Ha3Py

Release 0.0.2

Jan Wiszniowski, Andrzej Kijko

Feb 13, 2026

CONTENTS

CHAPTER ONE

INTRODUCTION

Hy3Py is the Python package for assessing earthquake recurrence parameters. The typical parameters include the mean activity rate λ, the b-value or β-value of the Gutenberg-Richter frequency-magnitude distribution, and the maximum possible earthquake magnitude, m_{max}, for a given area. Additional parameters can be evaluated when an alternative seismic magnitude occurrence model or a different magnitude distribution is defined. HyPy can incorporate any quality of catalogue, including paleo-earthquakes, historical records, and instrumental data, even if the catalogue is significantly incomplete. (Fig. 1.1) illustrates the data that can be used to determine the required recurrence parameters.

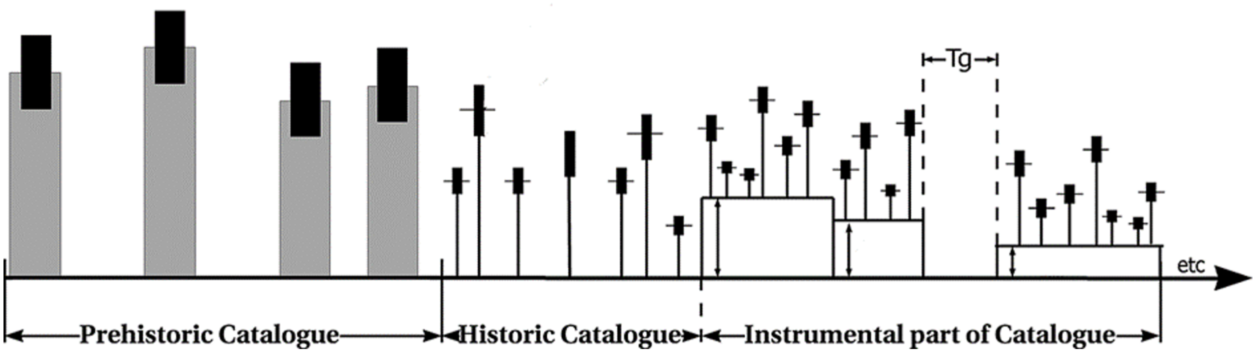

Fig. 1.1: Illustration of typical data used for assessment of model recurrence parameters based on prehistoric, historic, and instrumental datasets (after Kijko *et al.* [2016])

The prehistoric catalogue (or paleo-catalogue) contains earthquake records derived from geological investigations, a process fraught with significant challenges and uncertainties. The timing of these events is often ambiguous, and the catalogue is markedly incomplete. The paleo-catalogue primarily comprises very strong earthquakes that appeared at the surface.

The historical catalogue includes events gathered from written sources. This catalogue is characterised by a reasonably well-established timeframe for earthquake occurrence, but exhibits poor magnitude determination. It remains incomplete, encompassing only the most notable events.

The instrumental catalogues may include several sub-catalogues with different levels of completeness. The time gaps (Tg in Fig. 1.1) represent missing event records.

The Ha3Py code is written in Python and adopts an object-oriented approach (see OOD and OOP) to this issue. The estimation algorithm operates on abstract magnitude probability classes. It provides a highly flexible structure that enables the assessment of multiple probability distributions in various combinations. The library contains predefined classes for magnitude distributions, such as the double-truncated Gutenberg-Richter distribution, and classes for earthquake distribution probabilities, including the Poisson distribution. Refer to the section on the *Ha3Py Operation Diagram* for more information on how the code works.

The Ha3Py requires a working Python environment and the same Python packages to run (see *Installation*).

The key programme of the package is called *ha3*. This programme manages everything necessary for seismic hazard estimation. It defines computation coefficients, assesses earthquake recurrence parameters, and visualises the results. Its queries and outcomes are analogous to those of the HA3 programme written in MATLAB. Additionally, *ha3* generates the *configuration file* for other programmes. However, both *ha3* and the *configuration* programme do not provide all configuration options and cannot define external classes. Utilising all the features of the package requires the manual definition of the configuration file. All programmes, as command line tools, are detailed in the *Command line tools* section. In addition to executing the command line tool, it is also possible to build your hazard evaluation programme using predefined classes and methods. All Ha3Py methods and classes are outlined in the *Methods and Classes* sections.

The main programme in the package is called *ha3*. This programme handles all aspects necessary for seismic hazard estimation. It defines computation coefficients, evaluates earthquake recurrence parameters, and visualises the results. Its queries and outputs are similar to those of the *HA3* programme written in MATLAB. Additionally, *ha3* generates the *configuration file* for other programmes. However, both *ha3* and the *configuration* programmes do not offer all configuration options and cannot define external classes. To utilise all the package's features, the configuration file must be defined manually. All programmes, as command-line tools, are described in the *Command line tools* section. Besides running the command line tool, it is also possible to build your hazard evaluation programme using predefined classes and methods. All Ha3Py methods and classes are detailed in the *Methods and Classes* section.

CHAPTER

TWO

THEORETICAL BACKGROUND

The seismic hazard assessment conducted by the Ha3Py algorithm employs a hybrid approach. The maximum magnitude m_{max} is estimated separately using various methods. In contrast other parameters of the magnitude exceed probability, such as β (the equivalent of the b-value in the Gutenberg-Richter (G-R) magnitude distribution) and the λ coefficient of the Poisson events occurrence probability, are estimated independently using maximum-likelihood methods. Nevertheless, the estimation errors for all parameters are assessed using the maximum likelihood method.

2.1 The earthquake recurrence parameters and their estimation

The sought area-characteristic earthquake recurrence parameters (ERP) $\theta = (\lambda, \beta, m_{max})$ are estimated by the maximum likelihood method. The chosen assessment procedure requires knowledge of the likelihood function of the parameters $\theta = (\lambda, \beta, m_{max})$. Following the multiplicative property of the likelihood function (Rao [1973]), the joint likelihood $\mathcal{L}(\theta)$ based on prehistoric, historic, and complete parts of the catalogue, is of the form:

$$\mathcal{L}(\theta) = \mathcal{L}_P(\theta)^{w_P} \times \mathcal{L}_H(\theta)^{w_H} \times \prod_{i=1}^{N_C} \mathcal{L}_C^{(i)}(\theta)^{w_{Ci}}, \tag{2.1}$$

where $\mathcal{L}_P(\theta)$, $\mathcal{L}_H(\theta)$, and $\mathcal{L}_C^{(i)}(\theta)$ denote the likelihood functions based on prehistoric, historic, and complete sections of the catalogue (Fig. 1.1), N_C is the number of complete catalogues and w_P, w_H, w_{Ci} are weights of prehistoric (P), historic (H) and complete (Ci) sub-catalogues. The form of the likelihood functions $\mathcal{L}_P(\theta)$, $\mathcal{L}_H(\theta)$, and $\mathcal{L}_C^{(i)}(\theta)$ is provided, e.g. by Kijko *et al.* [2016] and Smit *et al.* [2019].

2.2 Maximum magnitude assessment methods

There is no universally accepted method for estimating the value of m_{max}. The presented software evaluates m_{max} by nine different methods. (For details, Kijko [2004]; Kijko and Singh [2011]; Vermeulen and Kijko [2017] and Kijko [2025]). However, the assessment of this parameter is only possible, if information about it is available and provided by seismic event catalogues or other independent sources.

2.2.1 Primitive Procedure

$$m_{max} = m_{max}^{obs} + 0.5 \tag{2.2}$$

2.2.2 Robson-Whitlock Procedure

Robson and Whitlock [1964] showed that, under very general conditions, when magnitudes of n events are arranged in ascending order $m_1 \leq m_2 \leq .... \leq m_{n-1} \leq m_{max}^{obs}$, the estimation of m_{max} takes the form:

$$\widehat{m}_{max} = m_{max}^{obs} + \left(m_{max}^{obs} - m_{n-1}\right) \tag{2.3}$$

with variance

$$\mathrm{VAR}\left(\widehat{m}_{max}\right) = 5\sigma_M^2 + \left(m_{max}^{obs} - m_{n-1}\right)^2,$$

where σ_M^2 denotes the standard error of the largest observed magnitude m_{max}^{obs} determination.

2.2.3 Robson-Whitlock-Cooke Procedure

Cooke [1979] showed that a reduction in the mean squared error of the Robson-Whitlock estimator is possible when some information about the shape of the upper tail of the probability distribution function $f_M(m)$ is known. Assuming that the observed magnitudes are sampled from a distribution that is truncated, as in the double truncated G-R relation (Gutenberg and Richter [1956]), the improved version of the Robson-Whitlock estimator is

$$\widehat{m}_{max} = m_{max}^{obs} + 0.5\left(m_{max}^{obs} - m_{n-1}\right) \tag{2.4}$$

with the variance

$$\mathrm{VAR}\left(\widehat{m}_{max}\right) = 1.5\sigma_M^2 + 0.25\left(m_{max}^{obs} - m_{n-1}\right)^2$$

Note that the *Primitive*, *Robson-Whitlock*, and *Robson-Whitlock-Cooke* procedures do not require specification of the magnitude distribution.

2.2.4 Gibowicz-Kijko Procedure

The procedure relies on the properties of end-point estimators of the uniform distribution. Since the values of the magnitude distribution $F_M(m)$ follow a uniform distribution, the following relation can be anticipated

$$F_M\left(m_{max}^{obs}\right) = \frac{n}{n+1}. \tag{2.5}$$

Since $F_M(m)$ depends on m_{max}, we can derive the estimator of the $\widehat{m}_{max}$ by solving equation (2.5).

2.2.5 Tate-Pisarenko Procedure

The Tate-Pisarenko (and following Kijko-Sellevoll) assess m_{max} by solving the equation

$$\widehat{m}_{max} = m_{max}^{obs} + \Delta, \tag{2.6}$$

where Δ depends on m_{max}. In the case of applying the Tate-Pisarenko procedure, the correction factor Δ takes the form

$$\Delta = \frac{1}{n f_M\left(m_{max}^{obs}\right)}, \tag{2.7}$$

where the probability density function of the magnitude distribution $f_M(m)$ depends on m_{max}. The approximate variance of the Tate-Pisarenko estimator is of the form (Kijko and Graham [1998]; Kijko [2004])

$$\mathrm{VAR}\left(\widehat{m}_{max}\right) = \sigma_M^2 + \frac{n+1}{n^3 f_M^2\left(m_{max}^{obs}\right)}$$

where σ_M^2 denotes the standard error of the largest observed magnitude determination.

2.2.6 Kijko-Sellevoll procedure

Similar to the previous Tate-Pisarenko procedure (2.6), the Kijko-Sellevoll procedure assesses m_{max} by solving the equation

$$\widehat{m}_{max} = m_{max}^{obs} + \Delta,$$

where

$$\Delta = \int_{m_{min}}^{m_{max}} F_M\left(m\right)^n dm, \tag{2.8}$$

The approximate variance of the Kijko-Sellevoll estimator of m_{max} for the frequency-magnitude G-R distribution is of the form

$$\text{VAR}\left(\widehat{m}_{max}\right) = \sigma_M^2 + \Delta^2$$

2.2.7 Procedure based on the largest few earthquakes

The procedure based on the largest few earthquakes is a special case of the Kijko-Sellevoll procedure. The non-parametric magnitude distribution model is applied, using only the largest earthquakes for model building.

2.3 Bayesian maximum magnitude assessment methods

The Bayesian m_{max} assessment methods incorporate any relevant information about m_{max}. This information is external and can come from geology, tectonics, or the seismicity of similar regions. The idea of m_{max} Bayesian estimation was first described by [Cornell, 1994]. It combines two sources of information:

- The information of the m_{max} prior distribution is described as $\pi\left(m_{max}\right)$. The information is based on observations and comes from external sources. Following [Coppersmith, 1994], it is assumed that $\pi\left(m_{max}\right)$ has the form of a double-truncated Gaussian distribution $\pi\left(m_{max}\right) = \mathcal{N}\left(m_{max}^{prior}, \sigma_{m_{max}^{prior}}, m_{max}^L, m_{max}^U\right)$, where $m_{\max}^U$ is the largest magnitude that might ever happen, and m_{max}^L is the magnitude that we are confident has occurred. Usually, it is the maximum observed magnitude.
- The likelihood function $\mathcal{L}\left(\mathbf{m}|m_{max}\right)$ of observed earthquake magnitudes.

Both the prior knowledge about m_{max} and the knowledge derived from the observed magnitudes $\mathbf{m}$, which is known as the posterior distribution of m_{max}, are summarised in the form

$$p_{m_{max}}\left(m_{max}|\mathbf{m}\right) = \begin{cases} 0 & : m_{max} < m_{\max}^L \\ C \cdot \pi\left(m_{max}\right)\mathcal{L}\left(\mathbf{m}|m_{max}\right) & : m_{\max}^L \leqslant m_{max} \leqslant m_{\max}^U \\ 0 & : m_{max} > m_{\max}^U \end{cases}, \tag{2.9}$$

where C is a normalising constant

$$C = 1/\int_{m_{max}^L}^{m_{max}^U} \pi\left(m_{max}\right) L\left(\mathbf{m}|m_{max}\right) dm_{max}.$$

Based on (2.9), three Bayesian analogues of the maximum likelihood (ML) point estimators are used:

- The maximum posterior estimate (MAP) value

$$p_{m_{max}}\left(\widehat{m}_{max}^{posterior}|\mathbf{m}\right) = \text{maximum}$$

- The posterior mean (PM) value (expected value)

$$\widehat{m}_{max}^{posterior} = \int \zeta\, p_{m_{max}}(\zeta) d\zeta$$

- The posterior median is defined by the solution of the equation

$$\int_{m_{\max}^L}^{\widehat{m}_{max}^{posterior}} p_{m_{max}}(\zeta) d\zeta = \frac{1}{2}$$

[Cornell, 1994] proposed

$$\mathcal{L}\left(\mathbf{m}|m_{max}\right) = \prod_{i=1}^{n} f_M\left(m_i|m_{max}\right). \tag{2.10}$$

However, if $f_M\left(m|m_{max}\right)$ is the G-R distribution, the MAP that fulfills (2.10) gives the maximum observed magnitude $m_{\max}^{obs}$ as the solution of (2.9), which is not expected as the maximum possible magnitude m_{max}. Therefore, several techniques were applied to correct the [Cornell, 1994] procedure.

2.3.1 Bayesian m_{max} assessment based on the shift of the likelihood function

In this method $m_{max}^{L} = \widehat{m}_{max}$, where $\widehat{m}_{max}$ can be assessed by any of those described in the *Maximum magnitude assessment methods* section. The likelihood function is

$$\mathcal{L}\left(\mathbf{m}|m_{max}\right) = \prod_{i=1}^{n} f_M\left(m_i|m_{max}\right). \tag{2.11}$$

2.3.2 Bayesian m_{max} assessment based on the Gaussian distribution

In this method $m_{max}^{L} = m_{max}^{obs}$, and the likelihood function is

$$\mathcal{L}\left(\mathbf{m}|m_{max}\right) = \mathcal{N}\left(\widehat{m}_{max}, \sqrt{\mathrm{VAR}\left(\widehat{m}_{max}\right)}\right) \tag{2.12}$$

where $\widehat{m}_{max}$ and $\mathrm{VAR}\left(\widehat{m}_{max}\right)$ can be assessed by any of methods described in the *Maximum magnitude assessment methods* section.

2.3.3 Bayesian m_{max} assessment based on the Fiducial m_{max} distribution

The method assumes that the database information on m_{max} (in our case, seismic event catalogue) is expressed in the form of the fiducial distribution [Pisarenko, 1991]

$$F_{M_{max}}^{FID}\left(m_{max}\right) = 1 - \left[F_M\left(m_{max}^{obs}|m_{max}\right)\right]^n$$

where n is the number of earthquakes of magnitude $m \geq m_C$. The probability density function is

$$f_{M_{max}}^{FID}\left(m_{max}\right) = n\left[F_M\left(m_{max}^{obs}|m_{max}\right)\right]^{n-1} \frac{\partial S_M\left(m_{max}^{obs}|m_{max}\right)}{\partial m_{max}} \tag{2.13}$$

and the likelihood $\mathcal{L}\left(\mathbf{m}|m_{max}\right) = f_{M_{max}}^{FID}\left(m_{max}\right)$. The equation (2.13) is written in the form that uses methods defined in the magnitude distribution classes (see *Ha3Py classes*).

