aomie

Release 0.0.0

Sep 17, 2019

Contents

CHAPTER 1

Overview

docs	
tests	
package	

aomie is a pure Python 3 open source library that helps you handle Iberian electricity market data published by OMIE.

Over 80 statistical indicators of historical data of the Iberian electricity market are published at http://www.omie.es/aplicaciones/datosftp/datosftp.jsp. These indicators are available as downloadable zip files containing text files of daily data with different levels of aggregation (by bidding unit, technology, etc.). To analyse these indicators over time or to make comparison between them you need to follow these steps:

- Download all the data files covering the time horizon of interest
- Extract the daily text files from the downloaded zip files
- Combine the content of potentially thousands of files

aomie automates this workflow. It downloads all files of the required metric over a user-specified time period, unzips the downloaded files and inserts their content into a SQLite database. Once in the database, data analysis can be conveniently performed using SQLite directly or with tools such as pandas in Python or dplyr in R.

- Free software: MIT license

1.1 Installation

```
pip install aomie
```

1.2 Usage

Import the aomie library into your Python libraries, scripts or applications as usual:

```
import aomie
```

amoie includes a succint command line interface that make OMIE data handling extremely easy. Some usage examples follow.

A typical aomie starts by jointly setting the required configuration parameters through a toml configuration file

```
omie -f myconfig.toml
```

The configuration settings included in myconfig.toml are now avalaible to omie commands without having to explicitly call the toml config file again, e.g. to download data just type

```
omie download
```

Obviously you can use a different config file at any time

```
omie -f otherconfig.toml download
```

or just change some of the configuration settings

```
omie -c end 200512
```

To check the current configuration settings type

```
omie -d
```

Once the zip files have been downloaded we can extract them like this

```
omie extract
```

To complete the workflow by inserting the extracted data into a SQLite database type

```
omie insert
```

The aomie commmand fetch bundles all the key data handling tasks. To run these tasks in a single step just type

```
omie -f myconfig.toml -c end 200512 fetch
```

Given the convenience of the fetch command, other commands that just perform one of the steps in omie workflow may seem redundant. Note however that omie data handling tasks covering long time horizons may involve downloading and processing hundreds of MBs that are disk and time consuming, and you may therefore prefer to proceed cautiously step by step.

More information can be found in the command line help, e.g. to learn more about aomie commands such as download type

```
omie download --help
```

From this help we learn that we can download and extract in a single step by typing

```
omie download -e
```

TIP: you can save your self some typing in the command line replacing omie with om, e.g. like this

```
om download -e
```

1.3 Documentation

https://aomie.readthedocs.io/

1.4 Development

To run all the tests run:

```
tox
```

Note, to combine the coverage data from all the tox environments run:

Windows	`set PYTEST_ADDOPTS=--cov-append` `tox`
Other	`PYTEST_ADDOPTS=--cov-append tox`

CHAPTER 2

Installation

At the command line:

```
pip install aomie
```

CHAPTER 3

Usage

To use aomie in a project:

```
import aomie
```

CHAPTER 4

Reference

4.1 aomie

CHAPTER 5

Contributing

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

5.1 Bug reports

When reporting a bug please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

5.2 Documentation improvements

aomie could always use more documentation, whether as part of the official aomie docs, in docstrings, or even on the web in blog posts, articles, and such.

5.3 Feature requests and feedback

The best way to send feedback is to file an issue at https://github.com/qheuristics/aomie/issues.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that code contributions are welcome :)

5.4 Development

To set up *aomie* for local development:

1. Fork aomie (look for the "Fork" button).
2. Clone your fork locally:

```
git clone git@github.com:qheuristics/aomie.git
```

3. Create a branch for local development:

```
git checkout -b name-of-your-bugfix-or-feature
```

   Now you can make your changes locally.

4. When you're done making changes, run all the checks, doc builder and spell checker with tox one command:

```
tox
```

5. Commit your changes and push your branch to GitHub:

```
git add .
git commit -m "Your detailed description of your changes."
git push origin name-of-your-bugfix-or-feature
```

6. Submit a pull request through the GitHub website.

5.4.1 Pull Request Guidelines

If you need some code review or feedback while you're developing the code just make the pull request.

For merging, you should:

1. Include passing tests (run `tox`)[1].
2. Update documentation when there's new API, functionality etc.
3. Add a note to `CHANGELOG.rst` about the changes.
4. Add yourself to `AUTHORS.rst`.

5.4.2 Tips

To run a subset of tests:

```
tox -e envname -- pytest -k test_myfeature
```

To run all the test environments in *parallel* (you need to `pip install detox`):

```
detox
```

[1] If you don't have all the necessary python versions available locally you can rely on Travis - it will run the tests for each change you add in the pull request.
It will be slower though ...

CHAPTER 6

Authors

- Guillermo Lozano Branger - glbanalysis.com

CHAPTER 7

Changelog

7.1 0.0.0 (2019-09-15)

- First release on GitHub.
- First release on PyPI (2019-09-17)

CHAPTER 8

Indices and tables

- genindex
- modindex
- search