so we usually use '///' to extend a command over more than one line.

```
bysort sex: tab nstays ///
   if age >= 20 & age <= 60 & ecstaa==1
```

<table>
<tr><td></td><td></td></tr>
<tr><td>Extension routines are often written by users and made available to the wider community. To exploit them, you need to run either ‘net install’ or ‘ssc install’</td><td>Example of finding and installing the ‘tabplot’ extension routine:


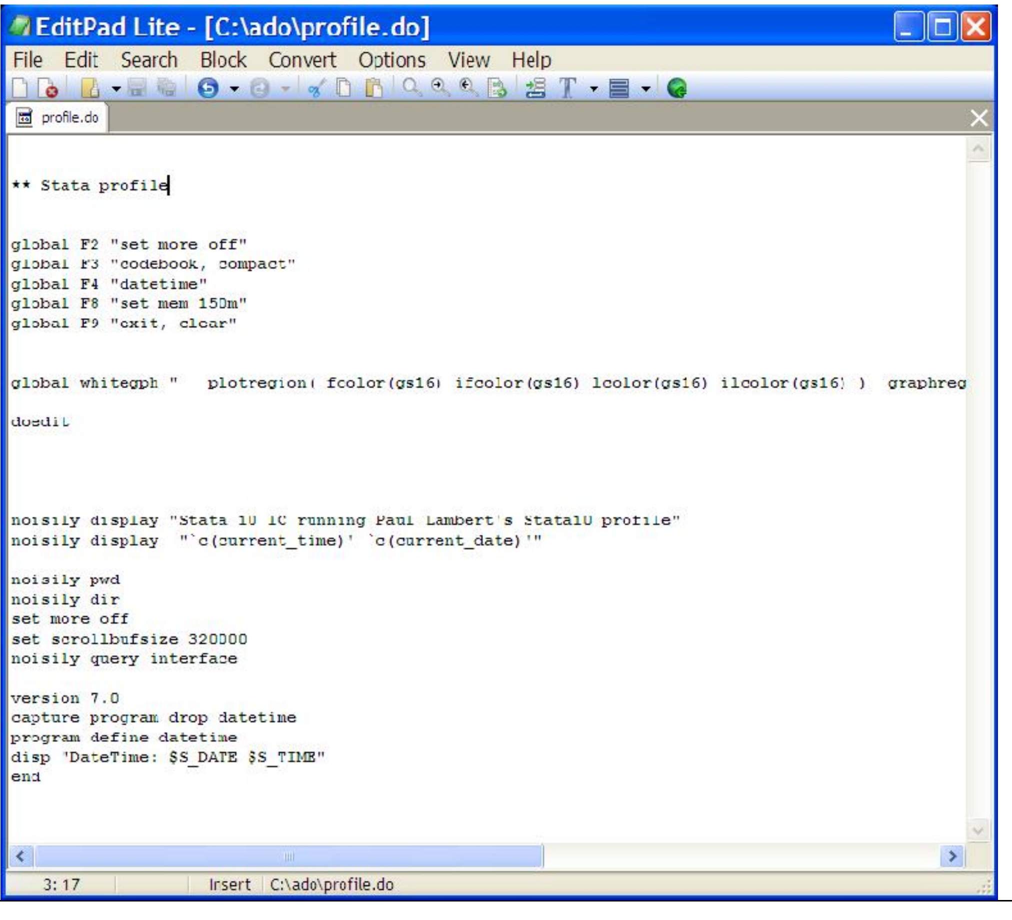
(the exact code needed may depend on which machine you are working on – you may have to define a folder for the installation that you have permission to write to)</td></tr>
<tr><td>We often run subfiles, or define macros or programmes, via calling upon other do files with the ‘do’ command</td><td>
</td></tr>
</table>

'est store' is a great way to collate and review multiple model results



Extension: You can write a '.profile' file to load some starting specifications into your session

For lots more on Stata, see the links and references given above, or the DAMES Node workshops at dames.org.uk

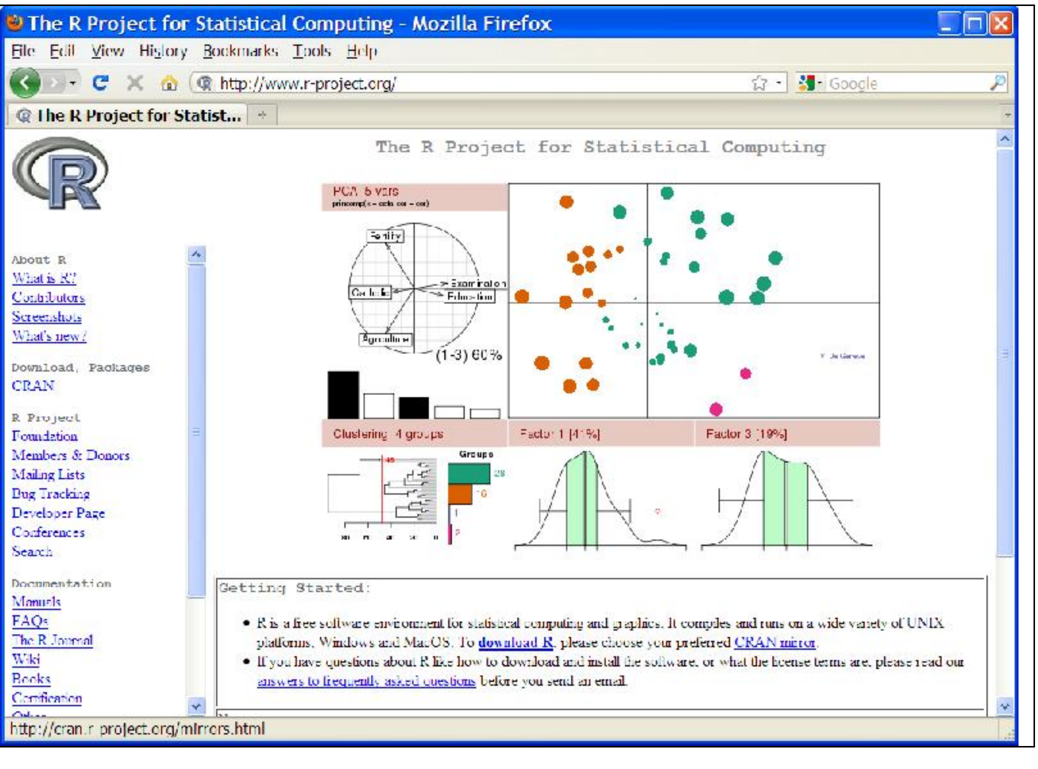


Background: Introducing R

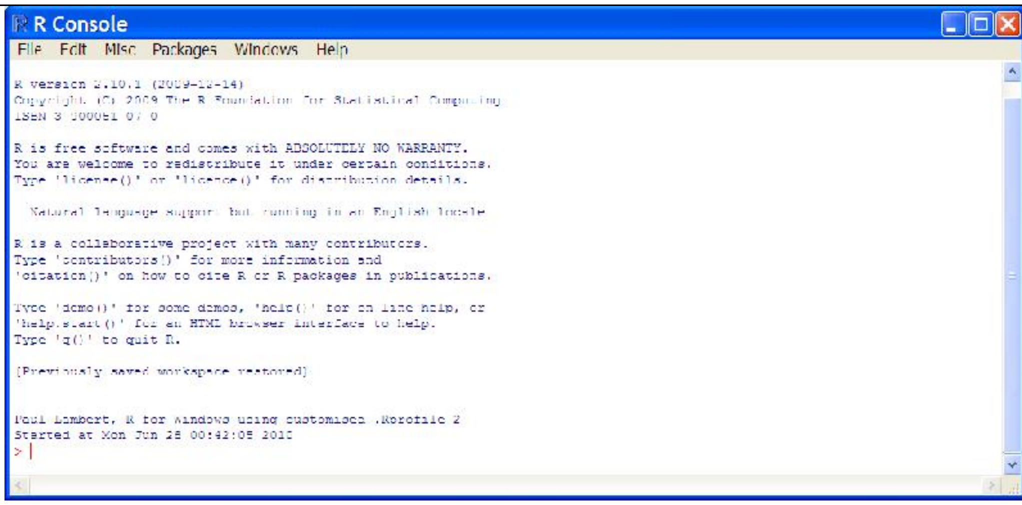
R is a freeware which is a popular tool amongst statisticians and a small community of social science researchers with advanced programming skills. It is an 'object oriented' programming language which supports a vast range of statistical analysis routines, and many data management tasks, through its 'base' or extension commands. Being 'object oriented' is important and means the package appears to behave in a rather different way to the other packages described above. The other packages essentially have one principal quantitative dataset in memory at any one time, plus they store metadata on the matrix and typically some other statistical results in the form other scalars and matrices. In the other packages, commands are automatically applied to the variables of the principal dataset. In R, however, different quantitative datasets ('data frames'), matrices, vectors, scalars and metadata, are all stored as different 'objects', potentially alongside each other. R therefore works by first defining objects, then second performing operations on one or many objects, however defined.

Some researchers are very enthusiastic about R, the common reasons being that it is free and that it often supports exciting statistical models or functions which aren't available in other packages. However, my perspective is that R isn't an efficient package for a social survey researcher interested in applied research, as the programming demands to exploit it are very high, and, because it isn't widely used in applied research, it hasn't yet developed robust and helpful routines, working interfaces, or documentation standards, to address popular social science data-oriented requirements.

An important concept in R is the 'extension library', which is how 'shortcut' programmes to undertake many routines are supplied. In fact, you will rarely use R without exploiting extension libraries. The idea here is that R has a 'base' set of commands and support, and that many user-contributed programmes have been written in that base language. Those extensions typically provide shortcut routes to useful outcome analyses. A few extension libraries in R are specifically designed to support random effects multilevel model estimation – e.g. the lme package (Bates, 2005; Pinhero & Bates, 2000).

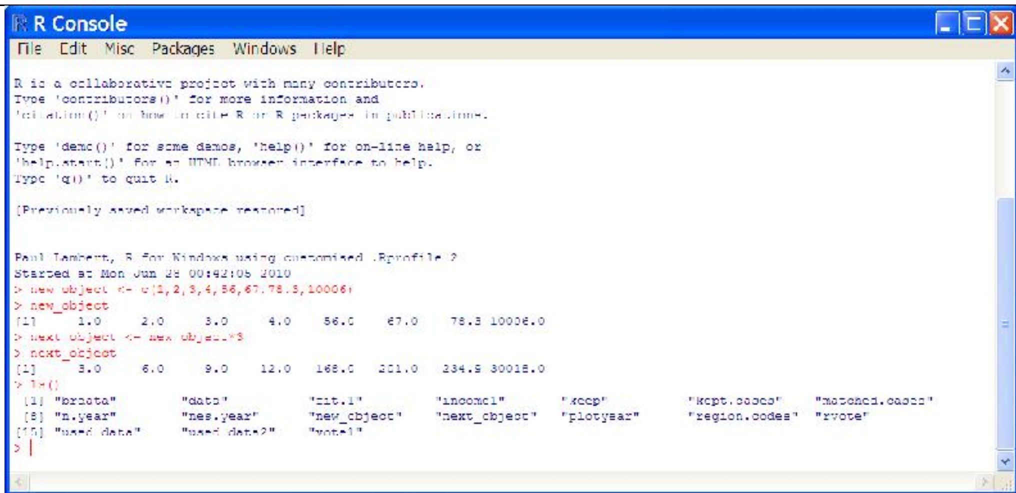
R is installed as freeware and since it is frequently updated it is wise to regularly revisit the distribution site and re-download	

When you open R, you will see something like this ‘R console’

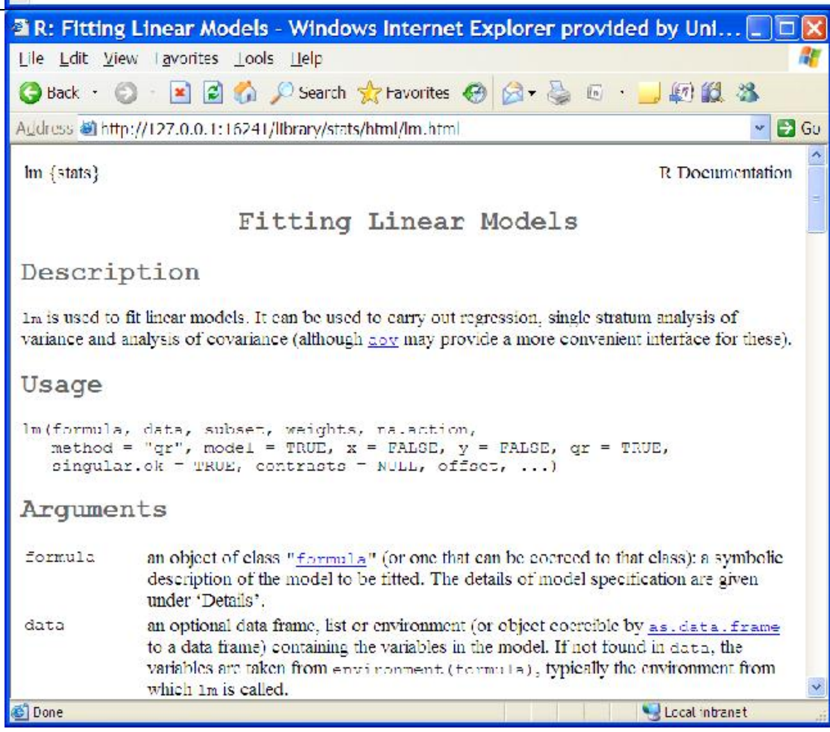


(on my machine I use a ‘.rprofile’ settings file so my starting display is marginally different to the default)

The first few lines show me defining a new object (a short vector) and listing the objects in current memory.

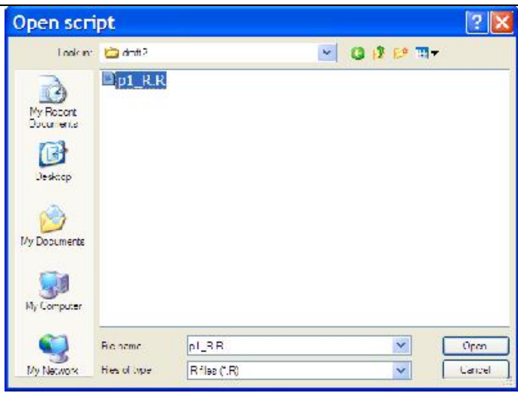


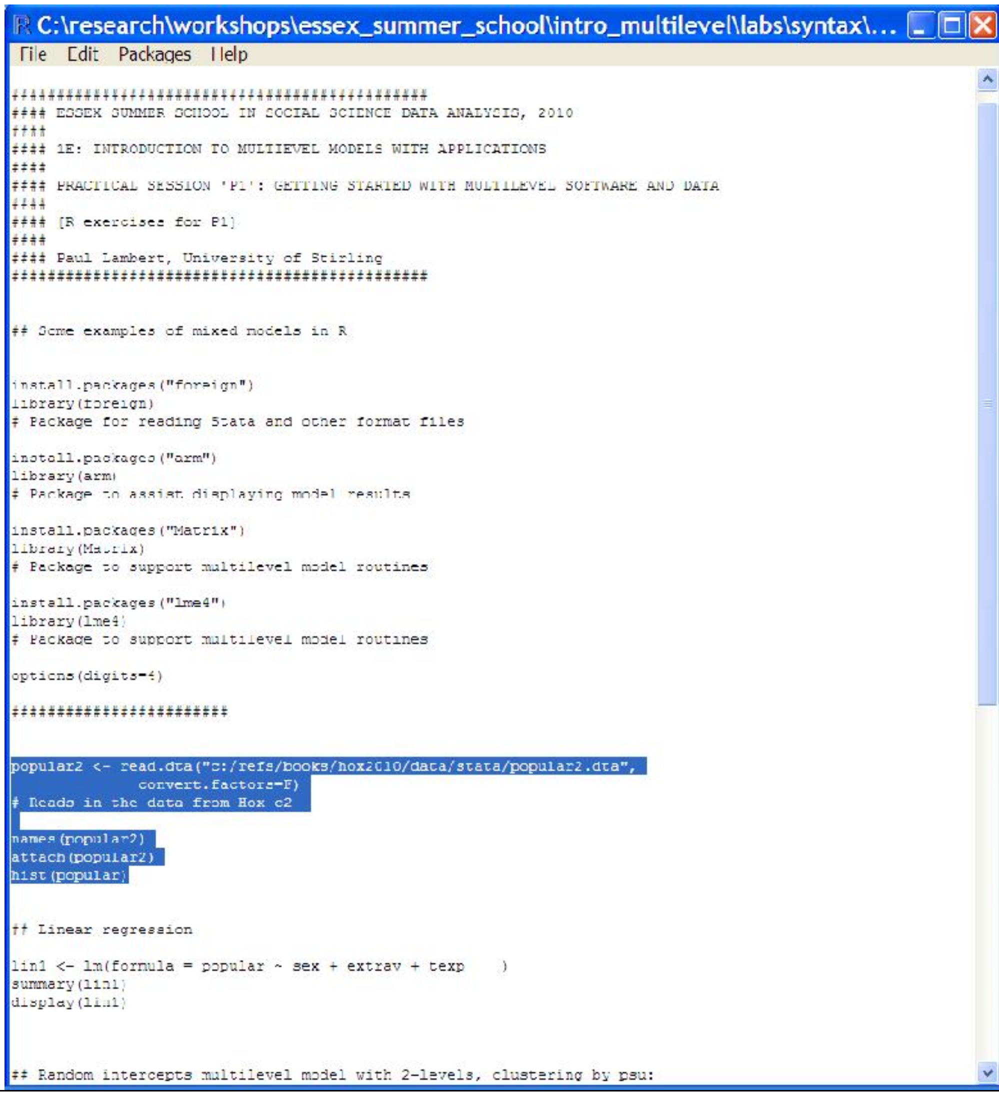
R’s basic help functions point to webpages.



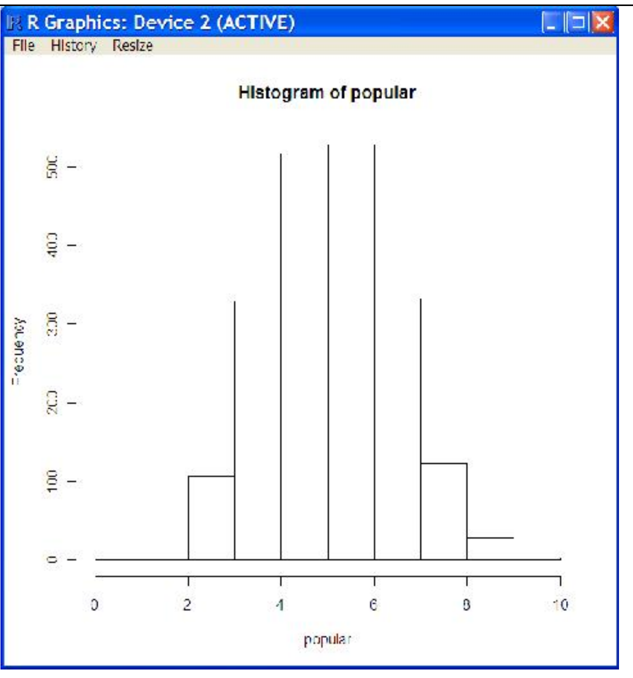
The general help pages mostly have generic information, and are not in general provided with worked examples. Many R users get their help from other online sources, e.g. http://www.statmethods.net/

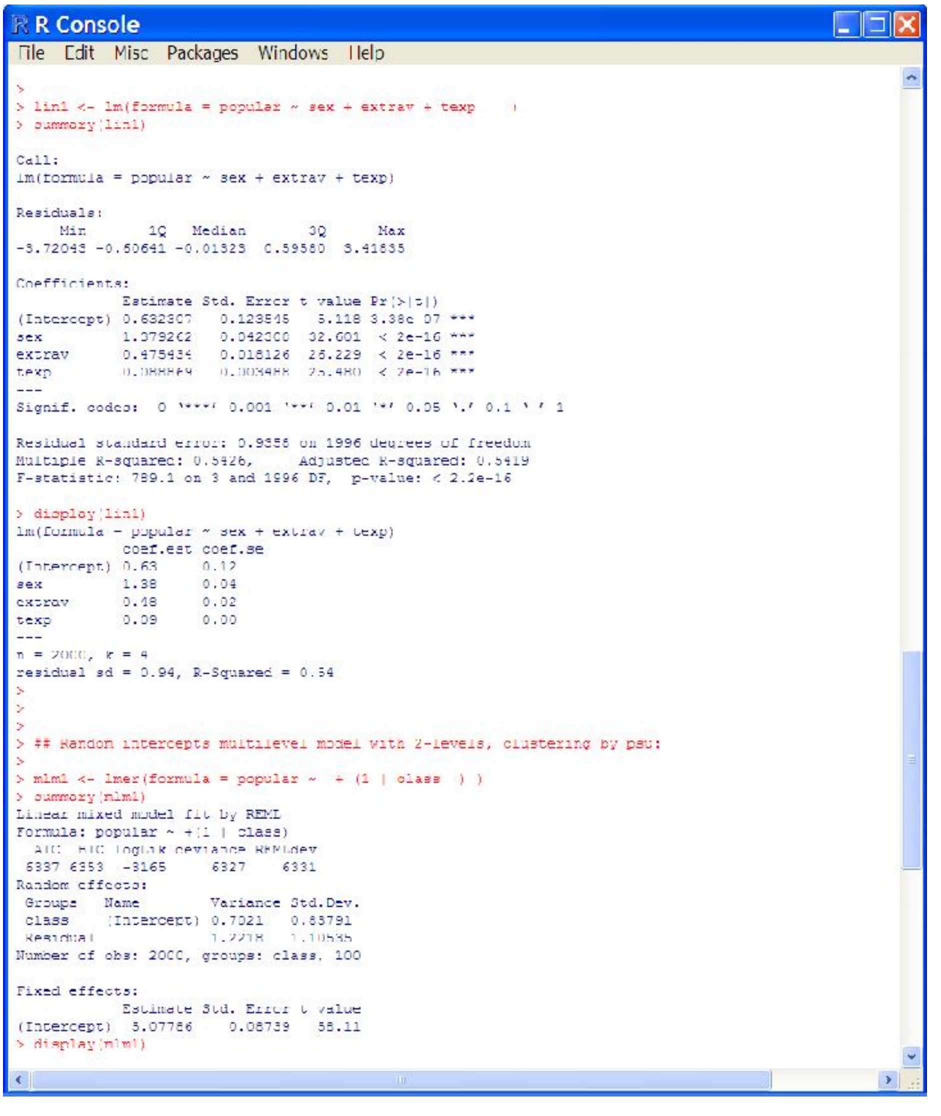
In general, with R, the first thing you should do is ask to open a new or existing script and work from that. Scripts in R work in a similar way to a syntax file in Stata or SPSS – highlight a line or lines, and press ‘ctrl+R’.