2.4 Estimation uncertainty assessment

$$\mathbf{COV}_{\boldsymbol{\Theta}} = {\mathbf{H}_{\mathcal{L}}}^{-1},$$

where $\boldsymbol{\Theta}$ is the vector estimated by the maximum likelihood coefficients, and $\mathbf{H}_{\mathcal{L}}$ is the Hessian of the likelihood logarithm

$$\mathbf{H}_{\mathcal{L}} = \begin{bmatrix} \frac{\partial^2 \ln(\mathcal{L})}{\partial\Theta_1 \partial\Theta_1} & \cdots & \frac{\partial^2 \ln(\mathcal{L})}{\partial\Theta_1 \partial\Theta_K} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 \ln(\mathcal{L})}{\partial\Theta_K \partial\Theta_1} & \cdots & \frac{\partial^2 \ln(\mathcal{L})}{\partial\Theta_K \partial\Theta_K} \end{bmatrix}$$

The standard deviation of the earthquake occurrence probability coefficients is

$$\sigma_{\boldsymbol{\Theta}} = \sqrt{\mathrm{tr}\left(\mathbf{COV}_{\boldsymbol{\Theta}}\right)}$$

- **Example:**

  When the coefficients are β, λ,

$$\mathbf{COV}_{\beta,\lambda} = \begin{bmatrix} -\frac{\partial^2 \ln(\mathcal{L})}{\partial \lambda^2} & -\frac{\partial^2 \ln(\mathcal{L})}{\partial \lambda \partial \beta} \\ -\frac{\partial^2 \ln(\mathcal{L})}{\partial \lambda \partial \beta} & -\frac{\partial^2 \ln(\mathcal{L})}{\partial \beta^2} \end{bmatrix}^{-1},$$

  where

$$\sigma_\lambda = \mathbf{COV}_{\beta,\lambda}[1,1]$$
$$\sigma_\beta = \mathbf{COV}_{\beta,\lambda}[2,2]$$

2.5 Return period computing methods

The return period ($T_R(m)$) is a function of magnitude. It is defined as the inverse of the annual probability of not occurrence, which is the survival function of the magnitude distribution.

$$T_R(m, \lambda_0, F_M) = \frac{1}{S_M^{max}(m)} \tag{2.14}$$

Sometimes, the annual probability of not-occurrence is described as λ (Not to confuse with the lambda notation as the occurrence probability coefficient). Then, e.g., the return period ($T_R(m)$) is

$$T_R(m) = \frac{1}{\lambda(m)}$$

where $\lambda(m) = \lambda_0 S_M(m|m_0)$, m_0 is the assumed minimum magnitude, λ_0 is the annual probability of not-occurrence corresponding to the minimum magnitude, and $S_M(m|m_0) = 1 - F_M(m|m_0)$ is the survival function of the *Magnitude distribution*.

The return period uncertainty is

$$(\nabla_{\mathbf{\Theta}} T_R)^T \mathbf{COV}_{\mathbf{\Theta}} \nabla_{\mathbf{\Theta}} T_R \tag{2.15}$$

where

$$\frac{\partial T_R(m)}{\partial \Theta_i} = \frac{-1}{(S_M^{max}(m))^2} \frac{\partial S_M^{max}(m)}{\partial \Theta_i}$$

The gradient of the survival function of magnitude distribution depends on not occurrence distribution models (see specific applications in *Seismic event occurrence probability* section). In the typical simplified case, when the magnitude and not-occurrence distribution coefficients are λ_0, β and m_{max}, gradient of the return period is

$$\begin{aligned} \frac{\partial T_R(m)}{\partial \lambda_0} &= \frac{\partial \left(\frac{1}{\lambda_0 \cdot S_M(m|m_0)}\right)}{\partial \lambda_0} = \frac{-1}{\lambda_0^2 \cdot S_M(m|m_0)}, \\ \frac{\partial T_R(m)}{\partial \beta} &= \frac{\partial \left(\frac{1}{\lambda_0 \cdot S_M(m|m_0)}\right)}{\partial \beta} = \frac{-1}{\lambda \cdot S_M^2(m|m_0)} \frac{\partial S_M(m|m_0)}{\partial \beta}, \\ \frac{\partial T_R(m)}{\partial m_{max}} &= \frac{\partial \left(\frac{1}{\lambda_0 \cdot S_M(m|m_0)}\right)}{\partial m_{max}} = \frac{-1}{\lambda \cdot S_M^2(m|m_0)} \frac{\partial S_M(m|m_0)}{\partial m_{max}}, \end{aligned} \tag{2.16}$$

where gradients of survival function $\partial S_M(m|m_0)/\partial m_{max}$ are defined with magnitude distribution classes.

2.6 Synthetic catalogues

Simulation of two types of synthetic catalogues is possible:

- Each seismic event is specified by its origin time and magnitude
- Each seismic event is specified by its magnitude

The first method should be applied for simulating paleo and historical catalogues. In contrast, the second method creates complete catalogues.

The catalogue, which contains both the origin times and magnitudes created by the simulation, with event times, can be used for all likelihood functions. The catalogue only includes seismic event magnitudes can be used by the likelihood function.

Two methods can calculate the catalog of earthquakes containing origin times and magnitudes:

Extreme events simulation
This method divides the catalog period into roughly equal time intervals with random margins, during which it counts a random number of earthquakes with random magnitudes following the specified occurrence probability. From these events, only one earthquake with extreme magnitude is selected and recorded in the catalog.

Full simulation incremental
This method is iterative with incremental event time. First, the time is set to the catalog's starting time. In each iteration, the algorithm generates random periods of non-occurrence of the seismic event drawn from a defined probability of occurrence. The time is incremented by the period and set as the event time. The earthquake with a random magnitude based on the defined magnitude distribution is recorded in the catalog and the time for the next iteration is set to the event time. This process repeats until the end of the catalog is reached.

The catalog of earthquakes containing only origin magnitudes without origin time can be calculated by one method:

Full simulation without date
The method generates a random number of earthquakes for the catalog period, assigns random magnitudes based on the specified occurrence probabilities, and then records all events with their magnitudes into the catalog.

CHAPTER

THREE

OPERATION DIAGRAM

3.1 Overview

The presented programme is a development of the written in Matlab language HA3 programme, which realises a generic methodology that is capable of utilizing different sources of information and uncertainty in the natural hazard based on the theory described among others in a series of works entitled "Estimation of Earthquake Hazard Parameters from Incomplete Data Files" (cite:t:*Kijko_atal_2016*). It applies to various types of natural hazards - prehistoric, historical, and instrumental data can be incorporated, and the programme can account for incomplete data, uncertainty in event sizes, and applied occurrence distributions. In addition, weighting information reflecting individual event validity is available.

3.2 Ha3Py classes

The presented code is written in Python and realises an object-oriented approach to this issue. The estimation algorithm works on abstract magnitude probability classes (Fig. 3.1). It provides a very flexible structure, allowing the assessment of multiple probability distributions in various combinations.

You can define your classes of the studied probability, which describes the probabilities of the *earthquake occurrence model*. Additionally, assuming independence of the magnitude of the seismic events of their occurrence, we can simplify the probability by defining the *magnitude distribution* class independent of the recurrence. Because each probability class inherits from the SciPy Statistical Library, you need only to define the probability density function or the cumulative distribution function. Additionally, you must define the survival function's gradient to assess parameter estimation errors.

Some classes of probability models, like *Gutenberg-Richter* and *compound Poisson distribution*, commonly used in seismic hazard probabilities, are predefined. They are presented in *application modules*.

3.3 Estimation diagram

The algorithm works on abstract probability classes (Fig. 3.1). The following classes are required:

- magnitude distribution class,
- probability of not-occurrence in a given time of the event with a magnitude greater than the given one probability of occurrence a given number of events with a magnitude greater than the given one in a specific time(one class supports two probabilities).
- classes supporting the m_{max} estimation

Some classes are predefined, but user can define their own classes by performing the following required methods of probability distribution classes:

- the probability density function (pdf) (probability mass function in case of discrete distribution - pmf)

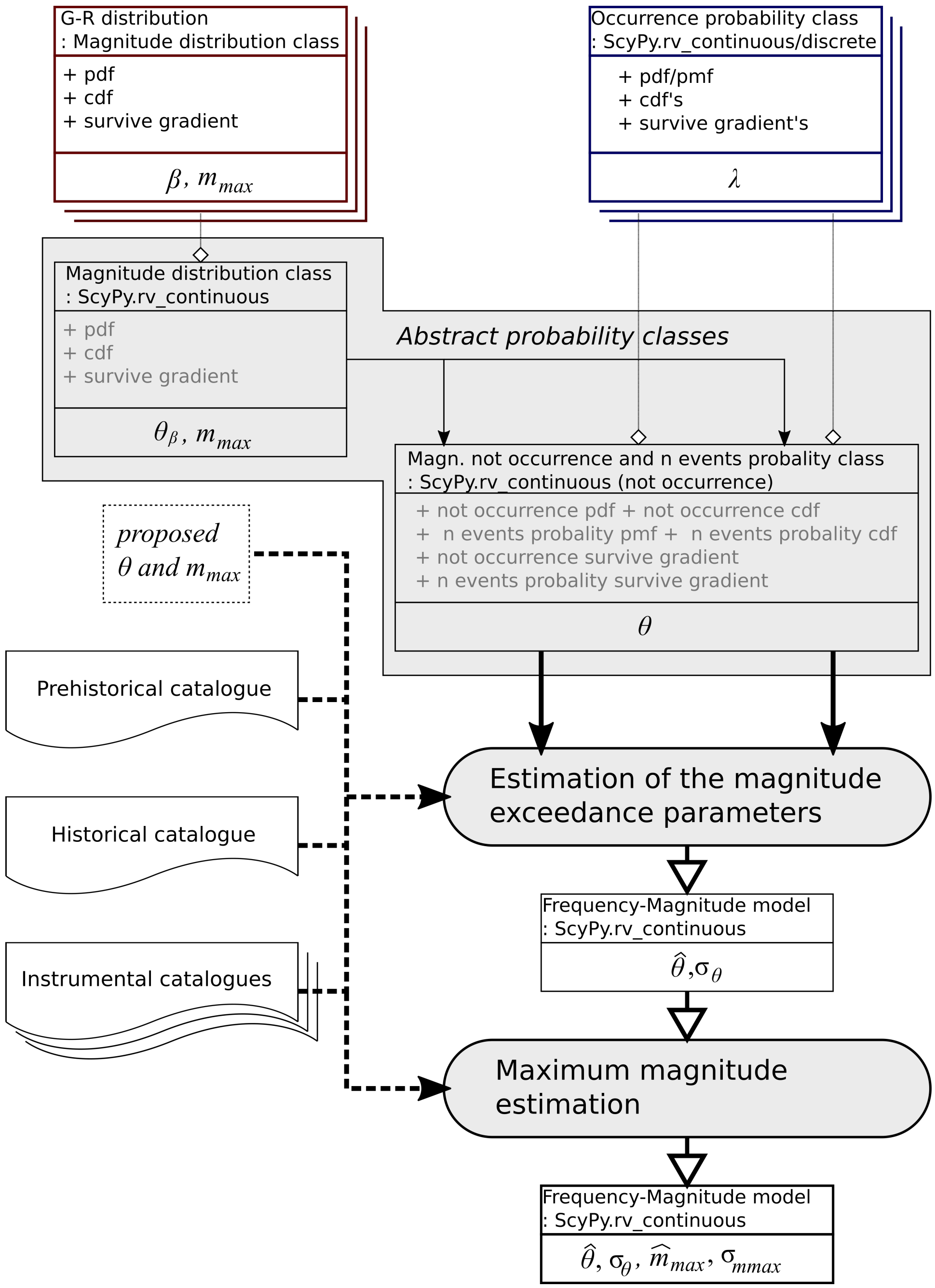

Fig. 3.1: The block diagram of magnitude recurrence parameters estimation.

- or the cumulative distribution function (cdf),
- the gradient of the survival function versus all probability parameters (∇S_M),

In the case of te probability of occurrence over time a given number of events, additional methods for discrete probability are required. Because the probability parameters can be other than lambda and beta, their names must also be defined.

The presented diagram describes all available HyPy seismic hazard assessment computation task. Still, it is possible to perform individual tasks, e.g., only maximum magnitude estimations, under the condition that all required parameters have been previously estimated or set manually in the configuration.

CHAPTER

FOUR

HA3PY INSTALLATION

Ha3Py requires at least Python 3.9. All the required dependencies will be downloaded and installed during the setup process.

4.1 Installing the latest release

The latest release of Ha3Py is available on the Python Package Index.

You can install it easily through `pip`:

```
pip install ha3py
```

To upgrade from a previously installed version:

```
pip install --upgrade ha3py
```

4.2 Installing a developer package

If you want to modify the source code, you should clone the project using `git`:

```
git clone https://github.com/JanWiszniowski/ha3py.git
```

Next, go into the `ha3py` main directory and install the code in "editable mode" by running:

```
pip install -e .
```

You can keep your local Ha3Py repository updated by running `git pull` commands at regular intervals. Thanks to `pip` editable mode, you don't need to reinstall Ha3Py after each update.

CHAPTER

FIVE

COMMAND LINE TOOLS

There are some Python programs, which you can run from the console, to compute seismic hazard parameters, get and visualize the results, or simulate catalogs. WARNING. It is important to have defined the Python scripts catalog in the system paths environment.

5.1 Main program for the assessment of seismic hazard parameters

The main program for assessment of seismic hazard parameters is *ha3*. It works similarly to the program *HA3*, written by A. Kijko in Matlab, which means it asks the user similar questions and generates the same results.

The *ha3* program uses the following modules, which can also be used as standalone programs:

1. **From the *configuration* module, the ha3 calls the procedure *questions***
   that sent the question to the user, and generates the parameter structure required to seismic hazard estimation. In order to operate, see the *Configuration of the Ha3Py* section.
2. **From the *compute* module, the ha3 calls the procedure *compute***
   that estimates the SH based on parameters previously defined. The results are added to the parameter structure. To operate, see the *The earthquake hazard parameters assessment* section.
3. **Results are printed on the console or to a file by procedures in the *print_info***
   and *print_results* module (see *Printing the estimation results*).
4. **Figures of results are plotted by the *plot_results* modules,**
   which consist of procedures that allows plot diagrams of SH properties (see *Plotting the estimation results*).

All presented modules can be called independently. We want to point out that the similarity between *ha3* and Matlab *HA3* makes it impossible to define all new Ha3Py options. If we were going to use them, we should not call *ha3* but: define the configuration file differently, e.g., by calling *ha_config* (see the *Configuration of the Ha3Py* section), modifying the configuration file (see the *Configuration description* section), and finally calling *ha_compute*.

Calling

```
ha3 [ <configuration_file.json> ]
```

where **<configuration_file.json>** is the optional input configuration file name. If *configuration_file.json* exists, the program fills only missing entries in the configuration dictionary, and then displays the message:

```
==============================================================
Loading configuration file for '<area_name>'
==============================================================
```

when **<area_name>** is defined in the configuration file or

```
================================================================
Loading configuration file for undefined area
================================================================
```

If configuration_file.json does not exist, the message will appear:

```
================================================================
The configuration will be created from the beginning
================================================================
```

If configuration_file.json is incorrect or empty, the message will appear:

```
================================================================
The configuration file error: *reason for file not being read if exist*
The configuration will be created from the beginning
================================================================
```

and configuration starts from the beginning.

Next, the list of questions appears for the user to answer. The whole series of queries appears when the input file name is not defined in the configuration. If *configuration_file.json* exists, the program does not prompt for most positions defined in the configuration file. **The *configuration_file.json* changes. The modified configuration is saved to thad file.** The first question is

```
Name of your output file >
```

The output file is the text file name, where the seismic hazard values and estimation info are saved. The extension 'txt' is added to the file name by default.

```
Name of the output configuration file >
```

This file contains the program's configuration, including all the choices made in the program. The output configuration file can be used as input configuration file in other Ha3Py programs. The output configuration file will be in JSON format. Therefore, it is recommended that the file name have theextension *json*. If extension is missing, the default 'json' is added to the file name.

```
Name of the area >
```

Name of the area that you are investigating. The name will appear in reports and results files The query appears when the area name is not defined in the configuration.

```
Pre-historic (paleo) data file name or enter if none >
```

The name of a prehistoric (paleo-) catalogue file if it is available. The query appears when the paleo-catalogue is not defined in the configuration. Putting ENTER indicates not processing a paleo-catalogue. The format of the paleo-catalogue file is similar to the Matlab HA3 formats. It is described in the *Input/Output data section*. Events with magnitude below the completeness level must be removed.

```
Historic data file name or enter if none >
```

The name of a historical catalogue file if it is available. The query appears when the historical catalogue was not previously configured. Putting ENTER indicates not processing a historical catalogue. The format of the historical catalogue file is similar to the Matlab HA3 formats. It described in the *Input/Output data section*. Events with magnitude below the completeness level must be removed.

```
Number of complete data catalogue files >
```

Indicate the number of complete catalogues. The query appears when complete catalogues were not earlier defined in the configuration. When a number greater than zero was entered, the following equation appears the number of catalogues times.

```
Name of the #<i> file with complete data >
```

The name of the <i>-th complete catalogue file. Putting ENTER is not allowed in this case. The format of the complete catalogue file is described in the *Input/Output data section*. The catalogue should be declustered and events with magnitudes below the completeness level must be removed.

```
Maximum (EVER!) observed magnitude determination less or equal to *m.m* (or enter to␣
↪confirm *m.m*) >

Maximum (EVER!) observed magnitude determination >
```

This value refers to the maximum observed magnitude. It is preset over all the provided catalogues and the maximum magnitude *m.m* is found. However, the higher value can be set based on other information. If not set (as in the second query), it must be defined. The program always asks for the maximum observed magnitude.

```
Standard deviation of the maximum observed magnitude (or enter to confirm *e*) >

Standard deviation of the maximum observed magnitude >
```

The standard deviation value refers to the maximum observed magnitude. If it is preset from catalogues, it is proposed as *e*. However, the other value can be set based on other information. If not set (as in the second query), it must be defined. The program always asks for the standard deviation value.

```
Minimum value of magnitude  less or equal to *m.m* (or enter to confirm *m.m*)
```

