USER MANUAL

An Open-Source Aggregation Program for the GTAP Database: PyGTAPAgg

Python Aggrigation Program for GTAP Databases

SUBMITTED BY
ImpactECON, LLC.

Month Year
June 2024

Table of Contents

Forward

PyGTAPAgg was motivated by the unprecedented growth in analytical tools which have accompanied the field of data sciences in the past ten years. In many ways, data sciences grew out of domains, such as economics and life sciences, but has come to encompass broader methods and data (for example, web traffic and social networks). This is evident in the development of Python, where the most popular data manipulation package, Pandas, grew from the needs of domains such as economists, but is now driven by an even larger numbers of users from fields outside economics. Just as important to CGE modelers, Python has benefited from the efforts in the development of highly efficient APIs for manipulating arrays, matrices and systems of equations such as NumPy.[1] Accessing lower-level programming languages, such as C, C++ and Fortran has always been accompanied with steep learning curves and long development lead times. Python procedures access these languages through APIs, which greatly reduce the learning curve and speeds the development cycle. This can be seen clearly in the implementation of the PyQt software, used to create the GUI in PyGTAPAgg, which is tightly linked to the core Qt procedures written in C and C++. [2] The HARPY API provides access to GEMPACK HAR files via NumPy.[3]

Linking Python with access to widely used CGE databases opens the potential for data analysis and development. Plotly provides a robust set of graphing and visualization tools.[4] The avenue of progress seemed clear, the aggregation of the GTAP database and supporting analytics is a springboard to greater ideas. PyGTAPAgg is inspired by the long and steadfast service provided by the legacy programs GTAPAgg and its successor GTAPagg2.[5] These programs still provide simple, reliable means for aggregating the GTAP database. The goal and inspiration of the current project is to replicate the consistent and reliable performance of those programs before jumping off to greater endeavors and what we hope will be a renaissance in software development related to the GTAP database, supported by community (open source) development.

While most GTAP applications focus on a particular analysis, model, or an area of data construction, it is expected that the distribution of PyGTAPAgg will support users across disciplines and enhance analysis for current users. It will also improve productivity, by bringing resources previously spread across documents and software into one place. The incorporation of analytics will further reduce cycle times of creating data and running models to refine aggregations. The use of a powerful, widely used, open-source program and language should encourage development of ad-on tools for future researchers. The goal will be to house

[1] Harris, C.R., Millman, K.J., van der Walt, S.J. et al. *Array programming with NumPy*. Nature 585, 357–362 (2020). DOI: 10.1038/s41586-020-2649-2.
[2] https://www.qt.io/.
[3] https://github.com/GEMPACKsoftware/HARPY.
[4] https://plotly.com/python/
[5] https://www.gtap.agecon.purdue.edu/products/packages.asp.

the open-source program on GitHub or similar version control system. Users can fork, modify, test, and improve the base code. Users can submit new code for review, approval, and integration via GitHub. In the cases of developing custom interfaces for databases which deviate significantly from the Standard GTAP Data Base, such as CO2 and Supply chains, forks can be made of the base program and customized as required.

Introduction

Python Agg for GTAP is a GUI aggregation utility, in development, for the GTAP Data Base. It is open source, accessible to any user and adaptable by Python programmers. PyGTAPAgg builds on the fundamental simplicity of GTAPAgg2 and adds a few features to facilitate and improve the experience of running routine aggregation programs. The additional features include spreadsheet like capabilities (such as sorting and cutting and pasting) along with robust data verification subroutines in the creation of aggregations of the GTAP Data Base. It also provides a library of common aggregation mappings, which may be applied at the push of a button (such as defining products as processed agricultural, light, and heavy manufactures and services based on standard classification systems). User defined mappings can be retrieved and applied for subsets of the standard GTAP 65 sectors and 158+ regions, in contrast to importing mappings for all GTAP sectors and regions as currently provided in GTAPAgg2 and FlexAgg2. Users can select a mix of aggregation criteria from prior projects. The program also facilitates the research of mapping products from the HS systems, CPC[6] and ISIC[7] coding system of product definitions (in contrast to searching for external documents in various formats such PDF and webpages). Finally, basic analytics of the GTAP aggregation will be provided, so users can iteratively improve their aggregation within the aggregation program, before moving it to the modeling process. These features, together, promise to improve the efficiency, quality, and accuracy of creating aggregations for the GTAP Data Base. PyGTAPAgg is open source: users may build and contribute add on modules for domain specific tasks, such as migration, supply chains, welfare, land use and climate change modeling, among others. The use of tabs in the main interface facilitates this flexible expansion. The following sections provide instructions for installing the software and its use. A separate document, stored on the GitHub repo includes code documentation and instructions on how to create add-ons and contribute to the open-source community.

