Brightway for beginners 2018

This document includes all the relevant info for the course from software installation to lectures and exercises. Just scroll down.

Learning Python

The more you can learn python before the course, the best. This is totally self-study.

Here some resources, give a look and find the one that feels best for you.

- Check a lot of options here on the Brightway2 website
- **NEW!** I like this one, very clear and for academics: A Byte of Python. Can download the pdf too.
- This is a good one: think python
- Another good one for beginners: learn python the hard way
- Sololearn python is a very good mobile App, in my opinion: download the Android version I am sure there is an iOS version too, search for it.

Python and Brightway2 installation

What we need for the course is:

- The latest version of python 3.
- Some scientific python packages like scipy and numpy
- Scientific python working environments, in particular Jupyter notebook (former Ipython notebook) and Spyder

(google these names to find out what they are)

Please make sure you have python, Brightway2, Jupyter, and Spyder up and running before we start (we won't waste lecture time to install stuff).

If nothing of the instruction below works or this is too hard, please write me an email and come to my office with your computer and I'll try to fix it.

On Windows

Follow **step by step** the instructions in the Brightway2 official page here. Here a couple of notes that may further help you with this installation, i.e. this is how I did it:

- Once you have installed miniconda, you need to open the 'Anaconda prompt' and only **there** type all the commands (e.g. the first one in the official tutorial is `conda update conda`). How to do that? Inside the search field from your taskbar, start typing 'Anaconda' and you should find it.
- You can give your virtual environment the name 'BW2' (replace `<change-me>` with `BW2` in the command lines suggested on the webpage).

After you have followed all the instructions, please close the Anaconda prompt and open it again, then...

Type the line below and you should now see '(BW2)' on the beginning of each line.

```
activate BW2
```

Type the lines below and this should open python, run Brightway2, then close python.

```
python

import brightway2

quit()
```

This below should install another scientific environment.

```
conda install spyder
```

You can now try to type either `spyder` or `jupyter notebook` and this should open a new window. When you close the window, the anaconda prompt will be full of stuff.

Press 'Ctrl + C' two times and it should return normal.

On Mac

Method 1 (recommended)

This is the recommended method. Follow **step by step** the instructions in the Brightway2 official page here. Here a couple of notes that may further help you with this installation, i.e. this is how I did it **(read this entirely before you start)**:

- As very first thing you should download the latest version of miniconda for mac here. Download it in the 'Downloads' folder.
- Then open the mac terminal and type these three commands (Note the `./` in front of the third one...):

```
cd ~/Downloads

chmod +x ~/Downloads/Miniconda3-latest-MacOSX-x86_64.sh

./Miniconda3-latest-MacOSX-x86_64.sh
```

- Do as in the Brightway2 website. After the installation of miniconda, you will need to navigate to your folder using `cd /Users/<your user name>/miniconda/bin/`
- Once you are in this folder, you need to use `./` in front of the first two commands: `./conda update conda` and then `./conda create -n <change-me> python=3.6`
- You can give your virtual environment the name 'BW2' (replace `<change-me>` with `BW2` in the command lines suggested on the webpage). When you will then type `source activate BW2` you should see '(BW2)' on the beginning of each line.
- Once you have activated the BW2 environment, follow the instructions in the website and install Brightway2.
- Once you are done with the Brightway2 installation, and you have the virtual environment still activated, try if everything works by typing:

```
python

import brightway2

quit()
```

- Use `source deactivate BW2` to close the virtual environment and close the terminal with `exit`.
- From now on, each time you want to use Brightway2, you need again to activate the virtual environment, and to do that need to navigate to your folder with e.g. `cd ~/miniconda/bin` and then type `source activate BW2`. Alternatively, just open terminal and e.g. use the commands `source ~/miniconda/bin/activate BW2` and `source ~/miniconda/bin/deactivate BW2` respectively.

Method 2 (worked for me, but it's at your own risk)

It should be simpler. Install everything via brew. Why? See here.

Open the mac terminal and copy paste the following commands one by one.

I am not sure if you will be asked to install other packages. Just say yes in case.

```
/usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"

brew install python3

pip3 install brightway2

pip3 install jupyter

pip3 install spyder
```

Try if everything works by closing and reopening the terminal and typing:

```
python3

import brightway2

quit()
```

If you don't get any error message, everything is alright. If you get an error message, try this: `pip3 install -U --pre brightway2` Hopefully will fix stuff.

Then type:

```
jupyter notebook

spyder3
```

This should open the two scientific environments in separate windows.