The minimum value of magnitude refers to the minimum (completeness) magnitude of catalogues. It is proposed to be the smallest completeness magnitude *m.m* of all catalogues. However, the other value, higher than *m.m*, can be set. The program always asks for the minimum completeness magnitude:

```
Year when time span starts less or equal to *yyyy* (or enter to confirm it) >
```

The beginning of the whole time span, if it is to be earlier than the first earthquake time (*yyyy*).

```
Time interval #1 = 1 year by default. You do not need define thr first time interval
Time interval #1 >
Time interval #2 >
Time interval #3 >
```

Time intervals for which seismic hazard will be estimated. Actually, four time intervals are used. The first 1 year time interval automatically provided. Others are defined here in units of years. E.g., times 50, 100 and 1000.

```
Assessment of the maximum regional magnitude m_max is based on:
 1: 'Gibowicz-Kijko (1994)',
 2: 'Gibowicz-Kijko-Bayes (Kijko and Singh, 2011)',
 3: 'Kijko-Sellevoll (1989)',
 4: 'Kijko-Sellevoll-Bayes/compound (Kijko, 2004)',
 5: 'Tate-Pisarenko (Kijko and Graham, 1998)',
 6: 'Tate-Pisarenko-Bayes/compound (Kijko and Singh, 2011)',
```

(continues on next page)

(continued from previous page)

```
 7: 'Non-Parametric-Gaussian/pseudo (Kijko, 2004)',
 8: 'Bayesian MEAN of shifted Likelihood Function & Gaussian Prior (Kijko, 2012)',
 9: 'Bayesian MEAN of Posterior Fiducial & Prior Gauss (Kijko, 2004)'
 10:'Fixed maximum magnitude'
Write proper number >
```

or

```
Assessment of the maximum regional magnitude m_max is based on:
 1: 'Gibowicz-Kijko (1994)',
 2: 'Gibowicz-Kijko-Bayes (Kijko and Singh, 2011)',
 3: 'Kijko-Sellevoll (1989)',
 4: 'Kijko-Sellevoll-Bayes/compound (Kijko, 2004)',
 5: 'Tate-Pisarenko (Kijko and Graham, 1998)',
 6: 'Tate-Pisarenko-Bayes/compound (Kijko and Singh, 2011)',
 7: 'Non-Parametric-Gaussian/pseudo (Kijko, 2004)',
 8: 'Bayesian MEAN of shifted Likelihood Function & Gaussian Prior (Kijko, 2012)',
 9: 'Bayesian MEAN of Posterior Fiducial & Prior Gauss (Kijko, 2004)'
 10:'Fixed maximum magnitude'
Write proper number (or enter to confirm *n*) >
```

Any one of these 9 procedures can be specified to calculate m_{max}. Based on the selected number, the maximum magnitude procedure is defined. Additionally, occurrence provability, magnitude distribution, and delta computation method are also chosen. The Kijko-Sellevoll-Bayes (4) procedure is recommended. The selected method does not agree with the described predefined *maximum magnitude assessment methods* described in *the theoretical background*, because it is a copy of the list of choices from the *HA3* program in Matlab. The selection made is translated to definitions of 'm_max_assessment', 'delta' 'occurrence_probability', and 'magnitude_distribution' (see *Configuration description*)

Then the complete analysis will be based on compound distribution principles. (If Bayesian maximum magnitude is not chosen the question does not appear).

```
Prior value of maximum possible earthquake magnitude >
```

The prior maximum regional magnitude m_{max} value for Bayesian m_{max} assessment methods. The prior maximum regional magnitude **must** be greated then magnitude estimated from the catalog. (If Bayesian maximum magnitude is not chosen the question does not appear)

```
Standard deviation of of prior m_max value >
```

The prior maximum regional magnitude $\sigma_{m_{max}}$ for Bayesian m_{max} assessment methods. If the magnitude distribution is the Compound Gutenberg-Richter distribution (During the selection of assessment of the maximum regional magnitude method 2, 4, 6, 8, or 9 was chosen), the program asks for :the math:*q_{beta}* value:

```
Define the Gutenberg-Richter parameter b uncertainty in percents (or enter for 25.0)  [
↪%] >
```

If the event's occurrence follows the Gamma Compound Poisson distribution, (During the selection of assessment of the maximum regional magnitude method 2, 4, 6, 8, or 9 was chosen), the program asks for :the math:*q_{labda}* value:

```
Define the mean activity rate 'lambda' uncertainty in percents (or enter for 25.0)  [%] >
```

In the case of the Non-Parametric-Gaussian distribution, calculation the seismic hazard from all magnitudes can be significantly time consuming. To minimize te calculation time, the assessment can be processed based on fewer number of magnitudes. For this purpose, the program asks for number of the largest magnitudes.

```
Number of largest magnitudes (or enter for all) >
```

The remaining parameters are defined.

```
Is provision for induced seismicity required (yes/no)? >
```

Induced seismicity here refers e.g., to reservoir triggered seismicity. If a provision is not required, enter *no*. If a provision is required and must be included in the hazard assessment, enter *yes*. Then the user will be asked to provide the multiplicative factor for λ.

```
Multiplicative factor of activity rate (lambda) >
```

The answer is the multiplicative factor value for λ Induced seismicity usually does not exceed the tectonic origin seismicity. Therefore the multiplicative factor for the λ is usually not more than 2.

```
Prior value of the Gutenberg-Richter parameter b (enter if not defined) >
```

The prior b-value of the Gutenberg-Richter distribution for Bayesian magnitude occurrence estimation methods. If you define the prior b-value, the next question is:

```
Standard deviation of the prior b value of >
```

Additional information for the b-parameter is valuable. It will help to stabilize the results in the case of poor quality catalogues. Information from similar tectonic areas can be used.

```
Choose optimization method (see scipy.optimize) >
```

Choosing the optimization method for ln_likelihood maximization. The following method names can be used (see scipy.optimize)

- Nelder-Mead
- Powell
- CG
- BFGS
- Newton-CG
- L-BFGS-B
- TNC
- COBYLA
- COBYQA
- SLSQP
- dogleg
- trust-ncg

At the end, the configuration program *ha3* asks, whether to save the configuration to another JSON file before starting seismic hazard estimation.

```
File to save the only configuration [enter if not save] >
```

It is helpful if you plan to do estimation many times. Next, the program starts computations. The m_{max} is assessed with the cooperation of the user. Other seismic hazard parameters are estimated by the maximum likelihood method. There can be many iterations of m_{max} assessment, which are notified by the message, e.g.

```
Likelihood estimation result: Optimization terminated successfully.
==================================================================
Run round number #1
------------------------------------------------------------------
lambda     =  6.295 (for m_min = 3.00)
beta       =  2.451 (b = 1.1)
m_max (current) = 6.3
Suggested value of m_max = 5.862 (sd_m_max = 0.260)
                           for m_max_obs = 5.80
==================================================================
```

The estimated coefficients list depends on the magnitude occurrence model. Here are beta and lambda. The the the information about the result is shown. Message:

```
You have reach an optimal solution !!!
```

means that an optimal solution was reached, and repeating calculation is not recommended. Otherwise a message:

```
To obtain the optimal solution for m_max, please re-run the process
according to the suggested value until the SUGGESTED and SOLUTION
values are the same.
```

will be displayed. Then the program asks an operator to set the current maximum magnitude based on the shown information:

```
NEW value of m_max (NOT LESS than 5.80) (or enter to accept current 6.30 and finish) >
```

Pressing ENTER accepts the **current** maximum magnitude and finishes the assessment. Putting a new current magnitude value, fixes its as the current, and starts the next estimation round. Next, the program assesses the estimation uncertainty, and calculates the hazard values. Results are saved to the output file as text, saved with the configuration to the JSON file, and printed on screen, e.g.

```
==================================================================
Information provided by each part of catalogue (in percent)
------------------------------------------------------------------
Historic catalogue:     lambda = 51.7%
                        beta   = 22.2%
Complete catalogue(s):  lambda = 48.3%
                        beta   = 77.8%
==================================================================
Final results
------------------------------------------------------------------
Area: Pret
Created on 2025-05-05 18:55:15
Computed on 2025-05-05 19:11:03
------------------------------------------------------------------
lambda     =   6.476 +/- 1.204, (for m_min = 3.00)
    LAMBDA(IS) = 6.476 +/-1.204
beta       =   2.388 +/- 0.108, (b = 1.04 +/- 0.05)
m_max      =   5.850 +/- 0.260
```

(continues on next page)

(continued from previous page)

```
-------------------------------------------------------------------
COV = [    1.449   -0.048 ]
      [   -0.048    0.012 ]
Corr(lambda,beta) = -0.368
==================================================================================
|                                SEISMIC HAZARD                                  |
==================================================================================
| Mag  |Lambda(sf)|   RP    | pr. T=    1| pr. T=   50| pr. T=  470| pr. T=10000|
|------|----------|---------|------------|------------|------------|------------|
| 3.00 | 6.48e+00 |     0.2 | 0.995650   | 1.000000   | 1.000000   | 1.000000   |
| 3.10 | 5.10e+00 |     0.2 | 0.988087   | 1.000000   | 1.000000   | 1.000000   |
| 3.20 | 4.04e+00 |     0.2 | 0.972655   | 1.000000   | 1.000000   | 1.000000   |
| 3.30 | 3.20e+00 |     0.3 | 0.945979   | 1.000000   | 1.000000   | 1.000000   |
| 3.40 | 2.55e+00 |     0.4 | 0.905898   | 1.000000   | 1.000000   | 1.000000   |
| 3.50 | 2.03e+00 |     0.5 | 0.852307   | 1.000000   | 1.000000   | 1.000000   |
| 3.60 | 1.62e+00 |     0.6 | 0.787223   | 1.000000   | 1.000000   | 1.000000   |
| 3.70 | 1.30e+00 |     0.8 | 0.714138   | 1.000000   | 1.000000   | 1.000000   |
| 3.80 | 1.05e+00 |     1.0 | 0.637110   | 1.000000   | 1.000000   | 1.000000   |
| 3.90 | 8.42e-01 |     1.2 | 0.559961   | 1.000000   | 1.000000   | 1.000000   |
| 4.00 | 6.79e-01 |     1.5 | 0.485773   | 1.000000   | 1.000000   | 1.000000   |
| 4.10 | 5.48e-01 |     1.8 | 0.416703   | 1.000000   | 1.000000   | 1.000000   |
| 4.20 | 4.43e-01 |     2.3 | 0.354023   | 0.999999   | 1.000000   | 1.000000   |
| 4.30 | 3.58e-01 |     2.8 | 0.298276   | 0.999994   | 1.000000   | 1.000000   |
| 4.40 | 2.90e-01 |     3.5 | 0.249471   | 0.999967   | 1.000000   | 1.000000   |
| 4.50 | 2.34e-01 |     4.3 | 0.207263   | 0.999846   | 1.000000   | 1.000000   |
| 4.60 | 1.89e-01 |     5.3 | 0.171103   | 0.999400   | 1.000000   | 1.000000   |
| 4.70 | 1.52e-01 |     6.6 | 0.140347   | 0.998004   | 1.000000   | 1.000000   |
| 4.80 | 1.22e-01 |     8.2 | 0.114328   | 0.994276   | 1.000000   | 1.000000   |
| 4.90 | 9.73e-02 |    10.3 | 0.092405   | 0.985677   | 1.000000   | 1.000000   |
| 5.00 | 7.70e-02 |    13.0 | 0.073984   | 0.968311   | 1.000000   | 1.000000   |
| 5.10 | 6.04e-02 |    16.5 | 0.058536   | 0.937201   | 1.000000   | 1.000000   |
| 5.20 | 4.67e-02 |    21.4 | 0.045596   | 0.887090   | 0.999999   | 1.000000   |
| 5.30 | 3.54e-02 |    28.2 | 0.034765   | 0.813586   | 0.999989   | 1.000000   |
| 5.40 | 2.61e-02 |    38.4 | 0.025699   | 0.714211   | 0.999888   | 1.000000   |
| 5.50 | 1.83e-02 |    54.7 | 0.018110   | 0.589005   | 0.998971   | 1.000000   |
| 5.60 | 1.18e-02 |    84.5 | 0.011754   | 0.440523   | 0.991533   | 1.000000   |
| 5.70 | 6.45e-03 |   155.1 | 0.006427   | 0.273309   | 0.937700   | 1.000000   |
| 5.80 | 1.96e-03 |   510.2 | 0.001958   | 0.093087   | 0.591688   | 0.999997   |
==================================================================================
```

Finally, the program plots the seismic hazard figures:

- Annual probability of exceedance versus magnitude
- Return period versus magnitude
- Probability of not exceedance for four defined time intervals

Closing the drawings ends the program.

The figures are also written to png files: *annual_probability.png*, *return_period.png*, and *probabilities.png*.

5.2 Maximum possible earthquake magnitude computation

The maximum magnitude is estimated independently of other magnitude recurrence parameters. Users can use one from the list of predefined magnitude estimation algorithms. The presented software evaluates the probability of exceeding a specified magnitude based on all the presented catalogues, if we have them. We can also apply it only to the historical or instrumental catalogue itself.

Of course, one must be reasonable, especially during the maximum magnitude estimation. (**The user must remember the principle, "*no mathematical tricks can replace the data*".**)

The software's adaptability is a key feature, as it can estimate any probability distribution of exceedance of a magnitude. This can be further divided into two probability distributions: the occurrence of an earthquake in time distribution and magnitude distribution. Importantly, there are no other prior assumptions about the probability function, showcasing the software's adaptability.

To run the maximum possible earthquake magnitude computation write:

```
ha_m_max <configuration_file.json>
```

Unlike the 'ha3' program, the maximum possible earthquake magnitude estimation requires complete configuration in the <configuration_file.json> JSON file. You can use the configuration program for that (see below), although not all available magnitude estimation methods can be defined there. The magnitude estimation configuration requires the method definition maximum observed magnitude, and its uncertainty. Depending on the method, it involves the description of event occurrence and magnitude distribution, the delta calculation method, time span, catalogues, etc. It must be checked in the earthquake magnitude estimation description.

In contrast to the *ha3* and *compute* programs, the *compute_m_max* processes only one round of m_{max} estimation, prints the result, e.g.

```
Result of m_max estimation: m_max = 7.32 (+/- 0.65)
```

and asks

```
Save the result? [yes/no] >
```

User has to put *yes* to save the results of m_{max} estimation. The results are saved to the output configuration JSON file. If the output name is not defined in the configuration, the output file name is 'Ha3Py.json'.

5.3 Configuration of the Ha3Py

Calling

```
ha_config [ <configuration_file.json> ]
```

where <configuration_file.json> is the optional input configuration file name. If configuration_file.json exists the configuration program fills only missing entries in the configuration dictionary. The message will then appear:

```
==============================================================
Loading configuration file for '<area_name>'
==============================================================
```

when <area_name> is defined in the configuration file or

```
==============================================================
Loading configuration file for undefined area
==============================================================
```

If configuration_file.json does not exist, an error will appear:

```
============================================================
The configuration will be created from the beginning
============================================================
```

If configuration_file.json is incorrect or empty, there will then appear the message

```
============================================================
The configuration file error: *reason for file not being read if exist*
The configuration will be created from the beginning
============================================================
```

Then the list of questions appears for the user to answer. The whole series of queries appears when the input file name is not defined in the configuration. If *configuration_file.json* exists, the program does not ask for most positions defined in the configuration. The first question is

```
Name of your output file >
```

The output file is the text file name, where the seismic hazard values and estimation info are saved. The extension 'txt' is added to the file name by default.

```
Name of the output configuration file >
```

This file contains the configuration with all the choices made in the program. The output configuration file can be used as input configuration file in other Ha3Py programs. The output configuration file will be in JSON format. Therefore, it is recommended that the file name have the extension 'json'. If extension is missing, the default 'json' is added to the file name.

```
Name of the area >
```

Name of the area that you are investigating. The name will appear in reports and results files The query appears when the area name is not defined in the configuration.

```
Pre-historic (paleo) data file name or enter if none >
```

The name of a prehistoric (paleo-) catalogue file if it is available. The query appears when the paleo-catalogue is not defined in the configuration. Putting ENTER indicates not processing a paleo-catalogue. The format of paleo-catalogue file is similar to the Matlab HA3 formats. It described in *Input/Output data section*. Events with magnitude below the completeness level must be removed.

```
Historic data file name or enter if none >
```

The name of a historical catalogue file if it is available. The query appears when the historical catalogue was not previously defined in the configuration. Putting ENTER indicates not processing a historical catalogue. The format of historical catalogue file is similar to the Matlab HA3 format. It is described in the *Input/Output data section*. Events with magnitude below the completeness level must be removed.

```
Number of complete data catalogue files >
```

Indicate the number of complete catalogues. The query appears when complete catalogues were not earlier defined in the configuration. When a number greater than zero was put, the following equation appears the enetered number times.

```
Name of the #<i> file with complete data >
```

The name of the <i>-th complete catalogue file. Putting ENTER is not allowed in this case. The format of complete catalogue file is described in the *Input/Output data section*. The catalogue should be declustered, and events with magnitude below the completeness level must be removed.

```
Maximum (EVER!) observed magnitude determination less or equal to *m.m* (or enter to␣
↪confirm *m.m*) >

Maximum (EVER!) observed magnitude determination >
```

This value refers to the maximum observed magnitude. It is preset over all the provided catalogues and the maximum magnitude *m.m* is found. However, the higher value can be set based on other information. If not set (second query), it must be defined. The program always asks for the maximum observed magnitude.:

```
Standard deviation of the maximum observed magnitude (or enter to confirm *e*) >

Standard deviation of the maximum observed magnitude >
```