After running commands, output is sent either to the main console or a separate graphics window




R Console

File Edit Misc Packages Windows Help

```
>
> lin1 <- lm(formula = popular ~ sex + extrav + texp    )
> summory(lin1)

Call:
lm(formula = popular ~ sex + extrav + texp)

Residuals:
     Min       1Q   Median       3Q      Max
-3.72043 -0.60641 -0.01323  0.59580  3.41635

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.632307   0.123545   5.118 3.38e 07 ***
sex         1.379262   0.042300  32.601  < 2e-16 ***
extrav      0.475434   0.018126  26.229  < 2e-16 ***
texp        0.088869   0.003488  25.480  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.9356 on 1996 degrees of freedom
Multiple R-squared: 0.5426,     Adjusted R-squared: 0.5419
F-statistic: 789.1 on 3 and 1996 DF,  p-value: < 2.2e-16

> display(lin1)
lm(formula = popular ~ sex + extrav + texp)
            coef.est coef.se
(Intercept) 0.63     0.12
sex         1.38     0.04
extrav      0.48     0.02
texp        0.09     0.00
---
n = 2000, k = 4
residual sd = 0.94, R-Squared = 0.54
>
>
>
> ## Random intercepts multilevel model with 2-levels, clustering by psu:
>
> mlm1 <- lmer(formula = popular ~  + (1 | class  ) )
> summory(mlm1)
Linear mixed model fit by REML
Formula: popular ~ +(1 | class)
  AIC  BIC logLik deviance REMLdev
 6337 6353  -3165     6327    6331
Random effects:
 Groups   Name        Variance Std.Dev.
 class    (Intercept) 0.7021   0.83791
 Residual             1.2218   1.10535
Number of obs: 2000, groups: class, 100

Fixed effects:
            Estimate Std. Error t value
(Intercept)  5.07786    0.08739   58.11
> display(mlm1)
```

Lab 1: Pajek exercises

Pajek is a social network analysis software package which has been developed by Vladimar Bataglj and Andrej Mrvar from the University of Ljubljana (hence the name, Slovenian for 'spider'). Some argue the software is not as advanced as competing generalist software such as UCINET or the Social Network package within R, but it has the following benefits:

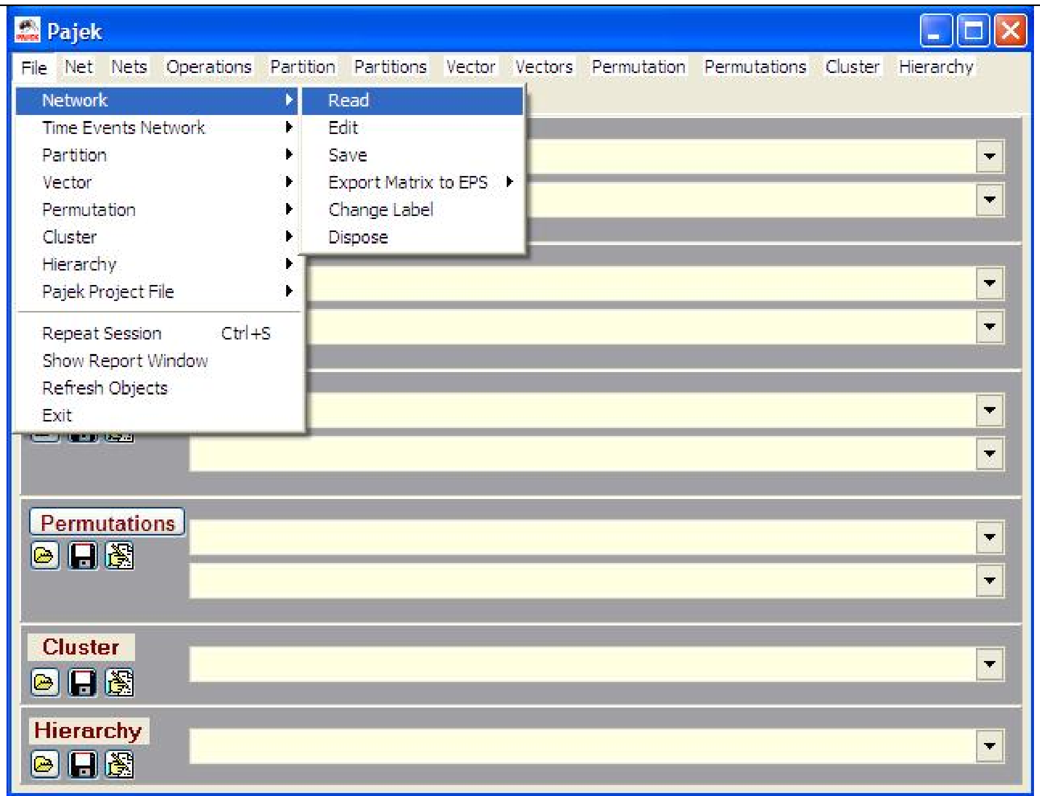
- It is simple and free to install (for non-commercial use)
- It has easy methods for importing data
- It is simple to use and covers most common network commands
- It is more robust than other packages for dealing with very large datasets.

There are several limitations with Pajek. Unlike R there is a requirement to use drop-down menus meaning it is not possible to run syntax files (although all processes can be saved). It cannot perform some of the emerging analyses such as random graph models, forcing users to use SIENA, PNET or other specialist package. However, Pajek performs the basic elements of network analysis in a very user-friendly manner, which makes it the ideal package for people unfamiliar with network methods (and, therefore, less likely to require the more advanced methods central to other packages). It retains sufficient sophistication to be utilised by many experienced researchers. Most other SNA packages (for instance, UCINET, Siena and PNET) have strong links to Pajek and enable data to be readily imported.

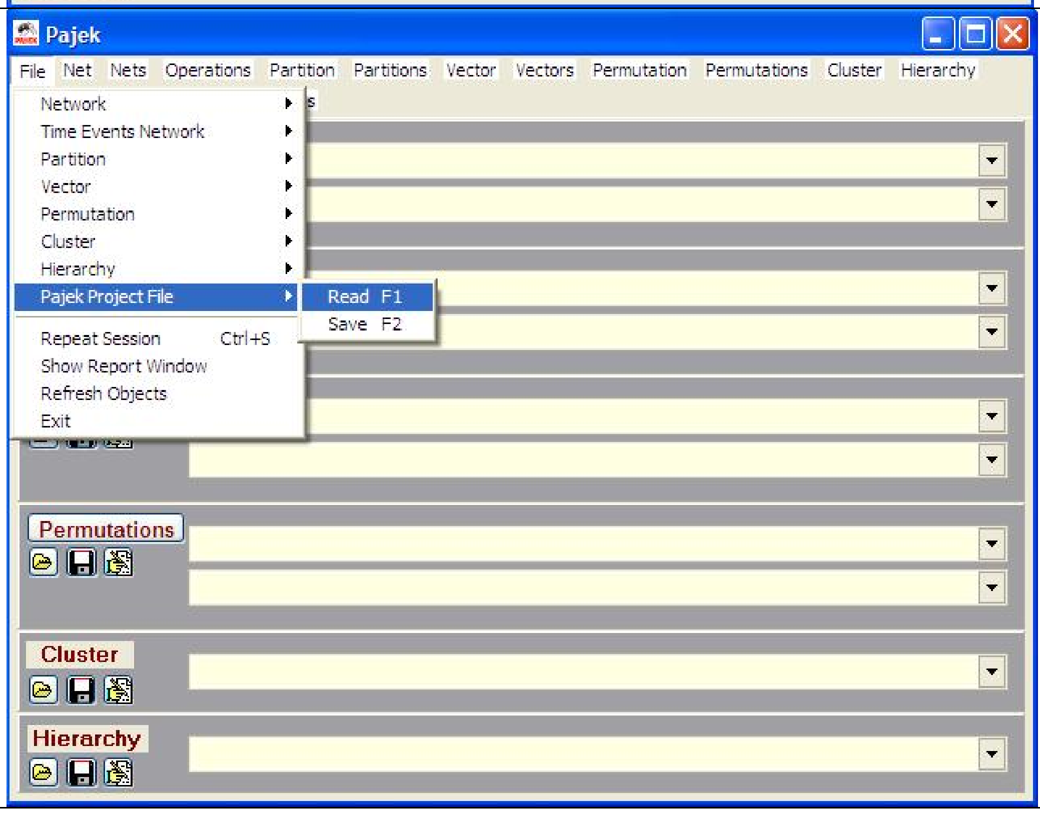
Pajek also benefits from having a comprehensive book providing good examples of how to use the software (de Nooy, W., Mrvar, A., & Batagelj. V. (2012) *Exploratory Social Network Analysis with Pajek*. Cambridge: Cambridge University Press. 2nd edition). This book is an excellent introduction to both Pajek and SNA more generally, providing an overview of each method described and working through examples which convey not only how to perform such analysis but also spells out the benefits of each technique.

The manual, however, is less helpful if you're unsure of how to use Pajek. It provides detailed information on Pajek but in a manner which assumes prior understanding of the operation. Therefore, it provides many useful resources for experienced users (such as the default colours for vertices and labels for triad censuses), providing in-depth knowledge of the finer points of the package, but the manual is more helpful for advancing your familiarity with the software.
Pajek frequently updates the software (usually fixing tiny glitches, adding new procedures or speeding up processes) so given the ease in installing it's often worth checking you're using the most up-to-date version before starting a piece of work. There is also a dedicated e-mail list which provides rapid answers to complex questions. A new development is the Pajek-XXL programme, which replicates Pajek but operates much faster on huge datasets (tens of thousands of nodes).

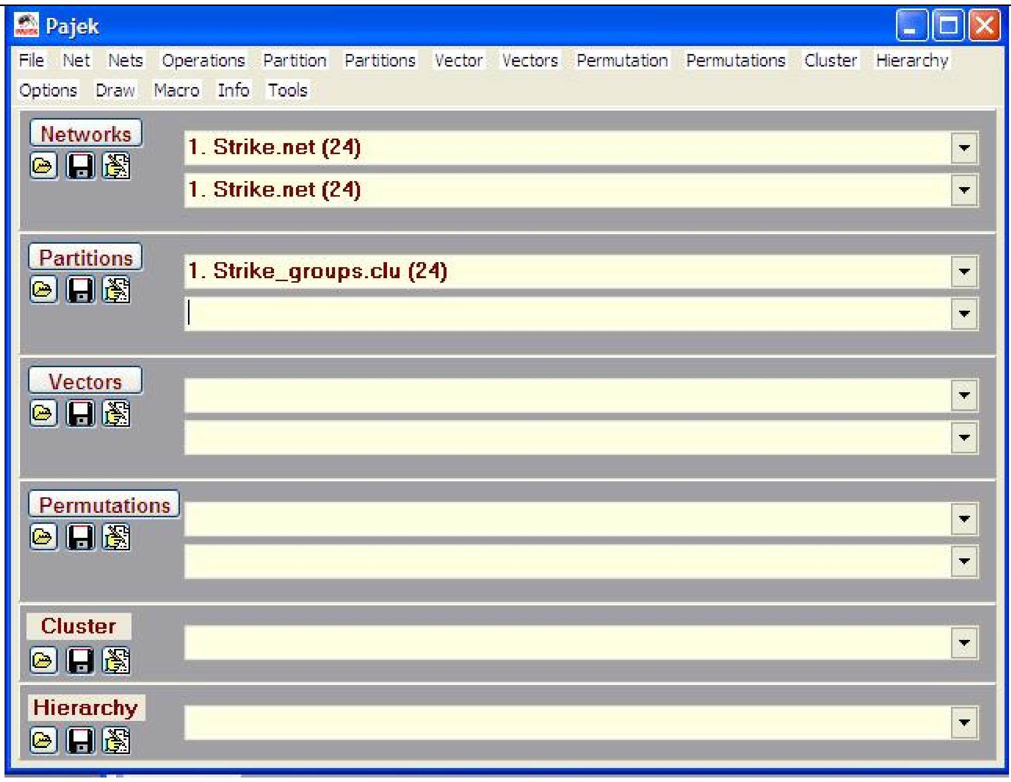
When you first open Pajek you need to import your data. You can import in individual networks one at a time using either the drop down menus, or the 'file open' button. You can then open any saved partitions, vectors and so on in the same way.



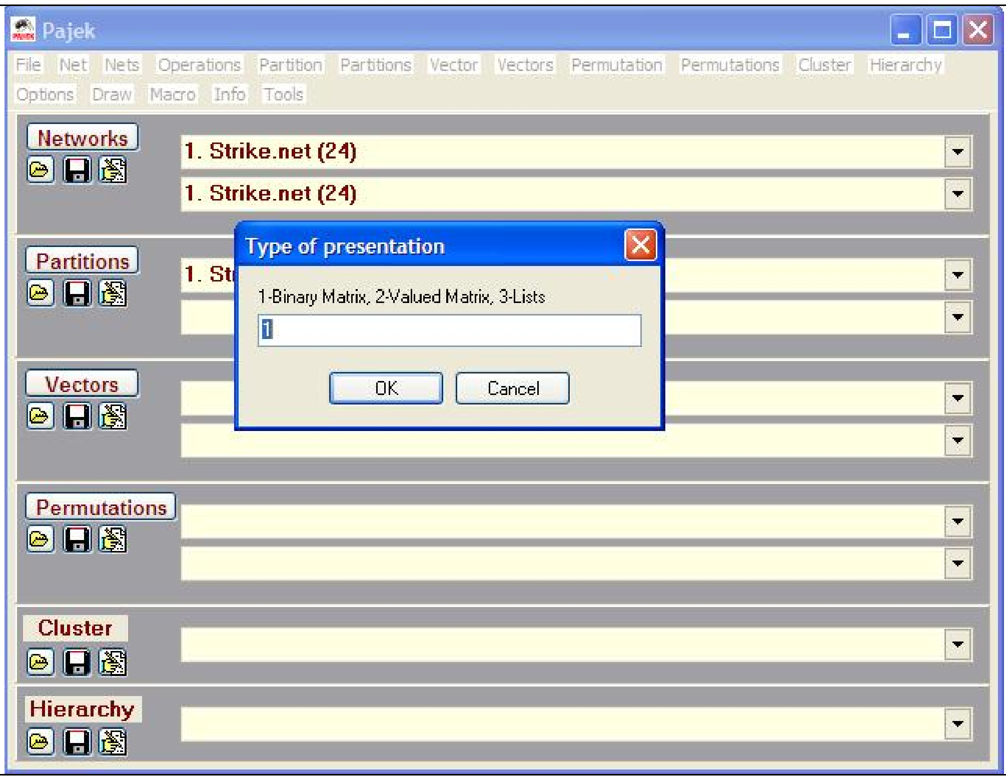
Alternatively, you could open a 'Pajek Project File' which imports all saved data relevant to the piece of work.



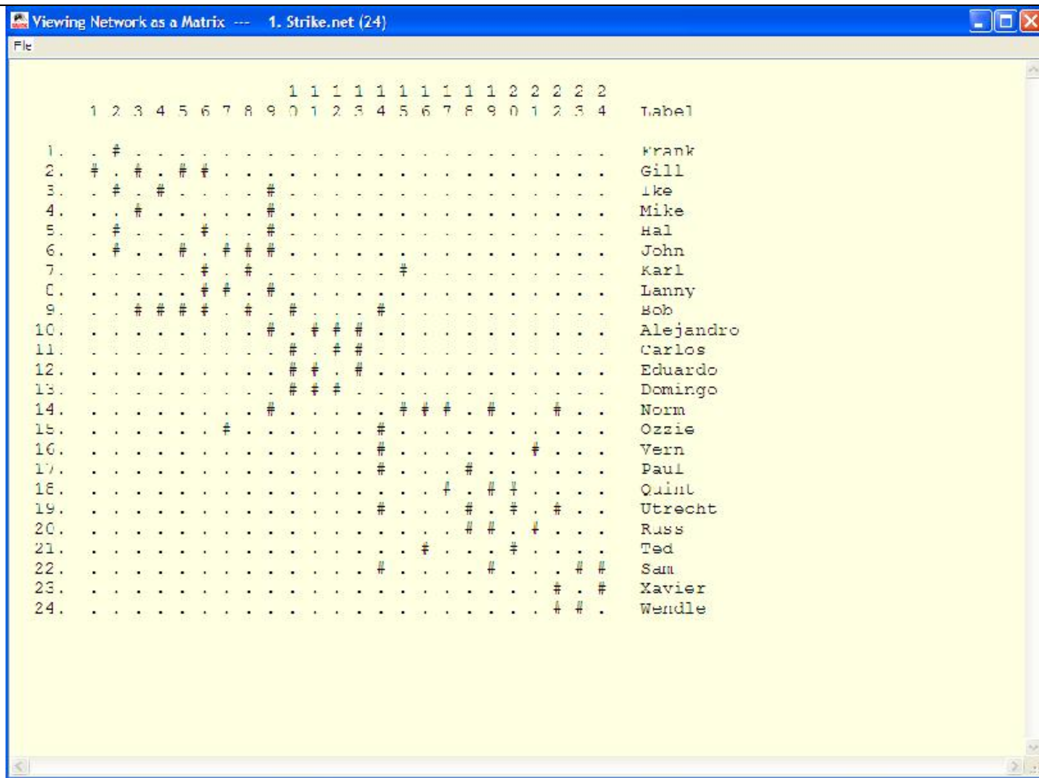
This provides the data useful for the research. We can simply add any new networks, partitions etc. to this file by simply opening additional individual data as before. However, opening a new ‘Pajek Project File’ will remove all the data from the package. Multiple files can be open at once in Pajek. The labels for each file have the structure for a number for the item in the drop list, the file name (or method of construction) and finally the number of cases in brackets.



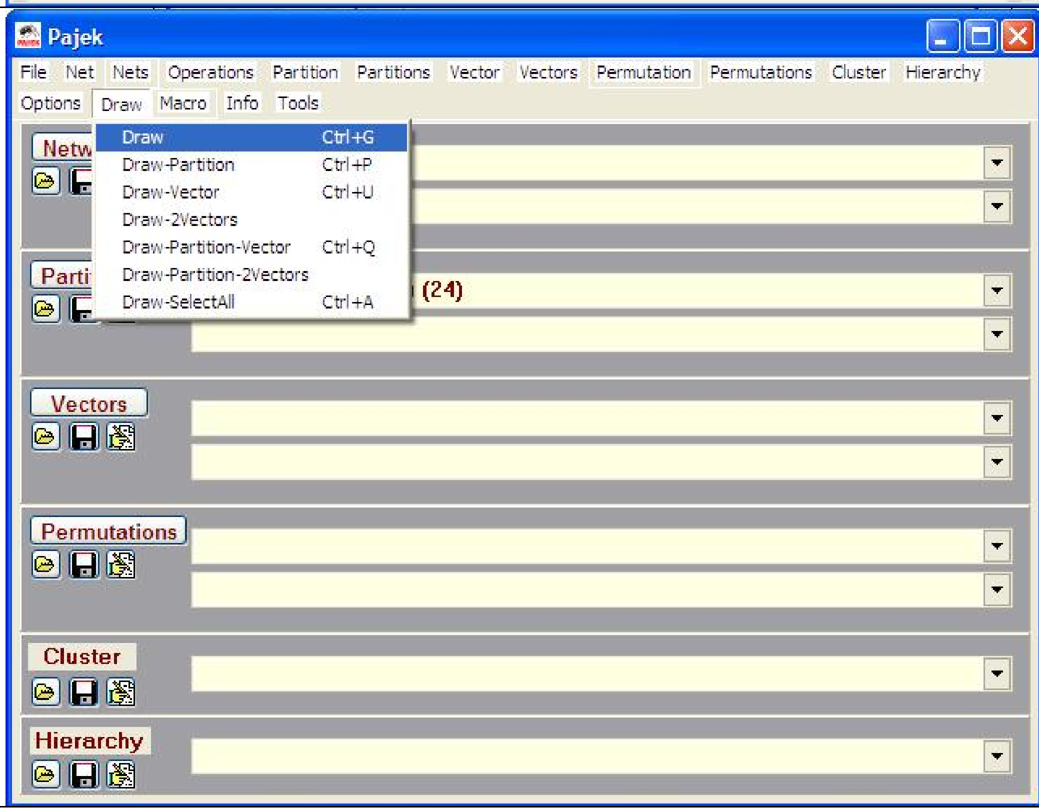
We can view the matrix which is providing all the information Pajek needs to operate. To do this, click on the actual name on the network in the yellow drop-drop section of the networks tab (the line marked ”1. Strike.net (24)” in this example). This produces a dialogue box enabling us to see whether a binary matrix (# marks a link), a valued list (showing the numeric value of the link) or a list (a list of the ties which are formed).

