brightway2-aggregated

Release 0.1

Sep 06, 2019

Contents

Brightway2-aggregated is an extension to the Brightway2 LCA framework to generate aggregated data.

It operates at two levels:

- aggregated LCI level (cradle-to-gate inventories), though this is not recommended because Brightway2 is optimized to work with sparse matrices, and the biosphere matrix of aggregated LCI databases are very dense)

- aggregated LCIA score (cradle-to-gate impact assessment score)

The `DatabaseAggregator` will calculate the LCI (and, optionally, the LCIA) for all activities in the database and store the results in a new database. The activities in the resulting database will:

- have a production exchange identical to the production exchange in the unit process database

- have no further technosphere (i.e. there will be no off-diagonal elements in the **A** matrix)

- Have either:

    - cradle-to-gate LCI results stored as biosphere exchanges (resulting in a dense biosphere matrix **B**), or

    - cradle-to-gate LCIA results stored as biosphere exchanges. The biosphere matrix (**B**) will therefore have as many rows as there are impact assessment methods of interest. This entails the creation of new "unit impact" biosphere exchanges, see *the section on preparing Brightway2 for use of cradle-to-gate scores*

While there are many reasons for using aggregated data, this module was specifically created to play well with pre-samples, where aggregated data can be associated with presampled data generated from e.g. Monte Carlo simulations.

Context

Inventory data usually contains three types of exchanges:

- Production exchanges, which represent the actual functional output of a given activity;

- Technosphere exchanges, which represent the exchange of products between activities;

- Elementary flows (called biosphere exchanges in Brightway2), which represent the exchanges between activities in the technosphere and the environment.

To calculate an impact score for a given functional unit, one needs to:

1) scale all activities so that they supply exactly one functional unit (i.e. solve a system of linear equations), which is computation-intensive.

2) Multiply the biosphere exchanges of each activity by the scaling factors obtained in the previous step, resulting in a life cycle inventory, normally containing hundreds or thousands of scaled biosphere exchanges.

3) Multiply the scaled biosphere exchanges with characterization factors (impact per unit biosphere exchange).

Brightway2-aggregated does two things that simplifies these steps:

1) It can store the solution to steps 1 and 2 for a given database, effectively eliminating the need to rescale the technosphere and biosphere exchanges each time one carries out an LCA. It does this by creating an "aggregated LCI database" (exactly like the "system" database supplied by ecoinvent).

2) It can store the solution to steps 1, 2 and 3 for a given database, i.e. calculate cradle-to-gate impact scores for all possible outputs of a database.

These levels of simplification can be useful in some contexts:

- Storing aggregated LCI data will slow down actual calculations in Brightway2 (because aggregated LCI are dense, and Brightway2 is optimised to work with sparse data). However, the LCI data can be associated with presamples, allowing the reuse of e.g. Monte Carlo results.

- The real strength of Brightway2-aggregated is when databases store LCIA scores. This allows one to lightweight databases and calculations, useful when creating LCA tools. LCIA scores also play very well with presamples.

Quickstart

This section provides a brief overview of the use of brightway2-aggregated.

- *Generating aggregated databases*
- *Preparing Brightway2 for use of cradle-to-gate scores*

Note: The package was developed as an extension to the Brightway2 LCA framework.

2.1 Generating aggregated databases

To generate an aggregated version of an existing unit process database, you need:

- to have brightway2 installed
- a project with a unit process database you want to aggregate

If you want to aggregate all the way to cradle-to-gate scores (which you probably do), you also need a list of method ids.

Warning: Aggregating at the LCI level generates very dense biosphere matrices. Because Brightway2 has been optimized to work with sparse matrices, this slows down calculations considerably.

```
>>> import brightway2 as bw
>>> from bw2agg.aggregate import DatabaseAggregator
>>> bw.projects.set_current('my project')

# Create an aggregated database generator
>>> agg_db = DatabaseAggregator(
...     up_db_name="Name of the unit process database",
...     agg_db_name="Name of the aggregated database",
```

(continues on next page)

```
...        database_type="LCIA", # or "LCI"
...        method_list=[('my', 'first', 'method'), ('my', 'second', 'method')], # Actual
→method ids
...        overwrite=False
... )

# Generate and write aggregated database
>>> agg_db.generate()
Writing activities to SQLite3 database:
0% [#############################] 100% | ETA: 00:00:00
Total time elapsed: 00:03:37
```

More details on the function can be found *in the technical reference page*.

2.2 Preparing Brightway2 for use of cradle-to-gate scores

In order to convert an LCI database to a cradle-to-gate LCIA scores database, and to use this aggregated database, two things are needed:

- New biosphere exchanges that represent the "unit impact scores". For example, say you want to have a database that contains precalculated cradle-to-gate climate change impact scores (i.e. carbon footprints), all the biosphere exchanges in the inventory need to be replaced with a single "unit score for climate change" biosphere exchange.