The standard deviation value refers to the maximum observed magnitude. If it is preset from catalogues, it is proposed as *e*. However, the other value can be set based on other information. If not set (second query), it must be defined. The program always asks for the standard deviation value.

```
Minimum value of magnitude  less or equal to *m.m* (or enter to confirm *m.m*)
```

The minimum value of magnitude refers to the minimum (completeness) magnitude of catalogues. It is proposed es the smallest completeness magnitude *m.m* of all catalogues. However, the other higher than *m.m* value can be set. The program always asks for minimum completeness magnitude.

```
Year when time span starts less or equal to *yyyy* (or enter to confirm it) >
```

The beginning of the whole time span, if it is to be earlier than the first earthquake time (*yyyy*).

```
Time interval #1 = 1 year by default. You do not need define thr first time interval
Time interval #1 >
Time interval #2 >
Time interval #3 >
```

Time intervals for which seismic hazard will be estimated. Actually, four time intervals are used. The 1 year time interval is given by default. Others are defined here in units of years. Suggested times** 50, 100 and 1000.

```
Assessment of the maximum regional magnitude m_max is based on:
 1: 'Gibowicz-Kijko (1994)',
 2: 'Gibowicz-Kijko-Bayes (Kijko and Singh, 2011)',
 3: 'Kijko-Sellevoll (1989)',
 4: 'Kijko-Sellevoll-Bayes/compound (Kijko, 2004)',
 5: 'Tate-Pisarenko (Kijko and Graham, 1998)',
 6: 'Tate-Pisarenko-Bayes/compound (Kijko and Singh, 2011)',
 7: 'Non-Parametric-Gaussian/pseudo (Kijko, 2004)',
 8: 'Bayesian MEAN of shifted Likelihood Function & Gaussian Prior (Kijko, 2012)',
 9: 'Bayesian MEAN of Posterior Fiducial & Prior Gauss (Kijko, 2004)'
 10:'Fixed maximum magnitude'
Write proper number >
```

or, when the maximum regional magnitude assessment is already defined,

```
Assessment of the maximum regional magnitude m_max is based on:
 1: 'Gibowicz-Kijko (1994)',
 2: 'Gibowicz-Kijko-Bayes (Kijko and Singh, 2011)',
 3: 'Kijko-Sellevoll (1989)',
 4: 'Kijko-Sellevoll-Bayes/compound (Kijko, 2004)',
 5: 'Tate-Pisarenko (Kijko and Graham, 1998)',
 6: 'Tate-Pisarenko-Bayes/compound (Kijko and Singh, 2011)',
 7: 'Non-Parametric-Gaussian/pseudo (Kijko, 2004)',
 8: 'Bayesian MEAN of shifted Likelihood Function & Gaussian Prior (Kijko, 2012)',
 9: 'Bayesian MEAN of Posterior Fiducial & Prior Gauss (Kijko, 2004)'
 10:'Fixed maximum magnitude'
Write proper number (or enter to confirm *n*) >
```

Any one of these 10 procedures can be defined to calculate m_{max}. Based on the selected number the maximum magnitude procedure is defined. Additionally, occurrence provability, magnitude distribution, and the delta computation method are also defined. The Kijko-Sellevoll-Bayes (4) procedure is recommended. The selected method does not agree with the described predefined *maximum magnitude assessment methods* described in *the theoretical background*, because it is a copy of the list of choices from the **HA3** program in Matlab. The selection made is translated to definitions of 'm_max_assessment', 'delta' 'occurrence_probability', and 'magnitude_distribution' (see *Configuration description*)

Then the complete analysis will be based on compound distribution principles. (If Bayesian maximum magnitude is not chosen the question does not appear)

```
Prior value of maximum possible earthquake magnitude >
```

The prior maximum regional magnitude m_{max} value for Bayesian m_{max} assessment methods. The prior maximum regional magnitude **must** be greated then magnitude estimated from the catalog. (If Bayesian maximum magnitude is not chosen the question does not appear):

```
Standard deviation of of prior m_max value >
```

The prior maximum regional magnitude $\sigma_{m_{max}}$ for Bayesian m_{max} assessment methods.

If the magnitude distribution is the Compound Gutenberg-Richter distribution, program asks for :the math:*q_{beta}* value:

```
Define the Gutenberg-Richter parameter b uncertainty in percents (or enter for 25.0)   [
↪%] >
```

If the event's occurrence follows the Gamma Compound Poisson distribution, the program asks for :the math:*q_{labda}* value:

```
Define the mean activity rate 'lambda' uncertainty in percents (or enter for 25.0)   [%] >
```

Calculation the seismic hazard from all Non-Parametric-Gaussian distribution magnitudes can be significantly time consuming. In the case of Non-Parametric-Gaussian the program asks for number of largest magnitudes limit used in calculations

```
Number of largest magnitudes (or enter for all) >
```

Next remaining parameters are defined.

```
Is provision for induced seismicity required (yes/no)? >
```

Induced seismicity here refers to water or dam induced seismicity. If a provision is not required, enter *no*. If a provision is required and must be included in the hazard assessment enter *yes*. Then the user will be asked to provide the multiplicative factor for λ.

```
Multiplicative factor of activity rate (lambda) >
```

The answer is the value of the multiplicative factor for λ Induced seismicity usually does not exceed the tectonic origin seismicity. Therefore the multiplicative factor for the λ is usually not more than 2.

```
Prior value of the Gutenberg-Richter parameter b (enter if not defined) >
```

The prior b-value of the Gutenberg-Richter distribution for Bayesian magnitude occurrence estimation methods. If you define the prior b-value, the next question is:

```
Standard deviation of the prior b value of >
```

Additional information for the b-parameter is valuable. It will help to stabilize the results in the case of poor quality catalogues. Information from similar tectonic areas can be used.

```
Choose optimization method (see scipy.optimize) >
```

Choosing the optimization method for ln_likelihood maximization. The following method names can be used (see scipy.optimize)

- Nelder-Mead
- Powell
- CG
- BFGS
- Newton-CG
- L-BFGS-B
- TNC
- COBYLA
- COBYQA
- SLSQP
- dogleg
- trust-ncg

At the end the configuration is saved to the output JSON file. Usually, it is the input configuration file. It is done only when something has changed in the configuration.

5.4 Import catalogue to Ha3Py

The program imports catalogues in other formats into the configuration file

To import catalogue enter

```
ha_import <configuration_file.json> <imported_catalogue> <format_name>
```

The program imports catalogues <imported_catalogue.extension> in all ObsPy formats and in MATLAB files from EPISODES Platform (<format_name> is 'EPISODES'). The <configuration_file.json> must exist created manually or by the *configuration* program. Already defined in the <configuration_file.json> prehistorical and historical catalogs remain unchanged. Complete catalog can be removed or stay depending ob the operato'r decision. Program list existing complete catalogs, e.g,

```
There exists 2 complete catalogs:
 1: c1.txt
 2. c2.txt
Write number of name of the catalog to remove it or ENTER to skip >
```

and the user can remove the catalogue by writing its number (e.g. 1 or 2) or its name (e.g. c1.txt or c2.txt). ENTER ends catalogues removing old catalogs, and the new catalogue will be added to the remaining catalogues. Catalogues need not contain magnitude uncertainties. Therefore, the user must define the default value

```
Default standard deviation of magnitude >
```

which be added to the event in the case of missing magnitude uncertainty. The catalogue must be declustered end events with magnitude below the completeness magnitude must be estimated using another tool. Because catalog do not contain the completeness magnitude after the question

```
Minimum magnitude >
```

the user must write the completeness magnitude value. All events in the catalogue having smaller magnitude wil be ignored.

WARNING Program loads only earthquakes having magnitude Mw.

The modified configuration is saved to the output configuration file. If the output name is not defined in the configuration, the output file name is 'Ha3Py.json'.

5.5 The earthquake hazard parameters assessment

To run the earthquake hazard parameters assessment, enter

```
ha_compute <configuration_file.json>
```

The procedure estimates the magnitude occurrence probability, including magnitude distribution, coefficients (e.g., estimates λ and β) and assesses the maximum possible event magnitude m_{max} in the investigated area. The assessment method depends on the configuration. Results are saved to the output configuration file. The existing in the configuration coefficients are replaced.

The m_{max} is assessed with the cooperation of the user. There can be many iterations of m_{max} assessment, which are notified by the message e.g.

```
Likelihood estimation result: Optimization terminated successfully.
===================================================================
Run round number #1
-------------------------------------------------------------------
lambda     =  6.295 (for m_min = 3.00)
beta       =  2.451 (b = 1.1)
m_max (current) = 6.3
Suggested value of m_max = 5.862 (sd_m_max = 0.260)
                           for m_max_obs = 5.80
===================================================================
```

The estimated coefficients list depends on the magnitude occurrence model. Here are beta and lambda. The information about the result is shown. Message:

```
You have reach an optimal solution !!!
```

means that the optimal solution equals the current magnitude and repeating calculation is not recommended. Otherwise a message:

```
To obtain the optimal solution for m_max, please re-run the process
according to the suggested value until the SUGGESTED and SOLUTION
values are the same.
```

will be displayed. Then the program asks en operator to set the current maximum magnitude based on the shown information

```
NEW value of m_max (NOT LESS than 5.80) (or enter to accept current 6.30 and finish) >
```

Pressing ENTER finishes the assessment. Putting a magnitude value starts the next estimation round. The results are saved to the output configuration JSON file. If the output name is not defined in the configuration, the output file name is 'Ha3Py.json'.

5.6 Printing the estimation results

The magnitude occurrence model estimation results can be printed to screen or to file.

For printing results enter

```
ha_print <configuration_file.json>
```

The program calculates and prints the hazard values. Results are saved to the output file as text, saved with the configuration to the JSON file, and printed on screen, e.g.

```
==================================================================
Information provided by each part of catalogues (in percent)
------------------------------------------------------------------
Historic catalogue:     lambda = 51.7%
                        beta   = 22.2%
Complete catalogue(s):  lambda = 48.3%
                        beta   = 77.8%
==================================================================
Final results
------------------------------------------------------------------
Area: Pret
Created on 2025-05-05 18:55:15
Computed on 2025-05-05 19:11:03
------------------------------------------------------------------
lambda      =   6.476 +/- 1.204, (for m_min = 3.00)
    LAMBDA(IS) = 6.476 +/-1.204
beta        =   2.388 +/- 0.108, (b = 1.04 +/- 0.05)
m_max       =   5.850 +/- 0.260
------------------------------------------------------------------
COV = [    1.449   -0.048 ]
      [   -0.048    0.012 ]
Corr(lambda,beta) = -0.368
=================================================================================
|                                SEISMIC HAZARD                                 |
```

(continues on next page)

(continued from previous page)

```
===============================================================================
| Mag  |Lambda(sf)|    RP    | pr. T=     1| pr. T=    50| pr. T=   470| pr. T=10000|
|------|----------|---------|------------|------------|------------|------------|
| 3.00 | 6.48e+00 |      0.2 |  0.995650   |  1.000000   |  1.000000   |  1.000000   |
| 3.10 | 5.10e+00 |      0.2 |  0.988087   |  1.000000   |  1.000000   |  1.000000   |
| 3.20 | 4.04e+00 |      0.2 |  0.972655   |  1.000000   |  1.000000   |  1.000000   |
| 3.30 | 3.20e+00 |      0.3 |  0.945979   |  1.000000   |  1.000000   |  1.000000   |
| 3.40 | 2.55e+00 |      0.4 |  0.905898   |  1.000000   |  1.000000   |  1.000000   |
| 3.50 | 2.03e+00 |      0.5 |  0.852307   |  1.000000   |  1.000000   |  1.000000   |
| 3.60 | 1.62e+00 |      0.6 |  0.787223   |  1.000000   |  1.000000   |  1.000000   |
| 3.70 | 1.30e+00 |      0.8 |  0.714138   |  1.000000   |  1.000000   |  1.000000   |
| 3.80 | 1.05e+00 |      1.0 |  0.637110   |  1.000000   |  1.000000   |  1.000000   |
| 3.90 | 8.42e-01 |      1.2 |  0.559961   |  1.000000   |  1.000000   |  1.000000   |
| 4.00 | 6.79e-01 |      1.5 |  0.485773   |  1.000000   |  1.000000   |  1.000000   |
| 4.10 | 5.48e-01 |      1.8 |  0.416703   |  1.000000   |  1.000000   |  1.000000   |
| 4.20 | 4.43e-01 |      2.3 |  0.354023   |  0.999999   |  1.000000   |  1.000000   |
| 4.30 | 3.58e-01 |      2.8 |  0.298276   |  0.999994   |  1.000000   |  1.000000   |
| 4.40 | 2.90e-01 |      3.5 |  0.249471   |  0.999967   |  1.000000   |  1.000000   |
| 4.50 | 2.34e-01 |      4.3 |  0.207263   |  0.999846   |  1.000000   |  1.000000   |
| 4.60 | 1.89e-01 |      5.3 |  0.171103   |  0.999400   |  1.000000   |  1.000000   |
| 4.70 | 1.52e-01 |      6.6 |  0.140347   |  0.998004   |  1.000000   |  1.000000   |
| 4.80 | 1.22e-01 |      8.2 |  0.114328   |  0.994276   |  1.000000   |  1.000000   |
| 4.90 | 9.73e-02 |     10.3 |  0.092405   |  0.985677   |  1.000000   |  1.000000   |
| 5.00 | 7.70e-02 |     13.0 |  0.073984   |  0.968311   |  1.000000   |  1.000000   |
| 5.10 | 6.04e-02 |     16.5 |  0.058536   |  0.937201   |  1.000000   |  1.000000   |
| 5.20 | 4.67e-02 |     21.4 |  0.045596   |  0.887090   |  0.999999   |  1.000000   |
| 5.30 | 3.54e-02 |     28.2 |  0.034765   |  0.813586   |  0.999989   |  1.000000   |
| 5.40 | 2.61e-02 |     38.4 |  0.025699   |  0.714211   |  0.999888   |  1.000000   |
| 5.50 | 1.83e-02 |     54.7 |  0.018110   |  0.589005   |  0.998971   |  1.000000   |
| 5.60 | 1.18e-02 |     84.5 |  0.011754   |  0.440523   |  0.991533   |  1.000000   |
| 5.70 | 6.45e-03 |    155.1 |  0.006427   |  0.273309   |  0.937700   |  1.000000   |
| 5.80 | 1.96e-03 |    510.2 |  0.001958   |  0.093087   |  0.591688   |  0.999997   |
===============================================================================
```

5.7 Plotting the estimation results

For plotting enter

```
ha_plot <configuration_file.json>
```

The program plots the seismic hazard figures, e.g Fig. 5.1, Fig. 5.2, Fig. 5.3.

Closing the drawings ends the program. The figures are also written to png files: *annual_probability.png*, *return_period.png*, and *probabilities.png*.

5.8 Creating synthetic catalogues

The program creates synthetic catalogues based on the existing catalogue description in the configuration file and defined seismic hazard parameters. The program does the opposite of calculating the parameters of exceeding the magnitude. Based on these parameters, it simulates catalogs. The program requires catalogue's periods (begin time and end time), completeness magnitude, and magnitude uncertainty for simulation.