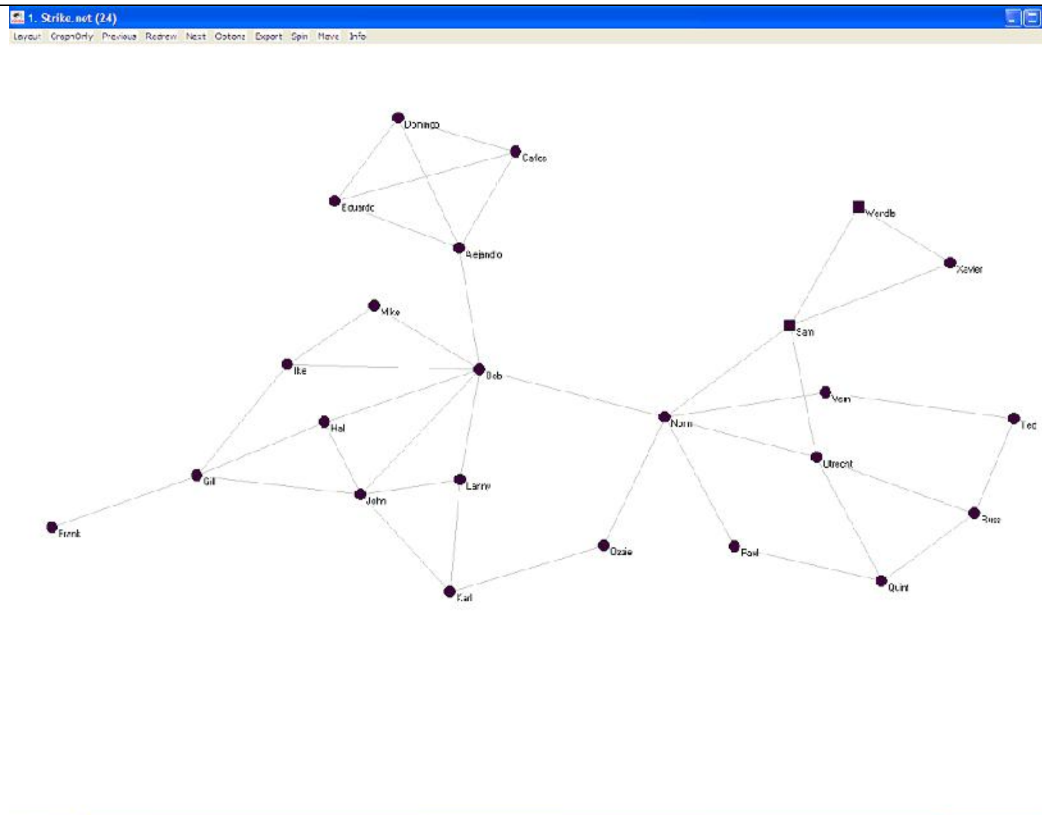
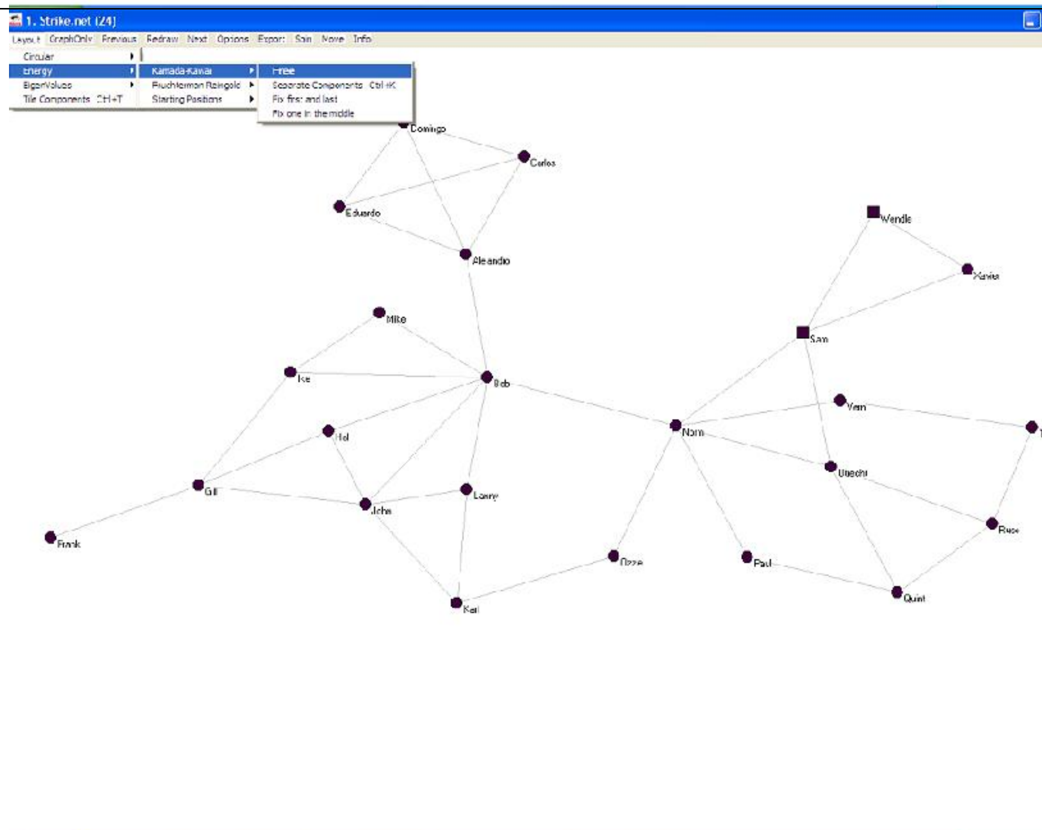


As this is non-numeric data, we have selected a binary matrix (1). This shows the presence of links between actors. The actors in the rows (which are labelled) send a link to the actors in the columns (which are in the same order). In this case, the links must be reciprocated (i.e., if A speaks to B, B speaks to A). This is not always the case (i.e., if A likes B, that doesn't necessarily mean B likes A).

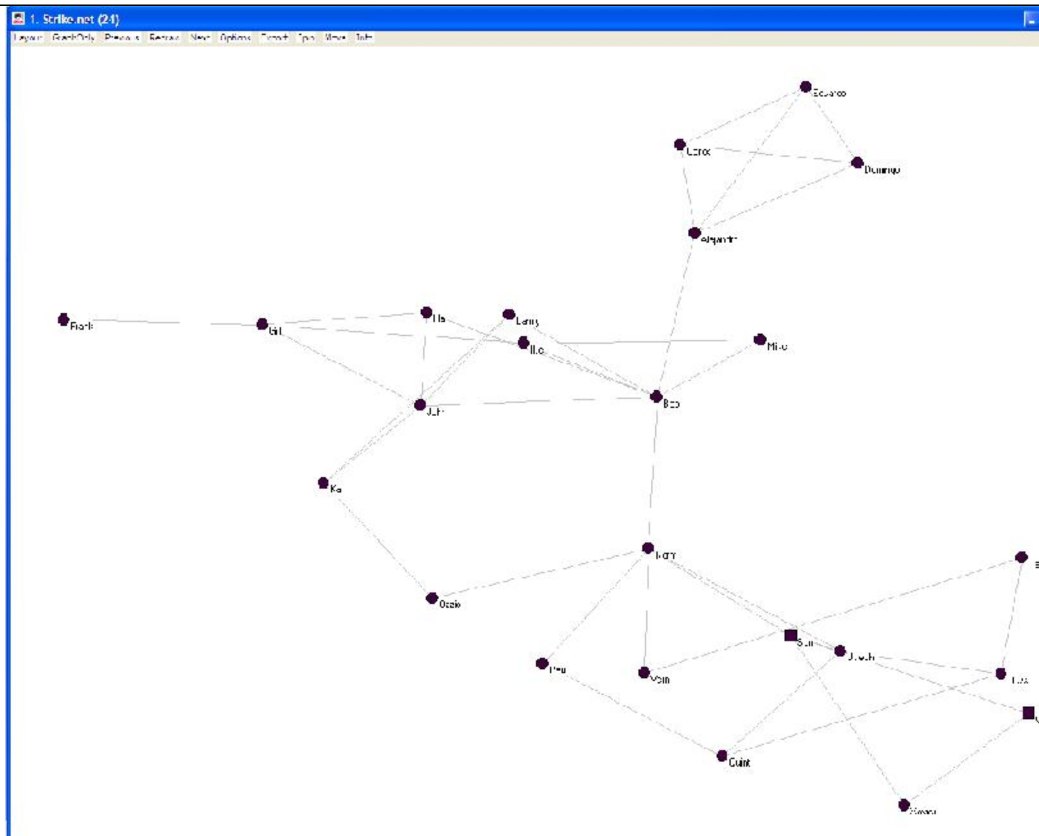


You can also visual the data as a network. Firstly, we will look at the basic structure of the network. This can be done through either using the 'DRAW' drop-down menu, or CTRL+G.

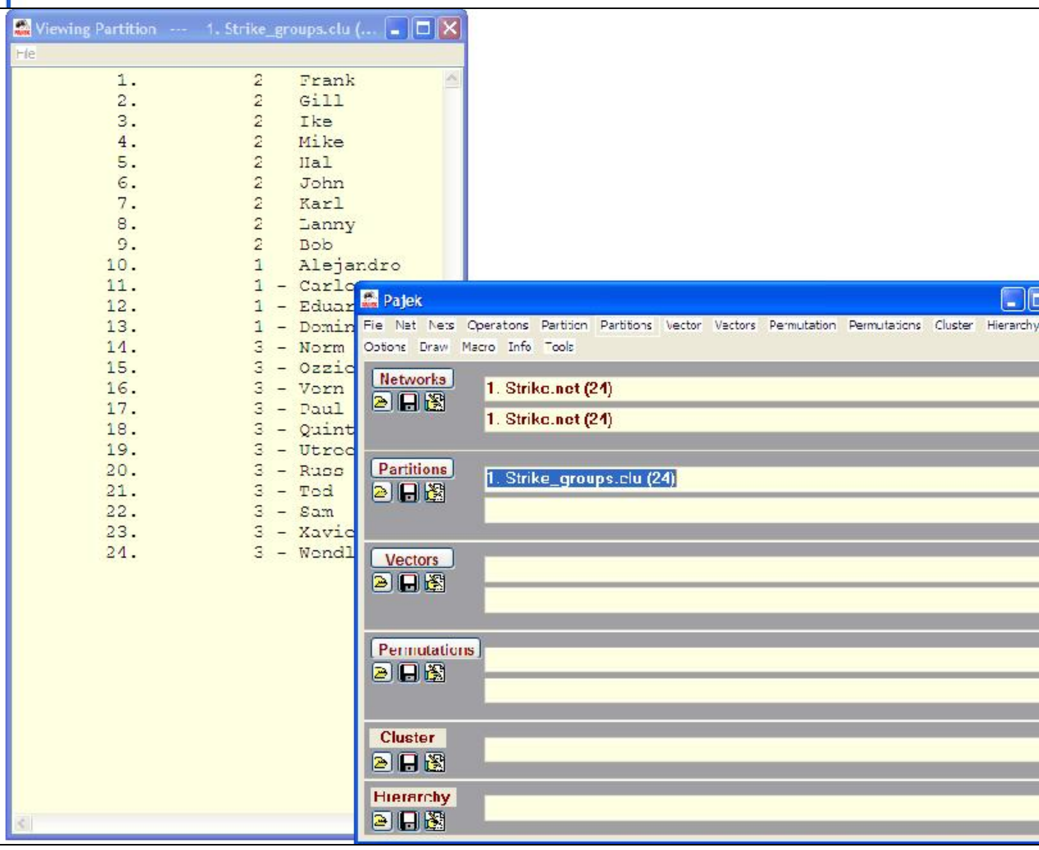


This produces a basic visualisation (in this case, who speaks with whom). Pajek places the nodes in what it believes is the most appropriate position. This is based upon a calculation which can be different each time. Sometimes the graph is displayed nicely the first time, as occurred for me in this example. The graph will be positioned slightly differently each time you generate it.	
Using the layout drop-down menu, you can use ‘energy’, then ‘Kamada-Kawai’, then ‘separate components’ (or CTRL+K) to rearrange the network using the same criteria as originally if you want a slightly different visualisation. The links will remain the same, but the nodes will be positioned slightly differently as it tries again to produce the most appropriate layout.	

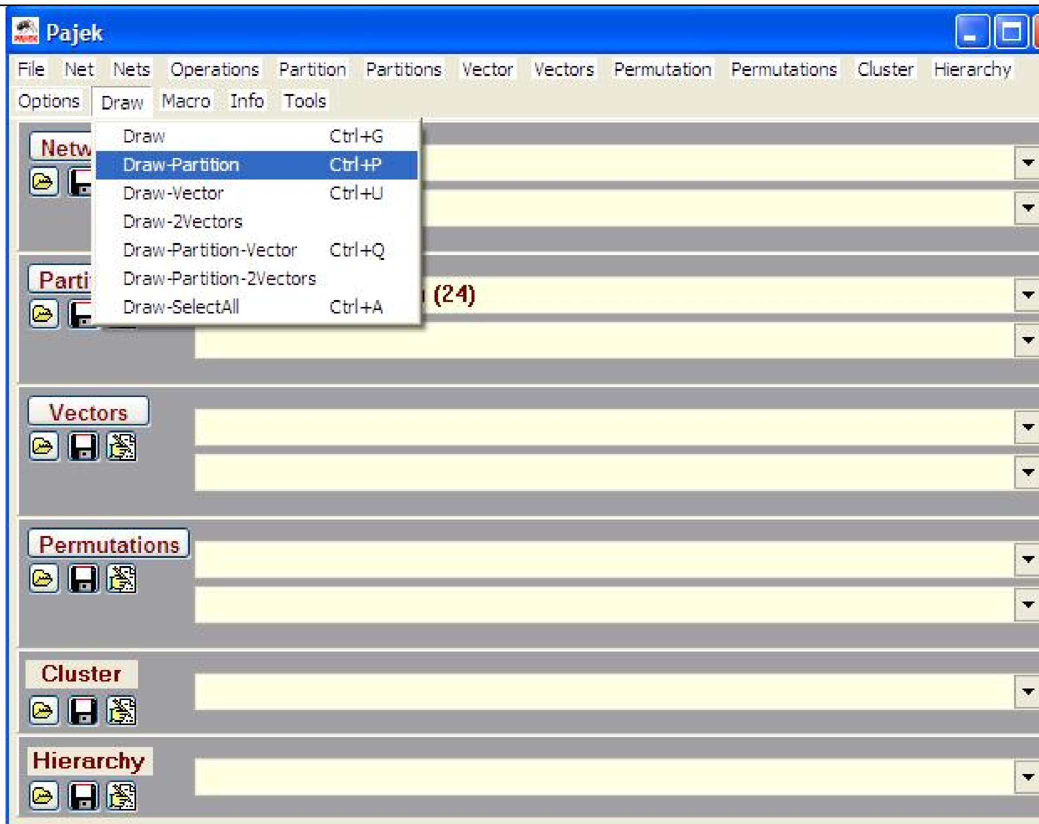
This is an example of how the network can look if repositioned using the Kamada-Kawai equation. Substantially there is no difference between the networks, but the first was spaced more nicely and made the links easier to read. Sometimes the layout is not optimal, therefore it is always useful to press CTRL+K a few times to see a few representations of the data.



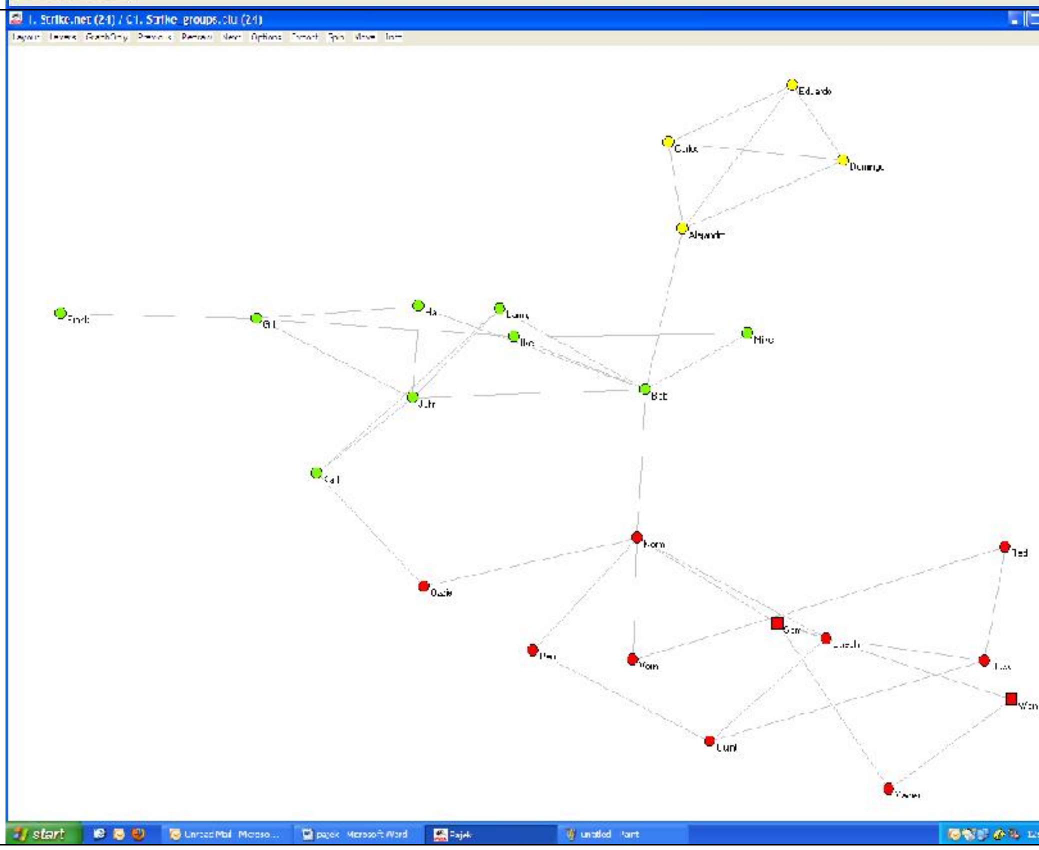
This data thus far has not distinguished between actors.. We might have some characteristics of the actors we wish to group them by. This project file contains a partition. Clicking on the name of the partition (highlighted on the Pajek window), brings up a list of the partitions and labels. We can see, therefore, these individuals are split into three groups.



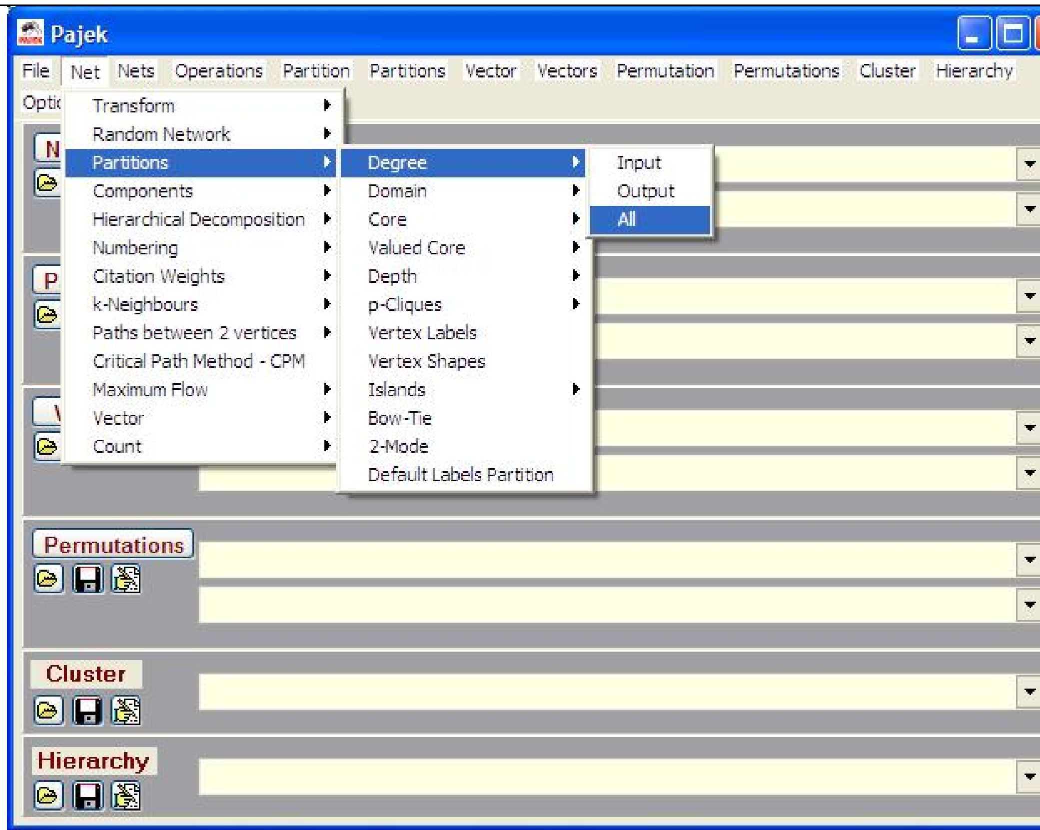
Therefore, we may wish to know how well the grouping within this partition corresponds with the network we've created. We can again go to the draw menu, but select 'draw-partition' (or CTRL+P). This shades the nodes dependent on their partition.



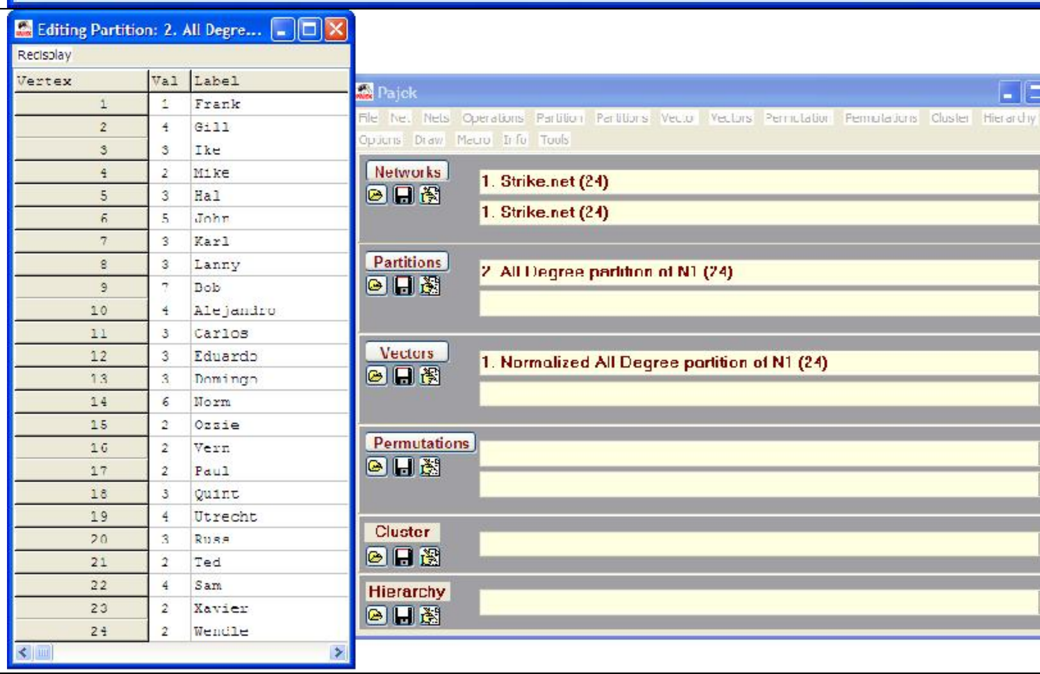
This graph shows we can explain very well the network positions by the characteristics used in creating the partition. In this example, the upper grouping are Spanish speakers, the middle grouping are younger English speakers and the lower group older English speakers. The hypothesis that age and language influences communication networks is clearly supported here.



