

GHG Emissions Tracker Documentation Manual

Contents

Objective	2
Data type and format	2
Setup.....	3
The process through the 12 R programs.....	3
1. coef6c.r (or coefXX.r – XX refers to file revision)	4
2. eia-api-11b.r (or eia-api-XXX.r – where XXX refers to the file revision).....	5
3. qwi33d.r (or qwiXXX.r – XXX refers to file revision).....	6
4. eia-elec-emi-calc-22e.r.....	7
5. Flight11f.r	8
6. ind11h.r	9
7. Com20f.r	10
8. res17c.r.....	11
9. Tran10h.r.....	13
10. Forest5i.r.....	14
11. Ag14g.r.....	15
12. Master34a.r.....	17
Maintenance details	18
1. File revisions	18
2. Data sanity check	18
3. Update and create a Power BI report	19
4. Create embed link and publish the dashboard.....	26

Objective

Drawdown Georgia Emission Tracker (“The Tracker”) is a visualization tool that provides month-by-month snapshots of emissions in both the State of Georgia and its 159 counties. The Tracker is built on the Microsoft Power BI platform, but the calculations are done in R statistical language. The end output from emission calculations in R is a single spreadsheet file called Extended County Master file or *ecomaster.csv* which will then become the input of Power BI. This *ecomaster.csv* file needs to be re-calculated and updated every month to keep the Tracker up to date.

Data type and format

The *ecomaster.csv* file is in long format (meaning each row refers to one variable-value pair). There are four columns making up this file:

1. geocode – Geocode contains one character and five-digit FIPS (Federal Information Processing Standard) code for the area. All geocode values in this dataset starts with either g13XXX which refers to counties in the state of Georgia or g01XXX which refers to counties in the state of Alabama. Alabama is included in this tracker as it is from where Georgia imports a sizeable amount of electricity each month.
2. ldate – This column contains a string of dates in year and month.
3. varname – This column contains a string of SEDS data (State Energy Data System). It comprises of four components, each separated by ‘.’ (period):
 - i. Abbreviated variable name – the first portion of this string described the datum in its corresponding row e.g., ‘emi’ stands for ‘CO₂ emissions’, ‘pop’ for ‘population’. It is varied in length.
 - ii. Fuel – this describes the energy source of the object in the first component, or action which generates the value in *datavalue* (usually fossil fuels used to generate electricity).
 - iii. The two-character EIA sector – this component in the string describes the origin of the object in the first component of this string. It is represented with abbreviated names of the responsible sectors.

<i>Abbreviation</i>	<i>Sector</i>
rc	Residential sector
cc	Commercial sector
ic	Industrial sector
ei/eg	Electricity
ac	Transportation
tc	Total?
ag	Agricultural sector
 - iv. Unit – the last component of this string provides the unit of measure of the amount in *datavalue*. It is represented by abbreviations such as ‘mt’ for ‘metric ton’, ‘mmt’ for ‘million metric ton’ or ‘mW’ for ‘million (mega) Watts’.
4. datavalue – This column contains numerical data for the variables in *varname*.

Setup

The ecomaster.csv file (the main output for Power BI) is the product of 12 R programs that need to be run in order. Setting the file directory to match what is currently written in the R codes will ensure that these programs can run smoothly with minimal edits to the codes. The reason behind this is that each R program will generate multiple output objects. So, in order to keep things separated, each subfolder is needed to receive these files. Ultimately, the work directory for the dashboard should be set up in a way that you have the subfolders as shown in the screenshot below. The main directory should be set in something like this:

C:/.../dashboard

You can name the main folder any way you like. Just please make sure that in that folder, you have all the following subfolders:

- /code – this folder contains .r files
- /outputs – this folder is the destination for all the output files
- /functions – this folder contains some custom r functions Dr. Drummond wrote for data wrangling
- /static – contains static, less-frequently updated, data needed as inputs for the r programs.

The process through the 12 R programs

As mentioned above, these R programs need to be executed in order as the outputs of certain programs will become inputs for the subsequent ones (See Figure 1). Each of these R programs will list the packages required to run at the top of the script, so make sure to install packages using:

```
install.packages('package1', 'package2'...)
```

If you use RStudio as the IDE for this project (recommended), it should recommend you install the packages listed in these R programs but not yet installed on your machine.

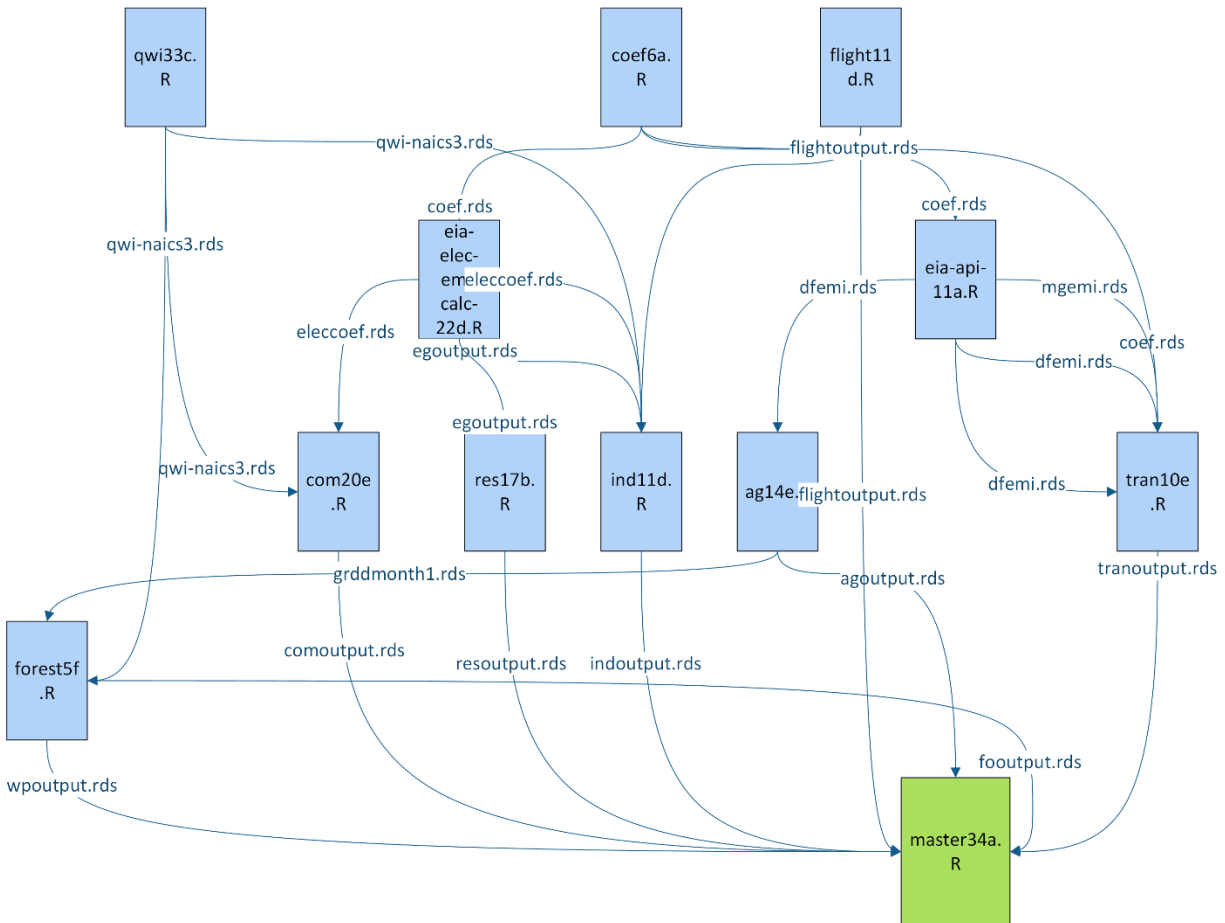


Figure 1 Tracker calculation diagram

1. coef6c.r (or coefXX.r – XX refers to file revision)

Inputs:

- /functions/apikeyeia.txt

Outputs:

- /outputs/coef.rds (and its “version” copy)
- /outputs/coefraw.rds (and its “version” copy)

The first file will calculate the emission coefficients for every process the dashboard will later introduce. The program uses EIA’s Open Data API to obtain the most up-to-date emission information directly from eia.gov. To obtain the API key, please go to www.eia.gov/opendata and follow the instructions there.

Once obtaining the API key, copy and paste the key into apikeyeia.txt in the *functions* folder. The program needs this key to access and obtain data from the EIA database. If you had set the working directory per step 3, you should see four output files in the folder *outputs*, two general ones (coef.rds and coefraw.rds) and two files with version number (coefXX.rds and coefrawXX.rds).

The file uses `geteia()` command to pull data from the EIA database based on the parameters the program specifies (CO2 coefficients series list from Georgia and Alabama). This command takes a brief moment to run, and your machine needs an Internet connection before you run this. (This returns ‘coef1’ data frame.)

The `coef1` data frame is then go through some data wrangling process in the next part of the code. The variable `varname` is generated here, using Dr. Drummond’s `makevar()` function which creates a standard four-part variable name by taking an input of four-character strings (or variable names representing character strings) and inserting periods between them. The variable `year` is created by pulling out year information from `ldate` variable. This returns ‘coef2’ data frame.

The final step in this R program is to create all the possible combinations of `varname` at all possible locations (counties, based on `geocode`) and at the different time periods (`year`). This breaks down the information obtained from the EIA database into different sectors, times, etc. The output data frame is ‘coef3’, which is then saved as ‘outputgeneral’ and ‘outputversion’ in the ‘outputs’ folder.

2. eia-api-11b.r (or eia-api-XXX.r – where XXX refers to the file revision)

Inputs:

- /functions/apikeyeia.txt

Outputs:

- /outputs/ngemi.rds (and its “version” copy)
- /outputs/mgemi.rds (and its “version” copy)
- /outputs/dfemi.rds (and its “version” copy)
- /outputs/rfemi.rds (and its “version” copy)
- /outputs/clemi.rds (and its “version” copy)
- /outputs/ffemi.rds (and its “version” copy)
- /outputs/diesel-sector-pcts.rds (and its “version” copy)

This program calculates emissions from different fuel types such as natural gas (ng) and distillate fuel oil (diesel, df) used in various sectors in the calculation. Similar to the first program, it uses EIA’s Open Data API to obtain emission data from the EIA website.

Part 1 – Natural gas

The first part of this program calculates Georgia natural gas emissions for residential, commercial, and industrial sectors. The program first pulls Georgia’s annual natural gas heat content in BTU per cubic feet (bcf). This is done for the entire state (`geocode` is set at “g13000”). The program then converts bcf into million metric tons (mmt) of CO₂-eq emissions and assigns to specific sector (`ic` = industrial, `rc` = residential, `cc` = commercial) using the coefficients calculated for different activities in these sectors in `coef.rds`. Lastly, the program formats this data frame (`ngemi5`) to match the structure of the `ecomaster` file. Notice the `makevar` command (from Dr. Drummond’s `functions.r` file), the command generates the 4-part `varname` that is used everywhere else in this project.