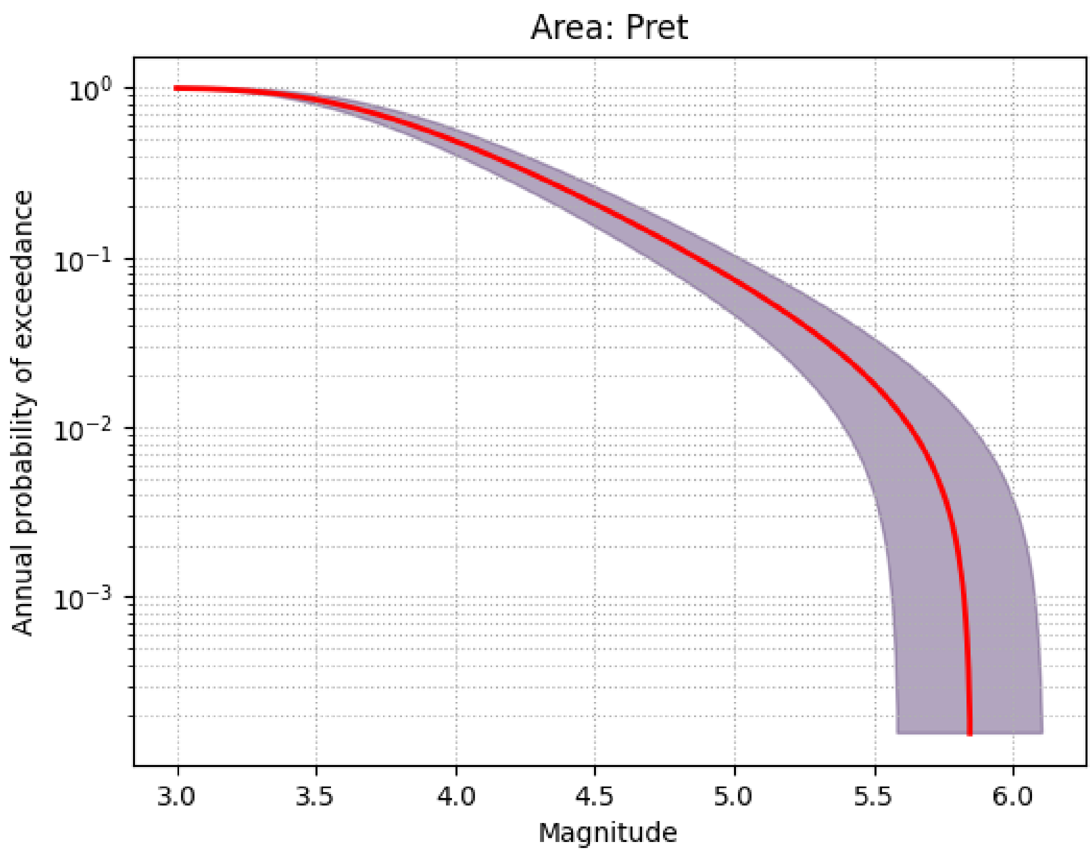

Fig. 5.1: Annual probability of exceedance versus magnitude

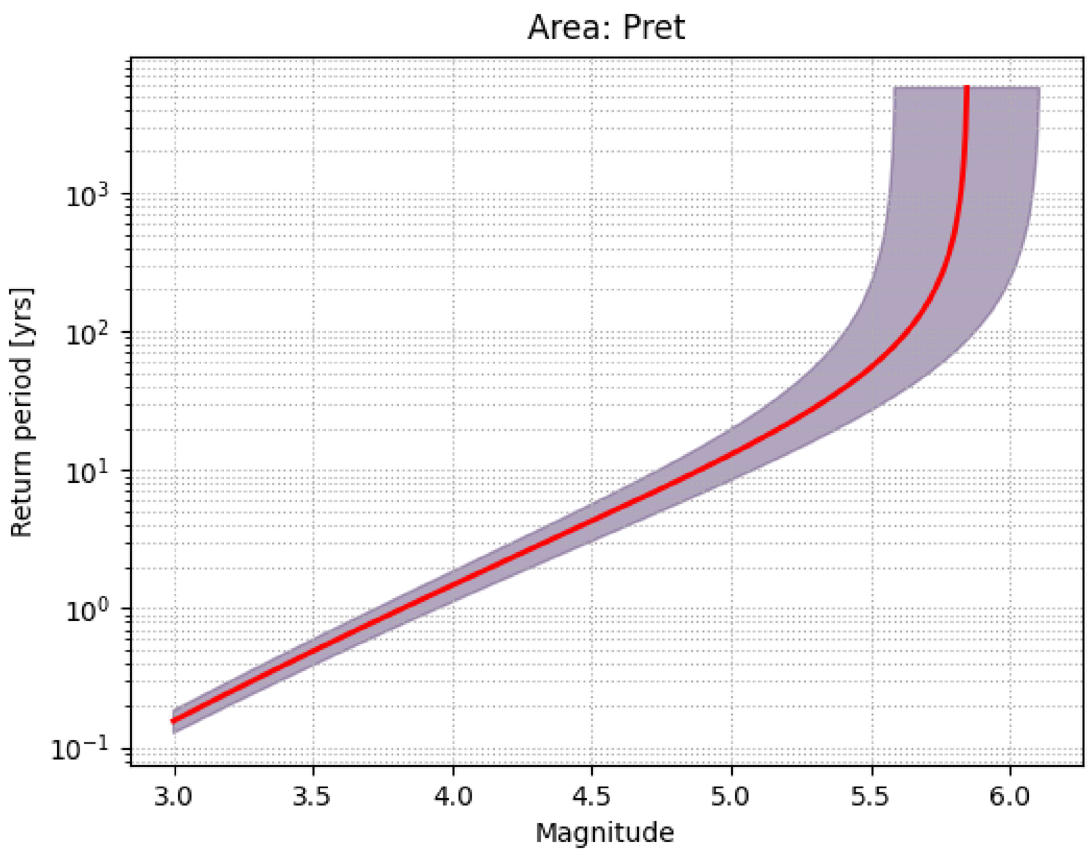

Fig. 5.2: Return period versus magnitude

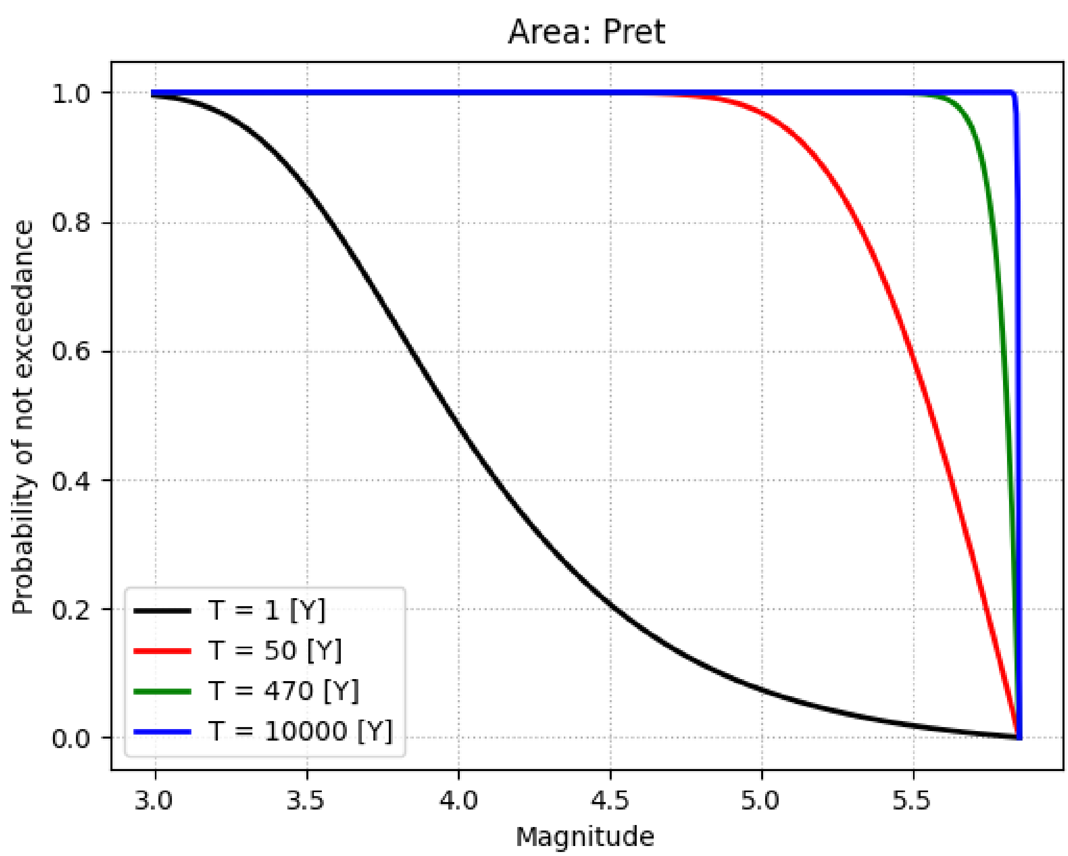

Fig. 5.3: Probability of not exceedance for four defined time intervals

For creating synthetic catalogues enter

```
ha_simulate <configuration_file.json>
```

WARNING! Existing earthquake lists in catalogs are overwritten.

CHAPTER

SIX

CONFIGURATION DESCRIPTION

The configuration is kept in the Python dictionary, where keys are strings and values depending on parameters: strings, float values, integer values, boolean values, sub-dictionaries, or lists. The configuration file (example name: *Ha3Py.json*) is a file in JavaScript Object Notation (JSON). Here is the example file:

```
{
    "paleo_catalog": {
        "begin": -10000.0,
        "end": 1800.0,
        "time_span": 11800.0,
        "sd": 0.5,
        "m_min": 5.94,
        "name": "p.txt",
        "earthquakes": [
            {
                "magnitude": 6.02,
                "date": -9650.0,
                "time_span": 350.0,
                "sd": 0.5
            },
            {
                "magnitude": 6.01,
                "date": -8887.0,
                "time_span": 763.0,
                "sd": 0.25
            },
...
...
            {
                "magnitude": 6.08,
                "date": 1203.0,
                "time_span": 247.0,
                "sd": 0.25
            }
        ]
    },
    "historic_catalog": {
        "begin": 1550.0,
        "end": 1800.0,
        "time_span": 250.0,
        "m_min": 3.5,
```

(continues on next page)

(continued from previous page)

```
        "name": "h.txt",
        "earthquakes": [
            {
                "magnitude": 4.16,
                "date": 1551.704109589041,
                "time_span": 1.7041095890410816,
                "sd": 0.25
            },
            {
                "magnitude": 4.97,
                "date": 1553.7452054794521,
                "time_span": 2.041095890411043,
                "sd": 0.25
            },
...
...
            {
                "magnitude": 4.59,
                "date": 1799.5205479452054,
                "time_span": 3.263013698630175,
                "sd": 0.25
            }
        ]
    },
    "complete_catalogs": [
        {
            "begin": 1800.0,
            "end": 1900.0,
            "time_span": 100.0,
            "m_min": 3.5,
            "sd": 0.1,
            "name": "c1.txt",
            "earthquakes": [
                {
                    "magnitude": 3.61,
                    "date": 1850.0,
                    "sd": 0.1
                },
                {
                    "magnitude": 4.86,
                    "date": 1850.0,
                    "sd": 0.1
                },
...
...
                {
                    "magnitude": 3.03,
                    "date": 1950.0,
                    "sd": 0.1
                }
            ]
        }
```

(continues on next page)

(continued from previous page)


```
    ],
    "begin": -10000.0,
    "end": 2000.0,
    "output_text_file": "Ha3PyMTest",
    "area_name": "Area",
    "created_on": "2023-06-04 09:42:14",
    "m_max_obs": 6.08,
    "sd_m_max_obs": 0.25,
    "induced_seismicity": "no",
    "induced_seismicity_coefficient": 1.0,
    "time_intervals": [
        1.0,
        50.0,
        470.0,
        1000.0
    ],
    "time_span": 12000.0,
    "m_min": 3.0,
    "procedure id": 8,
    "prior_m_max": 6.08,
    "sd_prior_m_max": 0.5,
    "m_max_prior": 7.0,
    "sd_m_max_prior": 0.25,
    "prior_b": "",
    "prior_beta": null,
    "sd_prior_beta": null,
    "likelihood_optimization_method": null,
    "m_max_current": 8.99,
    "q_beta": 16.0,
    "q_lambda": 16.0,
    "beta": 2.619477443131598,
    "lambda": 7.878695391805496,
    "magnitude_distribution": "GutenbergRichterBayes",
    "occurrence_probability": "Poisson",
    "delta": "Kijko-Sellevoll",
    "m_max_suggested": 7,
    "sd_m_max": 0.5,
    "cov_beta_lambda": [
        [
            0.0041074392844434576,
            -0.03865960610149208
        ],
        [
            -0.03865960610149208,
            1.050816279158931
        ]
    ],
    "sd_beta": 0.06408930709910553,
    "sd_lambda": 1.0250933026602658,
    "b": 1.0516286240033526,
    "sd_b": 0.02783363242214443,
    "lambda_is": 9.532098273353775,
```


(continues on next page)

(continued from previous page)


```
    "sd_lambda_is": 1.0250933026602658,
    "hazard": [
        {
            "mag": 3.0,
            "lambda_mag": 7.703958463289057,
            "return_period": 0.12980339974121163,
            "probabilities": [
                1.0,
                1.0,
                1.0,
                1.0
            ]
        },
        {
            "mag": 3.1,
            "lambda_mag": 5.8736839448960225,
            "return_period": 0.1702509037567397,
            "probabilities": [
                0.9932831012788538,
                1.0,
                1.0,
                1.0
            ]
        },
...
...
        {
            "mag": 8.900000000000006,
            "lambda_mag": 1.2599577011667585e-05,
            "return_period": 79367.74378012613,
            "probabilities": [
                1.259949267673477e-05,
                0.0006297680614141266,
                0.005903212762204291,
                0.012515638410797392
            ]
        }
    ],
    "m_max": 8.99
}
```


6.1 Parameters description

Below is a description of parameters. Not all described below parameters are in the example.

6.1.1 Catalogues

Three types of catalogues can be defined. The paleo-catalogue and historical catalogue can be described only once, whereas many complete catalogues having different periods and completeness magnitude can be determined. Catalogues should be declustered and can not contain events with magnitude below the defined minimum magnitude m_{min}. Magnitudes in all catalogs must be unified. We recommend the moment magnitude (Mw).

paleo_catalog
(dict) The prehistoric (paleo-) catalogue

historic_catalog
(dict) The historic catalogue

complete_catalogs
(list(dict)) The list of complete (instrumental) catalogues

6.1.1.1 Catalogue parameters

begin
(float) The beginnin of the catalogue time in years AD.

end
(float) The end of the catalogue time in years AD.

time_span
(float) The catalogue time span in years.

sd
(float) The standard magnitude deviation of earthquakes in the catalogue. This value is valid unless individual the standard deviation is defined for a specific earthquake.

m_min
(float, str) The minimum (completeness) magnitude m_{min}. If it is a float value, it means the m_{min}. WARNING. The catalogues can not contain earthquakes with magnitudes below m_{min}. If *m_min* is the text *variable*, it means that the completeness level is variable and is defined with events. The *variable* option is allowed only for complete catalogues.

name
(str) The name of the catalogue. Any name is allowed, though it depends on the imported catalogue format.

earthquakes
(list(dict)) The list of earthquakes parameters. It is described below.

6.1.1.2 Earthquake parameters

Specific earthquake parameter depends on the catalogue type. All possible fields are described below, though only magnitude can be sufficient for complete catalogs.

magnitude
(float) The earthquake magnitude. Magnitudes in all catalogs must be unified.

date
(float) The date of the earthquake in floating point years AD.

time_span
(float) The time_span to the previous event. It is required in historical and prehistoric catalogues.

sd
(float) The standard deviation of the earthquake magnitude.

m_min
(float) The minimum catalogue magnitude attributable to the period of occurrence of this earthquake. It is required in complete catalogue when *m_min* of the catalogue is *variable*.

6.1.2 Earthquake occurrence and magnitude distribution parameters and coefficients

6.1.2.1 Earthquake occurrence and magnitude distribution parameters

magnitude_distribution
(str) The name of the magnitude distribution model, e.g., *Compound Gutenberg-Richter*. Available options are:

- ***Gutenberg-Richter***
  The double truncated Gutenberg-Richter magnitude distribution, where $F_M(m) = \frac{1-\exp[-\beta(m-m_{min})]}{1-\exp[-\beta(m_{max}-m_{min})]}$ (see the full definition in the Gutenberg-Richter class description).
- ***Compound Gutenberg-Richter***
  The compound Gutenberg-Richter magnitude distribution where $F_M(m) = C_\beta \left[1 - \left(1 + \overline{\beta}\left(m - m_{min}\right)/q_\beta\right)^{-q_\beta}\right]$ (see the full definition in the CompoundGutenbergRichter class description). It requires definition of constant q_beta parameter.
- ***Nonparametric Gaussian kernel***
  The nonparametric magnitude distribution based on a Gaussian kernel (see the full definition in the NonparametricGaussianKernel class description)

occurrence_probability
(str) The name of the earthquake occurrence model, e.g., 'Poisson-gamma compound'. Available options are:

- ***Poisson***
  Poisson events occurrence probability
- ***Poisson-gamma compound***
  The combination of the Poisson distribution with the gamma distribution. It requires te definition of the constant *q_lambda* parameter.

q_lambda
(float) Constant coefficient required for the compound Gutenberg-Richter magnitude distribution $q_\lambda = \lambda^2/{\sigma_\lambda}^2$

q_beta
(float) Constant coefficient required for the compound Poisson-gamma earthquake occurrence probability $q_\beta = \beta^2/{\sigma_\beta}^2$

6.1.2.2 Earthquake occurrence and magnitude distribution coefficients

Earthquake occurrence and magnitude distribution coefficients depend on the earthquake occurrence and magnitude distribution classes used for seismic hazard assessment and defined by the parameters. The example below refers to a typical solution:

```
"beta": 2.619477443131598,
"lambda": 7.878695391805496,
"q_beta": 16.0,
"q_lambda": 16.0,
"prior_beta": null,
"sd_prior_beta": null,
"sd_beta": 0.06408930709910553,
"sd_lambda": 1.0250933026602658,
```

prior``_*`` coefficients are required, if we use Bayesian approach to the earthquake occurrence assessment. (` means the coefficient name)

6.1.3 Maximum magnitude parameters

procedure_id
(int) The number of the programs maximum assessment method, which is selected in the *ha_config* or *ha3* .

m_max_method
(str) The name of the defined maximum assessment method. It is set in the *ha_config* or *ha3* programs and used on for the report. If you configure manually, you should define or redefine it to have the correct report.

m_max_obs
(float) The maximum magnitude ever observed in the studied region.

sd_m_max_obs
(float) The standard deviation of the maximum observed magnitude.

prior_m_max
(float) The prior maximum magnitude used in Bayesian maximum magnitude assessments. The value is defined based on external information, like tectonic data. **The defined magnitude must be greater than the magnitude assessed from the catalogues.**

sd_prior_m_max
(float) The standard deviation of the prior maximum magnitude used in Bayesian maximum magnitude assessments.

m_max_suggested
(float) The temporary suggested maximum magnitude which is the result od maximum magnitude computation.

m_max_current
(float) The temporary suggested maximum magnitude, which used for the magnitude distribution definition used for computation before the final maximum magnitude is set. It is not the final maximum magnitude. An operator must confirm this value. So, when the final maximum magnitude is set, the m_max_current should equal the final maximum magnitude.