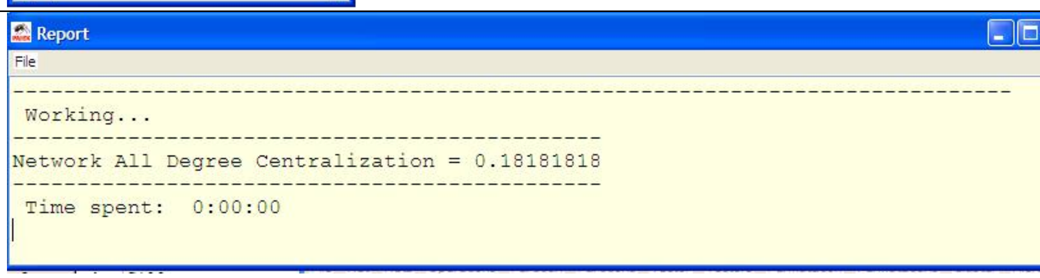
Visualisation is not all that Pajek can do. We might want to gather some statistics about our network. We can see who the most central actors are. We can do this by creating a new partition of the degree for each actor. *(Note: we have the option of input (incoming ties), output (outgoing ties) or all – as ties must be reciprocated within this network there are no differences, but if they were not reciprocated by design the choice would be important.)*

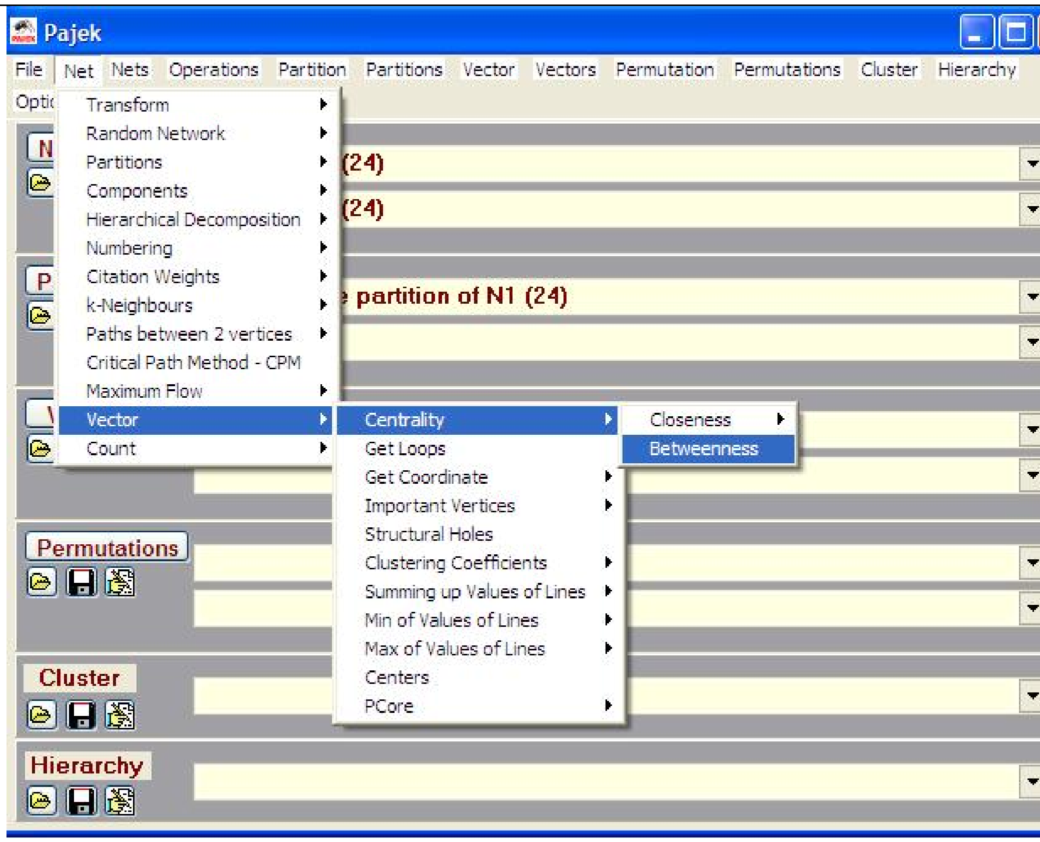
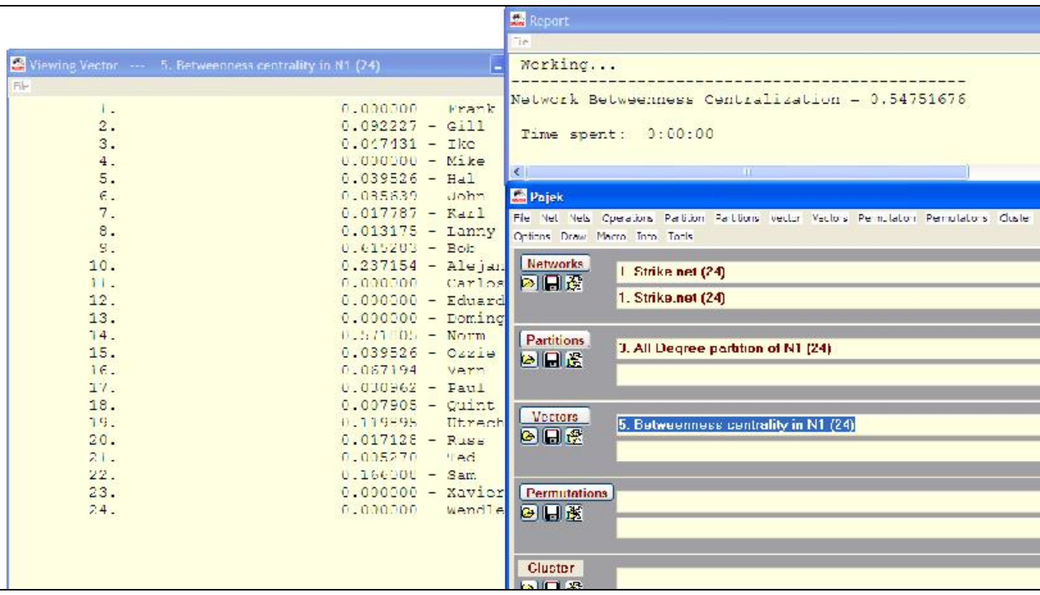


This creates a new partition. Clicking on the name of the new partition (2. All Degree partition of N1 (24)) enables us to see the number of ties each actor has. If we are interested in degree centrality (the number of people each individual speaks to in this case) we can get the information here.

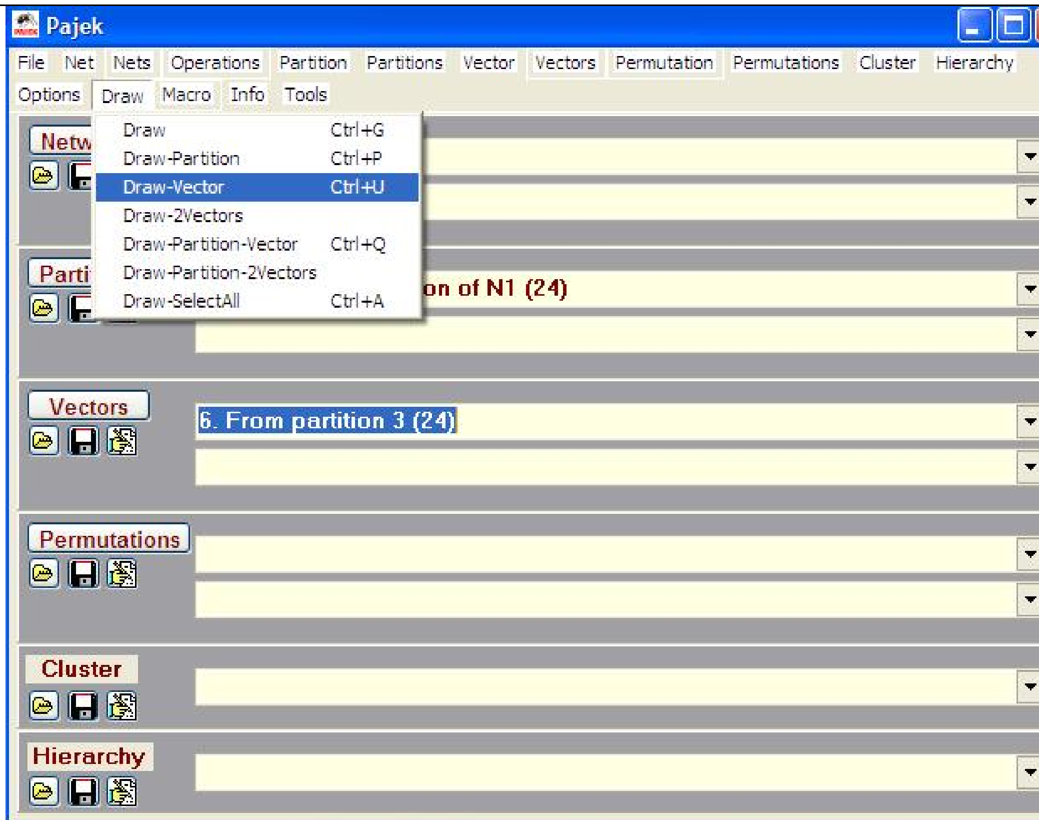


Also created is a report window. This explains what we have done, but also gives us the degree centralization of the network (i.e., the percentage of possible ties which were created). In this case it is 0.18, showing that

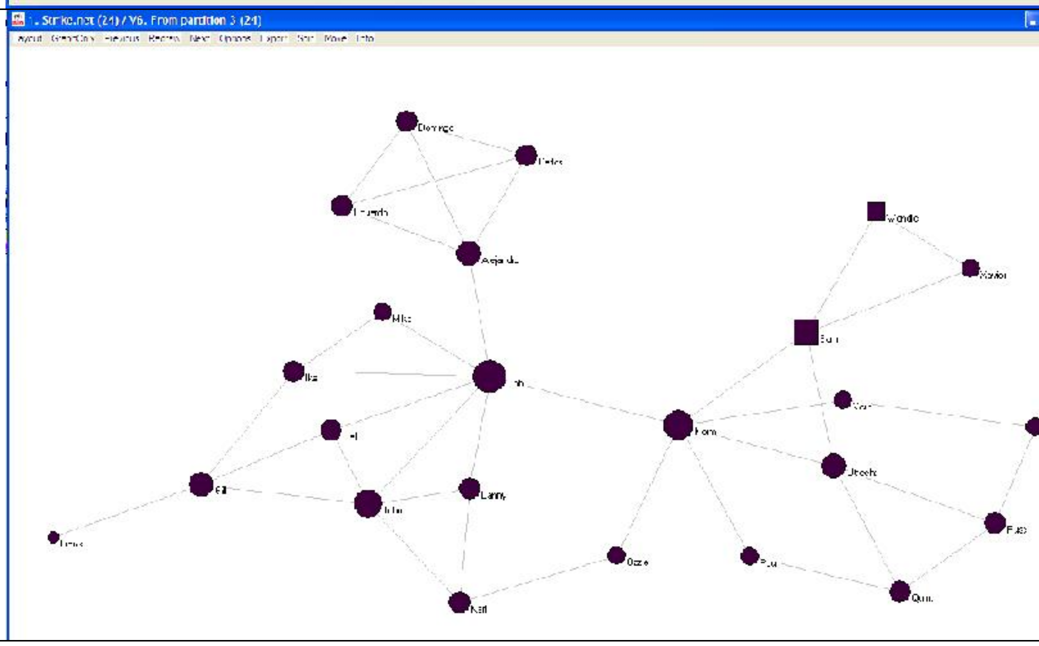


18% of all possible connections (i.e., where everyone speaks to everyone else) have been formed.	
We might be interested in other forms of centrality. Betweenness centrality measures how often an actor is part of a path between two other actors (i.e., where A and B can only communicate though C, whether directly or through others). The higher the betweenness centrality, the increased opportunity for actors to influence and control information which flows around a network. This can be found from the 'net', 'centrality', 'betweenneess' drop down menu.	
Again, the report window gives us a score for the overall network (.548) which can be compared to other similar networks. Clicking on the highlighted vector enables us to see the scores for each individual. Frank, in the first row, has a score of 0 as he never connects people. Bob (9) has the highest value of 0.61.	

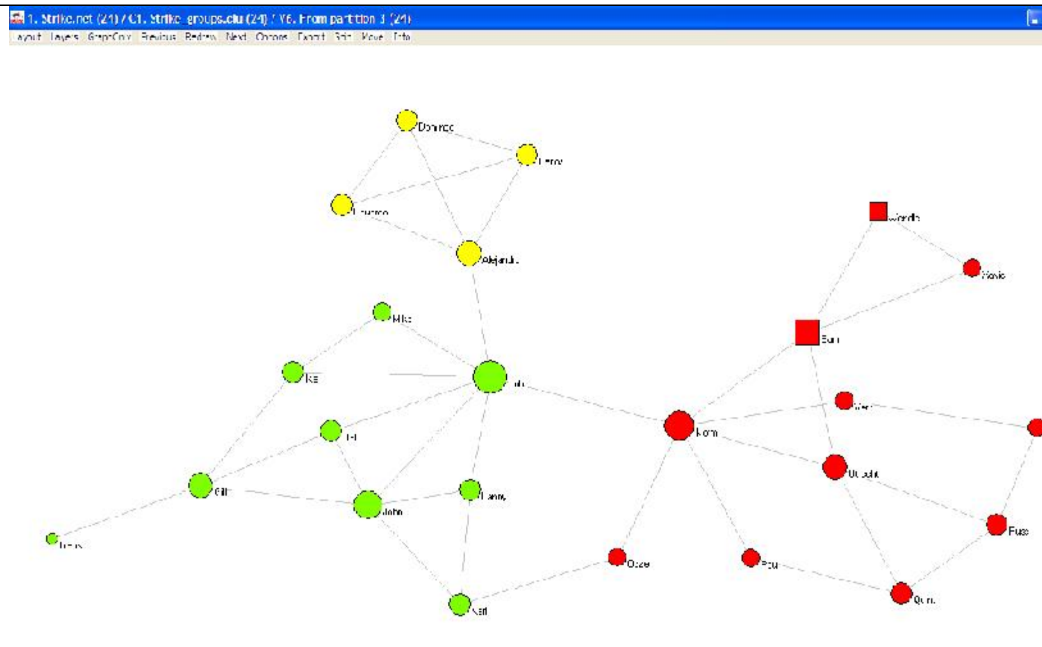
We might want to visualise this to understand the importance of betweenness in terms of the network. We can do this as we did with the partitions, but selecting ‘Draw-Vector’ (or CTRL+U).



Draw-Vector sizes the node by the scale of the vector. This demonstrates that, for instance, Gill has a reasonable level of influence despite being on the left of the diagram away from the more central actors. This is because they alone can communicate to Frank and manipulate his opinions to the group.

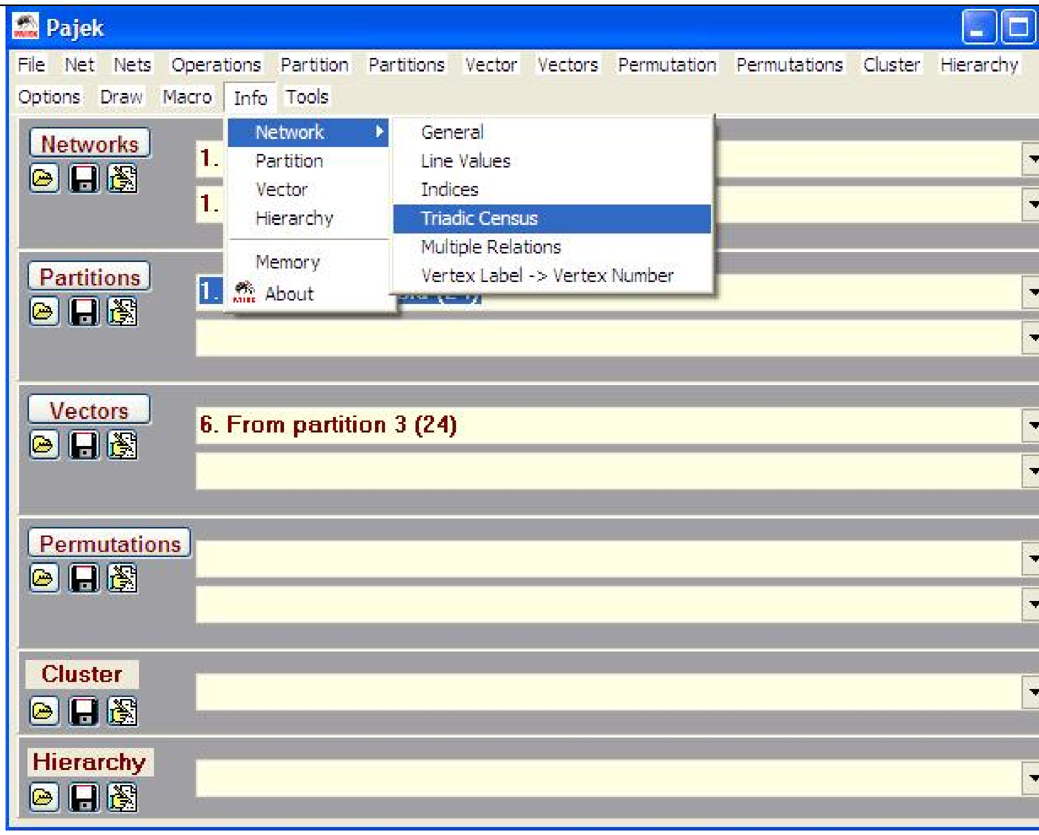


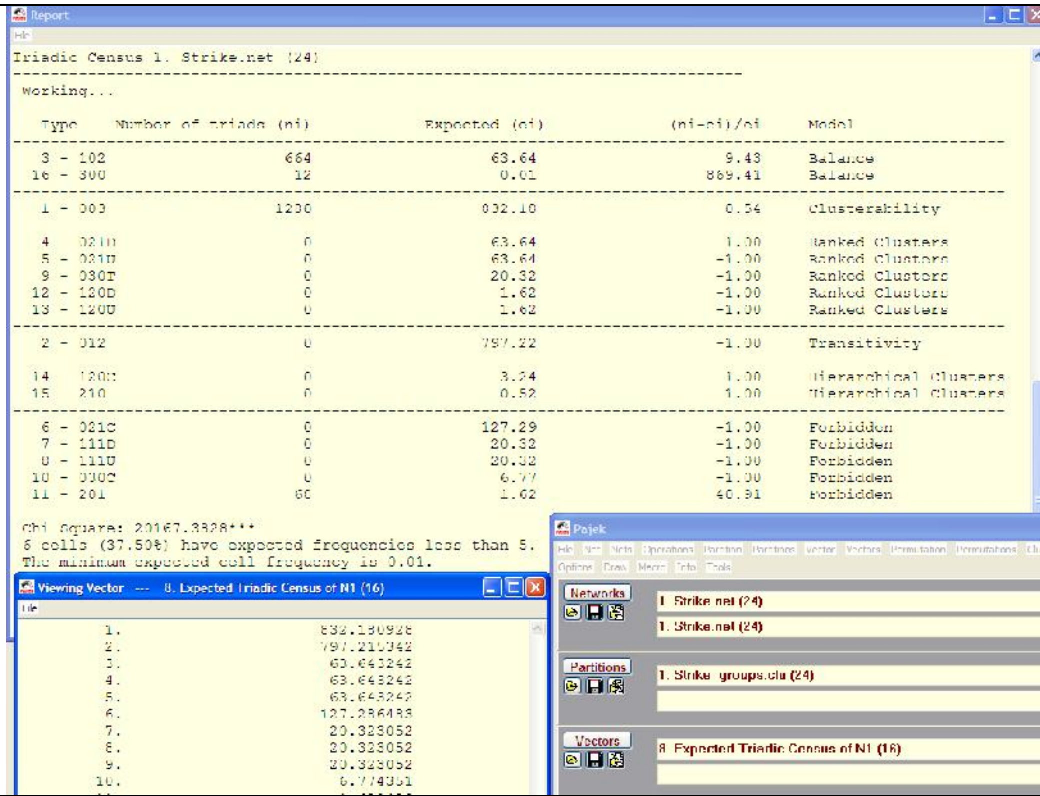
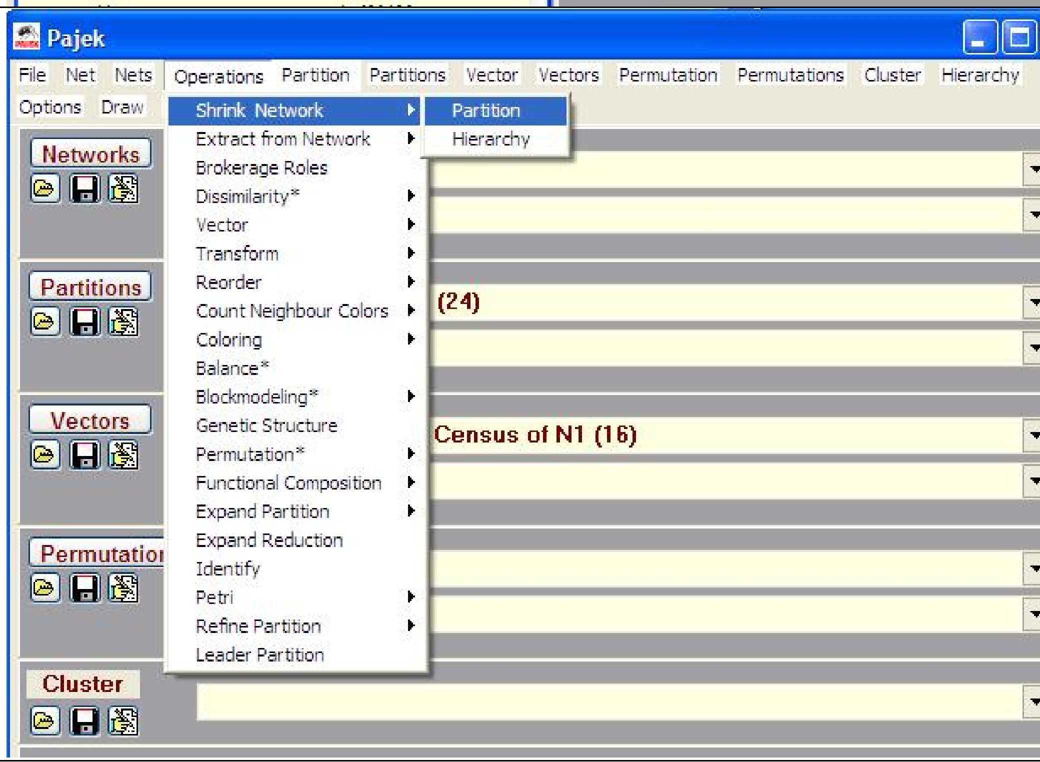
We can mix together different ways of visualising the data. We could look at both partition (in this instance the three groups) and a vector (in this case betweenness) to see how the groups differ. Whilst the upper group of Spanish speakers appears marginalised, using betweenness values we can see they have as much potential influence as the other groups.