Part 2 – Distillate fuel oil

This part of the program starts with obtaining annual distillate fuel oil consumption in the state of Georgia from the EIA website using the Open Data API. This information is already broken down into sectors and varieties of diesel fuel as noted in the program (right under `setwd()` command). The program then translates the `seriesid` from the EIA database to the abbreviated sector code used in this project. Since the EIA database breakdowns this diesel use finer than what DrawdownGA project is using, some of the EIA sectors will be assigned the same sector for the project (e.g. “comls”, “comhs”, and “comvls” are all assigned “cc” as in this project). Furthermore, since the diesel consumption obtained from the EIA is in thousands of gallons, the program first converts the values into millions of gallons before applying the emissions coefficients. The program converts gallons of diesel used into heat (BTU per gallon) and then CO₂ emission (million metric tons).

Similar processes are repeated for different fuel types (get the amount of fuel consumption for the state of Georgia from the EIA database, convert the fuel into emissions by applying heat coefficient and then emission coefficient): `mg` (gasoline – all assumed to be used in the transportation sector), `cl` (coal – industrial sector), and `rf` (refined fuel – all assumed to be used in the industrial sector). All these emissions are calculated at the state level (“g13000” geocode), for each month, fuel type, and sector.

3. `qwi33d.r` (or `qwiXXX.r` – XXX refers to file revision)

Inputs:

- `/static/qwi/qwi_current.csv`
- `/static/ldatemaster.rds`

Outputs:

- `/outputs/qwi-master.rds` (and its “version” copy)
- `/outputs/qwi-naics3.rds` (and its “version” copy)
- `/outputs/qwi-comindemp.rds` (and its “version” copy)
- `/outputs/qwi-comindtot-emp.rds` (and its “version” copy)

This file calculates Quarterly Workforce Indicators. The outputs of this R program provide information on workforce in industrial and commercial sectors, which, in turn, affect the emissions from these activities. The workforce data is obtained from the US Census Bureau, and it updated every quarter. The Census Bureau supports data access via API; however, it appears to be very slow. This R program instead makes use of a CSV file that requires direct download from the Census Bureau. To obtain this file, please use the LED extraction tool (<https://ledextract.ces.census.gov/static/data.html>) and follow the options below:

Data from LED extraction tool

- ```
https://ledextract.ces.census.gov/static/data.html
Check Georgia, Counties, Check All (below under "Industries")
Naics 3-digit, Check All, Select "All Ownership" under Firm Ownership
```

```
All workers
All for Emp variables (five checks)
All years/quarters 2000-present (left-most checkboxes)
Submit request
Download CSV & Metadata (ZIP)
```

Next, create a subfolder within folder *qwi* and name it *data*. Unzip the downloaded CSV file from the Census Bureau and put it in *data* folder. Copy the filename of this CSV file and edit the line of code below:

```
indata = "../static/qwi_0e0f26a654e840468be943f82a9ba05f.csv"
```

You need to change the highlighted part of this code to the filename of the downloaded CSV file, once this is done, the rest of the file should be executable.

The first part of this program (line 50-64) creates a data frame from the downloaded CSV with a structure similar to the final output *ecomaster.csv* i.e. geocode, varname, ldate, and datavalue. The file from the Census Bureau contains much more information than this. Thus this part of the code selects the relevant variables and formats them.

The next part of the program involves forecasting adjustment. The QWI data is quarterly, while our emission calculation is done monthly. Therefore, we use the *stl4* function to interpolate the workforce information in-between months not reported in QWI data. The function makes use of the package ‘forecast’ and include seasonal adjustment. The QWI data has the information for the beginning months of each quarter, and the program interpolates and fills the values for the middle months. This monthly workforce information is then assigned to specific sectors.

#### 4. eia-elec-emi-calc-22e.r

Inputs:

- /functions/apikeyeia.txt
- /outputs/coef.rds

Outputs:

- /outputs/egoutput.rds (and its “version” copy)
- /outputs/eleccoef.rds (and its “version” copy)

This program calculates emission coefficients from electricity. Since Georgia imports a considerable amount of electricity from Alabama, the emissions from electricity consumption for Georgia require more steps to reflect where and how electricity is generated and consumed. The steps to the calculations go from:

Part 1 – calculate each sector’s monthly percentage of Georgia retail electricity sales.

Similar to the previous program, the *setwd()* command needs to be edited to match the working directory on your machine and the hierarchy shown in the “Setup” part of this manual. This part

of the program obtains electricity sales numbers from the EIA open database and covers all the relevant sectors. The program then codes the EIA sectors (three letters) to the Drawdown GA format (two letters) for compatibility. The product of this stage are data frames containing electricity sales in Georgia and Alabama, for each month since January 2001, both separated by sectors (*elecsales2*) and total sales (*elecsalestotal*).

#### Part 2 – Calculate Georgia and Alabama fossil fuel CO<sub>2</sub> emissions

This step calculates the CO<sub>2</sub> emissions from electricity generation in both GA and AL. The strategy is similar to previous emissions calculations from the EIA Open Data. We convert the fossil fuels to heat and then emissions per the coefficients, then group them by state, month, year, and fuel types.

#### Part 3 – Calculate GA and AL export emissions per export unit

This step calculates the emissions from the power plants in both GA and AL. This step provides “the stock” of emissions based on the electricity generation, which, when subtract from the numbers in part 1, provide the information of how much emissions are from the imported electricity from AL. This step also accounts for power loss during the transmission and distribution of electricity (estimated 5.5% loss). Moreover, we also account for the actual sources of electricity GA purchases from AL by excluding the zero-emission electricity which is sold in TN rather than GA.

#### Part 4 – Calculate GA electricity emissions with imports

Using the information from part 3, this step combines the emissions from electricity generations in GA with those from AL. The numbers here represent the emissions from electricity consumptions in GA.

#### Part 5 – Distribute emissions to sectors

Lastly, this program uses the electricity sales proportion (calculated in part 1) to assign the emissions to each sector.

The output files from this R program are *eleccoef.rds* (GA electricity emissions and imports) and *egoutput.rds* (by-sector electricity emissions).

### 5. Flight11f.r

Inputs:

- /static/ghgp\_data\_by\_year\_2020.xlsx
- /static/gacounty.xls

Outputs:

- /outputs/flightoutput.rds (and its “version” copy)

This program calculates flight industry emissions i.e., emissions from large industrial facilities NOT including electricity generation (which we have a separate program for). The program utilizes the EIA’s GHG Reporting Program datasets that can be downloaded from:



<https://www.epa.gov/ghgreporting/ghg-reporting-program-data-sets>

Once you get to this site, download the Data Summary Spreadsheets zip file (the most current one is from 2020). The zip file contains spreadsheets on GHG emissions from large emitting facilities all the way back to 2010. And we will use the file called “ghgp\_data\_by\_year.xlsx” which covers every year that has been reported.

In the code, you’ll notice that the program takes in this file by the name “ghgp\_data\_by\_year\_2020.xlsx” (line 23). So, we’ll add “\_2020” to the file that we unzipped from the downloaded file and put this data file in the “flight” folder. This is done so we know which version of the GHG report it is used.

After renaming and putting the input file in the flight folder, run the rest of the program (don’t forget to setwd() to the correct file path first). The program will take in the emissions data from the input file, filter out the non-GA facilities, and calculate the emissions from those in Georgia. It then combines the emissions for specific sectors, geocode, and years as reported in this project. The output file (*flightoutput11f.rds* and *flightoutput.rds*) will be in the same four-column format as *ecomaster.csv*.

## 6. ind11h.r

Inputs:

- /static/qwi\_current.csv (this should be the same file used in qwi calculations)
- /outputs/coef.rds
- /outputs/eleccoef.rds
- /outputs/egoutput.rds

Outputs:

- /outputs/indoutput.rds (and its “version” copy)

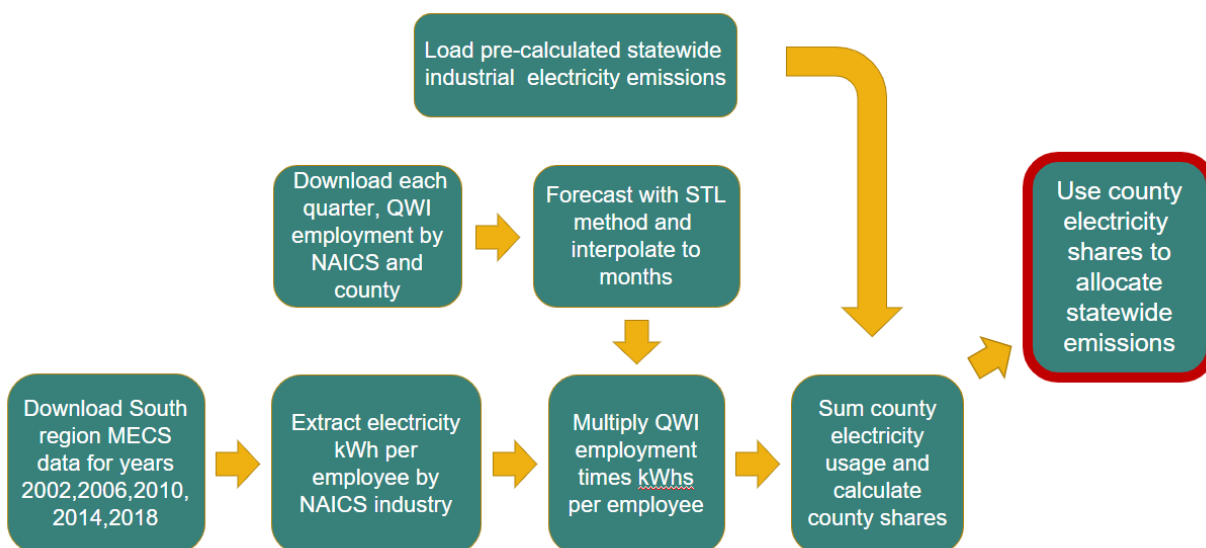


Figure 2 Calculation pathway for industry sector

This program calculates the emissions from electricity consumption from economic activities based on the Quarterly Workforce Indicators numbers. The program works in two steps. First, following the bottom line in the diagram, it calculates the electricity consumption per employee based upon the Manufacturing Energy Consumption Survey (MECS) total consumption of electricity (for electricity purchased) and QWI for employee number. In this step, there are two things to configure in the code.

- One, update the qwi CSV file to the latest one (the same one used in the QWI calculation).
- Two, the MECS data needs to be downloaded. Currently, the codes from lines 64-67 can get this data and feed it directly into the program (the program has these lines commented out by default). This is the MECS data from 2018. If there is a new version of this file, this will need to be addressed (consult with Dr. Drummond).