[6] https://unstats.un.org/unsd/classifications/unsdclassifications/cpcv21.pdf.
[7] https://unstats.un.org/unsd/publication/seriesm/seriesm_4rev4e.pdf

What Software Do I Need?

PyGTAPAgg was developed on Microsoft Windows operating system. The software has not been tested on widely used alternatives such as Linux and Apple OS. Until the software is extensively tested on these operating systems (with the accompanying GEMPACK HAR and Tab files), we recommend user employ Windows. The alternatives systems are not supported.

Essential software are:

- Python 3.7 or above;
- PyQt (free for open-source projects); and
- NumPy

The full environment can be loaded using a Python package manager such as pip or conda by applying the requirements file, as outlined in the instillation section following.[8]

Access to the GTAP Data Base can be obtained by purchasing an appropriate license from the GTAP Center at Purdue University.[9]

The current application requires that the GTAP database be extracted from its encrypted form. This is most easily done by accessing the FlexAgg database and programs.

Note that each software module comes with license requirements. For most open-source projects, licensing is not a major barrier.

[8] https://pip.pypa.io/en/stable/installation/
[9] https://www.gtap.agecon.purdue.edu/databases/v11/

Installing the Application

There are two methods of installing the application: 1) Installing the Python application and supporting modules; 2) Installing with an .exe file. Users with Python already installed on their system, running pip with the virtual environment and requirement file will ensure all programs are installed and of the correct version. The .exe file should be used by those interested only in the functional front end of PyGTAPAgg, without any of the Python source code or development capabilities. Given the widespread use of Python (it is installed by default on many Apple and Linux machies), most users will want to consider this as a first opportunity to start their learning path with Python by installing the full version.

SYSTEM MINIMUMS

Most Windows computers (laptop or desktop) running a version of Windows 10 or higher should be capable of running PyQTAgg. The installation of Python and the required modules will require close to 1GB of space for the complete installation. Note, an exe file may be downloaded, which includes only the required bytecode and interpreter which requires 200MB or less. However, the limited installation will not permit development.

Program	Size
Python 3.7	~600MB
PyQT 160MB	160MB
Numpy 61.5	61.5MB
HARPY 7.0MB	7.0MB
PIP 11.0 MB	11.oMB

System RAM of 4GB is required. 8 GB is recommended.

INSTALLING AS A PYTHON APPLICATION

The application can be run by typing c:> python .\PyGTAPAgg.py at the prompt in the directory in which you installed PyGTAPAgg. The file with that name must be present in the directory you type the name, PyGTAPAgg does not install itself in the system path at this time.[10]

INSTALLING WITH .EXE FILE

A ".exe" file may be downloaded from the ImpactECON web site at (xxxxx). The exe file will load files required by python and the those required by the PyGTAPApp. Users will not be able to see or change the source code with this approach.

Running an Aggregation

An aggregation in PyGTAPAgg proceeds in much the same way as using the legacy GTAPAgg2. There are major differences, which are subtle, and the user may not realize they are different, so emphasis will be added at those points. In general, GTAPAgg is a highly structured program, with data files having to be located in specific directly structures or the application will not see them. PyGTAPAgg was designed to be more flexible in specifying GTAP databases by simply pointing the system to a set of files. The system will do initial checks to be sure the essential data are available. Another point of difference is the use of the .agg files in GTAPAgg2 and prior versions. These were tab delimited files which had to be read in by GEMPACK using a rigid data structure. PyGTAPAgg employs the .json data format in common use today. JSON files are semi-structured data sets, which means there is no limit to the additional data which can be combined in the files, as long as the basic REG, PROD_COMM, TRAD_COMM, and ENDW_COMM source fields are present. Additionally, an aggregation file can contain regions and sectors from any current or past GTAP database version, the values will be matched up in accordance with the names in the database pointed too and non-matches result in values in the data tabs without associated descriptions.[11]