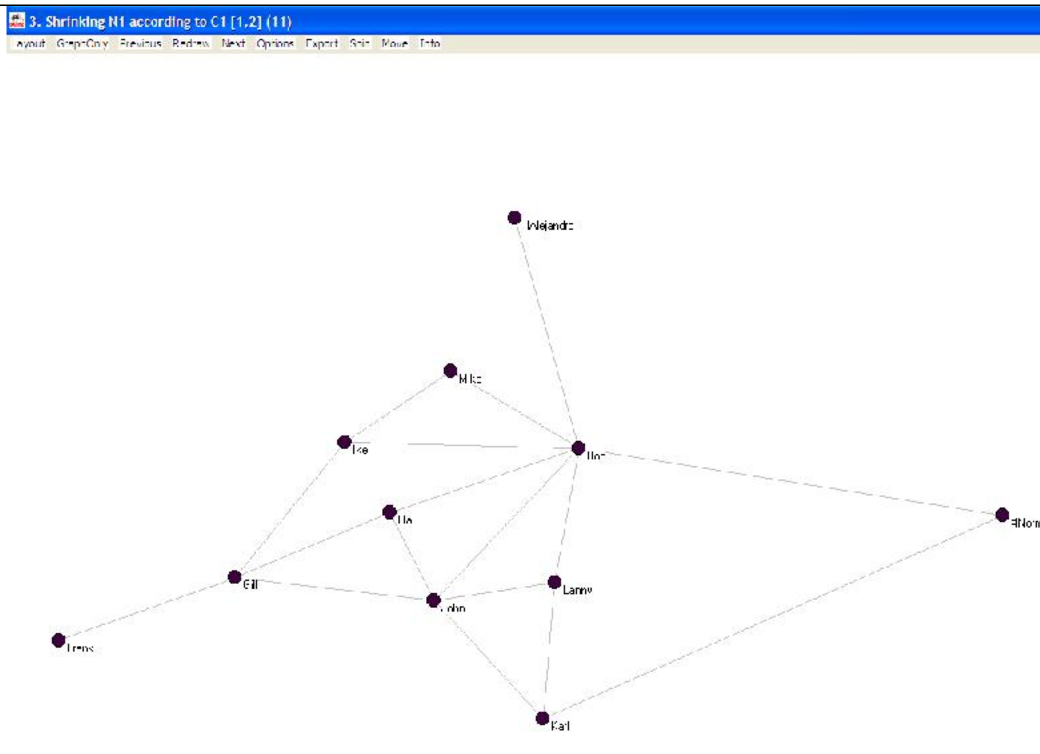
We can look at other elements of the network. We might be interested in a triadic census of the data. This can be performed using the info, network, triadic census drop down menu.

A triadic census takes every possible combination of three actors and looks at the structures between them.

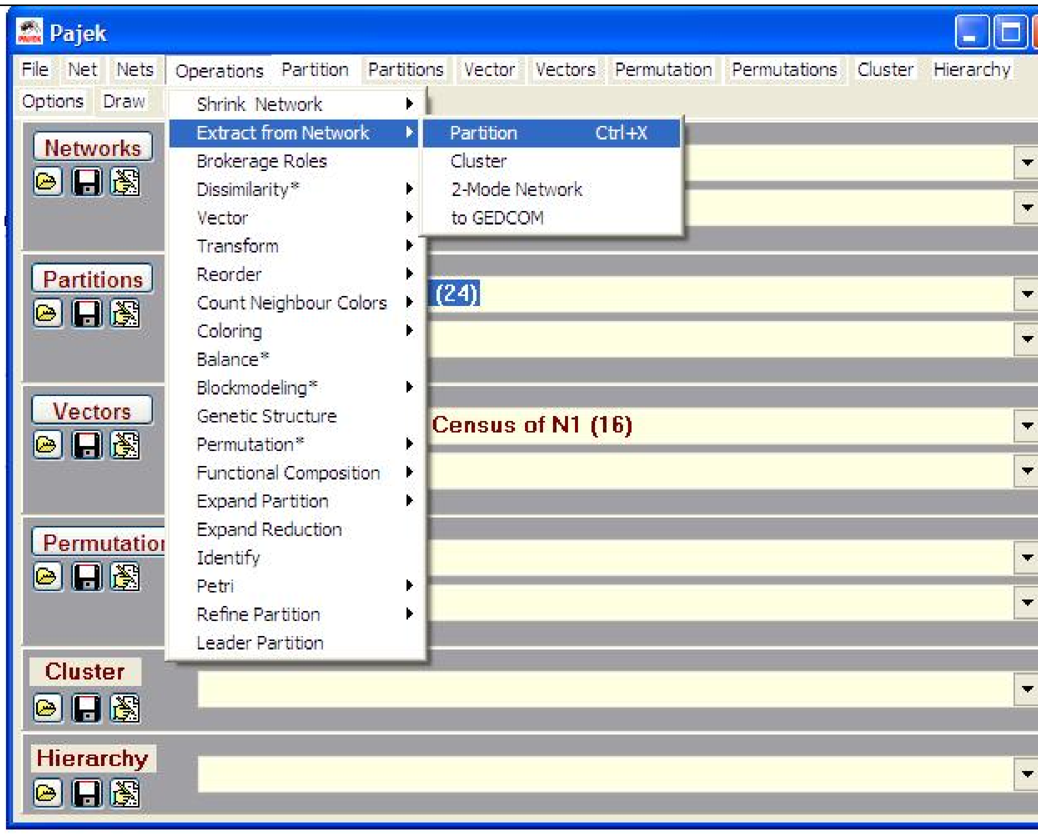


This produces a report of the number of each triad we have observed and the expected number given the number of actors and links. The triads are also labelled according to what they show. A chi-square test is conducted, which in this case shows a difference from what would be expected. There is a large percentage of triads which all connect to each other, showing there is balance within the network (i.e., my friends' friend is my friend).	
We can also perform analysis grouping together particular clusters. For instance, we can shrink the network into partition. This produces a dialogue box asking for the minimum number of ties to connect to another cluster (default of 1), before asking which partition(s) should not be shrunk. Once selected, it reduces all other clusters to a single actor (assuming the analysed cluster hold sufficient number of links to its members).	

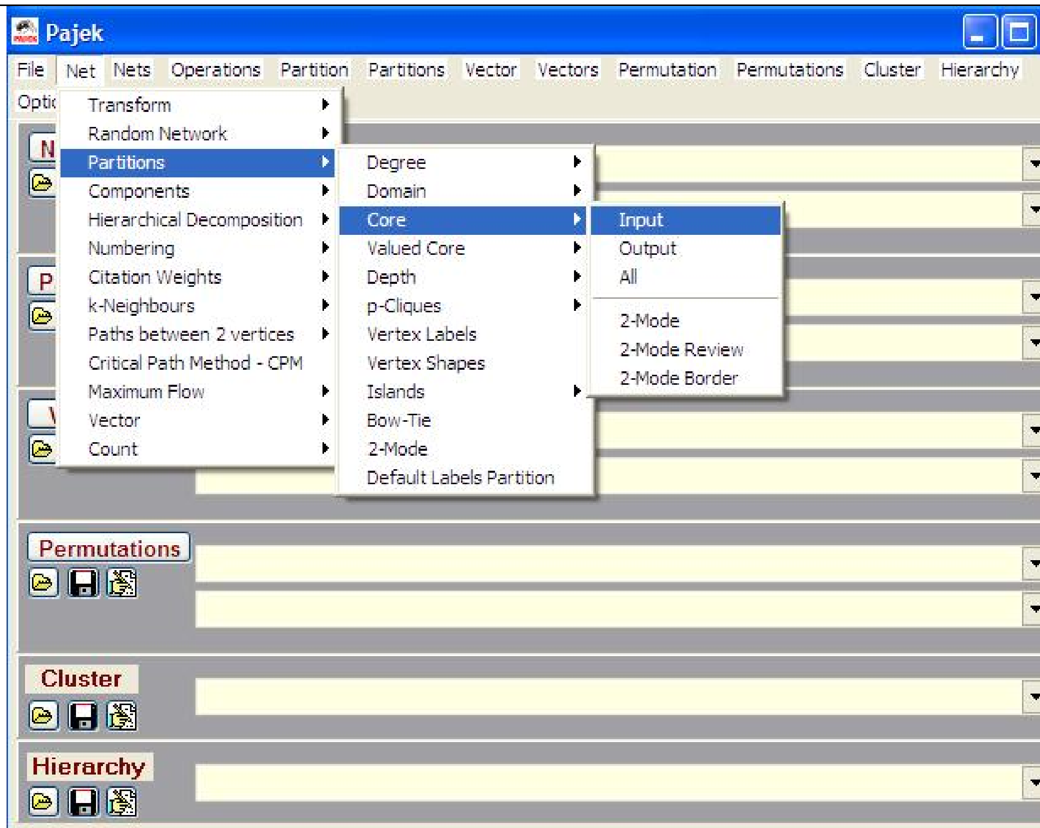
Here the network is shown, but only for the younger English cohort. The older English cohort have been shrunk into one group, and the Spanish cohort into another (labelled by #). We can now rerun any of our analysis (such as centrality methods) viewing the other clusters as sub-groups rather than their individual actors.



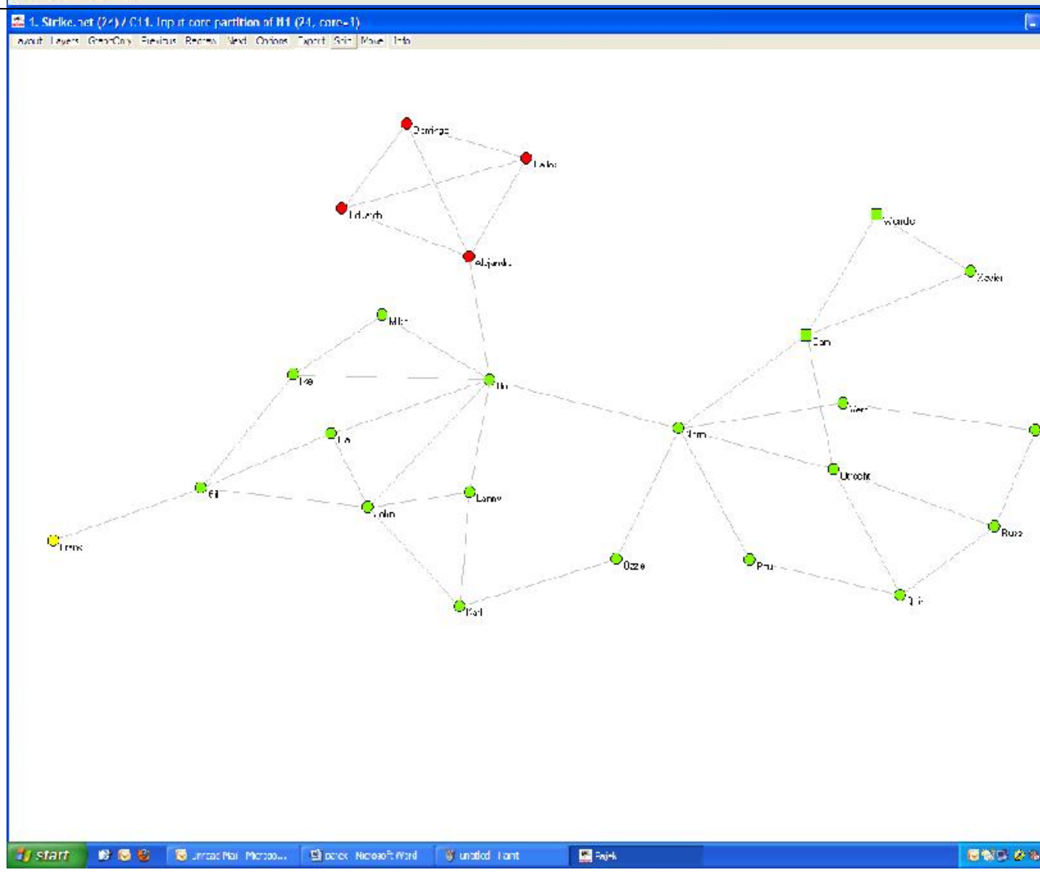
Alternatively, we can just decide to extract a particular partition from the network (by drop down menu or CTRL+X) which enables us to perform analysis irrespective of the wider network the actors are part of. (Note: this is particular useful if you have, for instance, international trade networks and you wish to look at Europe in isolation, or European countries by the other European countries and continents they trade with).



We could also look at the whole network and decide to look at the cores. This produces sub-graphs whereby each actor is connected to X actors who all have at least Y links. The core for each actor is the largest possible Y that they can be part of a core of.

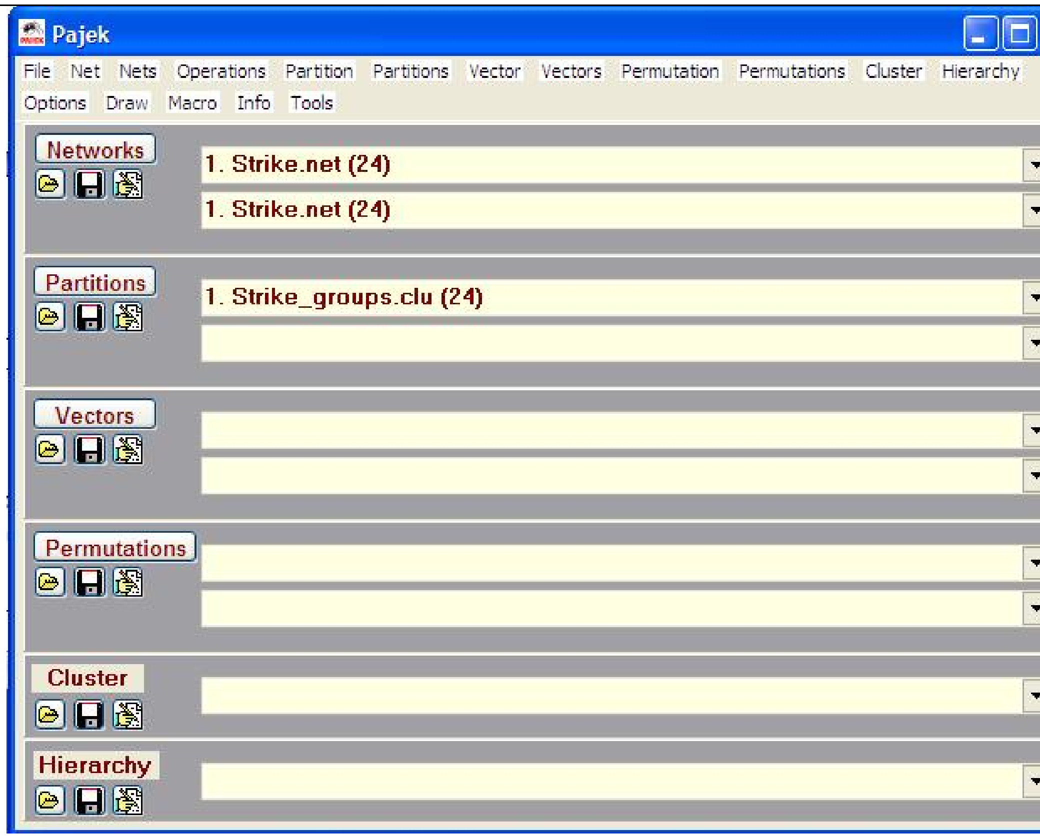


Therefore, we see that Frank can only be incorporated in a core measured by one link. The English speakers, Frank aside, all form part of the two core. The three Spanish speakers form a three-core. Whilst Karl has three connections, Ozzy does not. Therefore, removing Ozzy from the 3-core removes Karl, which in turn removes Lanny. Losing Mike loses Ike, which removes Gill and quickly the English structure falls apart. Therefore, this is ideal for spotting well-constructed elements of networks and identify cores (even if of non-central actors).

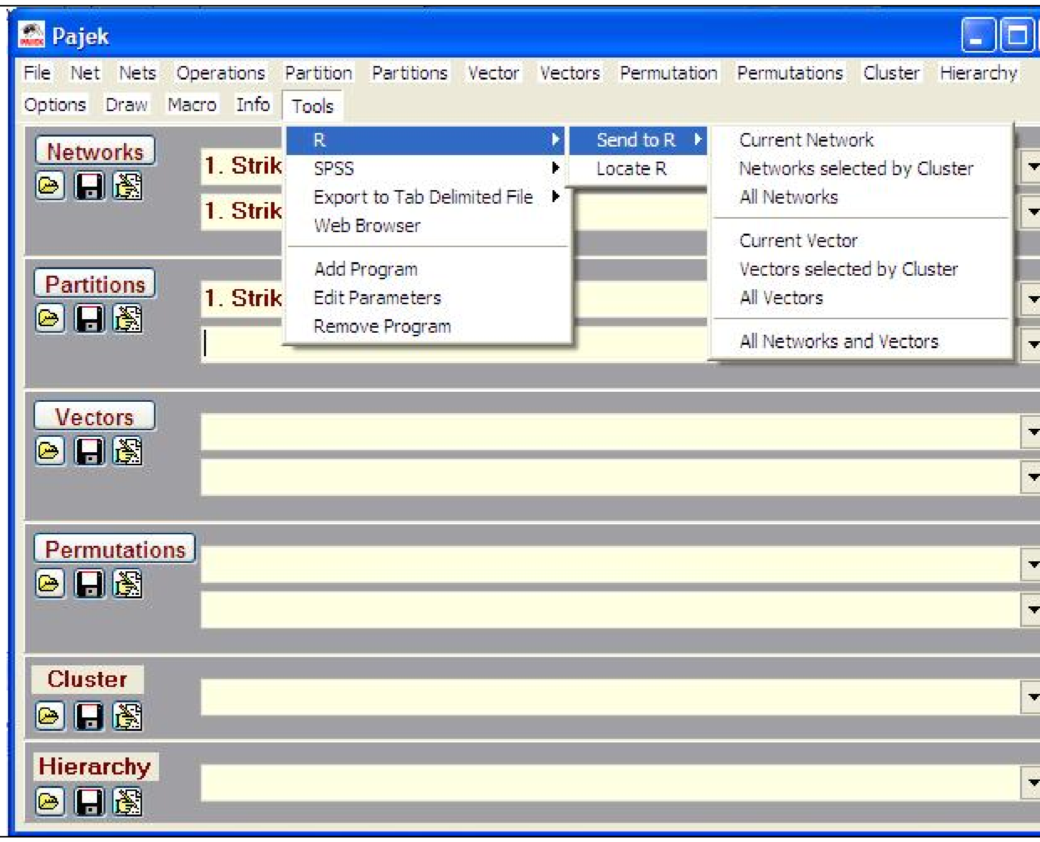


Saving data in Pajek involves using the file buttons under each tab (i.e., networks, partitions etc.), or the 'file' drop down menu. Each file you've used needs to be saved separately. You can move between them using the right-hand arrows. Alternatively, use 'file, Pajek project file, save' to save everything in one large file.

Note: Pajek keeps every file you've opened in its list which can become large if working on multiple tasks. Therefore, it is often beneficial to close and reopen the window to avoid confusion.



Pajek also has the options to export data directly to R and SPSS.



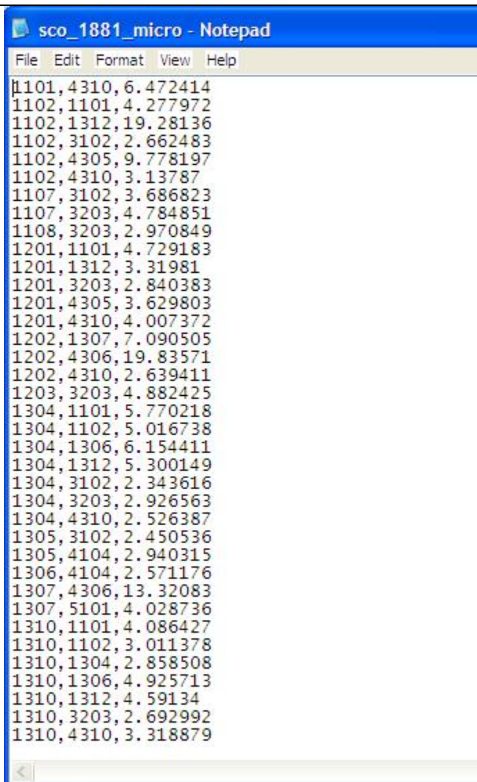
Preparing data for Pajek

The Pajek website (http://pajek.imfm.si) offers a range of useful datasets for exploring network theory. It also offers the Excel2Pajek and Text2Pajek tools for formatting data. Converting a dataset into Pajek format is simple.

Firstly, produce a dataset which has two columns of nodes, for instance, male and female occupations within married couples. A value for strength of line can also be saved. This can be produced in any format.

	hocc	wocc	val_min
1.	1101	4310	5.770497
2.	1102	1101	3.565027
3.	1102	1312	17.26048
4.	1102	3102	2.306733
5.	1102	4305	8.565514
6.	1102	4310	2.706895
7.	1107	3102	3.024689
8.	1107	3203	4.27487
9.	1108	3203	2.468718
10.	1201	1101	4.309639
11.	1201	1312	2.850363
12.	1201	3203	2.701401
13.	1201	4305	3.21621
14.	1201	4310	3.734819
15.	1202	1307	5.940358
16.	1202	4306	19.30663
17.	1202	4310	2.422515
18.	1203	3203	4.090449
19.	1304	1101	5.060044
20.	1304	1102	4.12995

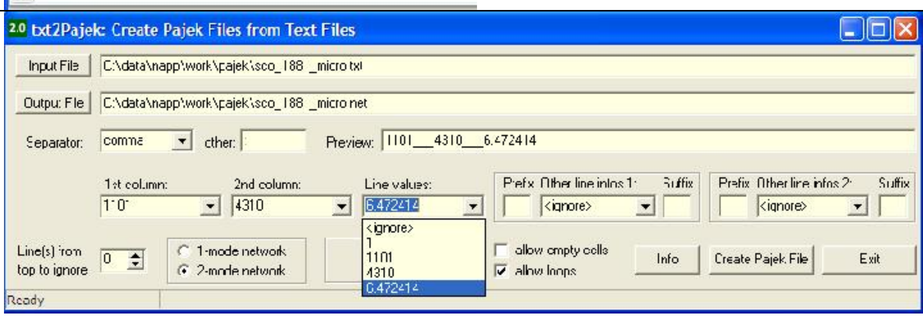
To use Text2Pajek, save the list as a text file. In this example it's a comma-separated version. Just the rows of data are required, with no additional information.