m_max
(float) The final maximum magnitude.

sd_m_max
(float) The standard deviation of the final maximum magnitude.

m_max_assessment
(str) The non Bayesian maximum magnitude assessment method. Available options:

- ***solve_delta***
  The maximum magnitude assessment by numerical solving the equation $\widehat{m}_{max} = m_{max}^{obs} + \Delta$. The coefficient *delta* must be set.
- ***Gibowicz-Kijko***
  The procedure relies on the properties of end-point estimators of the uniform distribution.
- ***iteration_delta***
  Similar to *solve_delta*, but the equation $\widehat{m}_{max} = m_{max}^{obs} + \Delta$ is solved by iteration
- ***momentum***
  Evaluate the maximum magnitude according to the moment estimator
- ***primitive***
  The maximum magnitude is 0.5 greater than the maximum observed magnitude,

- ***Robson-Whitlock***
  Robson-Whitlock maximum magnitude assessment procedure,
- ***Robson-Whitlock-Cooke***
  Robson-Whitlock-Cooke maximum magnitude assessment procedure,

bayesian_m_max_assessment
(str) The Bayesian maximum magnitude assessment method. When this parameter is not defined, the maximum magnitude is estimated by the method described in the 'm_max_assessment' parameter. When this parameter is not defined, the maximum magnitude is estimated by this method whereas the 'm_max_assessment' parameter is used in some Bayesian methods, which requires the pre-estimation of the maximum magnitude from data. When this parameter is defined, *prior_m_max* and *sd_prior_m_max* must be defined. Available options are:

- ***bayesian_normal***
  In this method the Bayesian likelihood function is normal truncated to the maximum observed magnitude and maximum possible magnitude,
- ***bayesian_normal_unlimited***
  In this method the Bayesian likelihood function is normal,
- ***bayesian_by_shift***
  In this method the Cornel Bayesian likelihood function is shofted by the maximum magnitude estimated by non Bayesian method,
- ***bayesian_fiducial***
  The method assumes that the database information on maximum magnitude (in our case, seismic event catalogue) is expressed in the form of the fiducial distribution,
- ***fixed_value***
  Primitive method, where the entered prior maximum magnitude is not modified.

bayesian_m_max_estimator
(str) The Bayesian maximum magnitude likelihood point estimator. Three estimators are available: *max* - maximum likelihood, *expected* - expected likelihood, and *median* - median likelihood, treated as a probability distribution.

m_min_ref
(float) The minimum magnitude definition defined for maximum magnitude estimation. It is optional. When is not defined, the minimum magnitude of all catalogs is used *m_min*. However, *m_min* can be small, which is numerically not recommended.

delta
(str) The method of Δ calculation in the maximum magnitude assessment, which allows solving the formula $\widehat{m}_{max} = m_{max}^{obs} + \Delta$. There are two available methods of Δ calculation: *Kijko-Sellevoll* and *Tate-Pisarenko*.

6.1.4 Other parameters

begin
(float) The beginning of all catalogues time in years AD (data and time converted to float year).

end
(float) The end of all catalogues time in years AD.

time_span
(float) The all catalogues time span in years.

output_text_file
(str) The name of the output text file, where the estimation results report is stored.

area_name
(str) The name of the investigated area.

created_on
(str) The date and time of data creation, e.g., *2023-06-04 09:42:14*.

induced_seismicity
(str) Information of the investigated seismicity. It is defined in the case of induced seismicity (*no* or *yes*).

induced_seismicity_coefficient
(float) The value of final λ correction used in the case of induced seismicity.

time_intervals
(list) The time interval of all catalogues. It is set while adding catalogues.

m_min
(float) The minimum magnitude of all catalogues. It is set while adding catalogues. Usually the event occurrence is assessed for this magnitude.

prior_b
(float) The prior b value for Bayesian estimation of magnitude distribution.

likelihood_optimization_method
(str) The optimisation method used for maximum likelihood computation.

COV
(np.array) The covariance matrix (Inversion of ln ln_likelihood hessian).

coefficients
(list(str)) The list of earthquake occurrence probability estimated coefficients names, e.g. ['lambda', 'beta', 'm_max'] (see *Earthquake occurrence and magnitude distribution coefficients*).

b
(float) The b-value of the Gutenberg-Richter magnitude distribution,

sd_b
(float) The standard deviation of the b-value.

lambda_is
(float) The annual λ in the case of induced seismicity.

sd_lambda_is
(float) The standard deviation of the induced seismicity λ.

hazard
(list(dict)) The list of magnitude exceedance probabilities, return periods, etc. for various magnitudes (see *Seismic hazard values*).

6.1.5 Seismic hazard values

mag
(float) The magnitude value for which the following parameters were calculated.

lambda_mag
(float) Lambda value for relevant magnitude,

return_period
(float) The return period in years

probabilities
(list(float)) List of non-exceeding the above magnitude *mag* probability in periods defined in the *time_intervals* list

6.1.6 Simulation configuration

simulation
(dict) It is the definition of parameter required for creating artificial catalogues.

Simulation parameters are analogues to catalogs parameters. The dictionary *simulation* consists of:

pre-historic data
(dict) which has a description how to generate the pre-historic catalogue,

historic data
(dict) which has a description how to generate the historic catalogue,

complete data
(list) which is the list of a few complete catalogues, each has the description how to generate the complete catalogue.

Description how to generate the catalogue, consists of:

generator
(str) The generator name. There are allowed three names: *full_simulation_incremental*, *extreme_simulation*, and *no_date_simulation*. For paleo- and historic catalogues 'full_simulation_incremental' and 'extreme_simulation' can be applied, whereas for complete catalogues 'extreme_simulation' and 'no_date_simulation' should be applied.

begin
(float) The beginning of the catalogues in floating point years AD.

end
(float) The end of the catalogues in years AD.

time_span
(float) The time interval of the catalogue in years

time_uncertainty
The event time uncertainty in the catalogue

magnitude_uncertainty
(float) The magnitude uncertainty in the catalogue

name
(str) The name of the catalogue

m_min
(float) The minimum magnitude in the catalogue (required)

CHAPTER

SEVEN

INPUT/OUTPUT DATA

The principal input data is the JSON file containing all the necessary data. Input and output files have the same structure, both for seismic hazard calculation and catalogue simulation. Different parameters are required and set then. In addition, you can use other catalogue files.

7.1 Catalogue formats converted to input data

7.1.1 HA3 catalogues

They are text files in free Format.

7.1.1.1 Pre-historic catalogue

Line 1
y1, m1, d1 = year, month, day of the beginning of the pre-historic part of the catalogue (e.g., -10000, 1, 1).

Line 2
y2, m2, d2 = year, month, day of the end of the pre-historic part of the catalogue (e.g., 800, 12, 31).

Lines 3 - n
For each earthquake year1, year2, magnitude, magnitude uncertainty, where year1 and year2 are lower and upper bounds of estimated occurrence time, the magnitude of the event and its uncertainty.

Example:

```
-10000   1   1
 1799  12  31
-9850 -9450   6.02   0.5
-9087 -8687   6.01   0.25
-7555 -7155   6.03   0.25
-6530 -6130   5.95   0.25
-6045 -5645   5.94   0.05
-4703 -4303   6.03   0.25
-3288 -2888   6.01   0.25
-2542 -2142   6.02   0.25
-1457 -1057   6.01   0.25
 -448   -48   6.01   0.25
  756  1156   6.08   0.25
 1003  1403   6.08   0.25
```

7.1.1.2 Historic catalogue

Line 1
y1, m1, d1 = year, month, day of the beginning of the historic part of the catalogue (e.g., 1550, 1, 1).

Line 2
y2, m2, d2 = year, month, day of the end of the historic part of the catalogue (e.g., 1550, 1, 1).

Line 3
Level of completeness for a magnitude (e.g., 5.0)

Line 4
se = standard error of magnitude determination (magnitude uncertainty). It is assumed that standard errors for each earthquake are the same (e.g., 0.25)

Lines 5 - n
For each earthquake: year, month, day, magnitude.

Example:

```
1550   1   1
1799  12  31
5.0
0.25
1570   3  29   5.8
1590   4  11   5.7
1633  11  28   5.2
1635   1   7   5.2
1668  11   2   5.1
1672   6  20   5.5
1680   5  10   5.0
1684   8  27   5.2
1692   6  24   5.3
1692   8   9   5.0
1711   8  27   5.3
1713   3  20   5.4
1716   1  10   5.4
1725   5   7   5.2
1744  10  14   5.1
1758   9   7   5.1
1764   4   6   5.3
1793   3  25   5.2
```

7.1.1.3 Complete catalogue

Line 1
y1, m1, d1 = year, month, day of the beginning of the complete part of the catalogue (e.g., 1800, 1, 1).

Line 2
y2, m2, d2 = year, month, day of the END of the complete part of the catalogue (e.g. 1899 12 31).

Line 3
Level of completeness for a magnitude (e.g., 3.5)

Line 4
se = standard error of magnitude determination (magnitude uncertainty). It is assumed that for each earthquake in the file, se-s are the same. (e.g., 0.2)

Lines 5 - n
Magnitudes of all earthquakes.

Example:

```
1800  1  1
1899 12 31
3.5
0.1
3.6
4.8
3.6
3.7
3.9
4.2
...
...
...
```

7.1.2 ObsPy formats of catalogues

The program imports catalogues in all ObsPy catalogues formats. They are formats:

CMTSOLUTION
Format of the *Global Centroid-Moment-Tensor (CMT) Project*

EVT
SeismicHandler EVT file format

FNETMT
Format of moment tensor files (TEXT format) provided for the F-net broadband seismograph network operated by the National Research Institute for Earth Science and Disaster Prevention in Japan (NIED)

FOCMEC
Format used to store data for focal mechanism determinations

GSE2
GSE2.1 Format developed for use in the GSE's Technical Test 3 (GSETT-3), which was a trial starting on 1st January 1995 for a system of global monitoring of nuclear explosions. Based on experience during GSETT-3, a new format was developed for use with the International Monitoring System (IMS) of the Comprehensive Nuclear Test Ban Treaty. Here is documentation of GSE2.1.

HYPODDPHA
HypoDD format.

IMS10BULLETIN
(IASPEI Seismic Bulletin Format) ISF is the IASPEI approved standard format for the exchange of parametric seismological data (hypocentres, magnitudes, phase arrivals, moment tensors, etc.). It was adopted as standard in August 2001 by IASPEI's Commission on Seismic Observation and Interpretation at the Scientific Assembly in Hanoi, Vietnam. The format is an extension of IMS1.0. Here is the brief online description of IASPEI Seismic Format (ISF).

MCHEDR
Machine-readable Earthquake Data Report format used to transmit data to the ISC by only the U.S. National Earthquake Information Service and is thus an agency-specific format.

NDK
File format used to store and distribute the Global Centroid-Moment-Tensor (CMT) catalog (formerly

the Harvard CMT catalog)

NLLOC_HYP
File format generated by the NonLinLoc events location format

NORDIC
NORDIC format was one of the first attempts to create a more complete format for data exchange and processing. The initiative came from the need to exchange and store data in Nordic countries and the so called Nordic format was agreed upon among the 5 Nordic countries. The format later became the standard format used in the SEISAN database and processing system and is now widely used. The format tried to address some of the shortcomings in the HYPO71 format by being able to store nearly all parameters used.

QUAKEML
QuakeML is a flexible, extensible and modular XML representation of seismological data. The documentation may be found at QuakeML page.

SC3ML
The SeisComp catalog format.

SCARDEC
The SCARDEC Source Time Functions Database format.

ZMAP
The format of the written for the MATLAB set of tools in order to the analysis of earthquake catalogues.

7.1.3 EPISODES Platform format

EPISODES Platform of the Anthropogenic hazard keeps catalogues in MATLAB file described at Anthropogenic hazard EPISODES Platform.

7.2 Output text format

The output is a text file containing the base information, estimated seismic hazard parameters, e.g., lambda, beta, m_max, and a table with magnitude exceedance : lambda, return period (RP), and probability in chosen period years.

Example:

```
==================================================================
Final results
------------------------------------------------------------------
Area: Pretoria
Created on 2025-05-23 07:06:40
Computed on 2025-05-23 07:34:22
------------------------------------------------------------------
lambda      =   4.887 +/- 0.812, (for m_min = 3.00)
m_max       =   5.860 +/- 0.260
------------------------------------------------------------------
COV = [    0.660 ]
=================================================================================
|                                 SEISMIC HAZARD                                |
=================================================================================
| Mag  |Lambda(sf)|   RP    | pr. T=    1| pr. T=   50| pr. T=  100| pr. T= 1000|
|------|----------|---------|------------|------------|------------|------------|
| 3.00 | 4.89e+00 |     0.2 |  0.985942  |  1.000000  |  1.000000  |  1.000000  |
```

(continues on next page)

(continued from previous page)

```
| 3.10 | 4.31e+00 |     0.2 | 0.977959  | 1.000000  | 1.000000  | 1.000000  |
| 3.20 | 3.66e+00 |     0.3 | 0.963076  | 1.000000  | 1.000000  | 1.000000  |
| 3.30 | 3.08e+00 |     0.3 | 0.940332  | 1.000000  | 1.000000  | 1.000000  |
| 3.40 | 2.60e+00 |     0.4 | 0.909838  | 1.000000  | 1.000000  | 1.000000  |
| 3.50 | 2.16e+00 |     0.5 | 0.868554  | 1.000000  | 1.000000  | 1.000000  |
| 3.60 | 1.76e+00 |     0.6 | 0.811631  | 1.000000  | 1.000000  | 1.000000  |
| 3.70 | 1.41e+00 |     0.7 | 0.740174  | 1.000000  | 1.000000  | 1.000000  |
| 3.80 | 1.13e+00 |     0.9 | 0.663117  | 1.000000  | 1.000000  | 1.000000  |
| 3.90 | 9.14e-01 |     1.1 | 0.588700  | 1.000000  | 1.000000  | 1.000000  |
| 4.00 | 7.44e-01 |     1.3 | 0.516762  | 1.000000  | 1.000000  | 1.000000  |
| 4.10 | 5.97e-01 |     1.7 | 0.443337  | 1.000000  | 1.000000  | 1.000000  |
| 4.20 | 4.70e-01 |     2.1 | 0.370768  | 0.999999  | 1.000000  | 1.000000  |
| 4.30 | 3.70e-01 |     2.7 | 0.306214  | 0.999995  | 1.000000  | 1.000000  |
| 4.40 | 2.94e-01 |     3.4 | 0.252915  | 0.999971  | 1.000000  | 1.000000  |
| 4.50 | 2.37e-01 |     4.2 | 0.209866  | 0.999860  | 1.000000  | 1.000000  |
| 4.60 | 1.94e-01 |     5.2 | 0.175365  | 0.999490  | 0.999997  | 1.000000  |
| 4.70 | 1.61e-01 |     6.2 | 0.147608  | 0.998503  | 0.999985  | 1.000000  |
| 4.80 | 1.33e-01 |     7.5 | 0.124281  | 0.996191  | 0.999938  | 1.000000  |
| 4.90 | 1.10e-01 |     9.1 | 0.103561  | 0.991051  | 0.999765  | 1.000000  |
| 5.00 | 8.89e-02 |    11.3 | 0.084827  | 0.980197  | 0.999149  | 1.000000  |
| 5.10 | 6.96e-02 |    14.4 | 0.067056  | 0.957027  | 0.996899  | 1.000000  |
| 5.20 | 5.06e-02 |    19.7 | 0.049309  | 0.904710  | 0.987724  | 1.000000  |
| 5.30 | 3.39e-02 |    29.5 | 0.033249  | 0.799882  | 0.953607  | 1.000000  |
| 5.40 | 2.17e-02 |    46.1 | 0.021449  | 0.649946  | 0.869279  | 0.999999  |
| 5.50 | 1.42e-02 |    70.6 | 0.014056  | 0.499859  | 0.742560  | 0.999961  |
| 5.60 | 9.27e-03 |   107.9 | 0.009223  | 0.366706  | 0.593820  | 0.999332  |
| 5.70 | 5.27e-03 |   189.6 | 0.005260  | 0.230155  | 0.404839  | 0.989523  |
| 5.80 | 1.71e-03 |   583.5 | 0.001712  | 0.081916  | 0.156739  | 0.803712  |
=================================================================================
```

CHAPTER

EIGHT

GETTING HELP

8.1 I need help

Join the SourceSpec Discussions and feel free to ask!

8.2 I found a bug

Please open an Issue.

When reporting any bugs, please be sure to attach the *configuration file*.

CHAPTER

NINE

HA3PY API

Ha3Py has a modular structure. Each module corresponds to a specific function or class of functions.

9.1 Ha3Py Programs

9.1.1 The base program (Python version of HA3)

The main module *ha3.py* contains functions, which perform all the operations needed to count the magnitude exceeding probability.

`ha3.main()`

The main function performs all the operations necessary to calculate the seismic hazard. It reads the configuration from a file or creates a configuration by asking the operator, calculates all earthquake recurrent parameters, and calculates, prints, and displays the seismic hazard for the studied region. Results are written to configuration like file.

9.1.2 Maximum magnitude assessment

The maximum magnitude assessment program api description is in the section *The m_max support procedures*.

9.1.3 Configuration (standalone)

The configuration program api description is in the section *Catalogues import and program configuration methods*.