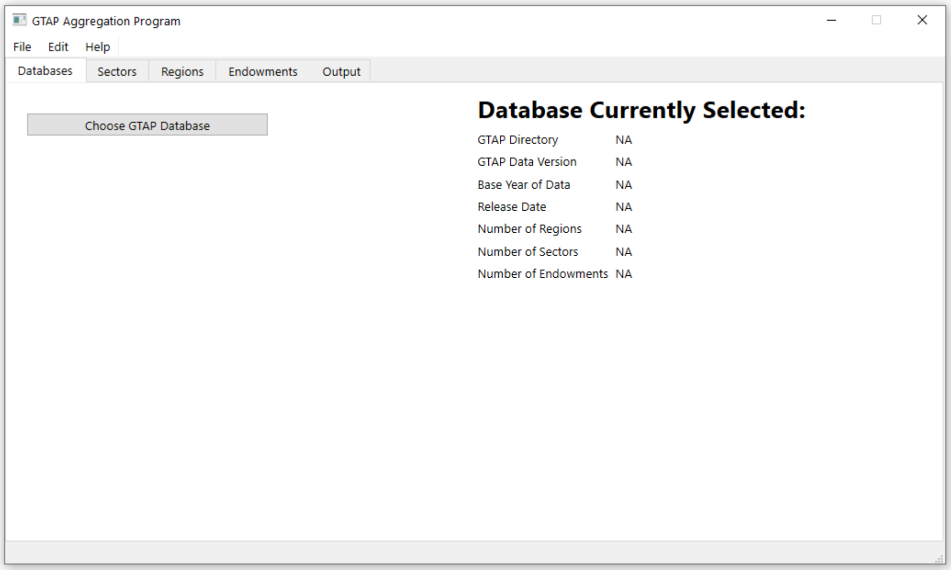
[10] Windows has shifted its terminal from the legacy command prompt (c:>) to the PowerShell prompt (PS:>). This manual uses the new PowerShell syntax which requires the entire relative path of the file, .\ and not simply the filename in the case of the command prompt. This is done for compatibility with Linux and Apple systems and for future proofing the directions.

[11] This might happen in the case that an aggregation file from an early GTAP version was employed to aggregate a newer version of the GTAP database.

Application Overview

The application is made up of a system menu which includes File, Edit and Help (Figure 1). The File menu allows the user to Open, Save, Save As and Quit the application (properly quitting will save any settings which have been changed, such as the current working directory etc.) Figure 2.

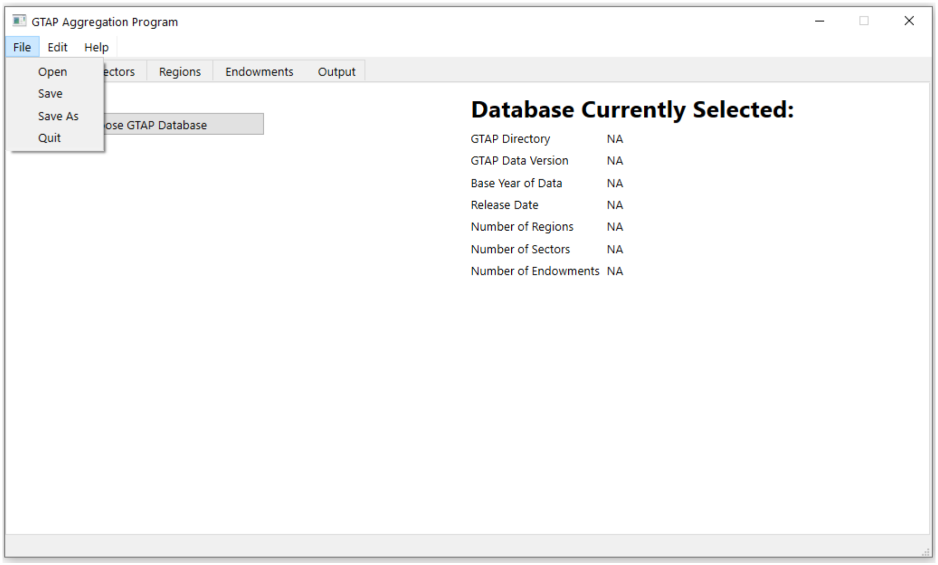
Figure 1
Opening Screen



Database Currently Selected:

GTAP Directory	NA
GTAP Data Version	NA
Base Year of Data	NA
Release Date	NA
Number of Regions	NA
Number of Sectors	NA
Number of Endowments	NA

Figure 2
File Menu



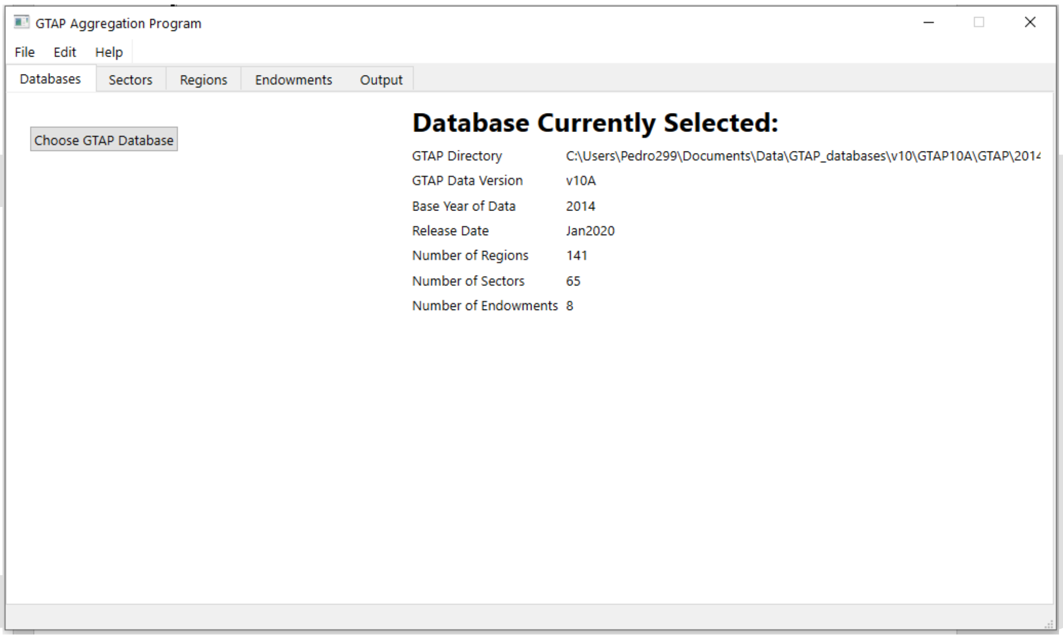
Database Currently Selected:

GTAP Directory	NA
GTAP Data Version	NA
Base Year of Data	NA
Release Date	NA
Number of Regions	NA
Number of Sectors	NA
Number of Endowments	NA

Choosing a Database (Data Tab)

An aggregation is started by selecting "Choose GTAP Database" (Figure 1). Navigate to the folder containing the GTAP Database to be aggregated. After choosing a database, the Databases tab will reflect the version, year, release data, number of regions, number of sectors and number of endowments in the database selected (Figure 3). If a database without a valid format is selected, a warning will be raised with information about why the database is not valid.

Figure 3
Selecting a Database to Aggregate



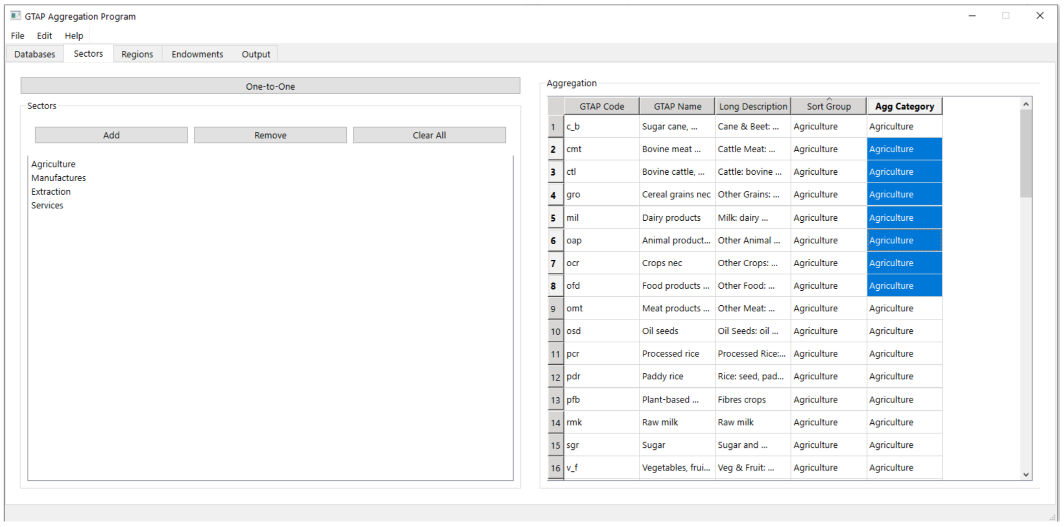
Database Currently Selected:

GTAP Directory	C:\Users\Pedro299\Documents\Data\GTAP_databases\v10\GTAP10A\GTAP\2014
GTAP Data Version	v10A
Base Year of Data	2014
Release Date	Jan2020
Number of Regions	141
Number of Sectors	65
Number of Endowments	8

Sector Mapping (Secor Tab)

Figure 4 shows the sectors tab. The right-hand pane shows the current state of the aggregation of sectors. In this example, there are five columns of data, which is different from the legacy application GTAPAgg2. We see the benefits of the JSON file format. The furthest right column is always the aggregation definition. The first column are the GTAP sectors. In between these two columns can be any data the user wishes to put in the JSON file, it will appear in this pane. Unlike GTAPAgg2, clicking on a column header will sort the data on that column, much as a spreadsheet. This allows for quick grouping and checking data. Next, a user can highlight multiple rows, and right click on them, like a spreadsheet and "paste" a value into all the cells. A menu with valid sector aggregations will appear.