In this first step, the qwi data is used to calculate numbers of employment by NAICs code and county. The employment number is then being interpolated to monthly employment number. Next, the electricity purchased data (from the MECS data) is used to calculate the electricity consumption per 1,000 employees.

The second step of this program calculates local industry electricity usage and county share from monthly QWI employment and kwhkperemp numbers (from step 1). These county shares will then multiply with statewide emissions to compute county-level industry sector emissions.

## 7. Com20f.r

Inputs:

- /functions/apikeyeia.txt
- /static/climate\_zones.csv
- /static/ldatemaster.rds
- /static/2012microdata\_codebook\_v2.xlsx
- /static/naicsbecswalk.xls
- /static/PBAvsNAICS.xls
- /static/2012\_public\_use\_data\_aug2016.xlsx
- /outputs/qwi-naics3.rds
- /outputs/eleccoef.rds
- /outputs/coef.rds

Outputs:

- /outputs/comoutput.rds (and its “version” copy)

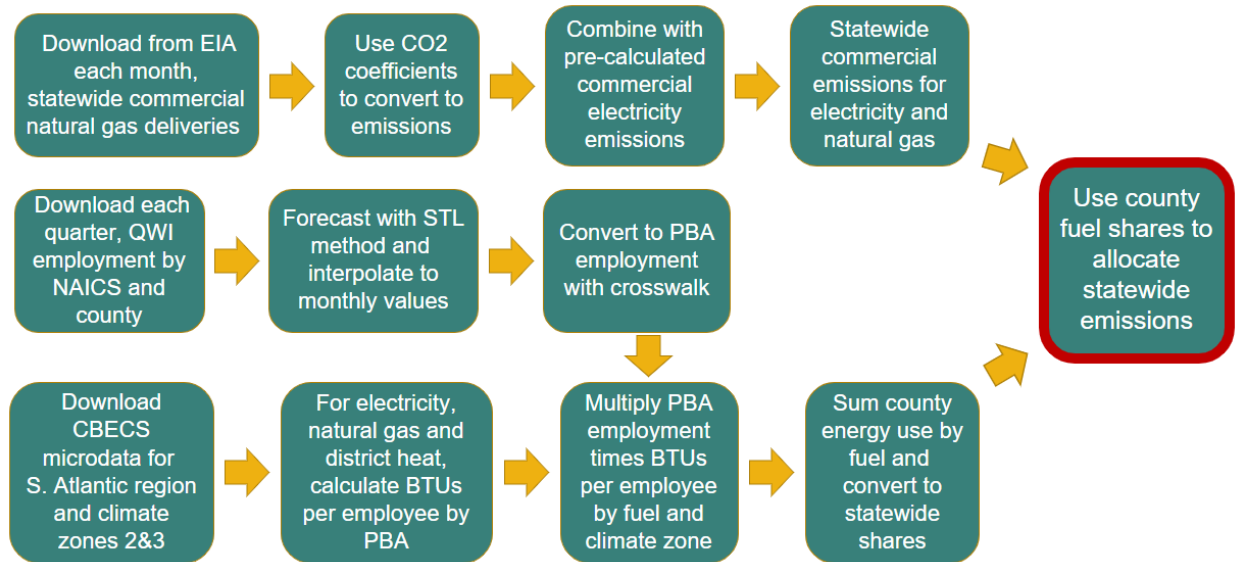


Figure 3 Calculation pathway for commercial sector

This program calculates emissions from energy consumption (multiple sources e.g., electricity, natural gas, and fuel oil) from activities in the commercial sector. The general strategy here is to calculate state-wide commercial emissions from energy consumption, then allocate these emissions down to the county level based on employment data. The program takes into account differing climate zones GA has as well as the different types of fuel used.

After the regular setup (adjust the `setwd()` to your work directory), the program will call several spreadsheet inputs from the *static* folder, so if certain lines of codes don't run properly, check if the input files are in the static folder (contact Dr. Drummond for the missing files). These inputs include:

- `climate_zones.csv` (GA has two climate zones, and they affect energy consumptions)
- `ldatemaster.rds` (it formats the date to match what the project uses)
- `2012microdata_codebook_v2.xlsx`, `naiccbecswalk.xls`, and `PBAvsNAICS.xls` are the codebook files to help translate NAICS county employment data (aggregate) to CBECS Principal Building Activities (PBA) microdata.
- `2012_public_use_data_aug2016.xlsx` contains PBA microdata per building, which is used to calculate energy use per employee factors for electricity, natural gas, and district heat.

The main output of this program is `comoutput.rds` (and `comoutputXXX.rds` for the version check). The output file follows the four-column format, with `varname` column represents emissions from the commercial sector in varying fuel types.

## 8. `res17c.r`

Inputs:

- /functions/apikeyeia.txt
- /outputs/coef.rds
- /outputs/egoutput.rds
- /static/recspercentcosts.rds
- /static/mly-hydd-normal.csv
- /static/mly-cldd-normal.csv
- /static/station-county.rds
- /static/dddata.rds
- /static/usa\_00067.xml
- /static/R12857510\_SL050.csv
- /static/R12857509\_SL050.csv
- /static/R12857507\_SL050.csv
- /static/R12857509\_SL050.csv
- /static/R12858850\_SL050.csv
- /static/R12858851\_SL050.csv
- /static/R12858852\_SL050.csv
- /static/R12859141\_SL050.csv
- /static/R12859148\_SL050.csv
- /static/R12859151\_SL050.csv
- /static/geocorr2014\_2118806872v2.csv
- /static/geocorr2014\_2118705668v2.csv

#### Outputs:

- /outputs/resoutput.rds (and its “version” copy)

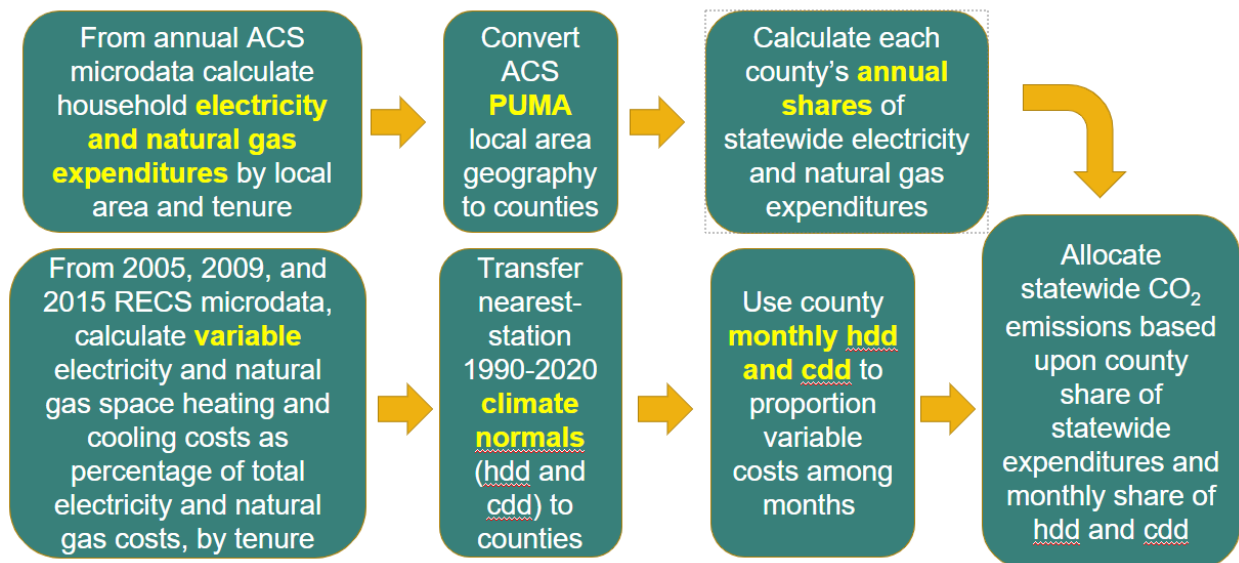


Figure 4 Calculation pathway for residential sector

This program calculates emissions in the residential sector, varying by fuel type. This calculation leverages the annual American Community Survey (ACS) microdata on electricity and natural gas

expenditures, the EIA’s Residential Energy Consumption Survey (RECS) microdata, and the NOAA Climate Normals data for local heating and cooling degree days (hdd and cdd, respectively) information to provide a close estimation of both variable (space heating and cooling per hdd and cdd) and fixed (such as lightings and refrigeration) energy expenditures.

In practice, the program has to reallocate much of these microdata to the county level (our unit of analysis). The ACS microdata has a smaller unit of analysis called PUMA, which is based on minimum of 100,000 households. In dense areas like Fulton County, it can have multiple PUMAs in a county. Whereas in less populated areas, it could take multiple counties to make up one PUMA, which makes allocating emissions to the county level tricky. The program thus uses 5-year aggregated ACS data for room numbers to help allocate households from multi-county PUMAs into individual counties.

## 9. Tran10h.r

Inputs:

- /static/trans/vmt\_statewide.rds
- /static/trans/vmt1.rds
- /outputs/mgemi.rds
- /outputs/dfemi.rds

Outputs:

- /outputs/tranoutput.rds (and its “version” copy)

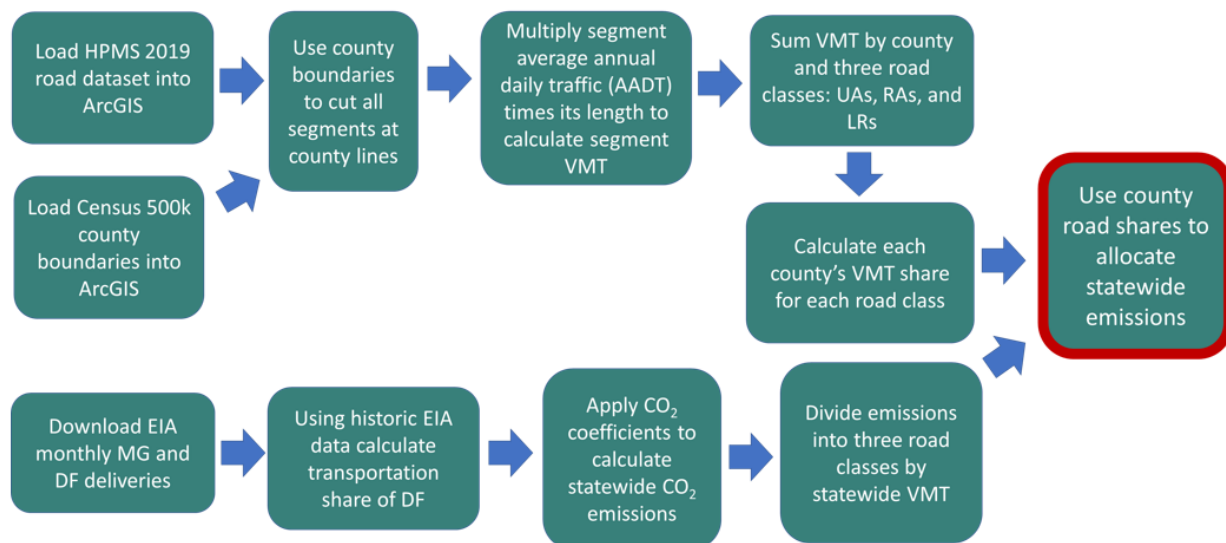


Figure 5 Calculation pathway for transportation sector

To calculate the emissions from the transportation sector, this R program follows two steps. Referring to the calculation step in the diagram above, the top line of calculation is to calculate the vehicle miles traveled (VMT), vary by three road types, in each county. The monthly Traffic Volume Trends (TVT) data from the Department of Transportation is used to divide traffic into shares for three different road types and by county. The bottom line of calculation total statewide

emissions based on fossil fuels consumed. The traffic shares per road type and county are then applied to the statewide transportation emissions to get the county-based transportation emissions.