9.1.4 Import

9.1.4.1 Import catalogues to the Ha3Py configuration

`class import_to_hapy.EPISODESCatalog`

The class EPISODESCatalog allows read the EPISODES catalogue. It keeps the catalogue events in the ObsPy format. Therefore, it is possible to operate on the EPISODESCatalog events similar to the ObsPy catalogue.

`import_to_hapy.get_magnitude(`*event*, *magnitude_type=None*`)`

Function get_magnitude extracts the magnitude of the event. If you want to extract a specific magnitude, you can define it as magnitude_type, e.g. *get_magnitude(event, magnitude_type='Mw')*, otherwise, any magnitude will be extracted. If the preferred_magnitude_id of the event is set, it returns the preferred origin. Otherwise, it returns the first magnitude from the list. The function is intended to extract the magnitude unconditionally and non-interactively. Therefore, if preferred_magnitude_id is not set and there are multiple magnitudes, the returned origin may be random.

If event magnitude does not exist, but station_name magnitudes exist, the new magnitude is computed as the mean value of station_name magnitudes.

Parameters

- **event** (*ObsPy Event*) – The event object
- **magnitude_type** (*str*) – Describes the type of magnitude. This is a free text.

Proposed magnitudes are:

- unspecified magnitude ('M') - function search for exactly unspecified magnitude,
- local magnitude ('ML'),
- moment magnitude ('Mw'),
- energy ('Energy'),
- etc.

Returns
The magnitude object or None if the function cannot find or create the magnitude. If only station_name magnitudes exist, the new ObsPy Magnitude object is created, but it is not appended to the event

Return type
ObsPy Magnitude

`import_to_hapy.`**`get_origin`**`(`*event*`)`

Function get_origin extracts the origin from the event. If the preferred_origin_id of the event is set, it returns the preferred origin. Otherwise, it returns the first origin from the list. The function is intended to extract the event origin unconditionally and non-interactively. Therefore, if preferred_origin_id is not set and there are multiple origins, the returned origin may be random

Parameters
event (*ObsPy Event*) – The event object

Returns
The origin object or None if no origin is defined for the event.

Return type
ObsPy Origin

`import_to_hapy.`**`obspy_to_ha3py`**`(`*obspy_catalog*, *magnitude_type='Mw'*`)`

It converts the ObsPy or EPISODESCatalog to the dictionary object accepted by the Ha3Py configuration.

Parameters

- **obspy_catalog** (*ObsPy Catalog or* `EPISODESCatalog`) – The input catalog
- **magnitude_type** (*str*) – The name of magnitude (The default is recommended magnitude Mw)

Returns
The complete catalogue

Return type
dict

`import_to_hapy.`**`read_episodes`**`(`*file_name*`)`

It reads the catalogue Matlab file and create the EPISODES catalog.

Parameters
file_name (*str*)

Returns
The catalog

Return type
EPISODESCatalog

`import_to_hapy.read_file`(*file_name*, *catalog_format*, *configuration*)

Procedure read the catalogue file.

Parameters

- **file_name** (*str*) – The path to the reading catalogue file
- **catalog_format** (*str*) – The catalogue format name. Allowed name is: EPISODES
- **configuration** (*dict*) – The dictionary of all Ha3Py parameters

Returns

`import_to_hapy.remove_catalogs`(*configuration*)

The procedure lists complete catalogues, and allows an operator to select, which one to delete.

Parameters
configuration (*dict*) – The configuration of Ha3Py including catalogues.

9.1.5 Simulation - synthetic catalogues generation

`class synthetic_catalogs.MagnitudeRandomise`(*occurrence_probability*, *magnitude_uncertainty*)

TODO The randomization of magnitudes does not work correctly.

Initialisation parameters:

occurrence_probability: (OccurrenceBase) The object describing the event occurrence probability.

magnitude_uncertainty: (float) The standard deviation of simulated magnitude uncertainty

`synthetic_catalogs.extreme_simulation`(*catalog_configuration*, *occurrence_probability*, *name*)

The function creates the synthetic extreme earthquakes. It simulates the historical or paleo catalogs, where data are incomplete. The time span of te catalogue is divided into random periods. In each period the random number of event with random magnitudes are generated and the maximum magnitude are added to the catalogue. The randomisation depends on the predefined occurrence probability.

Parameters

- **catalog_configuration** (*dict*) – The catalogue definition, part of the dictionary of all Ha3Py parameters.
- **occurrence_probability** (OccurrenceBase) – The object describing the event occurrence probability.
- **name** (*str*) – The name of the created synthetic catalogue.

Returns
The synthetic catalogue

Return type
dict

The simulation catalog configuration dictionary should contain following fields:

time_interval: (float) The time interval, from which the maximum magnitude are put to the synthetic catalogue.

time_uncertainty: (float of uniform probability) The uncertainty of periods time.

magnitude_uncertainty: (float) The uncertainty for magnitude simulation.

sd: (float) The standard deviation of catalog magnitudes. Required for compatibility, when magnitude_uncertainty is not defined. At last magnitude_uncertainty or sd must be defined.

m_min: (str) The minimum completeness magnitude of the simulated catalog. Event with lower magnitude are removed.

begin, end, time_span: The beginning, end, and time span of the catalog. At least two of them are required.

`synthetic_catalogs.`**`full_simulation_incremental`**`(`*catalog_configuration*, *occurrence_probability*, *name*`)`

The incremental catalog generation: it starts with the beginning of the interval time, first the non occurrence time of an event with magnitude greater the m_{min} is generated, then the event time is set, next the magnitude is random generated for that event, and next events times and magnitudes are generated until reaching the end of the interval time.

Parameters

- **`catalog_configuration`** (*`dict`*) – The catalogue definition, part of the dictionary of all Ha3Py parameters.
- **`occurrence_probability`** (`OccurrenceBase`) – The object describing the event occurrence probability.
- **`name`** (*`str`*) – The name of the created synthetic catalogue.

Returns
The synthetic catalogue

Return type
dict

The simulation catalog configuration dictionary should contain following fields:

magnitude_uncertainty: (float) The uncertainty for magnitude simulation.

sd: (float) The standard deviation of catalog magnitudes. Required for compatibility, when magnitude_uncertainty is not defined. At last magnitude_uncertainty or sd must be defined.

m_min: (str) The minimum completeness magnitude of the simulated catalog. Event with lower magnitude are removed.

begin, end, time_span: The beginning, end, and time span of the catalog. At least two of them are required.

`synthetic_catalogs.`**`full_simulation_without_date`**`(`*catalog_configuration*, *occurrence_probability*, *name*`)`

The full_simulation_none_date simulate the catalogue first by generation of random number events in the defined time interval based on the defined n-events occurrence probability and next for each event the magnitude is random generated based on the defined magnitude distribution.

Parameters

- **`catalog_configuration`** (*`dict`*) – The catalogue definition, part of the dictionary of all Ha3Py parameters.
- **`occurrence_probability`** (`OccurrenceBase`) – The object describing the event occurrence probability.
- **`name`** (*`str`*) – The name of the created synthetic catalogue.

Returns
The synthetic catalogue

Return type
dict

The simulation catalog configuration dictionary should contain following fields:

magnitude_uncertainty: (float) The uncertainty for magnitude simulation.

sd: (float) The standard deviation of catalog magnitudes. Required for compatibility, when magnitude_uncertainty is not defined. At last magnitude_uncertainty or sd must be defined.

m_min: (str) The minimum completeness magnitude of the simulated catalog. Event with lower magnitude are removed.

begin, end, time_span: The beginning, end, and time span of the catalog. At least two of them are required.

`synthetic_catalogs.`**`get_times`**`(`*catalog_configuration*`)`

The support function returns begin, end time of the catalogue, and the time span of the catalogue in years. At least two of these values must be defined in the configuration.

Parameters
`catalog_configuration` (*dict*) – The catalog definition, part of the dictionary of all Ha3Py parameters

Returns
The beginning, end, and time span of teh catalog definition as float years

Return type
tuple

`synthetic_catalogs.`**`remain_unchanged`**`(`*catalog*`)`

The procedure remains the catalogue almost unchanged, but finds the minimum magnitude, if not defined, and maximum observed in the catalogue magnitude. If m_min is defined, removes event with magnitude smaller than m_min. The procedure doesn't estimate the catalogue completeness magnitude.

Parameters
`catalog` (*dict*) – The full catalogue description with earthquakes - not only simulation definition.

Returns
The copy of the input catalog with event below m_min removed, maximum observed in the catalogue magnitude, and minimum magnitude in the catalogue.

Return type
tuple

`synthetic_catalogs.`**`synthetic_catalogs_generation`**`(`*configuration*`)`

The procedure generates pseudo paleo, historical, and complete synthetic catalogues based on magnitude occurrence probability definition and catalogs configurations defined in the simulation part of the configuration. *Warning* The older catalogs are replaced by synthetic ones and new minimum and maximum observed magnitudes of all catalogues are set.

Parameters
`configuration` (*dict*) – The dictionary of all Ha3Py parameters. The function doesn't change the input configuration.

Returns
The new configuration with synthetic catalogues

`synthetic_catalogs.`**`test_lambda_beta`**`(`*catalog_configuration*`)`

The function estimated and prints β and λ parameters of the catalogue assessed by the simplest method:

$$\beta = \frac{1}{n}\sum_{i=1}^{n} m_i - m_{min},$$

and

$$\lambda = \frac{n}{T},$$

where T id the catalogue time span, n is the number of events, and m_{min} the minimum magnitude of completeness.

Parameters
`catalog_configuration` (*dict*) – The catalog definition, part of the dictionary of all Ha3Py parameters

Returns
None

`synthetic_catalogs.time_cdf(`*t*, *m*, *occurrence_probability*`)`

The cumulative distribution function of non occurrence event probability given magnitude. If there exists time_cdf method in the event_occurrence object, the result of time_cdf is returned, else

$$F_T(t|m) = \frac{\int_0^t f_M(m,t)dt}{\int_0^\infty f_M(m,t)dt}$$

is returned.

Parameters

- **t** (*float*) – The time
- **m** (*float*) – The magnitude
- **occurrence_probability** (`OccurrenceBase`) – The object describing the event occurrence probability.

Returns
The CDF of the probability of the non occurrence of the event with the magnitude m or greater in the time t.

Return type
float

`synthetic_catalogs.time_rg(`*occurrence_probability*, *m=None*`)`

Next event time random generator. It generates random period of non occurrence of events with magnitude grater than selected. If none magnitude is set, the minimum magnitude of the event occurrence object is taken.

Parameters

- **occurrence_probability** (`OccurrenceBase`) – The object describing the event occurrence probability.
- **m** (*float*) – The magnitude of non occurrence

Returns
The random value of non occurrence period in years

Return type
float

9.2 Functions

Below are definition of all functions used in Ha3Py

9.2.1 Seismic event occurrence and magnitude assessment methods

9.2.1.1 Catalogues import and program configuration methods

Library for creating the configuration file

The procedure realises Bayesian MAP, Max of Posterior Fiducial & Gauss (Kijko, 2004) The algorithm name in the configuration is ‘bayesian_fiducial’.

copyright
Jan Wiszniowski <jwisz@igf.edu.pl>, Andrzej Kijko <andrzej.kijko@up.ac.za>

license
GNU Lesser General Public License, Version 3 (https://www.gnu.org/copyleft/lesser.html)

version 0.0.1
2025-01-01

`configuration.`**`date_years`**(*year*, *month*, *day*)

Coverts date (year, month, day) into years as float value :param year: :param month: :param day: :return: years

`configuration.`**`define_catalogs`**(*configuration*)

Procedure define catalogues fields if not exists, m_min as minimum completeness of catalogues, m_max_obs as maximum magnitude in catalogues, general date begin as the earliest beginning in catalogues, and general date end as the latest end in catalogues

Parameters
`configuration` (*`dict`*) – General configuration container, which is the dictionary of all parameters required for Ha3Py modules and results of all computations.

Returns

Return type

`configuration.`**`init_lambda_beta`**(*configuration*, *init_beta=True*, *init_lambda=True*, *m_max=None*)

Initialize $\beta' and : math : `lambda$ based on complete catalogs. If coefficient ‘beta’ exists, do not change it

Parameters

- **`configuration`**
- **`init_beta`**
- **`init_lambda`**
- **`m_max`**

9.2.1.2 Creating Ha3Py objects or selecting methods based on their names

Ha3Py (c) Jan Wiszniowski, Andrzej Kijko ver. 2024-01

`get_magnitude_distribution.`**`get_magnitude_distribution`**(*configuration*, *m_min=None*, *m_max=None*, *theta=None*)

Function get_magnitude_distribution returns magnitude distribution object based on the parameter ‘magnitude distribution’ in the dictionary of Ha3Py Parameters There are three currently available classes of magnitude distribution objects: ‘Gutenberg-Richter’

$$f_M(m) = \frac{\beta \exp[-\beta(m - m_{min})]}{1 - \exp[-\beta(m_{max} - m_{min}])}$$

$$F_M(m) = \frac{1 - \exp[-\beta(m - m_{min}])}{1 - \exp[-\beta(m_{max} - m_{min}])}$$

‘Compound Gutenberg-Richter’

$$f_M(m) = \overline{\beta} C_\beta \left(\frac{p}{p + m - m_{min}} \right)^{q+1}$$

$$F_M(m) = C_\beta \left[1 - \left(\frac{p}{p + m - m_{min}}\right)^q\right]$$

'Nonparametric Gaussian kernel'

$$f_M\left(m\right) = \frac{\left(h\sqrt{2\pi}\right)^{-1} \sum_{i=1}^{M} \frac{1}{T_i} \exp\left[-0.5\left(\frac{m-m_i}{h}\right)^2\right]}{\sum_{i=1}^{M} \frac{1}{T_i}\left[\Phi\left(\frac{m_{max}-m_i}{h}\right) - \Phi\left(\frac{m_{min}-m_i}{h}\right)\right]}$$

$$F_M\left(m\right) = \frac{\sum_{i=1}^{M} \frac{1}{T_i}\left[\Phi\left(\frac{m-m_i}{h}\right) - \Phi\left(\frac{m_{min}-m_i}{h}\right)\right]}{\sum_{i=1}^{M} \frac{1}{T_i}\left[\Phi\left(\frac{m_{max}-m_i}{h}\right) - \Phi\left(\frac{m_{min}-m_i}{h}\right)\right]}$$

A more detailed description of the magnitude distribution is included in the definition of a specific class

Parameters

- `theta`
- `m_max`
- `m_min`
- `configuration` (*dict*) – The dictionary of all Ha3Py parameters

Returns

the magnitude distribution object

The creating Δ computation objects method is described with the *Della classes* description.

9.2.1.3 Magnitude occurrence probability assessment methods

Main module for probability of exceeding the magnitude estimation

`compute.compute(configuration)`

The procedure estimates the earthquake occurrence probability including magnitude distribution coefficients (e.g. estimates λ and β) and asses maximum event magnitude m_{max} in the investigated area. The assessment method depends on the params. Results are saved to the output file. The existing in the params coefficients are replaced.

The m_{max} is assessed with the cooperation with the user. The procedure estimates and proposes the m_{max} value, which operator must confirm.

Parameters

`configuration` (*dict*) – General configuration container, which is the dictionary of all parameters required for Ha3Py modules and results of all computations.

Returns

None

`compute.compute_occurrence(configuration)`

The function estimates the events occurrence probability coefficients excluding m_max (e.g. λ and β) by finding the logarithm likelihood extremal value.

Parameters

`configuration` (*dict*) – General configuration container, which is the dictionary of all parameters required for Ha3Py modules and results of all computations.

Returns

names of estimated coefficients,

values of estimated coefficients

Return type
(tuple)

Likelihood coefficients estimation

copyright
Jan Wiszniowski <jwisz@igf.edu.pl>, Andrzej Kijko <andrzej.kijko@up.ac.za>

license
GNU Lesser General Public License, Version 3 (https://www.gnu.org/copyleft/lesser.html)

version 0.0.1
2025-01-01

`ln_likelihood.likelihood0`(*event_occurrence*, *catalogue*)

Compute natural logarithm of likelihood for a catalogue with constant completeness level based on the non occurrence event of magnitude m time distribution f_M^{max} with the catalogue completeness magnitude

$$ln\left(\mathcal{L}_{\mathbf{\Theta}}\right) = w_c \sum_{i=1}^{N} w_i \ln\left[f_M^{max}\left(m_i|\mathbf{\Theta}, t_i\right)\right],$$

where N is number of events in the catalogue, m_i is the i-th event magnitude, :param event_occurrence: :type event_occurrence: t_i is the between event time, $\mathbf{\Theta}$ are the probability coefficients, w_c is the weight of the current catalogue likelihood, and w_i is the weight of the i-th event

Parameters
catalogue (*dict*) – The catalogue must contain items:

- 'm_min' (float) the completeness magnitude,
- **'weight' (float, optional, default=1.0) the weight of the catalogue in the total ln likelihood,**
  (w_c)
- **'events' (list) list of events. Each event is a dictionary containing:**
  - 'magnitude' the event magnitude (m_i),
  - 'time_span' time span the events (t_i),
  - **'weight' (float, optional, default=1.0) the weight of the event in the catalogue ln likelihood,**
    (w_i).