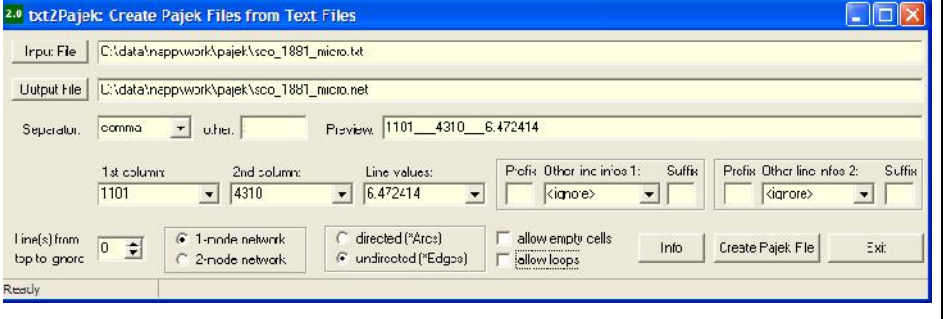

sco_1881_micro - Notepad

File Edit Format View Help

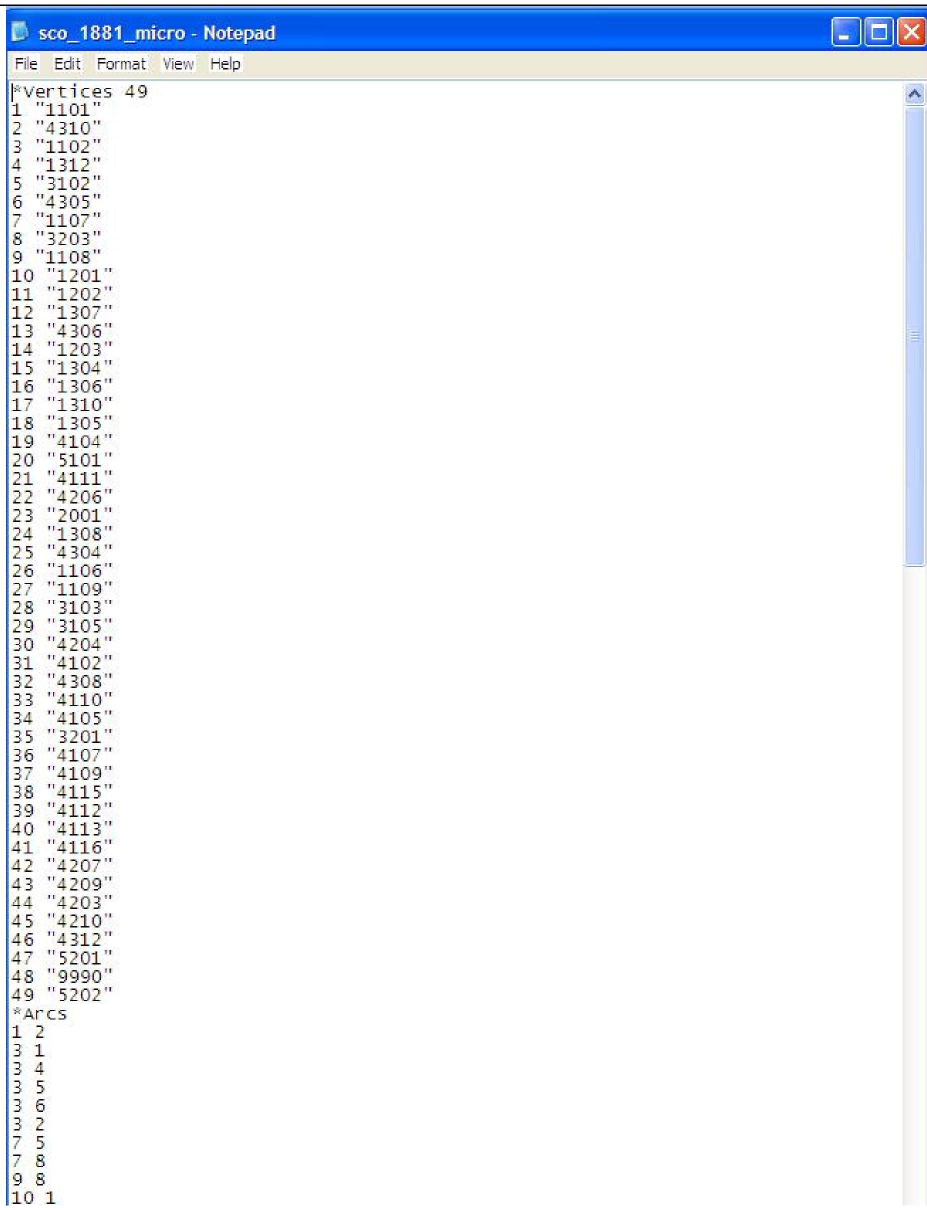
```
1101,4310,6.472414
1102,1101,4.277972
1102,1312,19.28136
1102,3102,2.662483
1102,4305,9.778197
1102,4310,3.13787
1107,3102,3.686823
1107,3203,4.784851
1108,3203,2.970849
1201,1101,4.729183
1201,1312,3.31981
1201,3203,2.840383
1201,4305,3.629803
1201,4310,4.007372
1202,1307,7.090505
1202,4306,19.83571
1202,4310,2.639411
1203,3203,4.882425
1304,1101,5.770218
1304,1102,5.016738
1304,1306,6.154411
1304,1312,5.300149
1304,3102,2.343616
1304,3203,2.926563
1304,4310,2.526387
1305,3102,2.450536
1305,4104,2.940315
1306,4104,2.571176
1307,4306,13.32083
1307,5101,4.028736
1310,1101,4.086427
1310,1102,3.011378
1310,1304,2.858508
1310,1306,4.925713
1310,1312,4.59134
1310,3203,2.692992
1310,4310,3.318879
```

We can then open the txt2Pajek software. Firstly, we use the 'input file' button to select the file. It then defaults to save the file in the same folder with the same name but



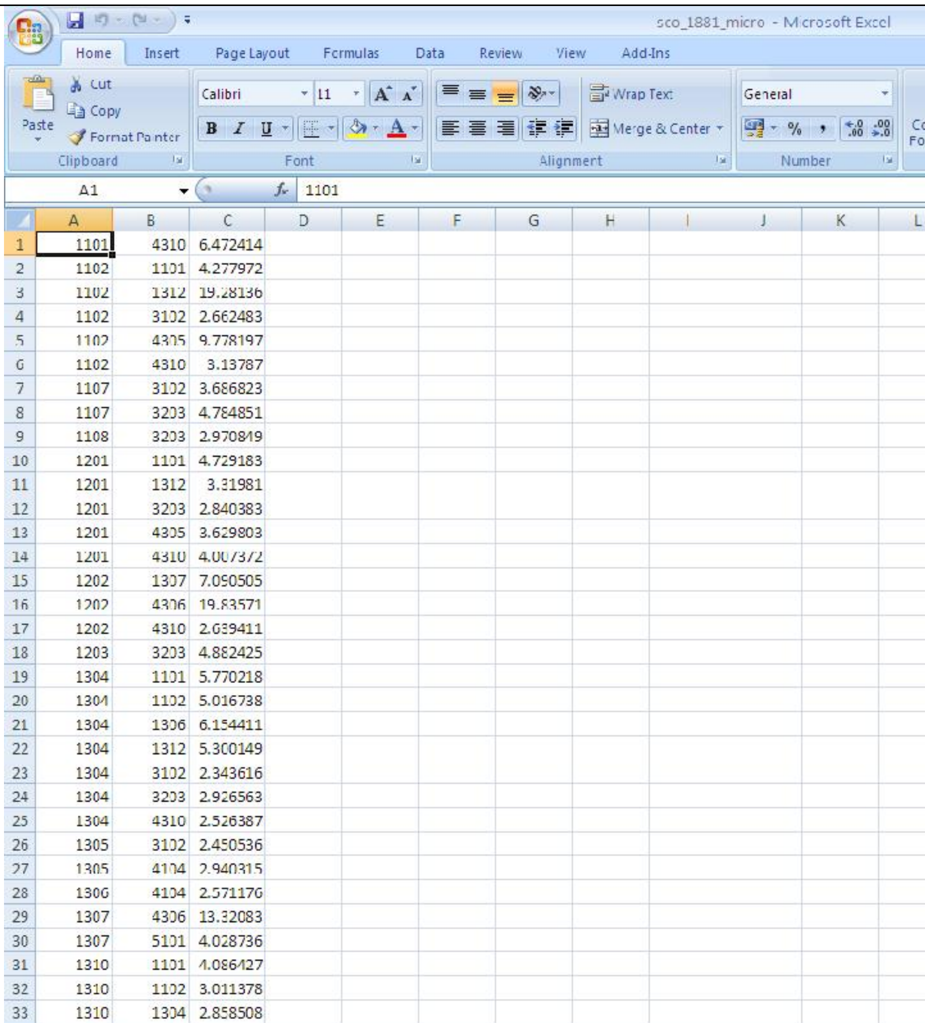
different suffix (clicking on the output name allows you to change it). We can specify the separator (comma) and select the 1st and 2nd columns from the available columns. We can include or ignore line values.	
We can specify if it is a one-mode (the same actors in both columns, such as jobs) or two-mode (different actors in each column, such as employees-employers) and whether the network is directed (i.e., if ties are assumed to always be replicated). We then click on 'Create Pajek File'.	

This produces a .net file which we need for Pajek. It starts off by specifying how many vertices exist, then giving a number to each label (as we saw above with the matrix). It also produces a list of the arcs/edges (links, whether directed or undirected), just showing the two numbers which are connected. In this example the labels are the occupational numbers. The data could be exported with the labels instead, which might in some cases be beneficial if they are to be shown in the graphs (obviously, with limitations on the size of what will be readable).



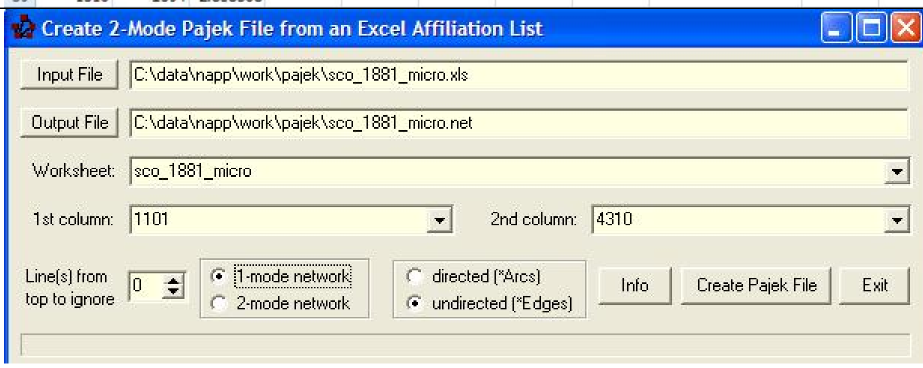

```
*Vertices 49
1 "1101"
2 "4310"
3 "1102"
4 "1312"
5 "3102"
6 "4305"
7 "1107"
8 "3203"
9 "1108"
10 "1201"
11 "1202"
12 "1307"
13 "4306"
14 "1203"
15 "1304"
16 "1306"
17 "1310"
18 "1305"
19 "4104"
20 "5101"
21 "4111"
22 "4206"
23 "2001"
24 "1308"
25 "4304"
26 "1106"
27 "1109"
28 "3103"
29 "3105"
30 "4204"
31 "4102"
32 "4308"
33 "4110"
34 "4105"
35 "3201"
36 "4107"
37 "4109"
38 "4115"
39 "4112"
40 "4113"
41 "4116"
42 "4207"
43 "4209"
44 "4203"
45 "4210"
46 "4312"
47 "5201"
48 "9990"
49 "5202"
*Arcs
1 2
3 1
3 4
3 5
3 6
3 2
7 5
7 8
9 8
10 1
```

Using the Excel2Pajek tool is just as simple. Start off with the data saved as an Excel file. Again, how two columns showing the linkages which form part of the network. You have the option to have a third variable for the strength of the line. It does not matter if there is additional information stored in the Excel file, as you will select the columns you wish. Therefore, it would be possible to have both the occupational codes and labels in the same file, which you could export as two different networks (if required).



	A	B	C
1	1101	4310	6.472414
2	1102	1101	4.277972
3	1102	1312	19.28136
4	1102	3102	2.662483
5	1102	4305	9.778197
6	1102	4310	3.13787
7	1107	3102	3.686823
8	1107	3203	4.784851
9	1108	3203	2.970849
10	1201	1101	4.729183
11	1201	1312	3.31981
12	1201	3203	2.840383
13	1201	4305	3.629803
14	1201	4310	4.007372
15	1202	1307	7.090505
16	1202	4306	19.83571
17	1202	4310	2.639411
18	1203	3203	4.882425
19	1304	1101	5.770218
20	1304	1102	5.016738
21	1304	1306	6.154411
22	1304	1312	5.300149
23	1304	3102	2.343616
24	1304	3203	2.926563
25	1304	4310	2.526387
26	1305	3102	2.450536
27	1305	4104	2.940315
28	1306	4104	2.571176
29	1307	4306	13.32083
30	1307	5101	4.028736
31	1310	1101	4.086427
32	1310	1102	3.011378
33	1310	1304	2.858508

Again, we can select the input file and it defaults in the same way. We can decide which worksheet we want to use, and which columns are important. We can save as a 1- or 2-mode network, and decide to ignore the top line (if it is merely column labels). Click on 'Create Pajek file' and the file will be ready to be opened in Pajek.



Lab 2: Creating CAMSIS scores for large-scale social surveys

This lab features some examples focussed upon using large scale occupational data in order to perform Social Interaction Distance analysis. Examples are given in Stata and in R. There are relatively more examples in Stata since that package supports a wider range of functionality, combining data analysis and data management, as relevant to the requirements.

Lab2_Stata.do
Lab2_R.R

There is online duplication of nearly all the commands covered in this lab at the CAMSIS project website (www.camsis.stir.ac.uk), under ‘construction’. Those pages feature information on using Stata and R for SID analsysis, including access to relevant downloadable files, and the CAMSIS pages also feature information on using SPSS and lEM for the same purposes.

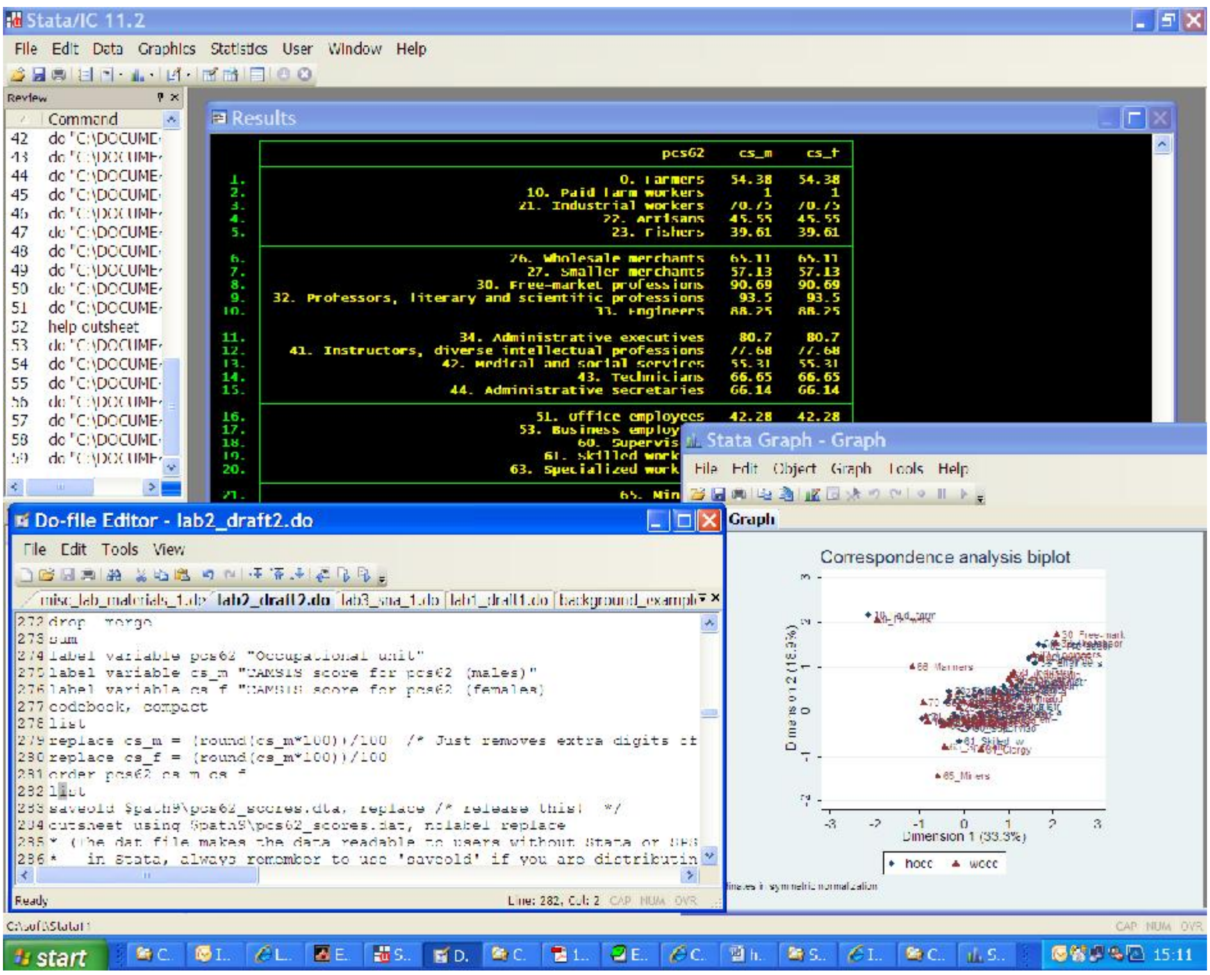


Image of a typical implementation of correspondence analysis results in Stata

Lab 3: Using SNA to analyse occupational structure

In this lab we explore ways we can use social network analysis to understand more about occupational stratification. Research in this area always benefits by consideration of the occupational structure and the national context of vocations. Research using social connections, similarly, can benefit from understanding underlying patterns of social interactions.
This analysis involves processes from both Stata and Pajek. Stata is utilised to generate the data which can be utilised by network analysis, which is processed within Pajek.

Our first step is to find a dataset consisting of pairs of occupations. For this lab, a dataset has been created consisting of within household combinations of microclasses. Only male ties with an age-gap of 16 years, aged between 16 and 75, are included. Combinations within the same microclass have been excluded.

```
. use $path9\scot81.dta, clear

. codebook, compact

Variable      Obs Unique      Mean   Min      Max  Label
------------------------------------------------------------------------
serial     258740 136853  403449.1     2   784262  Household index number
pernum1    258740     67  3.802242     1       87  Person index within ...
age1       258740     48  52.48311    28       75  Age
sex1       258740      1         1     1        1  Sex
occgb1     258740    384   210.265     1      413  Occupation, Britain ...
hocc       258740     64  4164.652  1101     9990
pernum     258740     76  6.289681     1      110  Person index within ...
age        258740     48  21.85543    12       59  Age
sex        258740      1         1     1        1  Sex
occgb      258740    382  199.2149     1      413  Occupation, Britain ...
wocc       258740     64  4061.158  1101     9990
pershh     258740     47  7.602937     2       49
pairshh    258740     47  62.56172     1     1176
aged       258740     48  30.62768    16       63
------------------------------------------------------------------------
```

Next we run a syntax file which automatically generates some information on pairs of occupations.

```
do http://www.camsis.stir.ac.uk/sonocs/do/pajek.do
```

This provides 16 variables. hocc is the older cohort occupation and wocc the younger cohort. freq is the number of connections for each combination. ewocc is the expected number of ties if the data was random. val_min is the value of over-representation (taking standard errors into account). These are the most important variables.

```
Variable      Obs Unique      Mean       Min       Max  Label
------------------------------------------------------------------------
hocc         3131     64  3458.181      1101      9990
wocc         3131     64  3401.974      1101      9990
freq         3131    395  82.63813         1     25594  (count) freq
tot          3131      1    258740    258740    258740  total number in ...
nhocc        3131     64  4849.383         4     39522  total number of ...
nwocc        3131     64  4828.742         5     33491  total number of ...
phocc        3131     64  .0187423  .0000155  .1527479  percentage of me...
pwocc        3131     64  .0186625  .0000193  .1294388  percentage of wo...
ewocc        3131   3129  79.17888  .0091598  5115.681  expected number ...
prop         3131    395  .0003194  3.86e-06  .0989178
staner       3131    395  .0000229  3.86e-06  .0005869  Standard error f...
pro_obs      3131    395  .0003194  3.86e-06  .0989178  Observed proport...
pro_exp      3131   3129   .000306  3.54e-08  .0197715  Expected proport...
pro_min      3131    395  .0002965  7.28e-12  .0983309  Lower confidence...
pro_max      3131    395  .0003423  7.73e-06  .0995048  Higher confidenc...
value        3131   3116  1.592005  .0257491   109.173  Observed value o...
val_min      3131   3131  .9458267  4.85e-08  24.38851  Value of represe...
val_max      3131   3131  2.238184  .0503272  218.3458  Value of represe...
------------------------------------------------------------------------
```

Next we need some criteria for which ties are needed. We are interested in cases occurring at least once in every 10,000 cases (therefore, frequency of at least 28) and which occur at least twice as often as we would expect.

```
. keep if freq>=27
(1942 observations deleted)

. keep if val_min>=2
(1081 observations deleted)
```

This produces 108 combinations which are both frequently constructed and occur more commonly than expected. As the val_min shows, these occur up to 20 times more than we would anticipate. There are 41 different microclasses for the older cohort and 37 for the younger cohort.

