Week 18 Exercises: Technology-adjusted CBA

(May 2nd 2022)

Objectives

- Calculate TCBA in Python
- Calculate Scope 1, 2, and 3 emissions in Python
- Visualize the results in Python

Python exercises: part 1

Technology adjusted consumption-based accounting

- In one Supplementary Information file of ‘Kander et al. 2015. National greenhouse-gas accounting for effective’, the authors illustrated the TCBA calculation and results in a spreadsheet.
- Download and go through the spreadsheet example to further understand the TCBA calculations.
- Implement the same calculations in Python. Note, when we use a real MRIO dataset (200 products and 49 regions/countries) for the IGA, performing the EIOA analysis in excel spreadsheet will be quite difficult.

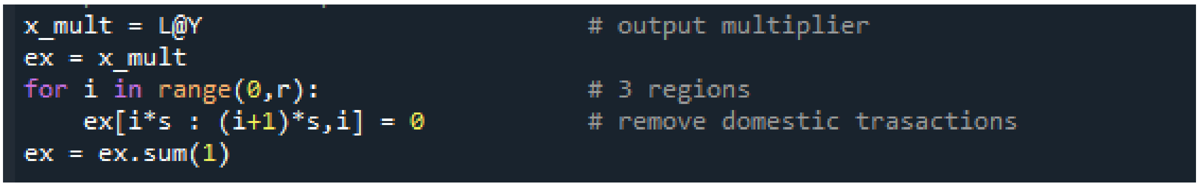
Coding tips:

(You may need to install "openpyxl". (code: conda install openpyxl))

1. Create folder path, select the data frame for Z, Y, F, etc, and calculate other IO variables.

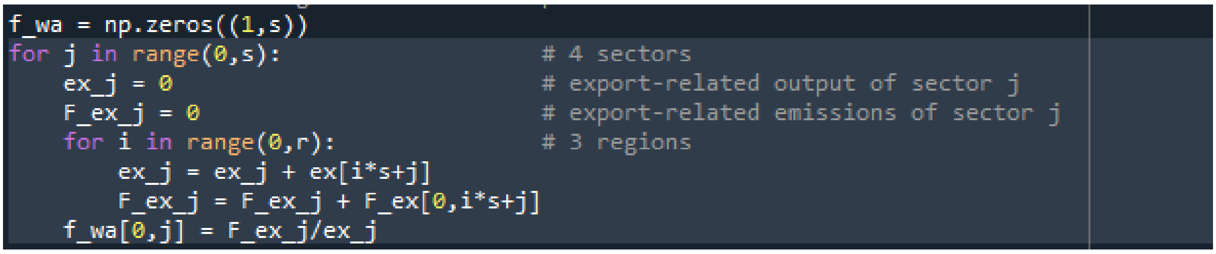
2. Adjust emissions embodied in export.

--Remove domestic transactions of each region



```
x_mult = L@Y                          # output multiplier
ex = x_mult
for i in range(0,r):                  # 3 regions
    ex[i*s : (i+1)*s,i] = 0           # remove domestic trasactions
ex = ex.sum(1)
```

--Calculate world market average emissions multiplier for each sector



```
f_wa = np.zeros((1,s))
for j in range(0,s):                    # 4 sectors
    ex_j = 0                            # export-related output of sector j
    F_ex_j = 0                          # export-related emissions of sector j
    for i in range(0,r):                # 3 regions
        ex_j = ex_j + ex[i*s+j]
        F_ex_j = F_ex_j + F_ex[0,i*s+j]
    f_wa[0,j] = F_ex_j/ex_j
```

--Calculate export-related emissions by TCBA

--Calculate TCBA

Python exercises: part 2

Economy-wide accounting of Scope 1, 2, and 3 emissions.

- In Figure 3 of "Hertwich, E. G., & Wood, R. (2018). The growing importance of scope 3 greenhouse gas emissions from industry", scope 1, 2, and 3 emssions are calculated by sector (**5 sectors**) and region.
- Calculate the scope 1, 2, and 3 emissions with the data from EXIOBASE for year 2015, by sector (**7 sectors**) and region (49 regions). Data source: https://zenodo.org/record/5589597 . ("IOT_2015_pxp.zip")
- Draw a figure (similar to figure 3) for the Netherlands of scope 1, 2, and 3 emissions.