In addition to the calculation strategy above, there are a few manual steps that require some edits in the R code. Since the Dashboard will update the numbers every month, some of these calculations need to be done monthly. Since this R program gets the TVT data from the USDOT website without the help of an API, these TVT data will need to be downloaded and added to the calculation manually. Dr. Drummond created the VMT dataset from 2005-2020 and stored it as `vmt1.rds` so this program doesn't need to load all these really old files. Anyway, to update this R program:

- Go to the URL for the monthly TVT data at [https://www.fhwa.dot.gov/policyinformation/travel\\_monitoring](https://www.fhwa.dot.gov/policyinformation/travel_monitoring) (also available in the R program). On this website, download the latest travel monitoring data file (the current month included in this `tran10h.r` file is April 2022).
- Once downloaded, change the R code per the following:
  - After the line that says “add new month(s) below” (currently line 80), add
 

```
mXX = getvmt(“newlydownloadedfile.xlsx”)
```

 where XX is the number of month added since January 2021). Basically, just follow the format of the existing code.
  - Also, add the new month(s) (what mXX that were just created) into `mnew1` object. (currently line 100)

## 10. Forest5i.r

Inputs:

- `/static/forest/sit-forest-8.xlsx`
- `/static/forest/gacounty.xls`
- `/static/forest/NLCDTotalsGACounties2000-2016.xlsx`
- `/static/forest/NLCD2019ForestryGACounty.xlsx`
- `/static/forest/grddmonth1.rds`
- `/outputs/qwi-naics3.rds`

Outputs:

- `/outputs/fooutput.rds` (and its “version” copy)
- `/outputs/wpoutput.rds` (and its “version” copy)

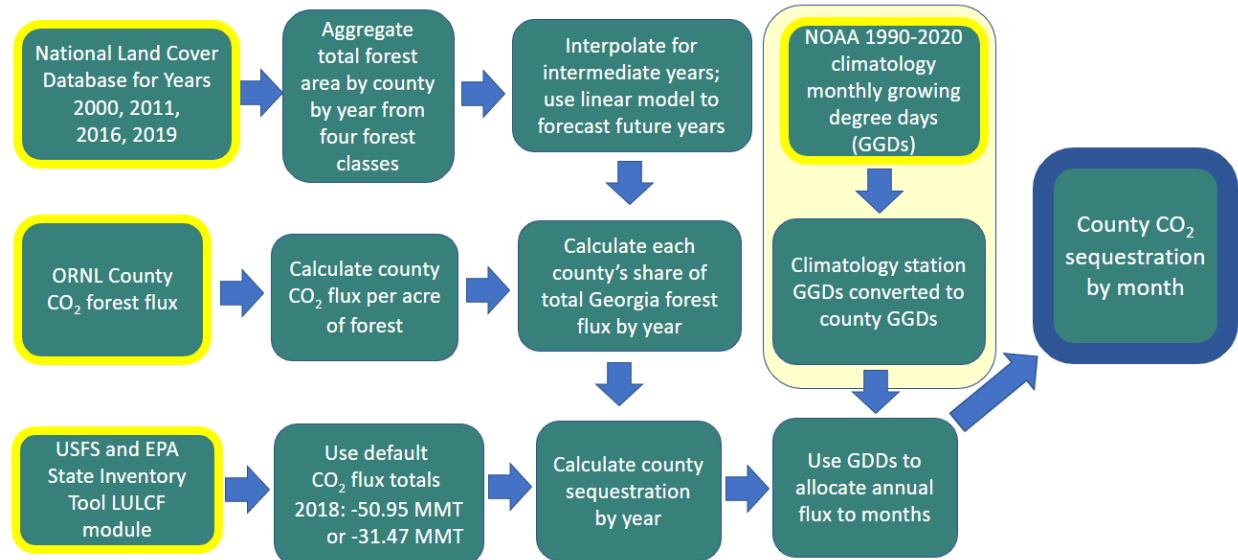


Figure 6 Calculation pathway for forestry sector

This program calculates carbon sequestration (uptake) in two forms: forest uptakes and wood products. Per the diagram above, the calculation in the program starts with the NLCD data (in the top row). The program aggregates total forest area by county and by year, from four classes of the forest. The program interpolates and fills the data between years that the data weren't available.

Secondly, the ORNL data is used to calculate county flux per acre, which, when multiplied by the forest area in each county, gives us the county flux. The proportion of each county's forest flux makes up the state's forest flux per year.

In the third row, use the State Inventory Tool data for the total statewide forestry flux. The product of statewide flux and the county flux share (from the second row) gives us the annual county sequestration numbers.

Finally, to allocate the amount of carbon sequestration for each month, we calculate each county's GDD (growing degree days), and assign the monthly share of GDD per year in each county. The product of this monthly GDD share per county and the annual county sequestration is monthly sequestration by county.

For the wood product sequestration, the State Inventory Tool data provides the amount of wood products the state produces each year. To assign this wood product to county and month, we will rely on the employment in the logging industry since the qwi calculation already has employment numbers by the industry, geocode, and months. The program first calculates the shares of logging employment in each county from the state's total logging employment. These shares will then be applied to the annual total wood product to calculate county wood product.

Both the *fooutput.rds* and *wputoutput.rds* will go to the *ecomaster.csv*.

## 11. Ag14g.r

Inputs:

- /static/ag/sit-ag-output-4.xlsx
- /static/ag/2017\_cdqt\_data.txt
- /static/ag/cropland.csv
- /static/ag/cows.csv
- /static/ag/hogs.csv
- /static/ag/chickens.csv
- /static/ag/sit-output-2.xlsx
- /static/stations.xlsx
- /static/ag/mly-grdd-normal.csv
- /static/ag/gacounty.shp
- /static/ag/station-county.rds
- /outputs/dfemi.rds

#### Outputs:

- /outputs/agoutput.rds (and its “version” copy)

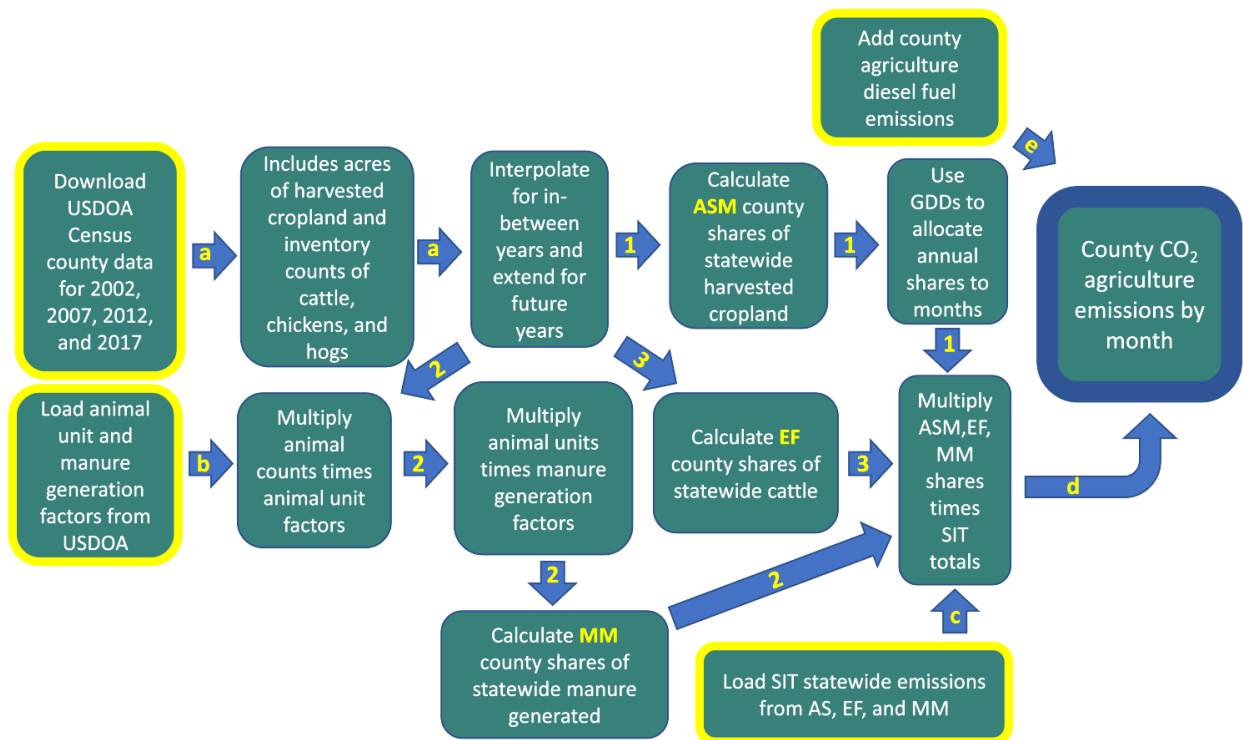


Figure 7 Calculation pathway for agricultural sector

This R program calculates the emissions from three different sources of greenhouse gases in the agricultural sectors: agricultural soil management (ASM), enteric fermentation (EF), and manure management (MM). All the calculations start at the same place, build the inventory of harvested cropland and livestock (cattle, chickens, and hogs). The data collected are in annual terms and thus they need to be interpolate for in-between years and extrapolate for the future years (following line “a” in the diagram above). Following line 1, once the inventory for the harvested cropland and livestock is done, the county annual shares of cropland is calculated and allocated the annual shares



to months based on the growing degree days. This establish the ASM shares for each county/month.

From the inventory, following line 2 of the diagram, the animal unit factors are obtained from USDOA and applied to the state livestock inventory numbers to calculate the state animal units. Then, the animal units are multiplied with manure generation factors to calculate county manure management (MM) share. Lastly, in line 3, the enteric fermentation (EF) share in each county is calculated from each county's shares of statewide cattle.

Total county agriculture sector emissions are calculated by multiplying shares of cropland, cattle, and manure generated with statewide SIT emissions. The county diesel fuel emissions are then added to make up the *agoutput.rds*, the output product of this R program.

