**Desk Notes for QITIS Sample Enlargement by Country and Industry**

**First Steps**

1. Start by opening RStudio.
2. Next, make sure your workspace is clear: either start a new project or navigate to *Session > Clear Workspace…*
   1. It may also be useful to clear plots (*Plots > Clear All…*) and the console (*Ctrl+L*)
3. Open *‘QITIS Sample Enlargement.R*’.
4. Make sure you have the *bit64* and *data.table* packages installed, if you haven’t, run the following code: 

**Preparing the data**

Notice the file paths at the top of ‘*QITIS Sample Enlargement.R*’. Each of these corresponds to one of the data files we’ll need. All filetypes are *csv*. Let’s specify what shape the data should appear in.

* *microdata.file* — This corresponds to the AITIS microdata for the year on which we’re basing our analysis. The formatting for this file is the same formatting used in the 2014 and 2015 data. Only 7 columns are required, all others are discarded: *'ru\_ref', 'current\_sic', 'q201', 'q202', 'q204\_i', 'q203', 'concern'.*

*ru\_ref* are the 11-digit referencing codes for the RUs. *current\_sic* should be each RU’s 5-digit 2007 sic code. *q201* refers to an import or export code which are *P* and *R*  respectively. *q203* refer to country codes which are integers. *q202* are product codes. *q204\_i* is imputed/returned trade in that product code measured in pounds-thousands. *concern* is a 2-digit integer.

* *dead.ru.list.file* — This corresponds to a file detailing all those RUs that have died. It contains two columns with no title headers. In the first column is a list of all those RU reference codes for the RUs that appear in the AITIS sample. The second column has a date of death next to all those RUs that are now dead and is blank otherwise (please make sure blank means that there is only white space—no tabs—there).
* *starting.sample.file —* This corresponds to the RUs in the sample before it is enlarged. The first column should have a header and consist of a list of those RUs’ reference numbers.
* *annual.total.country.import.estimates.file* — Corresponds to a list of countries and estimates for annual imports from that country to the UK in pounds-thousands. Should contain a column *‘Country code’* consisting of country codes consistent with those in the microdata file with each country being one of our *countries of interest*. Should also contain a column ‘*Total exports’* (despite them being imports)with the estimated totals in pounds- thousands.
* *annual.total.country.export.estimates.file* — Corresponds to a list of countries and estimates for annual exports from that country to the UK in pounds-thousands. Should contain a column *‘Country code’* consisting of country codes consistent with those in the microdata file with each country being one of our *countries of interest*. Should also contain a column ‘*Total exports’* with the estimated totals in £’s thousands.
* *countries.of.interest.file —* Corresponds to a file listing our *countries of interest* to which we shall be restricting our attention. This file should contain 2 columns, the first being called *‘Country’* being a list of the country names, the second being called *‘Code*’ which is a list of the country codes consistent with the microdata.
* *industry.totals.file —* Corresponds to a file listing both industries (*section groupings*) and their estimated total imports/exports. Contains 7 columns. The first, *‘Section’,* contains the industry names.

Industries consist of groupings of divisions (see details on SIC 2007 codes). To achieve this there are four columns—*‘Start’, ‘End’, ‘And’, ‘And.End’*—which contain 2-digit numeric values (or may be blank for *And* and *And.End*). Supposing we have a row i then any RU with a division in *Start[i]:End[i]*or *And[i]:And.End[i]* will be counted as being in division i. If *And* and *And.End* are blank then *And[i]:And.End[i]* will not be counted.

The last two columns are *‘Exports’* and *‘Imports’* which contain the total import and export estimates for each industry in pounds-thousands.

* *output.file —* Corresponds to the file in which we wish to write our resulting list of RUs. Should be a csv.

**Next Steps**

1. Set the relevant file paths at the top of the file.
2. Next press *Source* (*Ctrl+Shift+S*).
3. In the console enter the command…This should take around 30 seconds depending on your computer.

You’re now ready to begin analysis/enlargement so let me run you through the different commands.

**Performing the Analysis and Enlargement**

This file provides a suite of commands that give you a high degree of control over the enlargement process whilst still requiring little technical knowledge. It might help if you’ve read through the algorithm outline in the write up but it’s not necessary. As long as you know what coverage is you’ll be okay.

* This function has no arguments and should only be run once at the start of a session after file paths have been appropriately set. It imports and prepares all the data files.

* This variable is a numeric vector consisting of the ru reference numbers for the sample prior to enlargement.

* This function of two arguments plots a series of useful graphs for analysis when enlarging with respect to country combination coverages. These graphs are based on the properties of samples created by enlarging *inputrus* on the *thresholds* argument.

*inputrus* should be a numeric/integer64 vector of RUs.

*thresholds* should be a numeric vector of thresholds for which you would like values to be plotted. The more thresholds the longer computation will take but a few hundred values should be able to be plotted in under 10 seconds. For detailed graphs I would recommend plotting for something like…



* Similar to *plotCountryGraphs* but plots graphs for analysis when enlarging with respect to industry combination coverages. Somewhat slower than *plotCountryGraphs*.

* This function takes as input a numeric/integer64 vector of RUs, *inputrus*, and enlarges this sample based on country combination coverages, outputting another numeric/integer64 vector of RUs. (See the write up for details). Only one of *target.threshold* and *target.samplesize* need be specified.

*target.threshold* should be a single number between 0 and 100 which represents a percentage. If specified an enlarged sample is outputted based on this threshold.