Fingers crossed...

NOTE: Apparently the calculations will be slower in this case because of the solver used which is different from the one in conda. I don't know the details but I have checked and it's true. Almost twice faster with conda, even if you might not even notice the difference between 1 sec and 1.8 sec. But this is particularly relevant if you plan to do Monte Carlo simulation.

Learning objectives

Knowledge

At the end of the course you will know about: python data structures, python sicentific environments, Brightway2 data structures, Brightway2 functions, key concpets of statistical analysis for LCA (error propagation, statistical testing).

Skills

At the end of the course you will be able to: program in pythonb at a basic level, use Brightway2 at a level comparable to other commerical LCA software, including import foreground and background data into Brightway2, running calculations from a simple LCA to more complex simulations and comparative analyses, perform basic statistical analysis.

Competences

At the end of the course you will be ain a position to: apply LCA modelling in Brightway2 to analyze real-world case studies, by choosing the appropriate data, code, and workflow organization that solve the case-specific challenges.

Course plan

Lecture 1

- Surviving installation and ready to go
- Intro to general concepts in Brightway2 and resources
- Create a simple database

Lecture 02

- Create your own LCIA method
- Do LCA calculations
- Navigate the database

Lecture 03

- Import and navigate ecoinvent
- Import foreground LCI data from Excel

Lecture 04

- Run Monte Carlo simulation
- Run comparative Monte Carlo
- Statistical analysis of Monte Carlo results

Brightway2 resources

- Brightway2 official website https://brightwaylca.org/ contains info about installation and key features, and example jupyter notebooks
- On Stackoverflow you can ask questions, give answers, and find answers about Brightway2 using the tag `brightway`

- You can check the source code behind all Brightway2 functions. this means you can see how Brightway2 is written in python and the entire code is open source. You can even contribute.

Brightway2 elements (Script 00)

The most important data structures are represented here and described here. Look at the figure and try to make the parallel with what you know already (e.g. Simapro). I recommend that later you read carefully this documentation page. This is also the terminology to use when working with Brightway2. The script **00** (courtesy of Romain Sacchi) is for self-study and helps you setting a project and finding stuff.

A product system from scratch (Script 01.1 and Script 01.2)

First we start with a very simple example and we do everything manually to understand how the Brightway2 syntax works. The script **01.1** contains an example reproducing the product system described in Heijungs and Suh (2002). We will read it together and understand how such database is structured and written. This means that we will also touch on python-specific data structures (list, tuple, dictionary) and objects (variable, function, class). The script **02.2** shows a different way to link the technosphere and biosphere flows. Run the script and check that you get the same results as in the book in both cases.

Exercise: Take your own product system, select two or three activities that are linked together, and that have also some environmental exchanges associated with, and make by hand a database using the Brightway2 standard dict structure. Run the calculations on that. If you don't have data, use the data here.

Navigate the database (Script 02.1)

When you do an LCA you need to access the various activities and look at them to understand what are their inputs and outputs and how they are linked to other activities. The script **03** includes code to do this in different ways. Try it out and try it on your own product system as well.

Exercise: How to get the value of an exchange directly and without indexing? I.e. without using a number to indicate the position of this exchange in the list of exchanges of a specific activity, and instead by using the name of the exchange? E.g. what if I want to get the value of the CO2 emitted by electricity production?

Co-products (Script 02.2)

There are at least two ways do model co-products with the substitution method. Besides the exchange types `'technosphere'`, `'biosphere'`, and `'production'` there is a fourth type called `'substitution'`. You can use that (use **plus** sign!). See the docs about exchanges. Alternatively, you can simply create an exchange of the `'technosphere'` type but using the **minus** sign. I.e. a negative

input of some product. The script **02.2** includes code to do this. If you think about it, this is identical to SimaPro where you also have two options: either use the predefined line for co-products or insert a negative input from technosphere.

(I still don't know how/why BW allowed me to create an activity with two different exchanges of the `'production'` *type though...)*

NOTE: the signs issue is explained very clearly in the introduction docs under "Getting the signs right". Note that this is perfectly consistent with the Hejiungs and Suh book chapter. Diagonal values in the A matrix are positive, off-diagonal inputs are negative. Intervention matrix signs depend on the convention (you decide the sign or you have to follow the convention used of the database, e.g. the database may assume that +10 kg CO2 means the emission of CO2 and +10 kg crude oil means the extraction of oil).