Figure 4
Sectors Tab



Agriculture
Manufactures
Extraction
Services

	GTAP Code	GTAP Name	Long Description	Sort Group	Agg Category
1	c_b	Sugar cane, ...	Cane & Beet: ...	Agriculture	Agriculture
2	cmt	Bovine meat ...	Cattle Meat: ...	Agriculture	Agriculture
3	ctl	Bovine cattle, ...	Cattle: bovine ...	Agriculture	Agriculture
4	gro	Cereal grains nec	Other Grains: ...	Agriculture	Agriculture
5	mil	Dairy products	Milk: dairy ...	Agriculture	Agriculture
6	oap	Animal product...	Other Animal ...	Agriculture	Agriculture
7	ocr	Crops nec	Other Crops: ...	Agriculture	Agriculture
8	ofd	Food products ...	Other Food: ...	Agriculture	Agriculture
9	omt	Meat products ...	Other Meat: ...	Agriculture	Agriculture
10	osd	Oil seeds	Oil Seeds: oil ...	Agriculture	Agriculture
11	pcr	Processed rice	Processed Rice:...	Agriculture	Agriculture
12	pdr	Paddy rice	Rice: seed, pad...	Agriculture	Agriculture
13	pfb	Plant-based ...	Fibres crops	Agriculture	Agriculture
14	rmk	Raw milk	Raw milk	Agriculture	Agriculture
15	sgr	Sugar	Sugar and ...	Agriculture	Agriculture
16	v_f	Vegetables, frui...	Veg & Fruit: ...	Agriculture	Agriculture

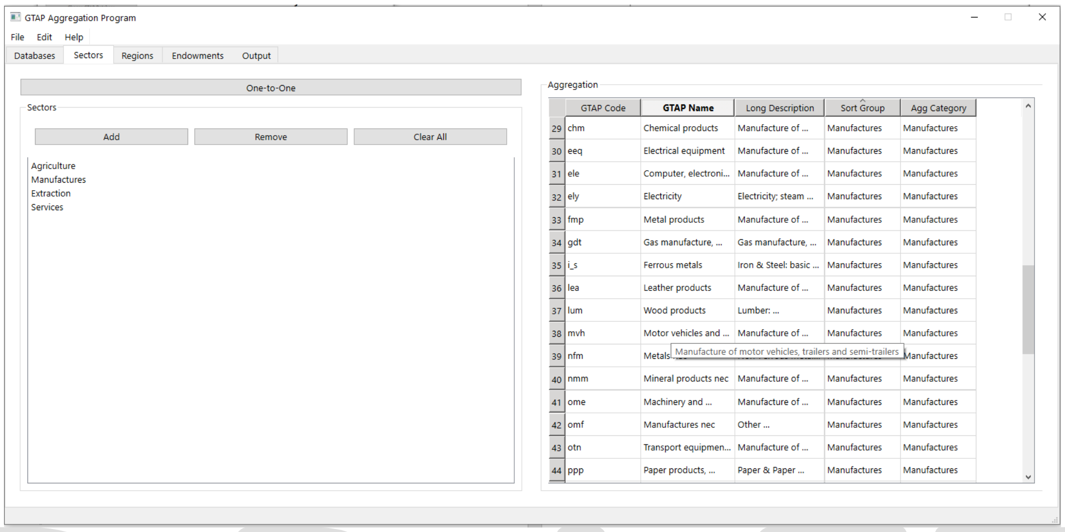
The headers may be resized or hidden, as in a spreadsheet. Hovering over a row will provide a long description of the GTAP sector.[12]

The sectors frame allows the user to define sectors for their aggregation. These sectors are used in the menus when a user "paints" a value in multiple rows and are used to validate entries (a helpful feature adopted from the legacy program GTAPAgg2). Buttons are provided to add, remove, and clear the entries.

[12] Future development will include one click access to the HS, CPC and ISIC codes associated with a sector, steps which previously required long look ups in international documentation.

Figure 5

Hovering for Descriptions



Agriculture
Manufactures
Extraction
Services

	GTAP Code	GTAP Name	Long Description	Sort Group	Agg Category
29	chm	Chemical products	Manufacture of ...	Manufactures	Manufactures
30	eeq	Electrical equipment	Manufacture of ...	Manufactures	Manufactures
31	ele	Computer, electroni...	Manufacture of ...	Manufactures	Manufactures
32	ely	Electricity	Electricity; steam ...	Manufactures	Manufactures
33	fmp	Metal products	Manufacture of ...	Manufactures	Manufactures
34	gdt	Gas manufacture, ...	Gas manufacture, ...	Manufactures	Manufactures
35	i_s	Ferrous metals	Iron & Steel: basic ...	Manufactures	Manufactures
36	lea	Leather products	Manufacture of ...	Manufactures	Manufactures
37	lum	Wood products	Lumber: ...	Manufactures	Manufactures
38	mvh	Motor vehicles and ...	Manufacture of ...	Manufactures	Manufactures
39	nfm	Metals			Manufactures
40	nmm	Mineral products nec	Manufacture of ...	Manufactures	Manufactures
41	ome	Machinery and ...	Manufacture of ...	Manufactures	Manufactures
42	omf	Manufactures nec	Other ...	Manufactures	Manufactures
43	otn	Transport equipmen...	Manufacture of ...	Manufactures	Manufactures
44	ppp	Paper products, ...	Paper & Paper ...	Manufactures	Manufactures