Coding tips:

(You may need to install package pymrio (code: conda install pymrio))

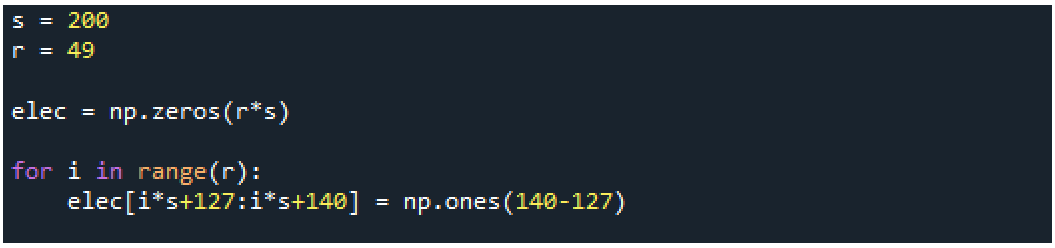
1. Prepare the excel file (IPCCsec.xlsx) from ***Brightspace****.*

Download the EXIOBASE data for year 2015 from https://zenodo.org/record/5589597 *.*

("IOT_2015_pxp.zip")

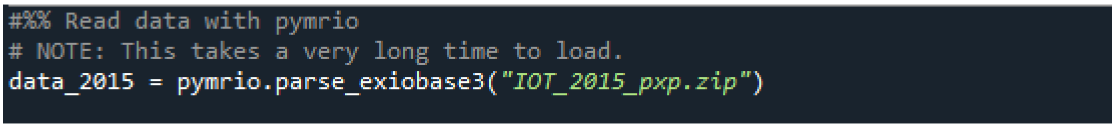
2. Assign 200 IO sectors to aggregate IPCC groupings, more specifically here in the paper 7 aggregated sectors. You can use the matrix from sheet "xio2detail" in excel file "IPCCsec.xlsx".

3. Create the electricity sectors in each economy.



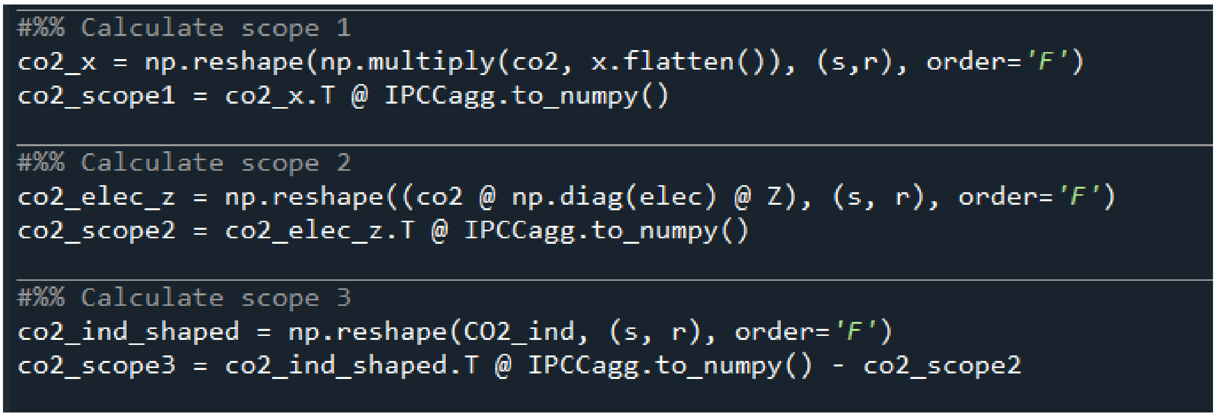
```
s = 200
r = 49

elec = np.zeros(r*s)

for i in range(r):
    elec[i*s+127:i*s+140] = np.ones(140-127)
```

4. Load the EXIOBASE data "IOT_2015_pxp.zip" .



```
#%% Read data with pymrio
# NOTE: This takes a very long time to load.
data_2015 = pymrio.parse_exiobase3("IOT_2015_pxp.zip")
```

5. Prepare the matrices you need for the scope 1, 2, and 3 calculation.



```
#%% Calculate scope 1
co2_x = np.reshape(np.multiply(co2, x.flatten()), (s,r), order='F')
co2_scope1 = co2_x.T @ IPCCagg.to_numpy()

#%% Calculate scope 2
co2_elec_z = np.reshape((co2 @ np.diag(elec) @ Z), (s, r), order='F')
co2_scope2 = co2_elec_z.T @ IPCCagg.to_numpy()

#%% Calculate scope 3
co2_ind_shaped = np.reshape(CO2_ind, (s, r), order='F')
co2_scope3 = co2_ind_shaped.T @ IPCCagg.to_numpy() - co2_scope2
```

6. Plot the figure. Be aware of the country you need to select.