## 12. Master34a.r

Inputs:

- /outputs/tranoutput.rds
- /outputs/comoutput.rds
- /outputs/indoutput.rds
- /outputs/resoutput.rds
- /outputs/agoutput.rds
- /outputs/fooutput.rds
- /outputs/wpoutput.rds
- /outputs/flightoutput.rds
- /static/master/msainfo06.xls
- /static/master/pop20.rds
- /outputs/masterdetail.rds

Outputs:

- /outputs/masterdetail.rds (and its "version" copy)
- /outputs/mastersector.rds (and its "version" copy)
- /outputs/comaster.csv (and its "version" copy)
- /outputs/ecomaster.csv (and its "version" copy)

This R program joins all the r objects calculated in previous steps. It also joins estimated populations in each geocode, at each time period (mm/yyyy), which will aid the presentation on the Dashboard. Notice that there are two .CSV files in the outputs. Both comaster.csv and ecomaster.csv (Extended comaster.csv) share the first five columns (*geocode*, *ldate*, *sector*, *datavalue*, *varname*). The comaster.csv contains all the emissions calculation in each sector (See *varname* description in "2. Data type and format" section above), and this file will be uploaded to [drawdownga.gatech.edu/data](http://drawdownga.gatech.edu/data) along with the updated Dashboard.

The extended comaster.csv (ecomaster.csv) contains more columns for the benefit of Power BI data visualization. Since R is much faster and more efficient at calculation than Power BI, the data are processed and calculated to ready-to-use format in R before being exported to Power BI. Most of the additional data in ecomaster.csv include redundant columns and rows that can technically

be derivable from the standard five columns data also found in `comaster.csv`. These columns include:

- year (derived from `ldate`)
- pop (population, new)
- `emimmt` (emissions in millions metric tons, derived from `datavalue`)
- `emimt` (emissions in metric tons, derived from `datavalue`)
- `emimtperpopk` (emissions in metric tons per 1000 people, derived from `datavalue` and `pop`)
- `emipctchange` (% year-over-year change in emissions, derived from `datavalue`)
- `msageocode` (present geocodes that are parts of Atlanta Metropolitan Statistical Area, derived from `geocode`)
- `emimilast12` (total emissions in metric tons in the last 12 months, derived from `datavalue`)
- `emimtyear` (total annual emissions in metric tons, derived from `datavalue`)
- `emimtyear2` (total annual emissions in metric tons with two-decimal points, derived from `datavalue`)

There are also row entries that represent the nuances of the data from certain sectors. The whole list can also be found on the [GaTech Dashboard page](#). These additional entries break down major sources of emissions, for example, agricultural soil management (ASM), diesel fuel (DF), enteric fermentation (EF), and manure management (MM) in the agriculture sector. These entries are separated from the total emissions and they are reported in ways that avoid double-counting the values. These broke-down emissions reports benefit the data visualization in Power BI as it provides more information than just lumping everything under one big category, as well as it helps put the spotlight on the right emissions sources.

## **Maintenance details**

### **1. File revisions**

- As a general rule, consult Dr. Drummond before making radical changes to the R codes.
- For non-master files, any small revisions will get an alphabet bump (ex. `res17c.r` → `res17d.r`). If there is a major change in the code (again, check with Dr. Drummond), the file will get a “number” update, and the alphabet gets a reset to “a” (ex. `res17c.r` → `res18a.r`)
- For the master files, the “number” update is reserved for the monthly run. Running the master file (after all the non-master files get their data updated) for each new month will result in a number update (and the alphabet reset). Ex.

`master33h.r` (old) → `master34a.r` (new calculations with updated data) → `master34b.r` (same data with some tweaks) → ...

### **2. Data sanity check**

The new `ecomaster.csv` should go through a data sanity check before using it to update the Dashboard. The following procedures are suggested, but they are not an exhaustive list. If the numbers in the data set seem to be majorly off, more sanity checks might be required.

Simple steps to do the data sanity check:

- Run `summary()` on the R object (in the `master.r` file, it's *m5plus* for *ecomaster* or *m5* for *comaster* files. This will present you with the summary statistics. These summary statistics from the updated *ecomaster/comaster* files shouldn't be too drastically different from the previous ones. This test checks for general oddities in the data.
- Compare the same last month from the two versions of the *ecomaster* files. Since updating the dataset mainly involves adding new month (s)/quarter of data into the calculation, it shouldn't affect the months in the past (or at least not so drastically). We can achieve this by filter out the numbers in the month of interest (in both files) and join them in R using `ldate`, `geocode`, and `varname` as keys. The joined table will should data entries in the same date, location, and variables. And the numbers between the 'old' (from the older *ecomaster*) and the 'new' from the updated *ecomaster*, should be fairly similar.
- Compare the numbers in the last month in the old *ecomaster* to the numbers from the latest month added in the new *ecomaster* file. Since we are now comparing two different months, the changes between the two sets of numbers are expected. However, the size of these emissions changes should be flagged if they are unreasonably large. (Dr. Drummond's suggested to pay attention to any emissions changes that are larger than 0.5 mmt (Mega metric ton)). Similar to the same-month data check, we can do this by joining numbers from the two data sets to form a new table, but `ldate` will no longer be a part of the joining key (since we are comparing two different months). So what we will be comparing is the emissions in the same location and same sector, but with different time periods. These two numbers will likely be different but pay attention to the degree of dissimilarity.
- If any insanity/irregularity in the data is spotted, try to see if there is a pattern in it e.g., which sector(s) show these oddities, then discuss your findings with Dr. Drummond or the team.

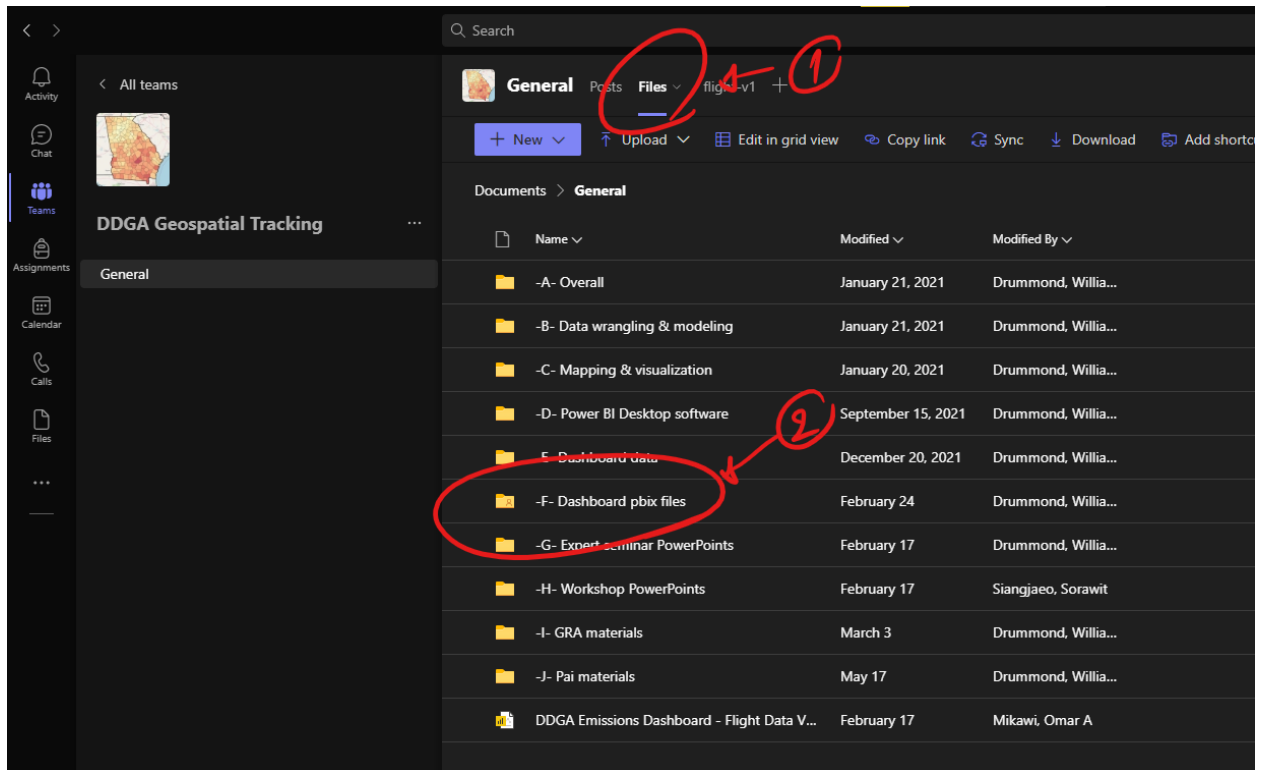
### 3. Update and create a Power BI report

Once the *ecomaster.csv* file is compiled and passed the data sanity check, the final task is to update the Drawdown Georgia Dashboard with the latest dataset.

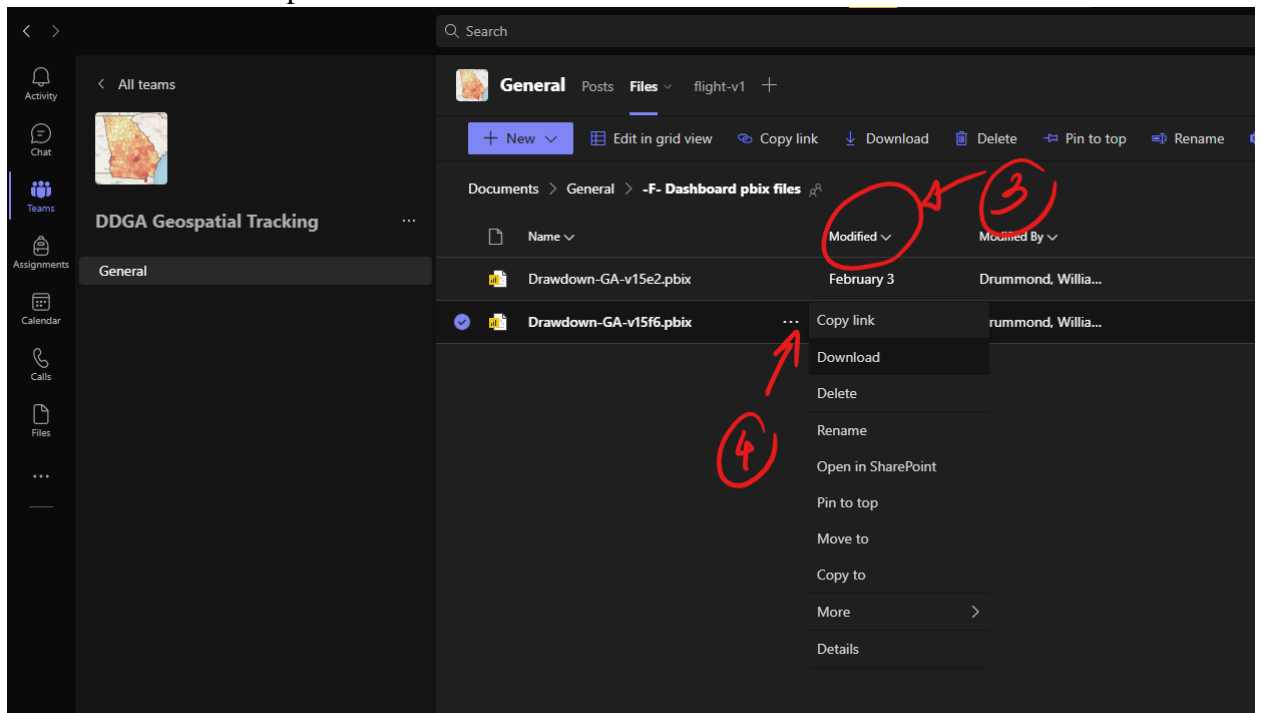
#### Update the Dashboard with the latest *ecomaster.csv*

This process will require you to have access to Power BI Desktop software (available only on Windows). Note that Power BI Desktop software is different from Power BI Service/App that is available on Microsoft Teams or [powerbi.microsoft.com](https://powerbi.microsoft.com). The two versions of software are not completely equivalent in terms of functionality, and you will need both to complete these tasks.