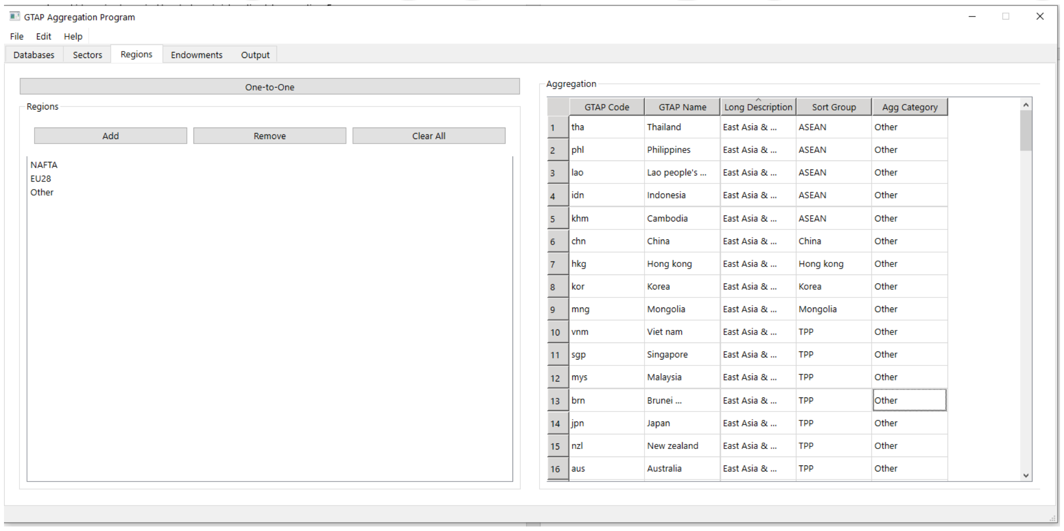
Manufacture of motor vehicles, trailers and semi-trailers

Regional Mapping (Region Tab)

Figure 6 shows the regional tab. It functions in a similar way to the sector tab. Features such as sorting based on headings, paint and paste fill-in, and augmenting fields for sorting are also included. Users may wish to change the .json file to include their own regional groupings. Like the Sectors tab, only the first and last columns must be specific data: the first column is the GTAP region codes and the last column is the aggregation codes.

