



# Preparing diachronic trade network data with Stata

Simon Fink<sup>1</sup>

**Abstract:** Stata is normally not used to work with social network analysis (SNA) data. Network data are usually analyzed using more specialized software like UCINET, Pajek, or visone. However, due to its ability to automatize the handling of large datasets, Stata is ideally suited to prepare new datasets for further analysis in network analysis software. The research note takes a “cookbook approach” and suggests some strategies to handle network data in Stata. The research example is the transformation of diachronic dyadic data from the UN ComTrade / World Integrated Trade Solution (WITS) database into a UCINET \*.dl file. However, the overall approach is valid for most kinds of dyadic data. Examples of Stata code are provided, and the do-files need only slight adjustments to account for other datasets, which can then be prepared for network analysis very easily.<sup>2</sup>

**Draft. Comments, ideas and stata code very welcome!**

**Version: 9<sup>th</sup> April, 2010.**

---

<sup>1</sup> Dr. Simon Fink is Assistant Professor for Comparative Politics, University of Bamberg, Faculty of Social Sciences and Economics, Feldkirchenstraße 21, 96052 Bamberg, Germany, Phone: ++49 (0)951 863 2552, E-Mail: [simon.fink@uni-bamberg.de](mailto:simon.fink@uni-bamberg.de).

<sup>2</sup> This paper is also supplementary material to the paper Fink, S., & Krapohl, S. (2010). Assessing the Impact of Regional Integration: Do regional trade institutions shape trade patterns?, ECPR Joint Sessions. Münster. The present paper outlines the methodological approach taken in Fink/Krapohl in more detail (and in a more hands-on manner) than could be done in the original paper.

## Introduction

Stata<sup>3</sup> is one of the most used statistics packages in social science. However, one area of social science is still dominated by more specialized software: Network analyses are usually conducted using specialized software like UCINET<sup>4</sup>, Pajek<sup>5</sup>, or visone<sup>6</sup>. However, due to its flexibility, its capability for automatization, and the ability to handle large datasets, Stata is ideally suited to prepare and modify network datasets for further analyses in more specialized software.

This paper shows how network data can be handled with Stata – from the acquisition of the data using an internet database to the ready-to-use \*.dl-file. The main steps of data preparation are outlined in detail, their rationales discussed and the commented Stata-code can be found in the appendix. The main advantage of using Stata is the replicability and transparency of the data-modifying processes. Using Stata do-files, the question “what was done to the data” (King, 1990) can be answered very transparently, and analyses can be easily replicated.

Furthermore, the paper introduces some ideas on how best to work with diachronic network data, that is, networks over time. While there has been considerable success in efforts to visualize dynamic networks<sup>7</sup>, a more classical approach – partly induced by the requirements to publish “on paper” – still requires some way to produce a series of snapshots of networks over time. The paper demonstrates how this may be done with the combination of Stata and visone. Additionally, as a look ahead, the paper suggests how Stata may be used to prepare network data for an analysis with the Social Network Image Animator (SONIA) (Bender-deMoll & McFarland, 2006).

The research example used is taken from the work of the research group “Regional intgration outside of Europe” at the University of Bamberg.<sup>8</sup> The overall aim of the project is to elucidate how trade patterns between nations change in the course of regional integration. The projects tries to find out how the building of institutions like NAFTA, the European Union, or ASEAN affects international trade

---

<sup>3</sup> <http://www.stata.com>

<sup>4</sup> <http://www.analytictech.com/ucinet/>

<sup>5</sup> <http://vlado.fmf.uni-lj.si/pub/networks/pajek/>

<sup>6</sup> <http://visone.info/>

<sup>7</sup> <http://www.stanford.edu/group/sonia/>

<sup>8</sup> [http://www.uni-bamberg.de/en/polib/forschung/research\\_focus\\_regional\\_intgration\\_outside\\_of\\_europe/](http://www.uni-bamberg.de/en/polib/forschung/research_focus_regional_intgration_outside_of_europe/)

patterns. Thus, the research problem is from international relations, but the basic data-handling procedures apply to all forms of network data.