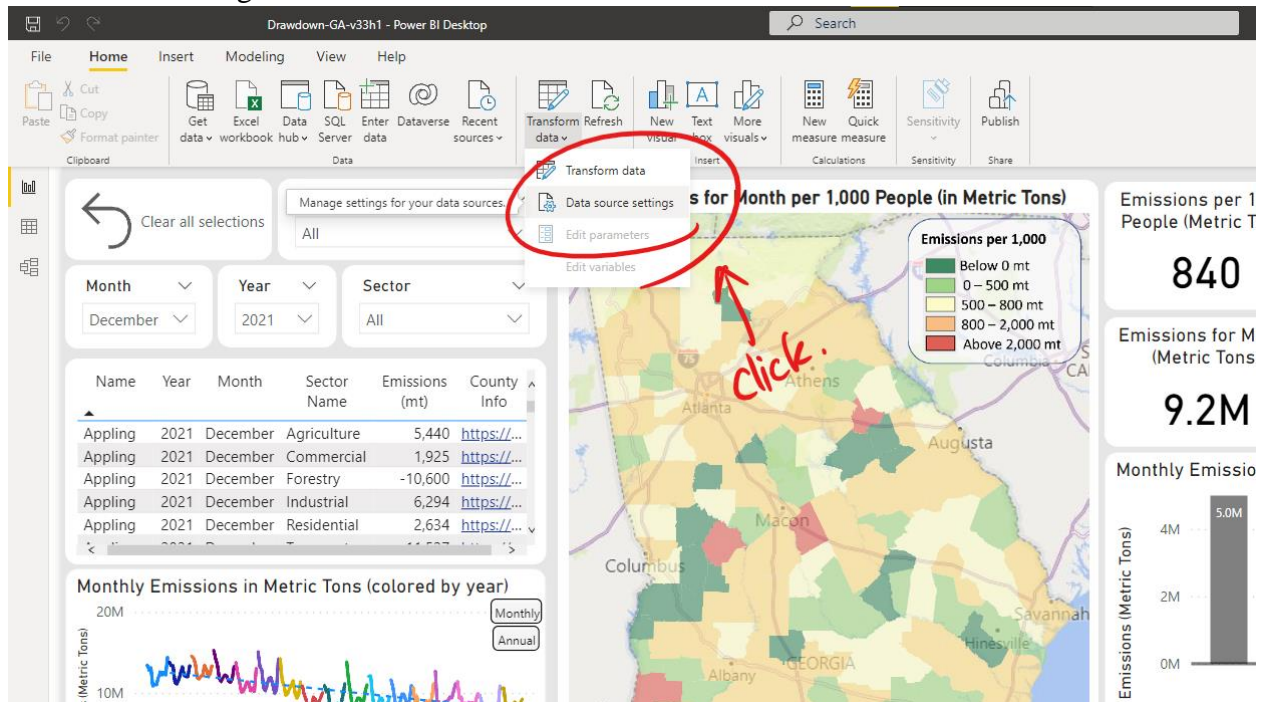
1. To update the Dashboard report file, first you will need to log in to Teams and go to DDGA Geospatial Tracking team. Click "Files" and select "-F- Dashboard pbix files" to download the Power BI report file (extension `.pbix`).



Once you are in the -F- folder, click on the “...” after the most recent file name (you may sort the file by the modified date) and download it. The most recent file should be Drawdown-GA-v34a.pbix.



- Open the .pbix file in Power BI Desktop software. The screen below should be the default screen. Otherwise, you can navigate the ribbon menu from Home → Transform data → Data source settings



The dialog box will pop up. Select the filename that you will update. In this case, it's the ecomaster.csv, but you can follow the same process if you need to update other files besides the ecomaster one. Click "Change Source..." to select the new ecomaster file to replace the existing one in the report.

**GHG Emissions for Month per 1,000 People (in Metric Tons)**

Emissions per 1,000 People (Metric Tons)

Emissions per Person (Pound)

**1,852**

ssions for Y (Metric Tons)

**03.2M**

ns

0.3M

-1.7

Agriculture Forestry

Sector Name

are building a movement in Georgia

Find your Electricity Large facility Forestry Back to Me

**Data source settings**

Manage settings for data sources that you have connected to using Power BI Desktop.

Data sources in current file  Global permissions

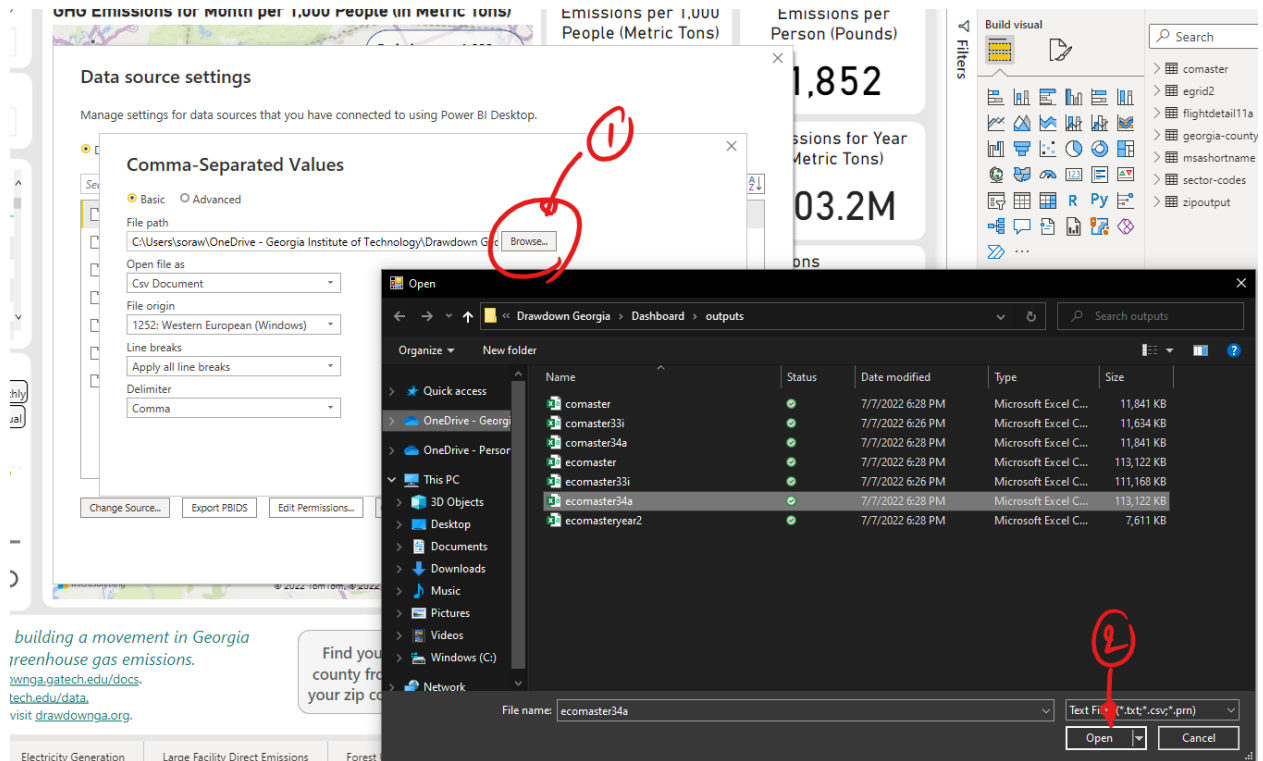
Search data source settings

- c:\research\climate\drawdown\outputs\ecomaster33h.csv
- c:\research\climate\drawdown\outputs\egrid3f.csv
- c:\research\climate\drawdown\outputs\flightdetail11b.csv
- c:\research\climate\drawdown\o...-counties-names-hlinks-v1.xlsx
- c:\research\climate\drawdown\outputs\msainfo8.xls
- c:\research\climate\drawdown\outputs\sector-codes.xlsx
- c:\research\climate\drawdown\outputs\zipoutput.csv

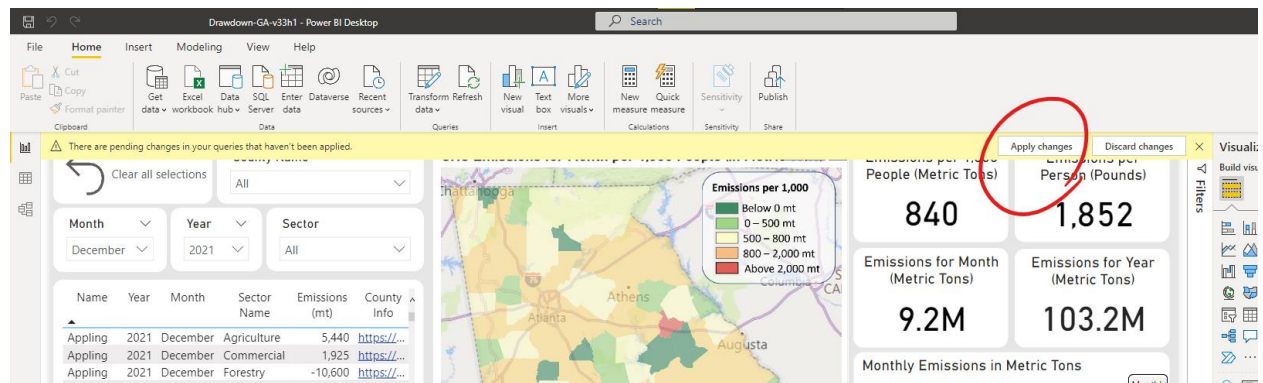
Change Source... Export PBIDS Edit Permissions... Clear Permissions