Figure 6

Regional Mapping



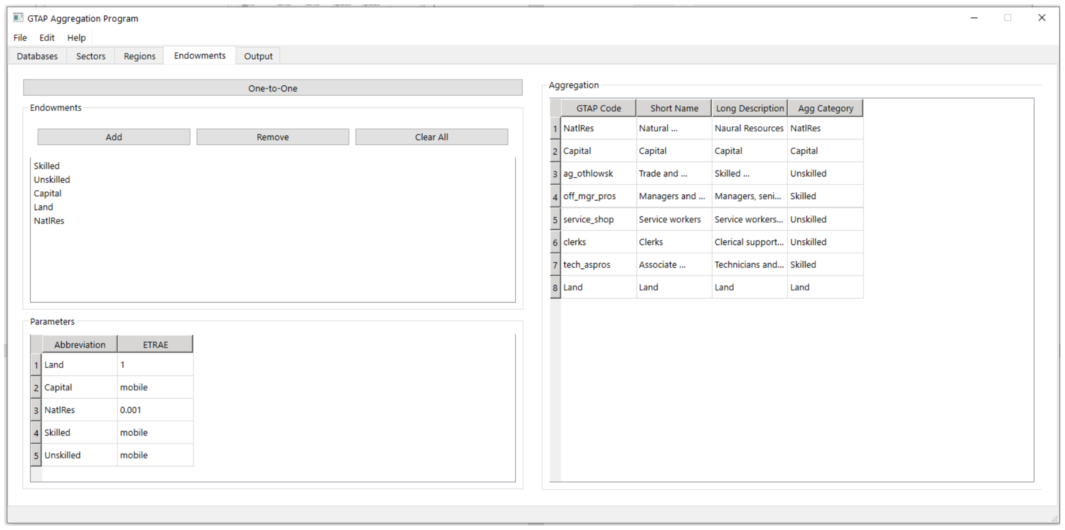
NAFTA
EU28
Other

	GTAP Code	GTAP Name	Long Description	Sort Group	Agg Category
1	tha	Thailand	East Asia & ...	ASEAN	Other
2	phl	Philippines	East Asia & ...	ASEAN	Other
3	lao	Lao people's ...	East Asia & ...	ASEAN	Other
4	idn	Indonesia	East Asia & ...	ASEAN	Other
5	khm	Cambodia	East Asia & ...	ASEAN	Other
6	chn	China	East Asia & ...	China	Other
7	hkg	Hong kong	East Asia & ...	Hong kong	Other
8	kor	Korea	East Asia & ...	Korea	Other
9	mng	Mongolia	East Asia & ...	Mongolia	Other
10	vnm	Viet nam	East Asia & ...	TPP	Other
11	sgp	Singapore	East Asia & ...	TPP	Other
12	mys	Malaysia	East Asia & ...	TPP	Other
13	brn	Brunei ...	East Asia & ...	TPP	Other
14	jpn	Japan	East Asia & ...	TPP	Other
15	nzl	New zealand	East Asia & ...	TPP	Other
16	aus	Australia	East Asia & ...	TPP	Other

Endowment Mapping (Endowment Tab)

Figure 7 shows the endowment mapping tab. This tab includes an extra frame to change the corresponding endowment mobility parameters. The aggregation frame works in a similar way as the sector and region aggregation frames, the first and last columns are the key information: the GTAP endowment code and the endowment aggregation on the right. Values in the middle may be used to sort and inform the user's aggregation decisions. Adding or removing a value from the "Endowments" frame adjusts the "Parameters" frame accordingly.

Figure 7
Endowment Mapping



Skilled
Unskilled
Capital
Land
NatlRes

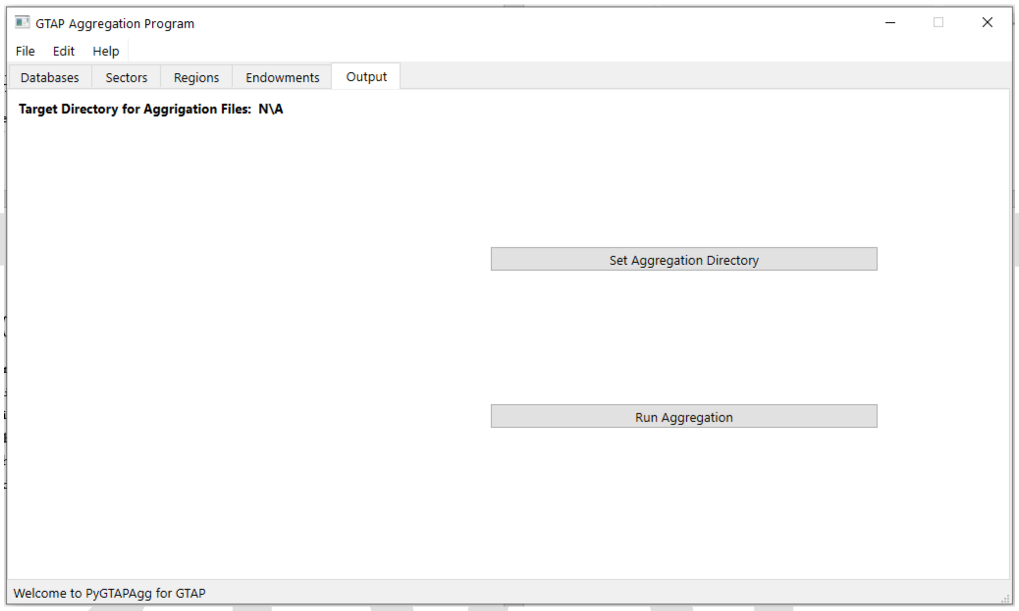
	Abbreviation	ETRAE
1	Land	1
2	Capital	mobile
3	NatlRes	0.001
4	Skilled	mobile
5	Unskilled	mobile

	GTAP Code	Short Name	Long Description	Agg Category
1	NatlRes	Natural ...	Naural Resources	NatlRes
2	Capital	Capital	Capital	Capital
3	ag_othlowsk	Trade and ...	Skilled ...	Unskilled
4	off_mgr_pros	Managers and ...	Managers, seni...	Skilled
5	service_shop	Service workers	Service workers...	Unskilled
6	clerks	Clerks	Clerical support...	Unskilled
7	tech_aspros	Associate ...	Technicians and...	Skilled
8	Land	Land	Land	Land

Creating a New Database (Output Tab)

Figure 8 shows the database output tab. This is where the user specifies the location of the aggregated files on their system. Click on Set Aggregation Directory and choose any valid directory on your system. The location will be indicated in the Target Directory for Aggregation Files space at the top of the tab.