### **The research and methodological problems**

Put in a stylized way, the research problem is to elucidate how trade network patterns in the world evolve in response to regional integration projects. The theoretical intuition is that these patterns should vary extremely between different regional integration projects, as countries of the North may exploit economies of scale and comparative cost advantages. Thus, interdependence within the region should exist from the outset, and be reinforced by regional integration. On the other hand, regional integration in developing and emerging regions of the world should not cause intraregional interdependence, as these Southern countries often do not constitute mutually attractive markets. Instead, regional integration should initially cause an increased dependence from the Northern trade partners (Fink & Krapohl, 2010).

The “classical” approach to this problem is to use highly aggregated data on the level of countries or regions, for example trade in relation to GDP, or intraregional trade in relation to extraregional trade. However, as useful as these aggregated indicators may be for some purposes, they conflate much of the complexity of interdependence relations in world trade (Frankel & Wei, 1998; Lombaerde, Fredrik Söderbaum, & Baert, 2009).

Network data, on the other hand, have the advantage to visualize country positions in the trade networks. Interdependence may be communicated without conflating too much complexity, and the visualizations can communicate the complex structural properties of the data more conveniently than highly aggregated indicators.

The data on trade are readily available using the UN ComTrade database.<sup>9</sup> Thus, the description of the analysis steps may start with the result of the UN ComTrade data query. The result is usually an Excel file. Using the WITS “Query View Designer” we can choose which variables should be in the Excel file. We are interested in Reporter Name, Partner Name, Year, and Trade Value. Following Feenstra et al. (2005) we chose the reports of the importers, as these are usually more valid than

---

<sup>9</sup> [http://wits.worldbank.org/witsnet/StartUp/Wits\\_Information.aspx?AspxAutoDetectCookieSupport=1](http://wits.worldbank.org/witsnet/StartUp/Wits_Information.aspx?AspxAutoDetectCookieSupport=1)  
Data documentation is available at <http://comtrade.un.org/>

the exporters' reports. Thus, the reporter is the importer, and the partner is the exporter. We thus have an Excel-file that looks roughly like in Table 1.

Reporter Name	Partner Name	Year	Trade Value (\$ '000)
Argentina	Argentina	2008	29.434,385
Argentina	Brazil	2008	17.976.759,481
Argentina	China	2008	7.103.885,723
Argentina	EU27 --- EU27	2008	8.927.057,050
Argentina	Paraguay	2008	1.782.956,138
Argentina	Uruguay	2008	540.145,806
Argentina	United States	2008	7.023.218,433
Brazil	Argentina	2008	13.257.932,120
Brazil	Brazil	2008	247.038,438
Brazil	China	2008	20.040.014,318

Table 1: Excel file output of the WITS database

In this Excel-file, only two things need to be done “by hand”: For some reason or the other, the variable year is coded as a string variable, and thus needs to be changed into numerical coding.<sup>10</sup> Additionally, it would be a good idea to format the “Trade Value” column as numerical, without any points or commas separating the numbers (our approach was to drop the three decimal places altogether). Thus, the Excel-file looks like in Table 2.

Reporter Name	Partner Name	Year	Trade Value (\$ '000)
Argentina	Argentina	2008	29434
Argentina	Brazil	2008	17976759
Argentina	China	2008	7103886
Argentina	EU27 --- EU27	2008	8927057
Argentina	Paraguay	2008	1782956
Argentina	Uruguay	2008	540146
Argentina	United States	2008	7023218
Brazil	Argentina	2008	13257932
Brazil	Brazil	2008	247038

Table 2: Modified Excel file

This is all that has to be done by hand. The rest can be handled automatically with some Stata code. Just copy and paste the Excel data into the Stata data editor. The

---

<sup>10</sup> You get alerted to this by a little exclamation mark in the column. Just use the context menu to change the format.

editor asks you how to treat the first row of data – you reply that Stata should treat it as variable names. The Stata dataset should look like in Table 3.

reportername	partnername	year	tradevalue000
Argentina	Argentina	2008	29434
Argentina	Brazil	2008	1.8e+07
Argentina	China	2008	7.1e+06
Argentina	EU27 --- EU27	2008	8.9e+06
Argentina	Paraguay	2008	1.8e+06
Argentina	Uruguay	2008	540146
Argentina	United States	2008	7.0e+06
Brazil	Argentina	2008	1.3e+07
Brazil	Brazil	2008	247038

Table 3: Raw Stata dataset

This raw dataset has still several problems that impede its use in network analysis software.