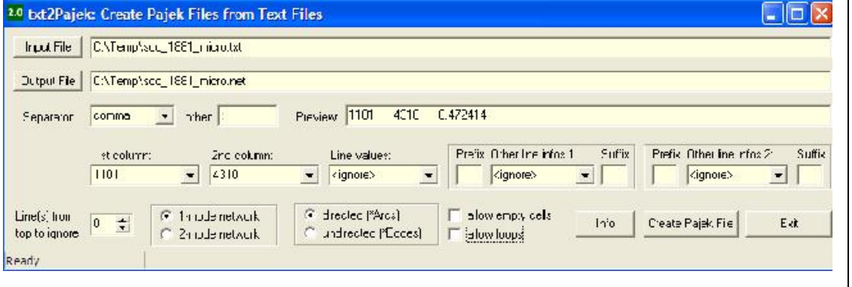
```
Variable     Obs  Unique        Mean         Min         Max   Label

hocc         108      41    2992.852        1101        9990
wocc         108      37    3223.398        1101        9990
freq         108      85    466.1852          29       25594   (count) freq
tot          108       1      258740      258740      258740   total number in s...
nhocc        108      41    4435.796         146       39522   total number of m...
nwocc        108      37    4578.713         281       33491   total number of f...
phocc        108      41    .0171438    .0005643    .1527479   percentage of men...
pwocc        108      37    .0176962     .001086    .1294388   percentage of wom...
ewocc        108     108    114.8665    3.142614    5115.681   expected number o...
prop         108      85    .0018018    .0001121    .0989178
staner       108      85    .0000512    .0000208    .0005869   Standard error fo...
pro_obs      108      85    .0018018    .0001121    .0989178   Observed proporti...
pro_exp      108     108    .0004439    .0000121    .0197715   Expected proporti...
pro_min      108      85    .0017506    .0000913    .0983309   Lower confidence ...
pro_max      108      85    .0018529    .0001329    .0995048   Higher confidence...
value        108     108    4.659531    2.122135    19.83571   Observed value of...
val_min      108     108    4.150986    2.011253    19.30663   Value of represen...
val_max      108     108    5.168075     2.22231    21.30224   Value of represen...
```

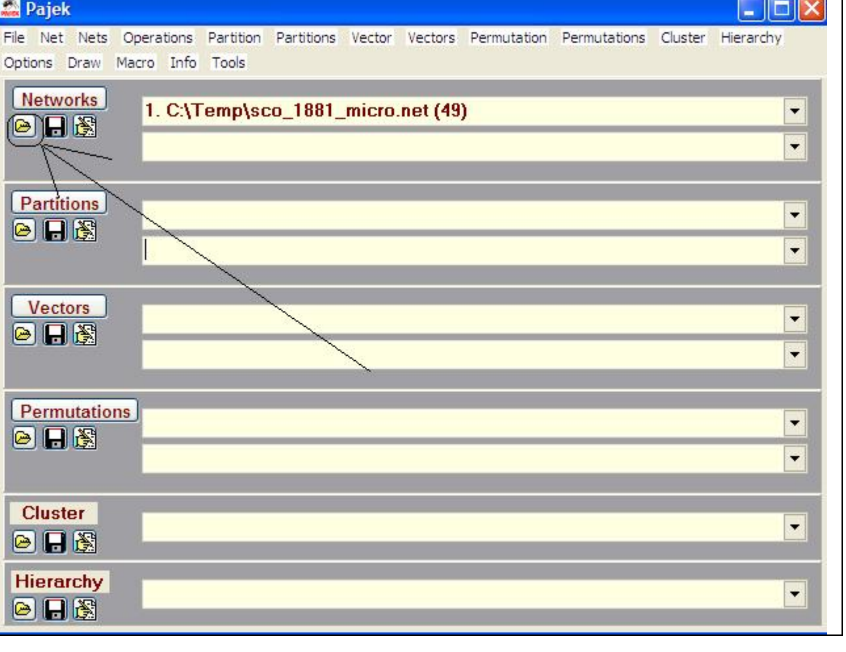
We can then export the data as a comma-separated text file, showing hocc wocc and val_min

```
outsheet hocc wocc val_min using ///
"$path9\sco_1881_micro.txt", ///
comma nonames nolabel replace
```

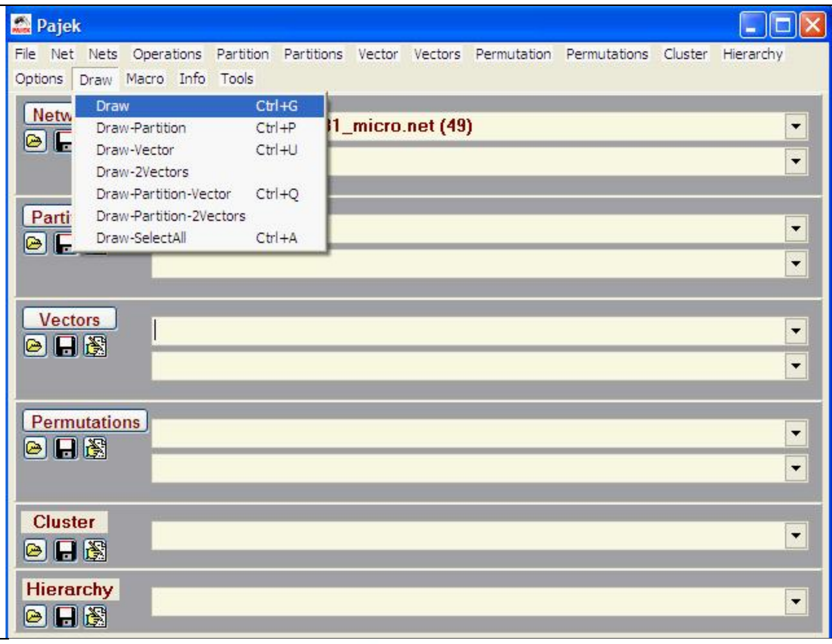
We can then use txt2Pajek to convert the data into a Pajek file. We select the input file, which then defaults to saving the output file to the same folder. We specify it is comma separated and select the two microclass labels. We then assert it is a one-mode directed network before clicking on 'create Pajek file'.



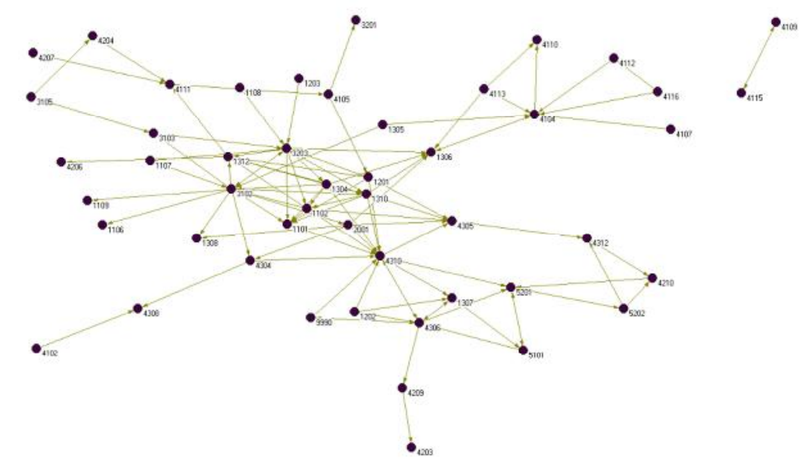
We can then open the file in Pajek, using either the file button below network, or the drop-down menu.



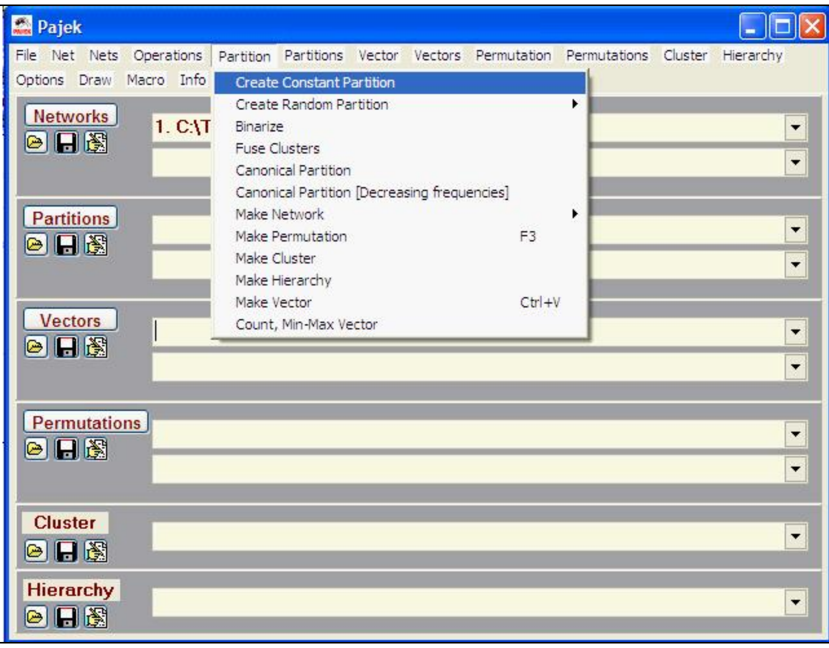
We can then select ‘draw’ or CTRL+G to visual the network.



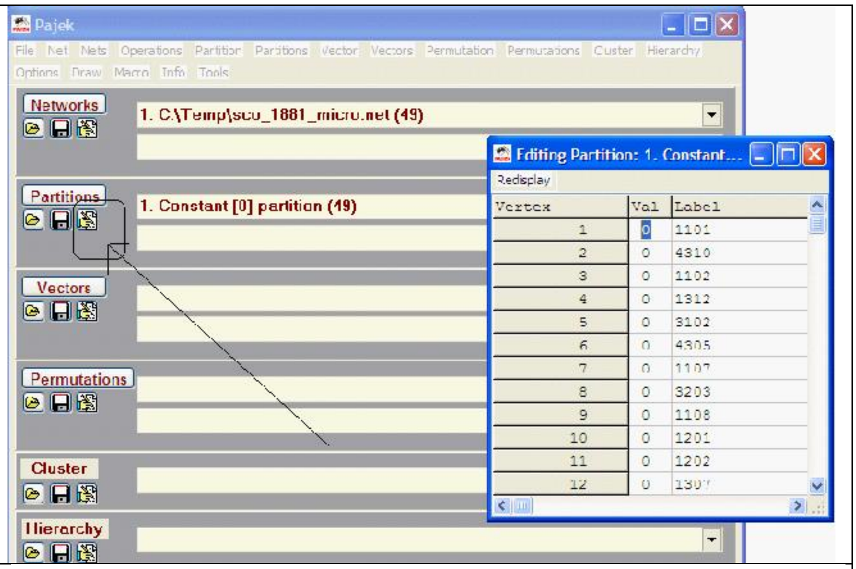
This provides a diagram of the network. The nodes are distributed where the software believes they are best placed. Sometimes it can be a little strange, so pressing CTRL+K will regenerate the network. Clicking on an individual node enables it to be moved around.



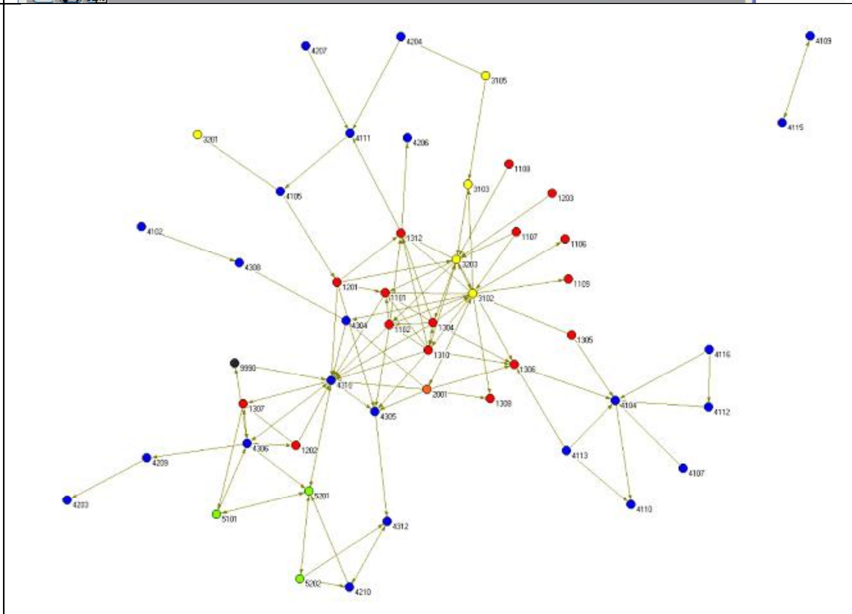
We can change the colour of the nodes to see differences more clearly. If we close the graph window we can click on ‘partition’, ‘create constant partition’ to create a blank partition. The first dialogue box regards the size of the partition (which defaults to the number of nodes, as it needs to be). The second dialogue box asks for the constant term to be used (default is 0).



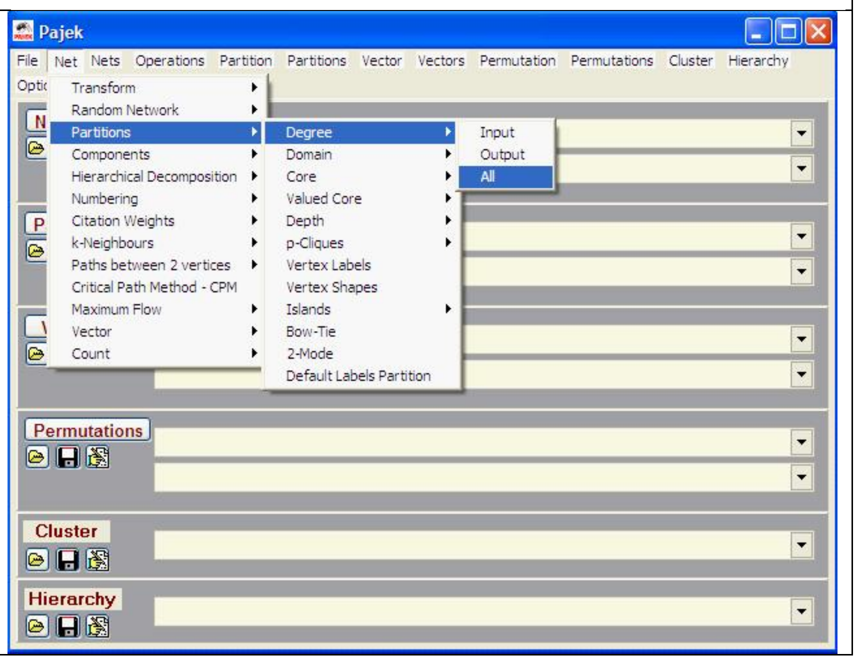
If we click on the last of the three buttons we can edit the partition. This enables us to group the nodes. In this example, we will place the first number of each label as the partition (which enables us to code the nodes by macroclass).



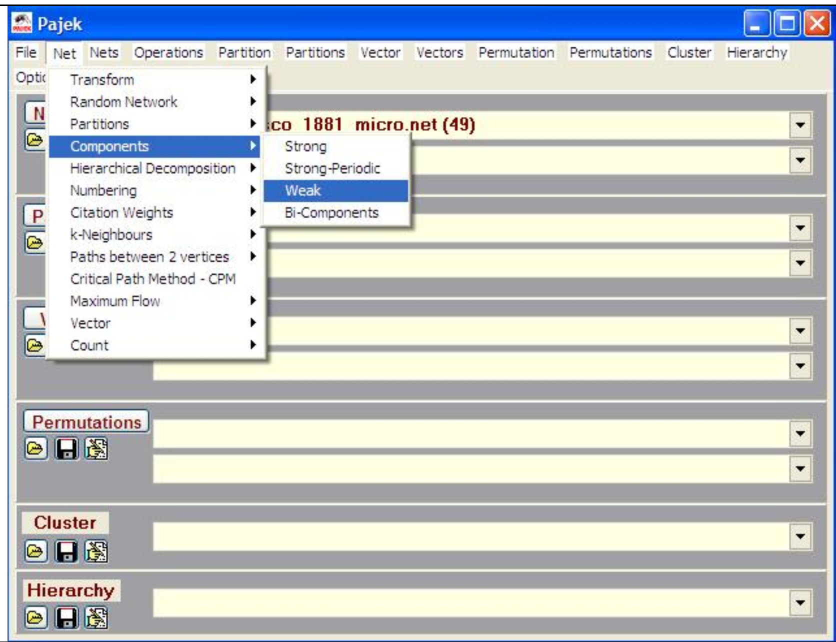
This enables us to see the differences between the macroclasses more clearly. The colours of the partitions can be altered using the 'options', 'colors', 'partition colors', 'of vertices' drop down menu within the draw window. Therefore, the colours might not be the same as displayed.



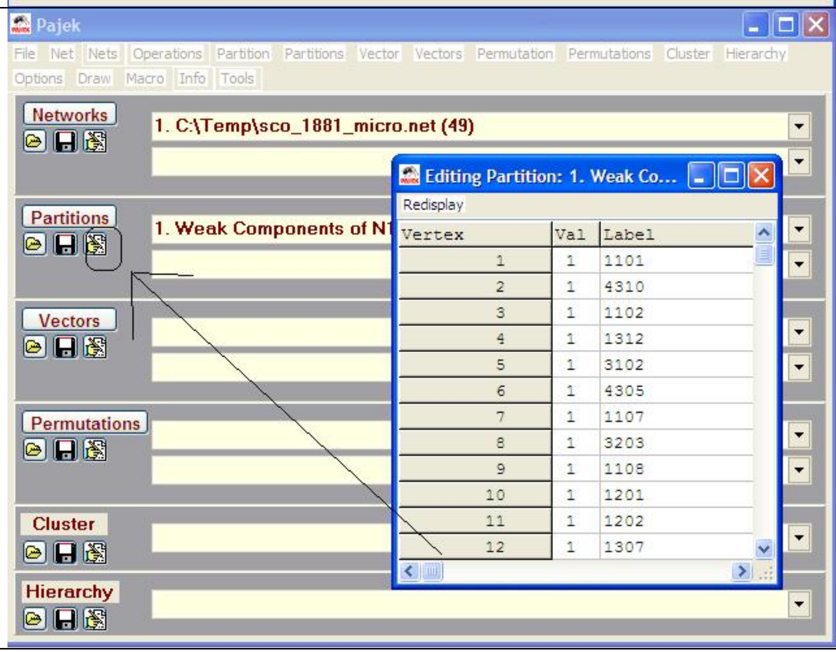
We can also find out statistics about the network. Using 'net', 'partition', 'degree', 'all' we can see the degree centrality of the network (as well as getting the degree for each occupation).



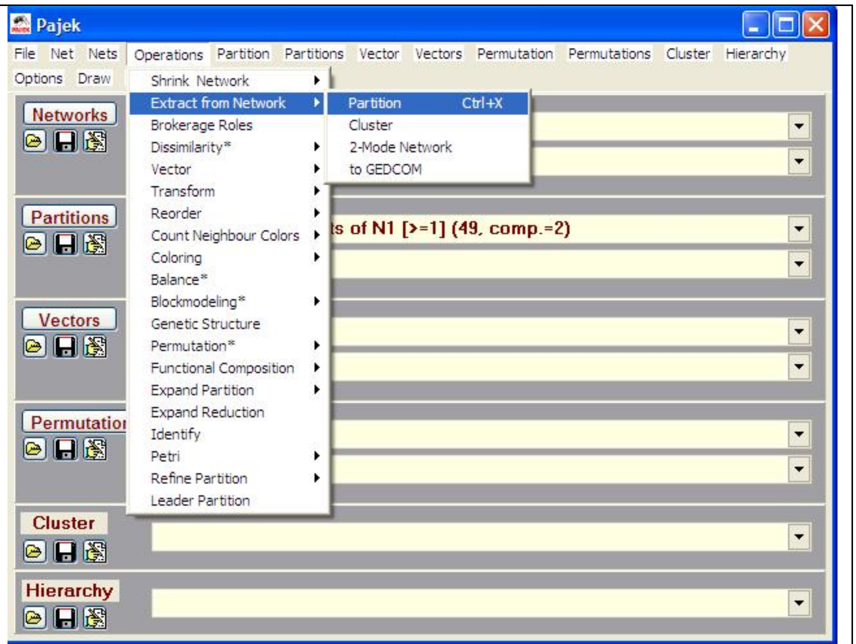
We can see how many components exist, using ‘net’, ‘components’, ‘weak’. From the dialogue box we select ‘1’ in this example (this is just about the strength of ties needed to form part of the community. We are interested here in nodes which have at least one tie to a member of the sub-population which are connected).



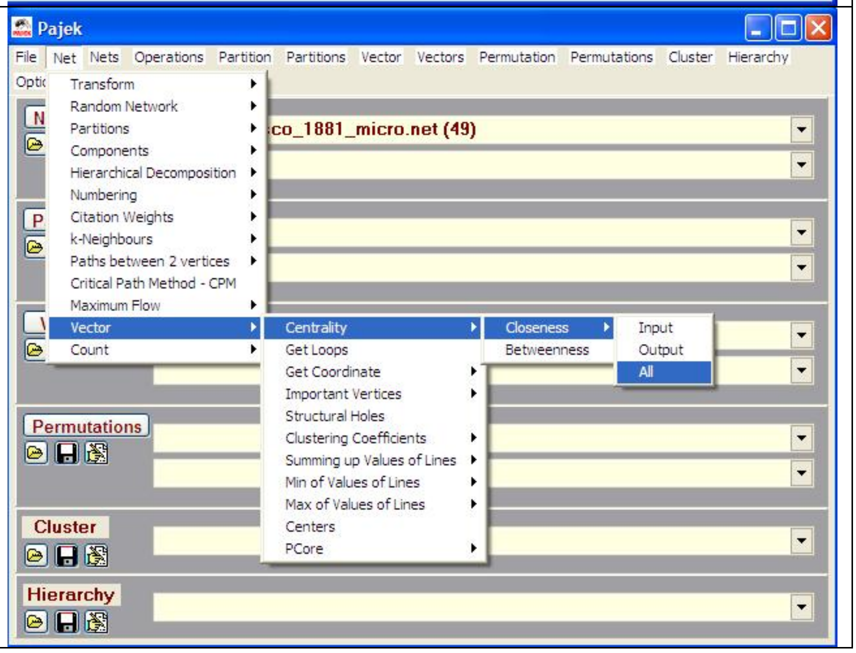
We can then reduce our analysis to just the largest component. This allows us to perform some further analysis. Firstly, we click on the partition named ‘components’ to see which is largest.