- For this new biosphere exchange to be used in future LCIA, a new characterization factor, equal to 1, needs to be added for the "unit score for climate change" biosphere exchange.

These new exchanges and characterization factors are added automatically for the selected methods by the `DatabaseAggregator`. However, it is possible to preemptively create these for a single method `bw2agg.scores.add_unit_score_exchange_and_cf(method_id)`, or all methods (`bw2agg.scores.add_all_unit_score_exchanges_and_cfs()`).

Technical reference

- *Generating aggregated databases*
- *Adding unit impacts exchanges to biosphere and unit impact characterization factors to methods*

3.1 Generating aggregated databases

class bw2agg.aggregate.**DatabaseAggregator**(*up_db_name*, *agg_db_name*, *database_type='LCIA'*, *method_list=[]*, *biosphere='biosphere3'*, *overwrite=False*)

Generator to aggregate and write an LCI database.

Can be used to generate either:

- an aggregated LCI database (i.e. a database with cradle-to-gate LCI stored in the biosphere matrix)
- an aggregated LCIA database (i.e. a database with cradle-to-gate characterized inventories stored in the biosphere matrix as "unit impact" biosphere exchanges). This requires the creation of unit impact characterization factors and biosphere exchanges.

Instantiating the class will generate the data for the new database. The *generate* method will write the database.

> **Parameters**
>
> - **up_db_name** (*str*) – Name of the unit process database from which results will be calculated. Must be a registered database name.
> - **agg_db_name** (*str*) – Name of the new aggregated database.
> - **database_type** (*{'LCI', 'LCIA'}, default 'LCIA'*) – Type of aggregated database to generate.
>   - 'LCIA' generates a database with cradle-to-gate inventories stored in the biosphere matrix as "unit impact" biosphere exchanges.
>   - 'LCI' generates a database with cradle-to-gate LCI stored in the biosphere matrix.

> **Note that performance of LCI aggregated databases is very poor due to Brightway2 assuming sparse matrices**

- **method_list** (*list, default list(bw.methods)*) – List of method ids (tuples) for which to generate LCIA scores. Default is all methods.

- **biosphere** (*str, default 'biosphere3'*) – Name of the biosphere database

- **overwrite** (*bool, default False*) – Determines whether an existing database with name *agg_db_name* will be overwritten.

The only method you will use after instantiating the DatabaseAggregator object is generate, see o *quickstart*.

DatabaseAggregator.**generate**()
    Generate data and write to database

3.2 Adding unit impacts exchanges to biosphere and unit impact characterization factors to methods

These steps are done automatically by the DatabaseAggregator. THey can however be called directly:

bw2agg.scores.**add_unit_score_exchange_and_cf**(*method, biosphere='biosphere3'*)
    Add unit score biosphere exchanges and cfs to biosphere and methods.

    Allows the storing of LCIA results in the B matrix for LCI datasets. Makes changes inplace and does not return anything.

        Parameters

- **method** (*tuple*) – Identification of the LCIA method, using Brightway2 tuple identifiers

- **biosphere** (str, default *biosphere3*) – Name of the biosphere database where biosphere exchanges are stored

---

    Note: This function is invoked directly by the DatabaseAggregator

---

bw2agg.scores.**add_all_unit_score_exchanges_and_cfs**(*biosphere='biosphere3'*)
    Add unit scores and cfs for all methods in project.

    Makes changes inplace and does not return anything.

        Parameters biosphere (str, default *biosphere3*) – Name of the biosphere database where biosphere exchanges are stored

Installing

4.1 Installing with pip

The Brightway2-aggregated package is hosted on Pypi and can be installed using pip:

```
pip install presamples
```

4.2 Installing with conda

The Brightway2-aggregated package is hosted on a conda channel and can be installed using conda:

```
conda install --channel pascallesage bw2agg
```

4.3 Install from github

The latest version of the Brightway2-aggregated package is hosted on Github and can be installed using pip:

```
https://github.com/PascalLesage/brightway2-aggregated/archive/master.zip
git clone https://github.com/PascalLesage/brightway2-aggregated.git
cd presamples
python setup.py install
```

Note: On some systems you may need to use `sudo python setup.py install` to install presamples system-wide.

4.4 Brightway2 dependency

Brightway2-aggregated is an extension to the Brightway2 framework, and is therefore dependent on that framework.

CHAPTER 5

Contributing

Contributions are welcome! The code is hosted on Github.

To contribute code, please follow the standard Github workflow. Make sure to add tests for any contribution you make.

Once your contribution has been merged, add yourself to the authors list on this page.

Authors

The core Brightway2-aggregated team is Pascal Lesage from CIRAIG and Chris Mutel from PSI.

Index

A

add_all_unit_score_exchanges_and_cfs()
(*in module bw2agg.scores*), 8
add_unit_score_exchange_and_cf() (*in module bw2agg.scores*), 8

D

DatabaseAggregator (*class in bw2agg.aggregate*),
7

G

generate() (*bw2agg.aggregate.DatabaseAggregator
method*), 8