- 1) The ordering of the variables is not correct. We want the exporter to be in the first column, followed by the importer, followed by the tradevalue (the link strength, in network parlance). Additionally, we don't yet know how to handle the year variable.
- 2) The names of the variables could be nicer, e.g. exporter and importer.
- 3) The country names often present problems for network analysis software, e.g. it would be better to have "USA" instead of "United States".
- 4) We are maybe not sure whether the network analysis software can handle the e+... scientific notation.
- 5) We would perhaps like to clean up the dataset a bit, e.g. we do not want all the connections in our data, but only the 3, 4, 5...most important trade links. Additionally, we may want to drop self-reflexive links.
- 6) We would like to know how many nodes are there in our dataset. NetDraw is rather friendly, letting you know if you have specified a wrong number of nodes in your dataset, and tells you the correct number, but visone simply doesn't open the dl-file if you have specified too little nodes.

Additionally, if we want to conduct a diachronic network analysis, several additional problems apply. These will be covered in the next section. Thus, the

present example applies for network data from one year only. Drop the other years if necessary.

To reorder and rename the variables (problems 1 and 2) just tell Stata to generate some new variables with the values of the old variables and drop the old ones. For the purpose of the static analysis, the variable year may be dropped.

```
generate exporter = partnername
generate importer = reportername
generate tradevalue = tradevalue000
drop reportername partnername year tradevalue000
```

After this little manipulation, the dataset should look like in Table 4. Alternatively, you might of course want to keep the year variable, if you want to go on to the dynamic analyses. In this case, do not drop year.

exporter	importer	tradevalue
Argentina	Argentina	29434
Brazil	Argentina	1.80e+07
China	Argentina	7103886
EU27 --- EU27	Argentina	8927057
Paraguay	Argentina	1782956
Uruguay	Argentina	540146
United States	Argentina	7023218
Argentina	Brazil	1.33e+07
Brazil	Brazil	247038

Table 4: Stata dataset re-ordered and with renamed variables

The country names can be handled using a do-file that systematically renames the exporters and importers like this:

```
replace exporter = "CongoDemRep" if exporter == "Congo, Dem. Rep."
replace exporter = "CongoRep" if exporter == "Congo, Rep."
replace exporter = "CotedIvoire" if exporter == "Cote d'Ivoire"
replace exporter = "Czechoslovakia" if exporter == "Czech Republic"
replace exporter = "Laos" if exporter == "Lao PDR"
```

```

replace exporter = "Egypt" if exporter == "Egypt, Arab Rep."
replace importer = "Iran" if importer == "Iran, Islamic Rep."
replace importer = "SouthKorea" if importer == "Korea, Rep."
replace importer = "NewZealand" if importer == "New Zealand"
replace importer = "Russia" if importer == "Russian Federation"
replace importer = "SaudiArabia" if importer == "Saudi Arabia"
replace importer = "Slovakia" if importer == "Slovak Republic"
replace importer = "Syria" if importer == "Syrian Arab Republic"
replace importer = "Taiwan" if importer == "Taiwan, China"

```

This is obviously a bit of work, as you have to find all the names with blanks in the dataset. However, the work is worth the trouble. Once you have the do-file, you can easily rename the countries in each new trade dataset you acquire, without having to do it “by hand” with search&replace in Excel or a text editor.