ALLOCATION What if you want to use the partitioning method? You need to calculate the allocated values (by mass/energy/revenue etc) for each exchange before importing the data into Brightway2 (or write a code that does that automatically). Just like in SimaPro.

Get ecoinvent 3.4 (Script 03.1)

Now that you know how to work with the foreground system, it's time to learn how to work with the background system. In particular it is useful to import and play around with two databases: *biosphere3* that contains all the exchanges and impact assessment methods, and *ecoinvent*. Run the script **03.1** to import biosphere3 and ecoinvent.

1. Open the ecoinvent website and login with the username and password you got via mail.
2. You should read somewhere: *To download LCI and LCIA cumulative matrices click here*. Click there.
3. Select *ecoinvent 3.4. (2017), current*
4. Download the file `ecoinvent 3.4_consequential_ecoSpold02.7z` in a folder of yours. Make sure you remember the full path to this directory. E.g. I have downloaded the file in: */Users/massimo/Documents/AAU/Research/Databases/ecoinvent v3.4*
5. The file you have downloaded is a compressed archive of many files (like with winzip or winrar). Extract the files from the .7z archive, e.g. by double clicking it. If it does not work, install a software that can do that. E.g. for mac users you can either download theunarchiver or, if you are using brew, just open terminal and do `brew install p7zip` and then from terminal find the folder and do `7z x 'ecoinvent 3.4_consequential_ecoSpold02.7z'` (here the p7zip instructions in case).
6. Now you an run the script **03.1**. Make sure you change the path line and replace it with the one where you have extracted the files. For example, I have extracted the files in a folder called "datasets". The path to this folder is: */Users/massimo/Documents/AAU/Research/Databases/ecoinvent v3.4/datasets/* You will see this same line in the script and you need to change it with your directory.

Working with background data (Script 03.2)

Look at the script **03.2** to understand how to navigate ecoinvent.

A key difference compared to previous exercises is that in ecoinvent each activity and exchange is defined by a **code** which are unique identifiers. So it is important to learn how to find both activity code and name and how to match them *(Actually we used the codes also in the previous lectures but they were identical to the activity names for simplicity)*.

Exercise: link the emissions of your previously defined product system to the biosphere3 database, and link some of the ecoinvent database activities to your product system.

Importing data from Excel (Script 03.3)

Brightway2 has a series of options for data import and export that you are invited to read about and try, they are on the official website and notebook. However, I have tried various of these and found them quite unpractical so I developed my own system, that fits with my workflow. In particular the file `lci_to_bw2.py` includes a code to convert a properly formatted csv file into a Brightway2 database dict. You need to install the Python Data Analysis Library pandas to make it work (do `conda install pandas` or `pip install pandas` if you are not using conda).

How does the importer work?

1. Prepare your inventory in MS Excel using the template. See the example file *test_db_excel_w_ecoinvent.xlsx*
2. Save the relevant MS Excel sheet as .csv file, see the example file *test_db_excel_w_ecoinvent.csv*
3. Import the module in your script with the command `from lci_to_bw2 import *`
4. Import the .csv file as a dataframe with the pandas function `.read_csv()`. Clean it up for unnecessary columns.
5. Convert the dataframe into a dict using the function `lci_to_bw2()`
6. Save the dict as a Brightway2 database in the usual way i.e. using Brightway's `Database()` and `.write()` functions.

See an example in the script **03.03**.

Uncertainties and Monte Carlo simulation (Script 04.1 and Script 04.2)

Now you are ready to start doing more intense simulations. In particular Brightway2 is great to perform fast error propagation with Monte Carlo simulation.

The script **04.1** shows how to add uncertainties to your home-made product system and run a Monte Carlo simulation.

The script **04.2** shows how to perform a more advanced comparative Monte Carlo simulation. This is the kind of simulation to be used in comparative LCAs, i.e. in analyses where different alternatives to provide the same Functional Unit are compared. In this comparative case it is important to randomly sample a common technology matrix **A** for all alternatives at each iteration, instead of having a different technology matrix per each alternative at each iteration. This allows for a smaller variance and shorter computational times, but is not possible with e.g. SimaPro. Moreover, this allows to optimally perform the statistical testing of the results with paired tests, to see if the two alternatives are significantly different from each other or not.

To better understand these scripts, I recommend reading read the paper by Limpert et al. (2001), that is a 'classic' and explains the log-normal distribution, and the paper by Henriksson et al. (2015) which is probably the best example of an LCA with comparative Monte Carlo simulation followed by statistical testing for significant differences between alternatives.

