

# **Read-me file for computations in**

## **Exogenous factors for order arrivals on the intraday electricity market**

Anke Kramer and Rüdiger Kiesel

### **1 Data Sources**

In our full-text paper, we use data from different sources:

- Order book data: The order book data from EPEX SPOT SE consist of orders with delivery area Germany. The data set is proprietary and thus not provided. Please refer to EPEX SPOT SE for ways to access the data (see, for example, the EEX webshop at <https://webshop.eex-group.com/>).
- Market data: The market data include imbalance volumes, day-ahead forecasts of solar and wind power generation and actual generation of solar and wind power. For the wind power generation, both onshore and offshore generation are used. The data are available from the Bundesnetzagentur | SMARD.de under a CC BY 4.0 license<sup>1</sup>. Please refer to <https://www.smard.de/en> for data access.

In section 2 of our full-text paper, the data used for the analysis are described in more detail.

---

<sup>1</sup>See <https://creativecommons.org/licenses/by/4.0/>.

## 2 Codes

All codes were used and tested under Python 3.7.6. The following packages are required:

- NumPy
- multiprocessing
- SciPy
- numdifftools
- statsmodels
- matplotlib
- pandas
- datetime

Table 1 gives an overview of the code structure. Please note that the functions are tailored for our data source and the data cannot be provided. Comments will be given on how to adapt the data import to other requirements.

For each model, the file `parameter_estimation.py` starts the parameter estimation. Switch between buy and sell orders by setting the variable `ob_side` in line 36 to *'buy'* or *'sell'*. We use four different sets of start parameters for the minimization of the negative log-likelihood function and choose the best estimate. The results are computed in parallel. To change the number of parallel workers, the argument of the `Pool` function needs to be adapted. To compute the results sequentially, the `pool.map` command can be replaced by a simple `map` command. Please note that the self-exciting model must be estimated prior to the full model, since the full model uses the estimated parameters as start parameters.

The inter-arrival times are computed by `get_interarrival_times.py`. Afterwards, figures and tables can be generated with `evaluation.py` and functions from `tools.py`.

Folder	File	Description
functions	database.py	Data import of order book data and market data.
	discretization.py	Class for parameter estimation and computation of standard errors.
	evaluation.py	Provides functions for the evaluation of results, such as figures and tables.
	model.py	Model class.
	optimization.py	Provides functions for minimization of objective function.
	tools.py	Provides auxiliary functions.
model_exogenous/ model_full/ model_selfexciting	model_specs.py	Model specification for the respective model.
	model_specs_noPV.py	Model specification in case no solar power generation variable is considered (e.g. at night).
	parameter_estimation.py	Parameter estimation for the respective model.
	get_interarrival_times.py	Computes inter-arrival times after parameter estimation.

**Table 1.** Source code files and descriptions.