Now, we may get rid of the scientific notation using

```

format %20.0g tradevalue
replace tradevalue = round(tradevalue)

```

Additionally, we might want to drop self-reflexive ties, and we want to keep only the 3 most important links in any dyad. We tell Stata to drop any observations, in which the name of the exporter is the same as the name of the importer. Then we sort the data by exporter and in descending value of tradevalue, and generate a rank variable that tells us the ranking of the export partners. After that, we may at will keep only the observations that are of interest to us, e.g. the three most important (with rank values of 1, 2, or 3). Of course, it may be wise to save the data before dropping observations.

```

drop if exporter == importer
gsort exporter -tradevalue
by exporter: generate rank = _n
keep if rank < 4

```

After the first three steps (before the dropping of observations), the dataset should look like in Table 5.

exporter	importer	tradevalue	rank
Argentina	EU27	16165173	1
Argentina	Brazil	13257932	2
Argentina	China	9361350	3
Argentina	USA	6177698	4
Argentina	Uruguay	2249961	5
Argentina	Paraguay	1289398	6
Brazil	EU27	53211152	1
Brazil	USA	32007380	2

Table 5: Stata dataset with renamed countries and ranked according to size of trade flows

Now, we would like to know how many nodes are present in the dataset to inform the header of our dl-file with the appropriate number. For this purpose, we can tell Stata to count the number of exporters:

```
egen nodes = group(exporter)
sort nodes
gsort -nodes
list nodes in 1
```

If our dataset is symmetric, the displayed highest value of the nodes-variable should be the number of nodes, if the dataset is not symmetric (i.e. if we have more importers than exporters), we replicate the procedure and take the higher value.<sup>11</sup>

```
egen inodes = group(importer)
sort inodes
gsort -inodes
list inodes in 1
```

---

<sup>11</sup> Of course, the number might still be wrong if our dataset is highly asymmetric to the extent that it contains exporters that are not importers, and importers that are not exporters. In any case, the number of nodes cannot be higher than nodes+inodes (if the set of importers and exporters does not overlap at all). I am currently working on a do-file that solves this problem and gives the exact number. However, for the most practical purposes the described counting procedure gives helpful results.

Now, the dataset is ready for further analysis. Just copy and paste the data from Stata (the columns exporter, importer, and tradevalue) into a standard dl-file, tell the dl file how many nodes there are, and use the dataset at will in a network analysis software of your choice.

Of course, other modifications of the data are easily implemented. For example, if you do not want to know the size of the trade flows, but a binary dataset that tells you whether in a country pair the importer belongs to the five most important partners (1) or not (0) (e.g. if you want to replicate Piana (2004)), you would use

```
generate top5 = 1 if rank < 6  
drop if top5 == 0
```

and copy the columns exporter, importer, and top5 into your dl-File – ready to use for further analyses.

### **Additional problems of diachronic network snapshots (and some solutions)**

Additional problems occur if you want to take diachronic snapshots of trade networks over time. The problem is one of proportionality. If you plot, e.g. separate networks for 1975, 1980, and 2000 in separate files, you cannot compare the visualization results over time. For example, you wish the height of the nodes to reflect the indegrees and the width of the nodes to reflect the outdegrees. If the values of the links are not binary but weighted, you cannot be sure whether, e.g. the double height of the node in 2000 compared to 1980 really reflects double indegrees – *if you have separate network plots*. However, if you treat all the snapshots for practical purposes as *one large network* – with network nodes designated, e.g. Brazil\_1975, Brazil\_1980, and Brazil\_2000 – you can be sure that the visualization takes into account the proportionality. Additionally, with trade data, the problem of inflation occurs, as the ComTrade Data are not adjusted for inflation. Thus, Stata needs to fulfill two tasks:

- 1) Adjust the trade values for inflation.
- 2) Combine years and countries to new node names

The first task is easily accomplished using a do-file. Again, the idea is that it is too much work to write a do-file if you only once transform one dataset. However, if you anticipate working with several datasets, the time lost in writing the do-file is balanced by the time saved with the second, third...and following datasets. In the example, I have taken data on US\$ inflation from an internet source<sup>12</sup>, and recalculated all trade values as 2008-US\$.

```
replace tradevalue = tradevalue*1.32 if year == 1998
replace tradevalue = tradevalue*1.28 if year == 1999
replace tradevalue = tradevalue*1.24 if year == 2000
replace tradevalue = tradevalue*1.22 if year == 2001
replace tradevalue = tradevalue*1.19 if year == 2002
replace tradevalue = tradevalue*1.17 if year == 2003
replace tradevalue = tradevalue*1.13 if year == 2004
replace tradevalue = tradevalue*1.09 if year == 2005
replace tradevalue = tradevalue*1.07 if year == 2006
replace tradevalue = tradevalue*1.04 if year == 2007
replace tradevalue = tradevalue*1.00 if year == 2008
```