Resources to understand LCA statistics

When applying a statistical approach to LCA, there are some key concepts that is important to understand in detail: distribution types, error propagation, statistical testing. If you are new to statistics, a first step is reading the wikipedia pages explaining Monte Carlo method, general statistical hypothesis testing, parametric and nonparametric statistics, normality tests, t-test, Wilcoxon signed-rank test. When implementing this in Brightway2, it is then useful to read the corresponding python documentation for statistical functions of the stats package. For example the following functions were used in the script **04.02**: Shapiro-Wilk test, paired t-test, and Wilcoxon signed-rank test.

If you want to learn more, buy a good introductory statistics book. Ideally one which has a good balance between mathematical expressions and pedagogic explanations. There are also many open source or free ones, I can recommend e.g. Statistics which explains testing in general and it's written in a way which is easy to understand, The Elements of Data Analytic Style which introduces to data analysis in general, and Statistical inference for data science which is is very practical and especially useful if you also know R. In general I would encourage to learn R too if you plan to work with stats - I honestly prefer it to python for doing statistical analysis and also for plots.

Brightway2 workflow

Brightway2 allows to automate and speed up stuff, but there is still a lot of manual work when doing LCA. Also, LCA is really iterative as often you find a small mistake and you need to correct the data and re-do the calculations, or you want to add a scenario, or repeat the analysis with different data, etc. This means that you have several **iterations** before you get the final results. It also means that you may want to speed up each iteration, and be able to reproduce everything after some months (e.g. if you have submitted a paper) so the stuff should be well-organised.

The ideal workflow depends on your needs and your current skills, but you should expect to use different tools. For example, in a current project of mine I have this workflow:

1. I organise the foreground system in MS Excel. I prepare one or more MS Excel spreadsheets with multiple calculation sheets that ultimately allow building the foreground inventory in the right format for later use in Brightway2 (see chapter about import from Excel). Every time I modify something, I export a new .csv file with the inventory.

2. I use Brightway2 for the LCA calculation and MC simulation. I import the inventory in .csv, run a python script with the calculations, and export the results the results in a tabular format, e.g as another .csv file with the LCIA results. At each iteration I simply run the script again, or modify it if necessary.

3. I use R for the statistical analysis and plots. For example I import the .csv file with the LCIA data, run the R script, and export either some new tables in .csv (e.g. p-values from the statistical tests) or the plots as .pdf files. Also in this case I can simply run the script again at each iteration, or modify it if necessary.

Brightway2 stuff that I can't do yet

Sometimes we don't do things because we don't even know that it is possible to do these things, or that they exist. Similarly, there is really a lot of unexplored potential in Brightway2. I will mention just a few things:

- There are many ore import **strategies** to automate the import of data into Brightway2. I am not very good at using them but these can potentially simplify the workflow and reduce the manual work substantially.

- It is possible to work with geographies and time-specific inventories. Thus allowing for **spatially and temporally** differentiated LCA results, building maps, etc.

- One can make **parametrised inventories**, where e.g. the output of wastewater from an activity is related to the input of water via a function, so that increasing the latter will result in a decrease of the former, automatically.

- In general one can combine all the python possibilities with Brightway2, e.g. the fact that one can take real-time data directly from the web and use them in a LCA to do real-time measurements.

You are welcome to try these things out.

Acknowledgements

I hope you enjoyed this course.

I would like to thank Romain Sacchi and Chris Mutel for helping me solving my Brightway2 problems so far. Some pieces of the code they wrote in response to my questions and queries ended up within the scripts provided in this course, so indirectly they are contributors. Romain also checked the scripts and wrote

specific ones.

References

Heijungs, R., & Suh, S. (2002). The basic model for inventory analysis. In A. Tukker (Ed.), The computational structure of Life Cycle Assessment (pp. 11–28). London: Kluver Academic Publisher.

Henriksson, P. J. G., Rico, A., Zhang, W., Ahmad-Al-Nahid, S., Newton, R., Phan, L. T., … Guinée, J. B. (2015). Comparison of Asian Aquaculture Products by Use of Statistically Supported Life Cycle Assessment. Environmental Science and Technology, 49(24), 14176-14183. https://doi.org/10.1021/acs.est.5b04634

Limpert, E., Stahel, W. A., & Abbt, M. (2001). Log-normal distributions across the sciences: Keys and clues. Bioscience, 51(5), 341-352. https://doi.org/10.1641/0006-3568(2001)051[0341:lndats]2.0.co;2