*target.samplesize* should be a single integer >0. A search will be performed to find a threshold which yields an enlarged sample of that size. This process will normally return a sample of the size specified as long as the *target.samplesize* is feasible.

If both are specified then *target.threshold* will always take priority.

* Like *enlargeByCountry* but enlarges on industry coverages.

* Takes as input a numeric/integer64 vector of RUs representing a sample and outputs that sample’s country combination coverages as a data frame. (It’s quite straightforward; I would recommend just trying it out, maybe with *oldrus* just to see what the output is like.) This function gives specific figures for analysing coverages of a sample.

If you’re interested in producing a spreadsheet of values either try writing the output to a file using… …or try the following and copy and paste the output…



* Once you have a vector of RUs that you want to write to a file us this function. *inputrus* takes a numeric/integer64 vector of RUs and *file.to.write.to* takes a file path to which it will write a csv.

**An Example Run**

The following is an example of a very standard run that you might perform to enlarge the sample by country to a sample size of 2000 followed by an enlargement by industry with a threshold of 0.2 before saving it, all topped off with some monitoring performance as we go.



**Data File Examples**

Your data files should look something like these…

***microdata.file***

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ru\_ref | concern | q202 | q203 | current\_sic | q201 | q204\_i | some column | some other column |
| 50100000000 | 70 | 0 | 167 | 42998 | P | 0 | a | u |
| 50100000000 | 70 | 0 | 167 | 42998 | R | 0 | b | v |
| 60500000404 | 56 | 49 | 119 | 45370 | P | 72 | c | u |
| 60500000404 | 56 | 49 | 131 | 45370 | P | 27 | d | v |
| 60500000404 | 56 | 49 | 132 | 45370 | P | 1536 | e | u |
| 60500000404 | 56 | 49 | 153 | 45370 | P | 2077 | f | v |
| 60500000404 | 56 | 0 | 167 | 45370 | P | 23272 | f | u |
| 60500000404 | 56 | 49 | 167 | 45370 | P | 23272 | f | v |
| 49800000000 | 56 | 51 | 15 | 45310 | R | 1 | f | u |
| 49800000000 | 56 | 6 | 64 | 45310 | R | 6 | f | v |
| 49800000000 | 56 | 8 | 64 | 45310 | R | 0 | f | u |

***dead.ru.list.file***

|  |  |
| --- | --- |
| 49900660355 |  |
| 49900660538 | 12/11/1993 |
| 49800660980 |  |
| 49800660983 |  |
| 49800661213 |  |
| 49800661226 | 10/06/1993 |
| 49800661505 |  |
| 49800661713 |  |
| 49800661863 |  |
| 49800662125 |  |
| 49800662155 |  |
| 49800662343 |  |
| 49800662375 | 08/01/1993 |
| 49800662445 |  |
| 49800663232 |  |
| 49800663443 |  |

***starting.sample.file***

|  |
| --- |
| Column Title |
| 49900660355 |
| 49900660538 |
| 49800660980 |
| 49800660983 |
| 49800661213 |
| 49800661226 |
| 49800661505 |
| 49800661713 |
| 49800661863 |
| 49800662125 |
| 49800662155 |
| 49800662343 |
| 49800662375 |
| 49800662445 |
| 49800663232 |
| 49800663443 |

***annual.total.country.import.estimates.file***

|  |  |  |
| --- | --- | --- |
| Country code | Country | Total exports |
| 0 | UNKNOWN |  |
| 1 | YEMEN | 13652 |
| 2 | ANGOLA | 530985 |
| 3 | ARGENTINA | 148752 |
| 4 | AUSTRALIA | 1786906 |
| 5 | AUSTRIA | 387548 |
| 6 | BELGIUM | 2077614 |
| 7 | BRAZIL | 781121 |
| 9 | CANADA | 1476007 |
| 10 | SRI LANKA | 42261 |
| 11 | CHILE | 105349 |
| 12 | CYPRUS | 287422 |

***annual.total.country.export.estimates.file***

|  |  |  |
| --- | --- | --- |
| Country code | Country | Total exports |
| 0 | UNKNOWN |  |
| 1 | YEMEN | 131652 |
| 2 | ANGOLA | 5301985 |
| 3 | ARGENTINA | 1418752 |
| 4 | AUSTRALIA | 17816906 |
| 5 | AUSTRIA | 3817548 |
| 6 | BELGIUM | 20717614 |
| 7 | BRAZIL | 7811121 |
| 9 | CANADA | 14761007 |
| 10 | SRI LANKA | 422161 |
| 11 | CHILE | 1053149 |
| 12 | CYPRUS | 2871422 |

***countries.of.interest.file***

|  |  |
| --- | --- |
| Country | Code |
| Austria | 5 |
| Belgium | 6 |
| Bulgaria | 127 |
| Croatia | 134 |
| Cyprus | 12 |
| Czech Republic | 143 |
| Denmark | 13 |
| Estonia | 136 |
| Finland | 14 |
| France | 15 |
| Germany | 17 |
| Greece | 20 |
| Hungary | 130 |

***industry.totals.file***

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Section | Start | End | And | And.End | Exports | Imports |
| A,B,D,E | 1 | 9 | 35 | 39 | 4730 | 4651 |
| C | 10 | 34 |  |  | 6904 | 88008 |
| F | 41 | 43 |  |  | 4575498 | 6754545.4 |
| G | 45 | 48 |  |  | 8888888 | 9999999 |
| H | 49 | 54 |  |  | 2 | 1 |