To combine the years and country names into new names, we would use the `concat` option of the (very useful) `egen` command.

```
egen exp_year = concat (exporter year)
egen imp_year = concat (importer year)
```

Thus, the two commands generate new node names that conflate the name of the exporter and the year, yielding a network dataset that looks roughly like this:

Vietnam2008	ChJpKor2008	13422806
Vietnam2008	EU272008	14757041
Vietnam2008	USA2008	13853633

---

<sup>12</sup> <http://www.westegg.com/inflation/>

Vietnam2003	USA2003	5743922
Vietnam2003	EU152003	6512999
Vietnam2003	ChJpKor2003	5325331
Vietnam1999	EU151999	4601066
Vietnam1999	USA1999	838167
Vietnam1999	ChJpKor1999	2962645
Vietnam1995	Singapore1995	628796
Vietnam1995	ChJpKor1995	2873369
Vietnam1995	EU151995	1861079

These new data may be treated like the static data, only the three most important partners kept, and the number of nodes counted. The resulting \*.dl-file can be imported in visone. If some layout algorithm of stress minimization is applied, the networks are neatly separated by year.

Going further, you might be interested in visualizing not only snapshots but making dynamic network visualization using the Social Network Image Animator (SONIA) (Bender-deMoll & McFarland, 2006). This paper cannot offer a comprehensive treatment on how to do it, but some first suggestions can be offered. SONIA works with pajek \*.net files, and uses numbers in brackets - [1], [2], [3] – to designate the time points. Thus, a snapshot of an appropriate dataset might look like this, with the first number designating the sender, the second number designating the receiver, the third number designating the value of the connection, and the number in brackets as the time point:

```

1 4 1 [1]
1 2 1 [1]
1 13 1 [1]
2 15 1 [1]
2 11 1 [2]
2 16 1 [3]

```

Stata may help generate a dataset like this from our original dataset. Suppose that we have data from the year 1995 onward. We first generate the timepoint variable so that instead of 1995 we have 1, instead of 1996 we have 2 and so on. Again, with the help of the concat option, we create a new variable that puts these time points into nice brackets.<sup>13</sup> Copy and paste the ensuing data into an appropriate \*.net file, and SONIA will recognize the time points.

```
generate timepoint = year - 1994
generate left = "["
generate right = "]"
egen timpoint2 = concat(left timepoint right)
drop left right timepoint
```

## Conclusion

This research note has demonstrated how Stata may be used to facilitate the handling of network datasets. Many researchers presumably use Stata for conventional statistics based on attributional data (regression models of all kinds, analysis of variance, graphics...), and specialized software for network analysis and visualization. Thus, the suggestions of the research note might be of interest to a considerable number of researchers. Of course, each researcher has her own strategies of handling and re-coding network data, and there is no one-size-fits-all approach. The procedures suggested in this paper are open to critique and improvement, there are certainly more elegant solutions. But the procedures outlined may serve as a starting point of a fruitful dialogue between users of “classical” statistics packages and network analysis software.

---

<sup>13</sup> Of course, Stata may also be used to designate the nodes with numerical codes instead of (or complementing) country names.