Close

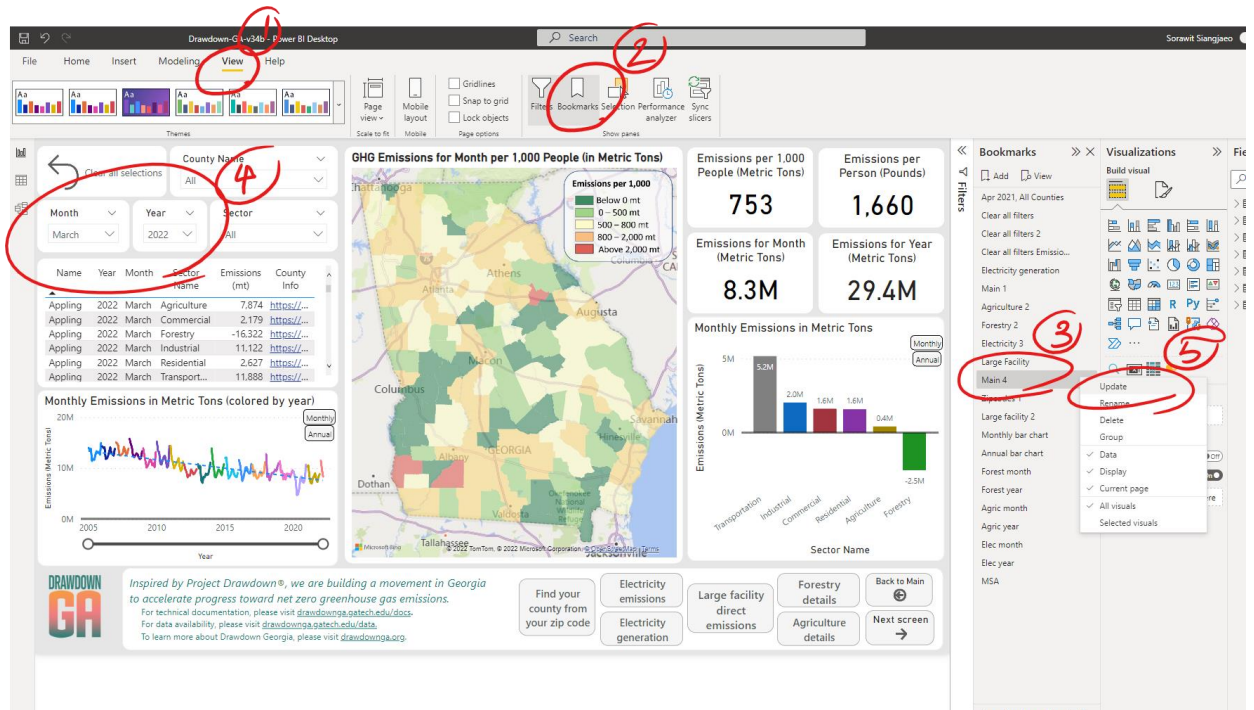
In the “Change Source...” screen, click “Browse...” after the File path line. A Windows Explorer box will pop up and you can navigate through your file directory to select the new ecomaster file. Please note that you want to select the “versioned” variant of the file (Ex. ecomaster33h.csv, ecomaster34a.csv,...). This will ensure that the updated ecomaster file is correctly selected for the report. Follow the “OK” and “Close” buttons to get out of these dialog boxes. There are multiple data sources on the list, but you do not need to change the sources for all of them. It seems Power BI embeds data files into the report. So you only need to update the source of the file that needs it.



- Power BI will prompt you that there are changes pending. Click “Apply changes” button and let the software run. It could take a moment.



- Once finished, the report will have the up-to-date numbers. However, the Drawdown report uses Power BI’s Bookmarks feature that show bookmarked screens when the Dashboard is published. Since we just updated the dataset, these bookmarks also need to be present the updated data points. To do this, go to View → Bookmarks from the ribbon menu. You will see bookmarks panel show up. Click on the bookmark you want to change the date (in this example is ‘Main 4’). Then, you will need to set the date (month and year) to match the data you recently updated in the ecomaster.csv file (in the picture below, data of March 2022 have been updated to the report, so we will set this as the “first page”).



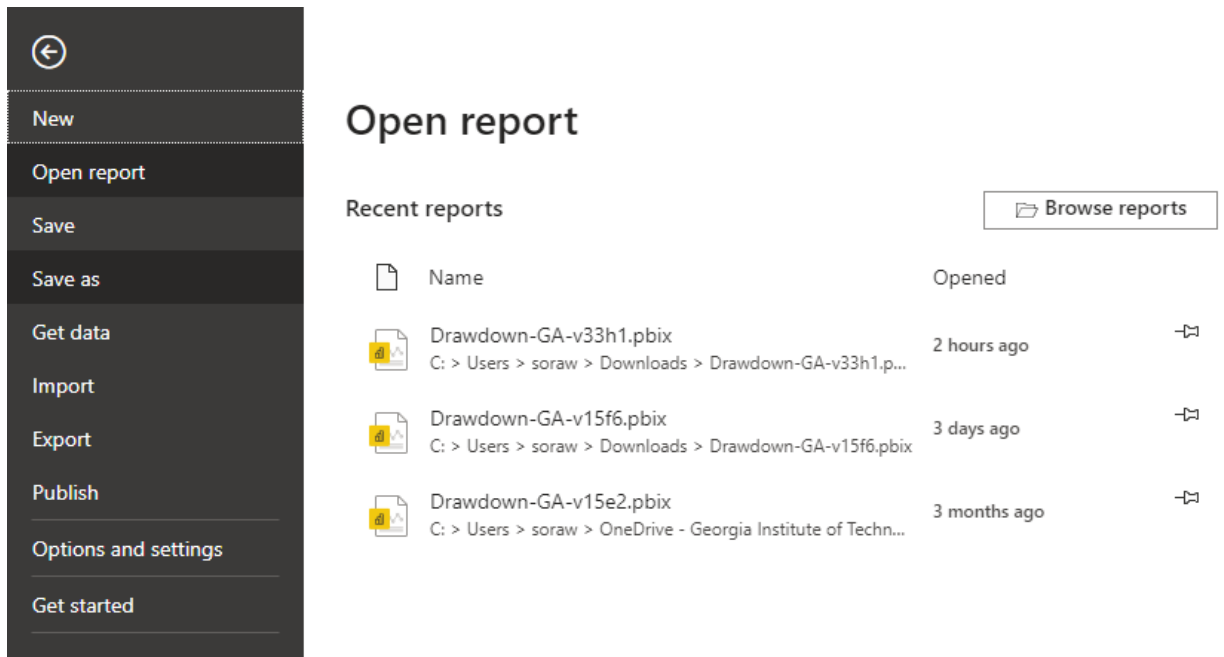
Once the date is set, right click (or click on the ellipsis) on the highlighted bookmark (the one you just changed the date) and click 'Update'. Repeat the process for the following bookmarks:

- Main 4
- Electricity 3
- Forestry 2
- Agriculture 2
- Monthly bar chart
- Annual bar chart
- Forest month
- Forest year
- Agric month
- Agric year
- Elec month
- Elec year

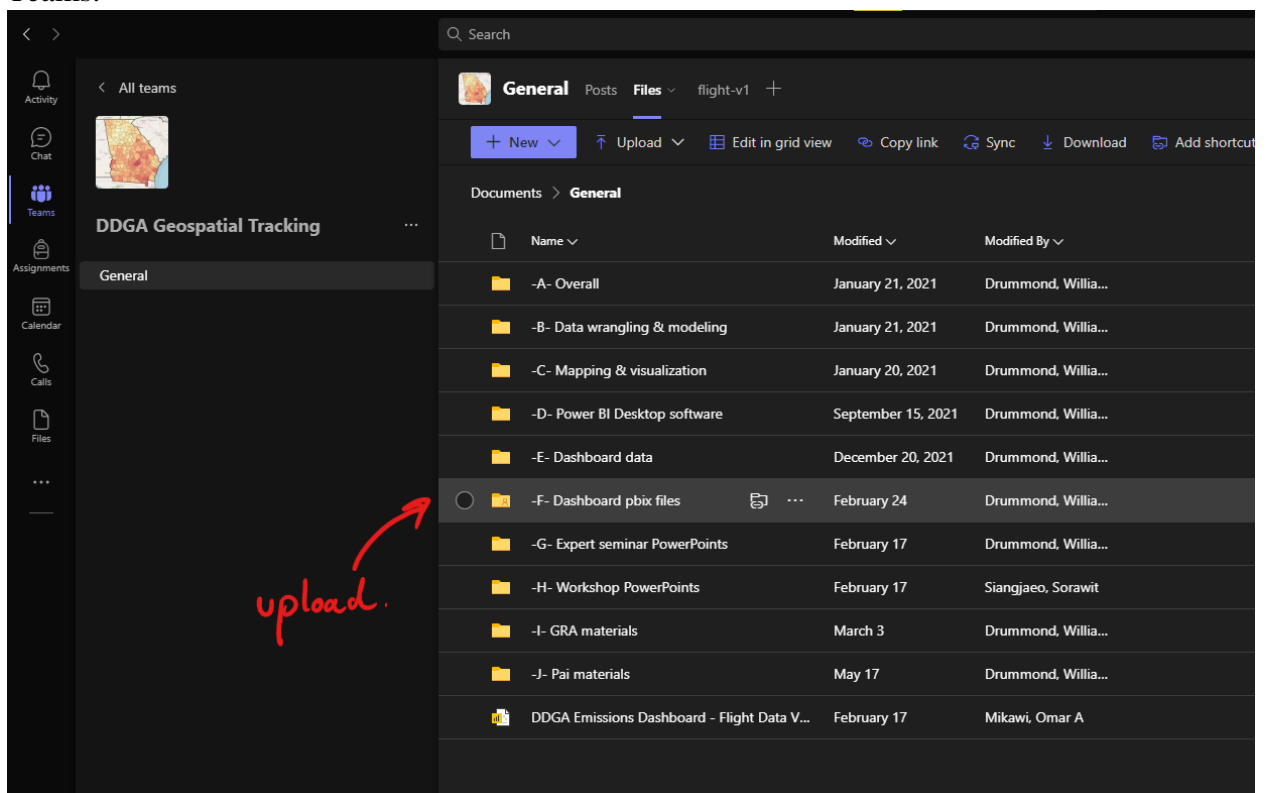
These bookmarks link to pages that we want to have the month up-to-date information for the dashboard users.

5. Save the report. Use "Save as" and name the new report in a similar naming scheme (Drawdown-GA-vXXXX.pbix) where XXXX refer to the version of the ecomaster file used to produce this report. E.g. ecomaster34a.csv → Drawdown-GA-v34a.pbix. If there's any small revisions that doesn't involve updating the ecomaster file, add numeric revision after the name (e.g. Drawdown-GA-v34a1.pbix).





Once saved, upload the .pbix report file into DDGA Geospatial Tracker on Microsoft Teams.