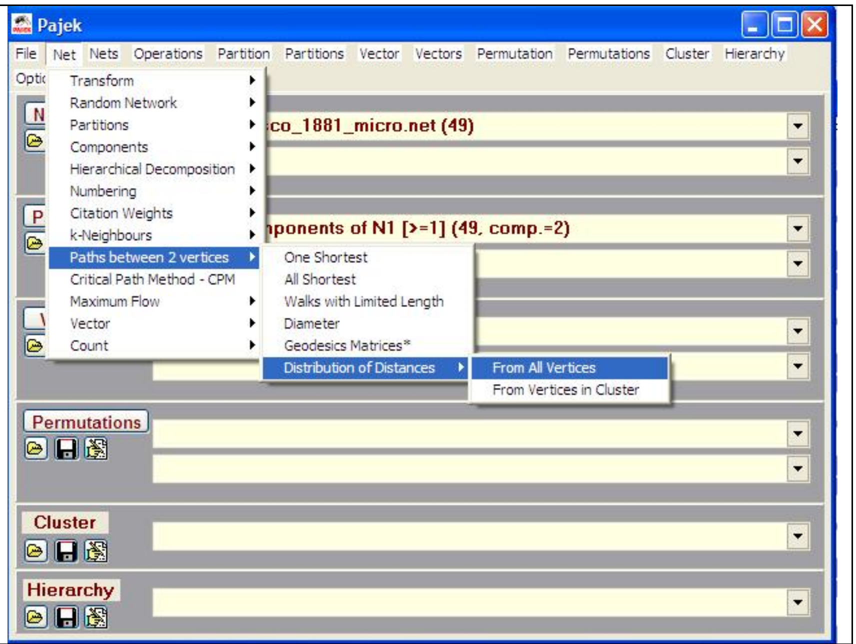
We then click on ‘operations’, ‘extract from network’, ‘partition’ (or CTRL+X) and decide to keep only ‘1’. This removes those cases which do not connect to the main component.



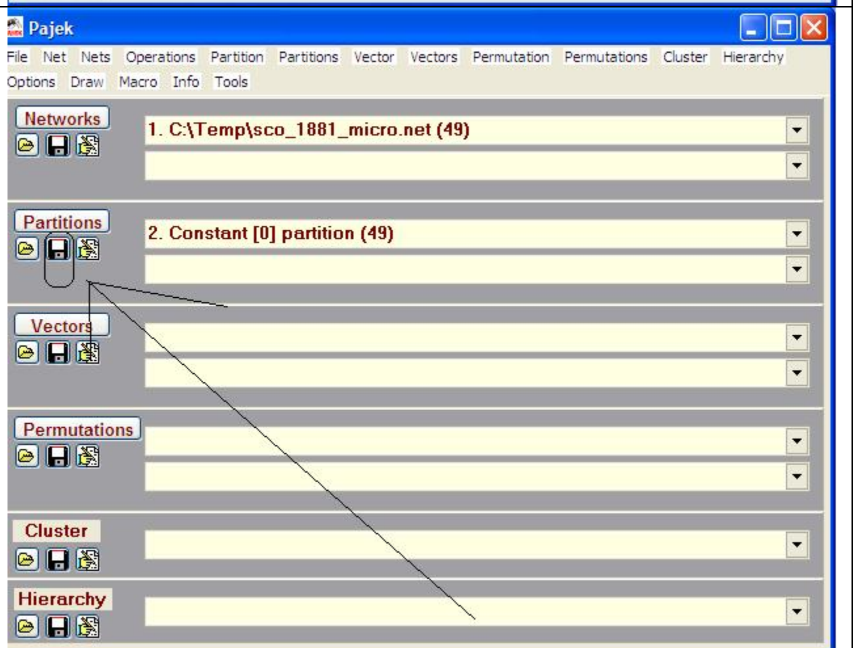
We can use ‘net’, ‘vector’, ‘centrality’, ‘closeness’, ‘all’ to see the closeness centrality.



We can use ‘net’, ‘Paths between 2 vertices’, ‘distribution of distances’, ‘from all vertices’ to see the average distance between the nodes.



We can then save the partition we created to ensure we have the data next time we analyse it.



We can analyse more than just microclasses. In this second example we will look at Canada 1891 data, looking at religion. Again, a dataset has been created.

```
. use $path9\canada91.dta, clear

. codebook, compact

Variable      Obs Unique      Mean       Min     Max  Label

serial      21699   7675  51919.54        12   76715  Household index ...
age1        21699     48   51.6469        28      75  Age
sex1        21699      1         1         1       1  Sex
hisco1      21699    312  60062.34      1110   98990
microclass1 21699     71  3658.492      1101    9990
age         21699     46  23.66455        12      57  Age
sex         21699      1         1         1       1  Sex
hisco       21699    337  62402.54      1110   98990
microclass  21699     70  3820.816      1101    9990
religion1   21699     66  3595.099      1100    9997  Religion, first ...
religion    21699     61  3609.406      1100    9997  Religion, first ...
hiscam1     21699     68  58.69302  45.69142      99  (mean) hiscam
hiscam      21699     68  57.72059  45.69142      99  (mean) hiscam
```

From the religion data, we can dichotomise the microclasses by placing a 1 in front of the microclasses for Catholics.

```
. capture drop cath*

. gen cath=religion==1100

. gen cath1=religion1==1100

. capture drop hocc

. gen hocc=microclass

. replace hocc=microclass+10000 if cath==1
(30927 real changes made)

. capture drop wocc

. gen wocc=microclass1

. replace wocc=microclass1+10000 if cath1==1
(29994 real changes made)
```

This enables us to run the analysis as above.

```
do http://www.camsis.stir.ac.uk/sonocs/do/pajek.do
```

This provides us with over 4,000 combinations, although many of these are of little relevance to us.

```
Variable     Obs Unique       Mean        Min        Max  Label
-------------------------------------------------------------------------
hocc        4471    137   7575.914       1101      19990
wocc        4471    138   7164.323       1101      19990
freq        4471     77    4.66115          1       1111  (count) freq
tot         4471      1      20840      20840      20840  total number in ...
nhocc       4471    107    265.926          1       1566  total number of ...
nwocc       4471    102   309.9512          1       2662  total number of ...
phocc       4471    107   .0127604    .000048    .075144  percentage of me...
pwocc       4471    102   .0148729    .000048   .1277351  percentage of wo...
ewocc       4471   3832   3.305061   .0004319   200.0332  expected number ...
prop        4471     77   .0002237    .000048   .0533109
staner      4471     77   .0000831    .000048   .0015562  Standard error f...
pro_obs     4471     77   .0002237    .000048   .0533109  Observed proport...
pro_exp     4471   3818   .0001586   2.07e-08   .0095985  Expected proport...
pro_min     4471     77   .0001406   1.15e-09   .0517548  Lower confidence...
pro_max     4471     77   .0003067    .000096   .0548671  Higher confidenc...
value       4471   3968   5.717776   .0432525   2315.556  Observed value o...
val_min     4471   4217   1.115635   1.04e-06   81.03288  Value of represe...
val_max     4471   4214   10.31992   .0691135   4631.055  Value of represe...
-------------------------------------------------------------------------
```

We set limits of frequency of 5 and val_min of 2.

```
.
. keep if freq>=5
(3546 observations deleted)

. keep if val_min>=2
(617 observations deleted)
```

This provides us with over 300 combinations of microclass_religion linkages.

```
Variable    Obs Unique       Mean        Min        Max  Label
-------------------------------------------------------------------------
hocc        308     92   8767.581       1101      15202
wocc        308     91   8542.578       1101      15202
freq        308     57   22.03571          5       1111  (count) freq
tot         308      1      20840      20840      20840  total number in s...
nhocc       308     85   273.1526         14       1566  total number of m...
nwocc       308     77     330.25         12       2662  total number of f...
phocc       308     85   .0131071   .0006718    .075144  percentage of men...
pwocc       308     77   .0158469   .0005758   .1277351  percentage of wom...
ewocc       308    304   4.442368   .0735605   200.0332  expected number o...
prop        308     57   .0010574   .0002399   .0533109
staner      308     57   .0001857   .0001073   .0015562  Standard error fo...
pro_obs     308     57   .0010574   .0002399   .0533109  Observed proporti...
pro_exp     308    304   .0002132   3.53e-06   .0095985  Expected proporti...
pro_min     308     57   .0008717   .0001326   .0517548  Lower confidence ...
pro_max     308     57   .0012431   .0003472   .0548671  Higher confidence...
value       308    304    9.38363   2.268401   109.2243  Observed value of...
val_min     308    306   6.628953   2.010274   81.03288  Value of represen...
val_max     308    306   12.13831   2.521529   137.4158  Value of represen...
-------------------------------------------------------------------------
```

We can then export the data as above.

```
outsheet hocc wocc val_min using ///
"$path9\ca_1891_micro_cath.txt", ///
comma nonames nolabel replace
```

We can then rerun the analysis in Pajek, as above.

Follow the steps from the txt2pajek example, using the Canadian data. The practices in Pajek are the same, regardless of the type of data.

(Note: When we create the partition to distinguish between Catholics and non-Catholics, if you ask it to set a constant partition with a value of 0, you can simply change the value to a '1' if it is a five-digit number (i.e., it has a one added to the front).

If you wish, you might want to code the macroclass additionally (the first number if 4 digits, first two if 5 digits). This will create potential values of 1-5, 9, 11-15 & 9. Using 'options', 'colors', 'partition colors', *‘of vertices’ in the graph window, you can click on the colors and assign them a new value. This would enable a ‘bluescale’ system for Catholics and a ‘redscale’ for non-Catholics akin to the usual ‘greyscale’ we see of the macroclasses being represented by brightness and the religion represented by colours).*

Performing the Pajek analysis is similar, regardless of the methods used to generate the data.

The analysis thus far has focussed on using the ‘Threshold’ method to extract data from Stata. We can also use the ‘Popularity’ method for looking at the top ties.

Firstly, we convert our dataset to Pajek, and set a limit for the number of ties we wish to use (to avoid a single instance being viewed as over-represented).

```
use $path9\usa2000.dta, clear

do $path5\pajek_ster.do

**Set a limit for number of ties
capture drop limit
gen limit = freq>=5
codebook if limit==1, compact

keep if limit==1
```

This gives us a dataset of 4,838 connections. There are 80 male occupations and 80 female. This is not always the case as sometimes occupations are too small for one gender to have any cases. If so, revise the frequency or consider dropping/merging the occupational group.

Variable	Obs	Unique	Mean	Min	Max	Label
hocc	4838	80	3264.186	1101	9900	Microclass
wocc	4838	80	3039.178	1101	9900	Microclass of spouse
freq	4838	1057	397.4161	5	42296	(count) freq
tot	4838	1	1924919	1924919	1924919	total number in sample
nhocc	4838	80	28550.33	733	154477	total number of males in occupation
nwocc	4838	80	30948.27	35	428777	total number of females in occupation
phocc	4838	80	.014832	.0003808	.0802512	percentage of men in occupation
pwocc	4838	80	.0160777	.0000182	.2227507	percentage of women in occupation
ewocc	4838	4838	385.5598	.2015259	34409.86	expected number of partnerships
prop	4838	1057	.0002065	2.60e-06	.0219729	
staner	4838	1057	6.69e-06	1.16e-06	.0001057	Standard error for tie
pro_obs	4838	1057	.0002065	2.60e-06	.0219729	Observed proportion of all ties
pro_exp	4838	4838	.0002003	1.05e-07	.017876	Expected proportion of all ties
pro_min	4838	1057	.0001998	1.44e-06	.0218672	Lower confidence interval of observed proportion
pro_max	4838	1057	.0002131	3.76e-06	.0220785	Higher confidence interval of observed proportion
value	4838	4833	1.122118	.0684662	26.00483	Observed value of representation
val_min	4838	4838	.9455958	.0425884	24.29394	Value of representation for lower confidence interval
val_max	4838	4838	1.29864	.0908308	35.90638	Value of representation for higher confidence interval
threshold	4838	1	1	1	1	

However, if there is a missing occupation for women but you are studying husband’s ties to wives, it is not important as that occupation will simply have no incoming ties. There are likely others which receive no incoming ties also. If you wish to compare male and female networks, however, that occupation should be removed from both.

Next we need to order each occupation in descending order by the over-

```
capture drop hcase
gsort hocc -val_min
by hocc: gen hcase=_n
```

representation (val_min) and assign a score to each occupation so we only have the most over-represented. We can then decide what level we wish to set the threshold for popularity to (3 in this case).

Deciding whether to sort by male (hocc) or female (wocc) occupation is consequential here. They do not necessarily provide the same results. You can export the data for men and for women separately if you wish to compare them.

```
keep if hcase<=3
codebook, compact
```

This gives us 240 cases. As there are 80 occupations and 3 ties from each occupation, then as 80*3=240 we can be assured the process has worked.

The early step showing 80 occupations did not show at least 3 ties for all 80 occupations, so it's worth checking this number and ensuring the numbers are as expected.

Variable	Obs	Unique	Mean	Min	Max	Label
hocc	240	80	3220.363	1101	9900	Microclass
wocc	240	75	3353.375	1101	9900	Microclass of spouse
freq	240	180	442.5208	5	12037	(count) freq
tot	240	1	1924919	1924919	1924919	total number in sample
nhocc	240	80	24061.49	733	154477	total number of males in occupation
nwocc	240	75	23507.99	35	428777	total number of females in occupation
phocc	240	80	.0125	.0003808	.0802512	percentage of men in occupation
pwocc	240	75	.0122125	.0000182	.2227507	percentage of women in occupation
ewocc	240	240	217.1488	.2015259	6715.202	expected number of partnerships
prop	240	180	.0002299	2.60e-06	.0062532	
staner	240	180	7.84e-06	1.16e-06	.0000568	Standard error for tie
pro_obs	240	180	.0002299	2.60e-06	.0062532	Observed proportion of all ties
pro_exp	240	240	.0001128	1.05e-07	.0034886	Expected proportion of all ties
pro_min	240	180	.000222	1.44e-06	.0061964	Lower confidence interval of observed proportion
pro_max	240	180	.0002377	3.76e-06	.0063101	Higher confidence interval of observed proportion
value	240	240	3.404909	1.255762	26.00483	Observed value of representation
val_min	240	240	2.888906	1.084877	24.29394	Value of representation for lower confidence interval
val_max	240	240	3.920912	1.324216	35.90638	Value of representation for higher confidence interval
threshold	240	1	1	1	1	
limit	240	1	1	1	1	
hcase	240	3	2	1	3	

Now that we have our data, it's a simple process of exporting our results, so they can be converted for us in Pajek.

```
outsheet hocc wocc using $path4\us00.txt, ///
comma nonames nolabel replace
```

As an aside, it's possible to dichotomise the data prior to exporting. This can be performed in Stata at the first stage.

```
use $path9\usa2000.dta, clear

keep if gpc_h>=20 & gpc_h <=80
keep if gpc_w>=20 & gpc_w <=80
```

The compact codebooks show that we have dropped from 1,925 cases to 3279k cases by looking only at couples both working in occupations with between 20% and 80% graduates. 279k is a large dataset so

Variable	Obs	Unique	Mean	Min	Max	Label
edattan	1924919	4	3.165295	1	4	Educational attainment, international recode [g...
edattan_sp	1924919	4	3.180714	1	4	Educational attainment, international recode [o...
hocc	1924919	80	3168.676	1101	9900	Microclass
wocc	1924919	81	2653	1101	9900	Microclass of spouse
gpc_h	1924919	80	23.36023	1.436552	91.79865	(mean) gpc
gpc_w	1924919	81	27.12525	1.436552	91.79865	(mean) gpc

Variable	Obs	Unique	Mean	Min	Max	Label
edattan	279143	4	3.573838	1	4	Educational attainment, international recode [ge...
edattan_sp	279143	4	3.526877	1	4	Educational attainment, international recode [of...
hocc	279143	27	1879.82	1101	9900	Microclass
wocc	279143	27	1676.914	1101	9900	Microclass of spouse
gpc_h	279143	27	43.56248	21.27431	74.08017	(mean) gpc
gpc_w	279143	27	43.96049	21.27431	74.08017	(mean) gpc

this is fine. Beware that these methods can make datasets too small to analyse. We've also dropped from 80 to 27 occupations, principally as 53 are outside of our range and therefore do not have sufficient numbers of both graduates and non-graduates to meaningfully analyse them.	
The usually processes can now be analysed. This method can be performed multiple times to create different datasets (i.e., one for male graduates irrespective of wife's education and one for male).	`do $path5\pajek_ster.do`

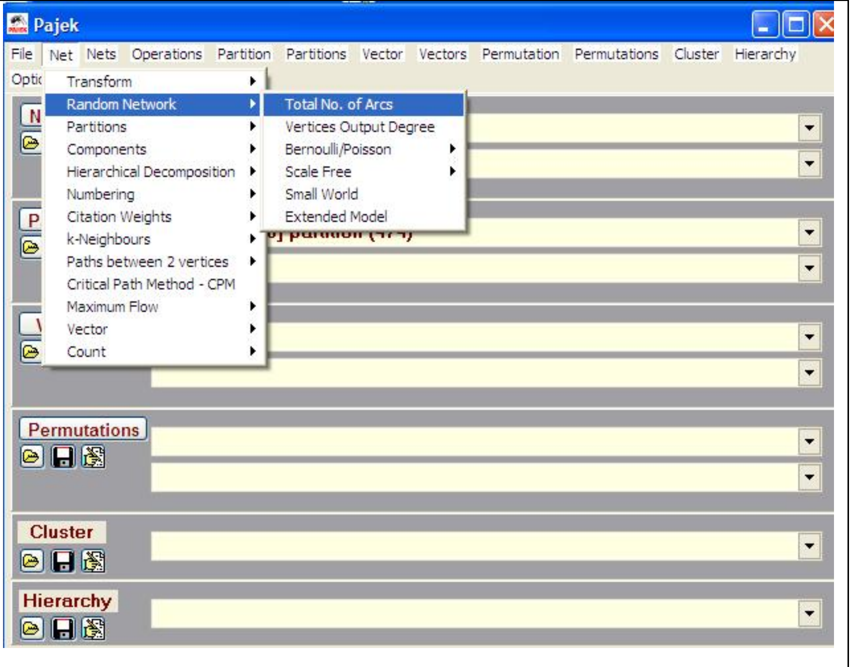
Combined method
The combined method brings together these techniques.
 	
Start off by using the popularity method.
	`use $path9\usa2000.dta, clear`
	
	`do $path5\pajek_ster.do`
	
	`**Set a limit for number of ties`
	`capture drop limit`
	`gen limit = freq>=5`
	`codebook if limit==1, compact`
	
	`keep if limit==1`
	
	`capture drop hcase`
	`gsort hocc -val_min`
	`by hocc: gen hcase=_n`

Create a new variable which enables you to code if it is the top (or top 3 etc.) occupations. Keep just those and export them to another dataset.
 	
Note: in Stata preserve and restore allows you to make changes which does not affect how the data was before the 'preserve' command was operated.
	`capture drop real`
	`gen real=num==1`
	
	`preserve`
	`keep if real==1`
	`save $path9/top.dta, replace`
	`restore`

Remove those cases from the dataset and then move to using the threshold	`drop if real==1`
	`gsort -val_min`
	`capture drop real`

technique. Sort all combinations by their over-representation score and assign an ordered number to each. Then, just keep however many is required to have the correct number of ties.	`gen real=_n` `keep if _n <=4528`

You can now add back the initial ties and you will have your data ready to be exported in the normal way.	`append using $path9\top.dta`
	
	`outsheet hocc wocc using $path9\combined.txt, ///`
	`comma nonames nolabel replace`

Creating a random network. This is simple in Pajek. Go to ‘random network’, ‘total no. of arcs’. The first dialogue box asks how many vertices are needed (number of occupations). The second dialogue box asks for number of arcs (how many linkages). Then, it randomly generates a network based on those numbers. There are other options for types of networks to generate if required.	
The QAP analysis has been performed in Ucinet. Pajek cannot yet do QAP analysis, whilst the system in R is not as user-friendly for comparing networks.	

Selected references

Stata

Kohler, H. P., & Kreuter, F. (2009). *Data Analysis Using Stata, 2nd Ed* College Station, Tx: Stata Press.

Leckie, G., & Charlton, C. (2011). *runmlwin: Running MLwiN from within Stata.* Bristol: University of Bristol, Centre for Multilevel Modelling, http://www.bristol.ac.uk/cmm/software/runmlwin/ [accessed 1.6.2011].

Long, J. S. (2009). *The Workflow of Data Analysis Using Stata*. Boca Raton: CRC Press.

Rabe-Hesketh, S., & Skrondal, A. (2008). *Multilevel and Longitudinal Modelling Using Stata, Second Edition.* . College Station, Tx: Stata Press.

Treiman, D. J. (2009). *Quantitative Data Analysis: Doing Social Research to Test Ideas*. New York: Jossey Bass.

Web: http://www.longitudinal.stir.ac.uk/Stata_support.html

R

Bates, D. M. (2005). Fitting linear models in R using the lme4 package. *R News, 5*(1), 27-30.

Fox, J., & Weisberg, S. (2011). *An R Companion to Applied Regression, 2nd Ed.* London: Sage.

Gelman, A., & Hill, J. (2007). *Data Analysis using Regression and Multilevel/Hierarchical Models*. Cambridge: Cambridge University Press.

Spector, P. (2008). *Data Manipulation with R (Use R)*. Amsterdam: Springer.

(There is also a useful guide to using R, within SPSS, in the body of Levesque & SPSS Inc 2010, cited above).

Web: http://www.ats.ucla.edu/stat/r/

Pajek

de Nooy, W., Mrvar, A., & Batagelj, V. (2011). *Exploratory Social Network Analysis with Pajek*. Cambridge: Cambridge University Press.

Other references of relevance

Crouchley, R., Stott, D., & Pritchard, J. (2009) *Multivariate generalised linear mixed models via sabreStata (Sabre in Stata).* Lancaster: Centre for e-Science, Lancaster University.

Dale, A. (2006). Quality Issues with Survey Research. *International Journal of Social Research Methodology, 9*(2), 143-158.

Freese, J. (2007). Replication Standards for Quantitative Social Science: Why Not Sociology? *Sociological Methods and Research, 36*(2), 153-171.

Rafferty, A., & Watham, J. (2008). *Working with survey files: using hierarchical data, matching files and pooling data*. Manchester: Economic and Social Data Service, and http://www.esds.ac.uk/government/resources/analysis/.