Figure 8
Database Output Tab

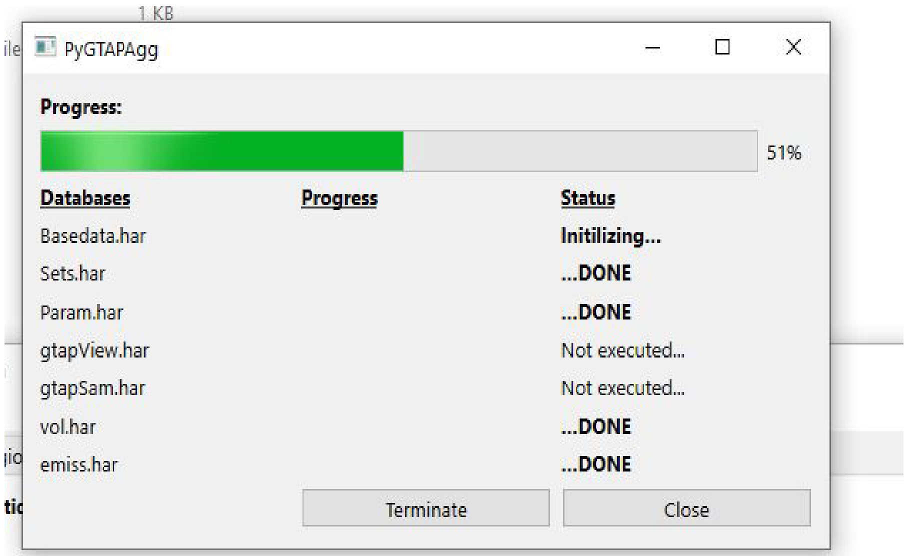


Target Directory for Aggrigation Files: N\A

Welcome to PyGTAPAgg for GTAP

Running the aggregation will bring up a status window, so the user can see the progress of the aggregation. Although they aggregation process looks the same as GTAPAgg2, it is different in PyGTAPAgg for several reasons and hence why the status window is important. In the past, running an aggregation would occasionally fail and it would take time examining log files to understanding where in the aggregation process something went wrong. The status window provides immediate feedback to the user on the progress of the various aggregation programs. Perhaps more importantly, PYGTAPAgg employees multithreading to implement concurrent processes (parallel processing). This greatly speeds up the aggregation of large data sets, since processes do not have to be run in a series and can utilize the now common, numerous cores on a computer. It's important to understand that some of the aggregation programs may at times try to access the same files before they are released by another system. This may cause an error (race conditions are very hard to predict). While these types of errors are no more common than any other run time processes, they may occur the user must be aware of when a process did not complete.[13] Rerunning the aggregation will usually solve the error. In the case a program "hangs" for some reason (it could be a race condition, a runtime error in the aggregation program, the data etc.), the user will have to press "terminate" to execute a termination signal to the operating system.

Figure 9
Aggregation Status



Databases	Progress	Status
Basedata.har		Initilizing...
Sets.har		...DONE
Param.har		...DONE
gtapView.har		Not executed...
gtapSam.har		Not executed...
vol.har		...DONE
emiss.har		...DONE

Finally, to view the files, the standard process of opening up the directory and using viewHar is the recommended method. PyGTAPAgg does not maintain a separate viewing application for HAR files.

[13] Numerous processes protections and careful sequencing of concurrent processes have been undertaken to minimize these events, but they cannot, in theory, be eliminated without greatly reducing the advantages (speed) or multithreading the operations.

Productivity Features

Several productivity enhancing features have been introduced into PyGTAPAgg and they are reviewed here.

Aggregation Files (JSON)

The JSON format for data is a semi-structured file format. A complete description of the JSON format is beyond the scope of this document. Users can find documentation on the web. For a person unfamiliar with the JSON format, it might take 30 minutes to review the basics. Semi-structured data provides a means for storing data and retrieving it, without strict requirements that each element have the same format. That is the strength of JSON, its flexibility and that is leveraged in the PyGTAPAgg program. Its important to understand that the JSON format more closely follows concepts of programming data objects and hierarchy then a simple square table. Python users will immediately recognize the direct analogy to lists and dictionaries. As such, the best method for editing JSON files is via a program like Python, R or a database program. Most programing languages provide robust tools for importing and exporting data as JSON format. Excel user may also import JSON with queries; however, this is not the preferred method. Users may also find editing the JSON file directly helpful, it is not as sensitive to editing errors as the legacy tab or comma delimited format. Users may enter their own fields within the usual headers and data file, as long as the first and last value the GTAP code and the aggregation code respectively.

Sorting

Perhaps the most powerful, and underappreciated, visualization and analysis tool is the simple “sort”. PyGTAPAgg leverages sorting by implementing simple sorting by clicking the column heads. Sorting can greatly speed the creation of aggregation. It can also speed the identification of outliers and errors.

Customized Groupings

As mentioned in the JSON section, user can add columns to their aggregation window by inserting them into the JSON file between the first and last columns.

Customization and Development

File Organization and Code Documentation

Open-Source Contributions

Licenses