- Finally, publish the new report by clicking on the “Publish” button. You will need to publish it into “Geospatial tracking 1” which will upload the report to Power BI service where we can create embedded code for web publication to both [drawdownga.gatech.edu/](http://drawdownga.gatech.edu/) and

[drawdownga.org](http://drawdownga.org).

The screenshot shows the Power BI Desktop interface for the 'Drawdown-GA-v34b1 - Power BI Desktop' project. The dashboard is titled 'Agriculture Emissions for Month per 1,000 People (in Metric Tons)'. It features a map of Georgia, a table of emissions data, and several summary cards. A 'Publish to Power BI' dialog box is open, allowing the user to select a destination for publishing the dashboard. The 'Publish' button in the top right corner of the dashboard is circled in red.

| Name     | Year | Month | Subsector Name | Emissions (mt) | County Info                           |
|----------|------|-------|----------------|----------------|---------------------------------------|
| Appling  | 2022 | March | Ag. soil mg... | 3,755          | <a href="https://...">https://...</a> |
| Appling  | 2022 | March | Diesel fuel    | 1,235          | <a href="https://...">https://...</a> |
| Appling  | 2022 | March | Enteric ferm.  | 1,210          | <a href="https://...">https://...</a> |
| Appling  | 2022 | March | Manure m...    | 1,675          | <a href="https://...">https://...</a> |
| Atkinson | 2022 | March | Ag. soil mg... | 1,754          | <a href="https://...">https://...</a> |
| Atkinson | 2022 | March | Diesel fuel    | 613            | <a href="https://...">https://...</a> |

If you are authorized to publish the dashboard, follow the instructions in the next step. Otherwise, inform Dr. Drummond that the dashboard report has been updated and uploaded to Teams.

#### 4. Create embed link and publish the dashboard

This step requires that you are authorized to publish the dashboard. If you are tasked with this, speak with Dr. Drummond first to see if you have access to the features required for this step.

1. This step is done on Power BI service. You can access it through either Power BI app on Microsoft Teams or [powerbi.microsoft.com](https://powerbi.microsoft.com). You may need to use your GT login to use Power BI on the web.
2. Find "Geospatial tracking 1" workspace. This is where the latest .pbix files are stored. Click on "Refreshed" to arrange the files from the most recent to the oldest to make selecting the file easier.

| Name                     | Type    | Owner                 | Refreshed            | Next refresh | Endorsement | Sensitivity | Include in app |
|--------------------------|---------|-----------------------|----------------------|--------------|-------------|-------------|----------------|
| Drawdown-GA-v34b1        | Report  | Geospatial tracking 1 | 7/13/22, 11:29:35 AM | —            | —           | —           | Yes            |
| Drawdown-GA-v34b1        | Dataset | Geospatial tracking 1 | 7/13/22, 11:29:35 AM | N/A          | —           | —           | —              |
| Drawdown-GA-v34b (1)     | Report  | Geospatial tracking 1 | 7/12/22, 3:42:11 PM  | —            | —           | —           | Yes            |
| Drawdown-GA-v34b (1)     | Dataset | Geospatial tracking 1 | 7/12/22, 3:42:11 PM  | N/A          | —           | —           | —              |
| Drawdown-GA-v34b         | Report  | Geospatial tracking 1 | 7/12/22, 2:56:13 PM  | —            | —           | —           | Yes            |
| Drawdown-GA-v34b         | Dataset | Geospatial tracking 1 | 7/12/22, 2:56:13 PM  | N/A          | —           | —           | —              |
| DDGA Solution Solar      | Report  | Geospatial tracking 1 | 6/23/22, 10:43:39 AM | —            | —           | —           | Yes            |
| DDGA Solution Solar      | Dataset | Geospatial tracking 1 | 6/23/22, 10:43:39 AM | N/A          | —           | —           | —              |
| phase3-drummond-modified | Report  | Geospatial tracking 1 | 6/23/22, 10:43:39 AM | —            | —           | —           | Yes            |
| Atlanta emissions        | Report  | Geospatial tracking 1 | 5/20/22, 7:26:44 PM  | —            | —           | —           | Yes            |

- Click on the report file to be published. Once it's loaded, click on File → Embed report → Publish to web (public)

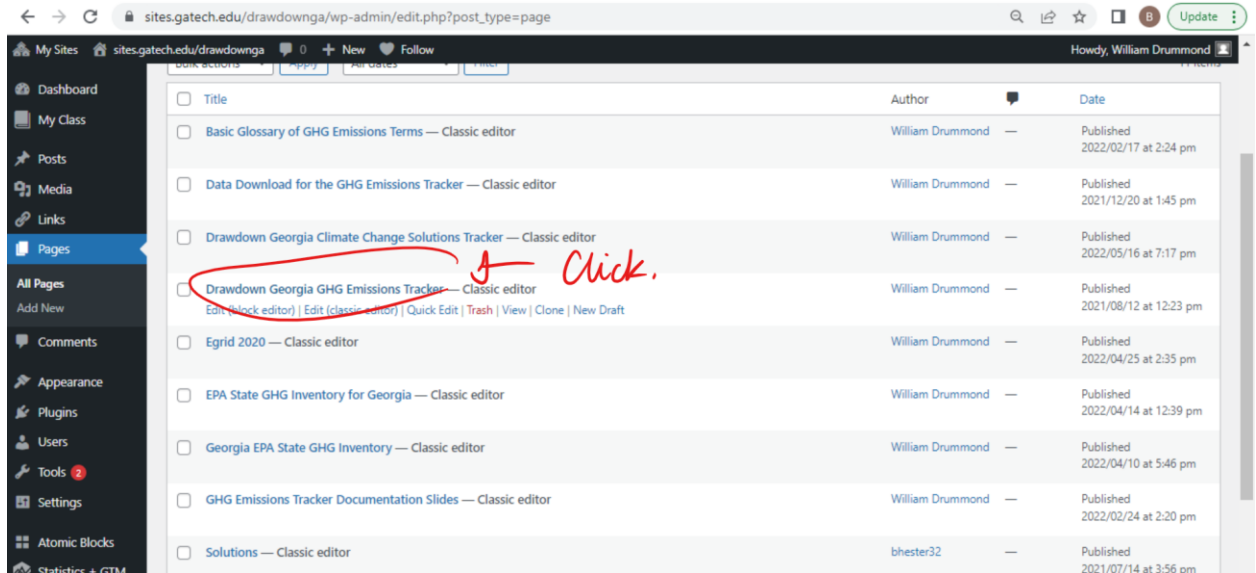
- The embed code window will pop up. Select the size to 1024 × 612 px from the dropdown box. Then copy the HTML code by clicking on the second Copy button from the top.

The screenshot shows a Microsoft Power BI dashboard titled "GHG Emissions for Month per 1,000 People (in Metric Tons)". The dashboard features a map of Georgia, a table of emissions by county and sector, and several summary cards. A "Share" dialog box is open, showing the "Embed code" section with a link and an HTML embed code. A dropdown menu for "Size" is also visible, with "1,024 x 612 px" selected.

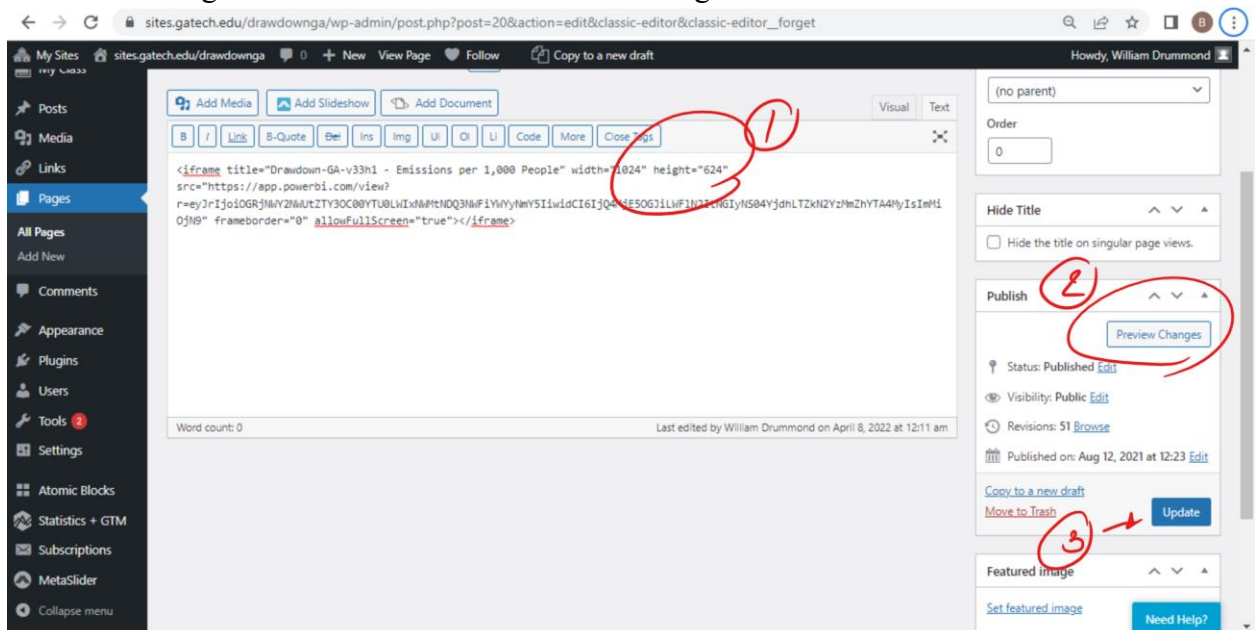
- Go to [drawdownga.gatech.edu/wp-admin](https://drawdownga.gatech.edu/wp-admin) (again, provided that you have privileges to the admin site. If not, contact Dr. Drummond). You'll be greeted with Wordpress Dashboard panel. Select Pages → All Pages to see the list of pages on this site.

The screenshot shows the WordPress dashboard for the site "drawdownga.gatech.edu". The dashboard includes a sidebar with navigation options, a main content area with site information, and a "My Account" section. The "Pages" menu item is highlighted in the sidebar, and a dropdown menu is open, showing "All Pages" and "Add New".

- Click on "Drawdown Georgia GHG Emissions Tracker" page. This will direct you to the Wordpress editor. Make sure to select the "Classic editor".



Delete the existing HTML code and replace it with the one you copied from Power BI service. Change the parameter height to 624. Then, in the Publish panel on the right, select “Preview Changes” to see that the Dashboard looks right.



If it does, click “Update” to publish the Dashboard. Copy the HTML code (with edited height parameter) and paste it in a Word document.

7. The new Dashboard will be published on drawdownga.gatech.edu. This is the internal Dashboard site. Spend about 90 minutes testing the new Dashboard: go to every page, use all the filters, etc. and see if it functions properly, the numbers are presented sensibly. This is the last step of testing.
8. If the site works properly, send the Word document with the HTML code and inform Dr. Drummond that it is ready to be sent the HTML code for the updated dashboard to [www.drawdownga.org](http://www.drawdownga.org)