## Appendix: Stata code – just cut & paste into do-file editor

Code for the basic manipulation of the dataset to make it dl-file ready:<sup>14</sup>

```
*** generate new variables with new names and in the correct order
generate exporter = partnername
generate importer = reportername
generate tradevalue = tradevalue000
drop reportername partnername year tradevalue000

*** get rounded values for the tradevalue variable, without scientific
notation
format %20.0g tradevalue
replace tradevalue = round(tradevalue)

*** drop self-reflexive ties
drop if exporter == importer

*** sort by the size of trade value, rank the partners accordingly, keep
only the three most important
gsort exporter -tradevalue
by exporter: generate rank = _n
keep if rank < 4

*** count the number of nodes
egen nodes = group(exporter)
sort nodes
gsort -nodes
list nodes in 1

egen inodes = group(importer)
sort inodes
gsort -inodes
list inodes in 1
```

---

<sup>14</sup> If Stata does not recognize some of the commands, then your Stata version does not yet have the appropriate ado-files. You can search and install them using the Stata web search.

Code for the re-naming of the countries (example, fit it to your own needs and preferences):

```
replace exporter = "CongoDemRep" if exporter == "Congo, Dem. Rep."
replace exporter = "CongoRep" if exporter == "Congo, Rep."
replace exporter = "CotedIvoire" if exporter == "Cote d'Ivoire"
replace exporter = "Czechoslovakia" if exporter == "Czech Republic"
replace exporter = "Laos" if exporter == "Lao PDR"
replace exporter = "Egypt" if exporter == "Egypt, Arab Rep."

replace importer = "Iran" if importer == "Iran, Islamic Rep."
replace importer = "SouthKorea" if importer == "Korea, Rep."
replace importer = "NewZealand" if importer == "New Zealand"
replace importer = "Russia" if importer == "Russian Federation"
replace importer = "SaudiArabia" if importer == "Saudi Arabia"
replace importer = "Slovakia" if importer == "Slovak Republic"
replace importer = "Syria" if importer == "Syrian Arab Republic"
replace importer = "Taiwan" if importer == "Taiwan, China"
```

Code for the creation of a binary dataset that tells only whether a country is among the top5 import partners of an exporter:

```
generate top5 = 1 if rank < 6
drop if top5 == 0
```

### Example code for inflation adjustment:

```
replace tradevalue = tradevalue*1.32 if year == 1998
replace tradevalue = tradevalue*1.28 if year == 1999
replace tradevalue = tradevalue*1.24 if year == 2000
replace tradevalue = tradevalue*1.22 if year == 2001
replace tradevalue = tradevalue*1.19 if year == 2002
replace tradevalue = tradevalue*1.17 if year == 2003
replace tradevalue = tradevalue*1.13 if year == 2004
replace tradevalue = tradevalue*1.09 if year == 2005
replace tradevalue = tradevalue*1.07 if year == 2006
replace tradevalue = tradevalue*1.04 if year == 2007
replace tradevalue = tradevalue*1.00 if year == 2008
```

### Example code for conflating name of exporter (importer) and year:

```
egen exp_year = concat(exporter year)
egen imp_year = concat(importer year)
```

### Example code for preparing diachronic network data for analysis with SONIA (starting year 1995).

```
*** generate new variable that denotes time points 1, 2, 3...
generate timepoint = year - 1994
*** put the time points in brackets
generate left = "["
generate right = "]"
egen timpoint2 = concat(left timepoint right)
drop left right timepoint
```

## References

- Bender-deMoll, S., & McFarland, D. A. (2006). The Art and Science of Dynamic Network Visualization. *Journal of Social Structure*, 7(2).
- Feenstra, R. C., Lipsey, R. E., Deng, H., Ma, A. C., & Mo, H. (2005). World trade flows: 1962-2000. *NBER working paper*.
- Fink, S., & Krapohl, S. (2010). Assessing the Impact of Regional Integration: Do regional trade institutions shape trade patterns?, *ECPR Joint Sessions*. Münster.
- Frankel, J. A., & Wei, S.-J. (1998). Open regionalism in a world of continental trade blocs. *IMF Working Paper*, 98(10).
- King, G. (1990). On Political Methodology. *Political Analysis*, 2, 1-29.
- Lombaerde, P. D., Fredrik Söderbaum, L. V. L., & Baert, F. (2009). The Problem of Comparison in Comparative Regionalism, *3rd Annual Research Conference "The EU in a comparative Perspective"*. Halifax.
- Piana, V. (2004). Hierarchy Structures in World Trade. *Economics Web Institute*.