Returns
The likelihood of the catalogue

Return type
(float)

Example, the gamma compound Poisson distribution case

$$ln\left(\mathcal{L}_{\lambda}\mathcal{L}_{\beta}\right) = ln\left[\lambda time\left(1+\frac{\lambda time\left(1-F_M\left(m\right)\right)}{q_{\lambda}}\right)^{-(q_{\lambda}+1)}\right] + ln\left(f_M\left(m\right)\right)$$

`ln_likelihood.ln_likelihood`(*x_v*, *configuration*, *m_max=None*)

Parameters

- **x_v** (*numpy.array*) – Array of event distribution and occurrence coefficients. They must agree with the definition of distributions

- **configuration** (*dict*) – General configuration container, which is the dictionary of all parameters required for Ha3Py modules and results of all computations.
- **m_max** (*float*) – Maximum magnitude. If missing the maximum magnitude is taken from configuration

Returns
The natural logarithm of likelihood computed ever all catalogues.

Return type
(float)

9.2.1.4 The m_{max} estimation procedures

These procedures estimate the maximum possible magnitude in the area. They all use m_{max}^{obs}, but some procedures require the earthquake occurrence probability, delta estimation methods, or catalogues.

The m_{max} estimation by solving $m_{max} = m_{max}^{obs} + \Delta$

m_{max} is assessed by the solving the $m_{max} = m_{max}^{obs} + \Delta$ equation by the *SciPy.fsolve* function. The algorithm name in the configuration is 'solve_delta'

copyright
Jan Wiszniowski <jwisz@igf.edu.pl>, Andrzej Kijko <andrzej.kijko@up.ac.za>

license
GNU Lesser General Public License, Version 3 (https://www.gnu.org/copyleft/lesser.html)

version 0.0.1
2025-01-01

`fsolve_delta.`**`m_max_solve_equation`**(*configuration*, *magnitude_distribution=None*, *m_max=None*, *delta=None*)

The function m_max_determination asses maximum regional magnitude by numerical solving the formula (Kijko, 1983, 1985; Pisarenko, 1991; Pisarenko et al., 1996)

$$m_{max} - m_{max}^{obs} + \Delta = 0$$
$$\sigma_{m_{max}} = \sqrt{\sigma^2_{m_{max}^{obs}} + \Delta^2}$$

where :math`Delta` is a class defined outside the function and the object ot the class is one of the Parameters of the call.

Parameters

- **configuration** (*dict*) – General configuration container, which is the dictionary of all parameters required for Ha3Py modules and results of all computations. Required parameters for m_max_solve_equation in the dictionary are (keys are strings):
  - m_max_obs
  - sd_m_max_obs
  - m_min
  - m_max_current
  - time_span
  - theta coefficients, e.q. beta, lambda
  - constant coefficients, e,q, q_beta, q_lambda

- m_max_current: starting m_max for solving the formula
- **magnitude_distribution**
- **m_max**
- **delta**

Returns

Returns
m_max - estimated maximum magnitude sd_m_max - standard deviation of maximum magnitude

Return type
(float, float)

The m_{max} estimation by the iteration

m_{max} is assessed by the solving the $m_{max} = m_{max}^{obs} + \Delta$ equation by the iteration. The algorithm name in the configuration is 'iteration_delta'.

copyright
Jan Wiszniowski <jwisz@igf.edu.pl>, Andrzej Kijko <andrzej.kijko@up.ac.za>

license
GNU Lesser General Public License, Version 3 (https://www.gnu.org/copyleft/lesser.html)

version 0.0.1
2025-01-01

iteration.**m_max_solve_by_iteration**(*configuration*, *magnitude_distribution=None*, *m_max=None*, *delta=None*)

The function m_max_determination asses maximum regional magnitude by the iteration according to the formula (Kijko, 1983, 1985; Pisarenko, 1991; Pisarenko et al., 1996)

$$m_{max}^{(i+1)} = m_{max}^{obs} + \Delta_{m_{max}^{(i)}}$$

where i is the iteration step, $m_{max}^{(0)} = m_{max}^{obs}$. Standard deviation is assumed as

$$\sigma_{m_{max}} = \sqrt{\sigma^2_{m_{max}^{obs}} + \Delta^2}$$

where the delta is a class defined outside the function and the object ot the class is one of the Parameters of the call.

Parameters

- **delta**
- **m_max**
- **magnitude_distribution**
- **delta** – Delta object (e.g. GuRiBaKiSe), which combines the magnitude distribution (e.g. Gutenberg-Richter-Bayes) and delta calculation method (e.g. Kijko-Sellovoll).
- **configuration** (*dict*) – General configuration container, which is the dictionary of all parameters required for Ha3Py modules and results of all computations. Required Parameters in the dictionary are (keys are strings):
  - m_max_obs
  - sd_m_max_obs

- m_min
- m_max_current
- beta
- lambda
- time_span
- q_beta
- q_lamb
- m_max_current: starting M_max in the iteration

Returns

M_max - estimated maximum magnitude Sd_mag_max - standard deviation of maximum magnitude

The m_{max} estimation by momentum

The algorithm name in the configuration is 'momentum'.

copyright

Jan Wiszniowski <jwisz@igf.edu.pl>, Andrzej Kijko <andrzej.kijko@up.ac.za>

license

GNU Lesser General Public License, Version 3 (https://www.gnu.org/copyleft/lesser.html)

version 0.0.1

2025-01-01

`momentum.m_max_by_momentum(configuration, magnitude_distribution=None)`

m_{max} evaluation according to moment estimator (SEE "Estimation of Paramers of a Right Truncated Exponential Distribution" by U.J. Dixit and P.N. Nasiri published in Statistical Papers Vol 49 (2008) pp.225-236)

Parameters

- **configuration** (*dict*) – General configuration container, which is the dictionary of all parameters required for Ha3Py modules and results of all computations.
- **magnitude_distribution** (*MagnitudeDistribution*) – object

Returns

Return type

Function history:

- **Created by P.J. Vermeulen on FEB 2014**
  (Created as additional procedure for the program mmax.m)
- **MAR 2014: Calculation of standard deviation by bootstrap method**
  included. Done by means of subfunction f_boot_var.
- 2025.01.01 code in Python

`momentum.m_max_by_momentum_compute(mag_a)`

Parameters

mag_a

Returns

`momentum.`**`sd_m_max_by_momentum_compute`**`(`*mag_v*`)`

function var_m_max = f_boot_var(mag_v)

Subfunction f_boot_var calculates variance of moment estimator of M_max by means of bootstrap resampling

Parameters

`mag_v` – Vector of seismic event magnitudes

Returns

Estimated variance from bootstrap procedure

The m_{max} assessment by Gibowicz-Kijko procedure

The algorithm name in the configuration is 'Gibowicz-Kijko'.

copyright

Jan Wiszniowski <jwisz@igf.edu.pl>, Andrzej Kijko <andrzej.kijko@up.ac.za>

license

GNU Lesser General Public License, Version 3 (https://www.gnu.org/copyleft/lesser.html)

version 0.0.1

2025-01-01

`gibowicz_kijko.`**`m_max_by_gibowicz_kijko`**`(`*configuration*, *magnitude_distribution=None*, *m_max=None*, *m_min=None*`)`

Gibowicz-Kijko procedure estimates m_{max} by numerically solving the equation

$$F_M\left(m_{max}^{obs}\right) - \frac{n}{n+1} = 0,$$

where n can be assumed as

$$n = \lambda time.$$

Standard deviation is assumed as

$$\sigma_{m_{max}} = \sqrt{\sigma^2_{m_{max}^{obs}} + (m_{max} - m_{max}^{obs})^2}$$

Parameters

- **`m_min`**
- **`configuration`** (*dict*) – General configuration container, which is the dictionary of all parameters required for Ha3Py modules and results of all computations.
- **`m_max`**
- **`magnitude_distribution`**

Returns

estimated maximum magnitude, standard deviation of maximum magnitude.

Return type

(float, float)

m_{max} assessment by Robson-Whitlock and Robson-Whitlock-Cooke procedure

m_{max} is assessed by adding to m_{max}^{obs} the difference between m_{max}^{obs} and the second maximum magnitude.

copyright
Jan Wiszniowski <jwisz@igf.edu.pl>, Andrzej Kijko <andrzej.kijko@up.ac.za>

license
GNU Lesser General Public License, Version 3 (https://www.gnu.org/copyleft/lesser.html)

version 0.0.1
2025-01-01

`robson_whitlock.`**`m_max_by_robson_whitlock`**`(`*configuration*`)`

Robson-Whitlock procedure

$$m_{max} = m_{max}^{obs} + \left(m_{max}^{obs} - m_{n-1}\right)$$
$$\sigma_{m_{max}} = \sqrt{5\sigma_{m_{max}^{obs}}^2 + \left(m_{max}^{obs} - m_{n-1}\right)^2}$$

The algorithm name in the configuration is 'Robson-Whitlock'.

Parameters
`configuration` (*`dict`*) – General configuration container, which is the dictionary of all parameters required for Ha3Py modules and results of all computations.

Returns
m_max - estimated maximum magnitude sd_m_max - standard deviation of maximum magnitude

Return type
(float, float)

`robson_whitlock.`**`m_max_by_robson_whitlock_cooke`**`(`*configuration*`)`

Robson-Whitlock-Cooke procedure

$$m_{max} = m_{max}^{obs} + 0.5\left(m_{max}^{obs} - m_{n-1}\right)$$
$$\sigma_{m_{max}} = \sqrt{1.5\sigma_{m_{max}^{obs}}^2 + 0.25\left(m_{max}^{obs} - m_{n-1}\right)^2}$$

The algorithm name in the configuration is 'Robson-Whitlock-Cooke'.

Parameters
`configuration` (*`dict`*) – General configuration container, which is the dictionary of all parameters required for Ha3Py modules and results of all computations.

Returns
m_max - estimated maximum magnitude sd_m_max - standard deviation of maximum magnitude

Return type
(float, float)

Module bayesian maximum magnitude computation assuming normal distribution

In this method we assume the normal distribution of a prior maximum magnitude $\mathcal{N}\left(m_{max}^{prior}, \sigma_{m_{max}^{prior}}\right)$ and normal distribution of a maximum magnitude estimated based on catalogues $\mathcal{N}\left(m_{max}, \sigma_{m_{max}}\right)$.

copyright
Jan Wiszniowski <jwisz@igf.edu.pl>, Andrzej Kijko <andrzej.kijko@up.ac.za>

license
GNU Lesser General Public License, Version 3 (https://www.gnu.org/copyleft/lesser.html)

version 0.0.1
2025-01-01

`bayesian_normal.`**`get_bayesian_truncnorm`**(*configuration*, *magnitude_distribution=None*)

The m_max_by_bayesian_norm assumes the truncated normal distribution of a prior maximum magnitude $\mathcal{N}\left(m_{max}^{prior}, \sigma_{m_{max}^{prior}}, m_{max}^{L}, m_{max}^{U}\right)$ and normal distribution of a maximum magnitude $\mathcal{N}\left(m_{max}, \sigma_{m_{max}}\right)$ estimated based on catalogues, where m_{max}^{U} is the maximum possible magnitude that we believe might ever happen and m_{max}^{L} is the magnitude that we are sure the maximum magnitude is greater. Coefficients m_{max} and $\sigma_{m_{max}}$ are assesses by one of non bayesian method.

Parameters

- **`configuration`** (*dict*) – General configuration container, which is the dictionary of all parameters required for Ha3Py modules and results of all computations.
- **`magnitude_distribution`**

Returns
The

Return type
tuple(float, float)

`bayesian_normal.`**`m_max_bayesian_norm`**(*configuration*, *magnitude_distribution=None*)

The m_max_by_bayesian_norm assumes the normal distribution of a prior maximum magnitude $\mathcal{N}\left(m_{max}^{prior}, \sigma_{m_{max}^{prior}}\right)$ and normal distribution of a maximum magnitude $\mathcal{N}\left(m_{max}, \sigma_{m_{max}}\right)$ estimated based on catalogues. Coefficients m_{max} and $\sigma_{m_{max}}$ are assesses by one of non bayesian method. The posterior maximum magnitude and its standard deviation are then

$$m_{max}^{posterior} = \frac{\sigma_{m_{max}^{prior}}^{2} m_{max} + \sigma_{m_{max}}^{2} m_{\mathrm{max}}^{prior}}{\sigma_{m_{max}^{prior}}^{2} + \sigma_{m_{max}}^{2}},$$
$$\sigma_{m_{max}^{posterior}} = \frac{\sigma_{m_{max}^{prior}} \sigma_{m_{max}}}{\sqrt{\sigma_{m_{max}^{prior}} + \sigma_{m_{max}}}}$$

Parameters

- **`configuration`** (*dict*) – General configuration container, which is the dictionary of all parameters required for Ha3Py modules and results of all computations.
- **`magnitude_distribution`** (*object*)

Returns
The posterior maximum magnitude and its standard deviation.

Return type
tuple(float, float)

Module Bayesian by shift maximum magnitude computation

class `bayesian_by_shift.`**`BayesianByShift`**(*configuration*, *magnitude_distribution=None*)

The class for estimation of the m_{max} by the Bayesian by shift method. In the case of single catalog, the class define the likelihood function

$$\mathcal{L}\left(\mathbf{m}|m_{max}\right) = \prod_{i=1}^{n} f_{M}\left(m_{i}|m_{max}\right),$$

used in the Bayesian posterior distribution of m_{max}

$$p_{m_{max}}\left(m_{max}|\mathbf{m}\right)=\begin{cases}0 & : m_{max}<m_{\max}^{L}\\ C\cdot\pi\left(m_{max}\right)\mathcal{L}\left(\mathbf{m}|m_{max}\right) & : m_{\max}^{L}\leqslant m_{max}\leqslant m_{\max}^{U}\\ 0 & : m_{max}>m_{\max}^{U}\end{cases},$$

where C is a normalising constant

$$C=1/\int_{m_{max}^{L}}^{m_{max}^{U}}\pi\left(m_{max}\right)L\left(\mathbf{m}|m_{max}\right)dm_{max},$$

and $m_{max}^{L}=\widehat{m}_{max}$. In the case of many catalogs $\mathcal{L}$ is computed by the ln_likelihood function.

Parameters
: **`configuration`** (*`dict`*) – General configuration container, which is the dictionary of all parameters required for Ha3Py modules and results of all computations.

`bayesian_by_shift.`**`get_bayesian_by_shift`**(*configuration*, *magnitude_distribution=None*)

Parameters

- **`configuration`** (*`dict`*) – General configuration container, which is the dictionary of all parameters required for Ha3Py modules and results of all computations.
- **`magnitude_distribution`**

Returns
: The BayesianByShift object

Return type
: BayesianBase derived object

Module bayesian fiducial maximum magnitude computation

class `bayesian_fiducial.`**`BayesianFiducial`**(*configuration*, *magnitude_distribution=None*)

The class that estimate the m_{max} by the Bayesian fiducial method. In the case of single catalog, the class define the likelihood function

$$\mathcal{L}\left(\mathbf{m}|m_{max}\right)=f_{M_{max}}^{FID}\left(m_{max}\right)$$

used in the Bayesian posterior distribution of m_{max}

$$p_{m_{max}}\left(m_{max}|\mathbf{m}\right)=\begin{cases}0 & : m_{max}<m_{\max}^{L}\\ C\cdot\pi\left(m_{max}\right)\mathcal{L}\left(\mathbf{m}|m_{max}\right) & : m_{\max}^{L}\leqslant m_{max}\leqslant m_{\max}^{U}\\ 0 & : m_{max}>m_{\max}^{U}\end{cases},$$

where C is a normalising constant

$$C=1/\int_{m_{max}^{L}}^{m_{max}^{U}}\pi\left(m_{max}\right)L\left(\mathbf{m}|m_{max}\right)dm_{max},$$

and $m_{max}^{L}=m_{max}^{obs}$.

The probability density function is

$$f_{M_{max}}^{FID}\left(m_{max}\right)=n\left[F_{M}\left(m_{max}^{obs}|m_{max}\right)\right]^{n-1}\frac{\partial S_{M}\left(m_{max}^{obs}|m_{max}\right)}{\partial m_{max}}$$

In the case of single catalog cumulate density function is

$$F^{FID}_{M_{max}}(m_{max}) = 1 - \left[F_M\left(m^{obs}_{max}|m_{max}\right)\right]^n$$

In the case of many catalogs

$$F^{FID}_{M_{max}}(m_{max}) = 1 - \prod_i^k \left[F_M\left(m^{obs}_{max}|m_{max}, m^{(i)}_{min}\right)\right]^{n_i},$$

where :math:'k' is the number of catalogues, :math:'m_{min}^{(i)}' is the completeness level in the catalogue, and :math:'n_i' is the number of events in the catalogue.

Parameters
`configuration` (*dict*) – General configuration container, which is the dictionary of all parameters required for Ha3Py modules and results of all computations.

`bayesian_fiducial.`**`get_bayesian_fiducial`**(*configuration*, *magnitude_distribution=None*)

Parameters

- **`configuration`**
- **`magnitude_distribution`** (*object*)

Returns

Return type

Trivial m_{max} assessment

The algorithm name in the configuration is 'primitive'.

The primitive m_{max} is assessed by adding the value 0.5 to m^{obs}_{max}.

copyright
Jan Wiszniowski <jwisz@igf.edu.pl>, Andrzej Kijko <andrzej.kijko@up.ac.za>

license
GNU Lesser General Public License, Version 3 (https://www.gnu.org/copyleft/lesser.html)

version 0.0.1
2025-01-01

`primitive.`**`m_max_primitive`**(*configuration*)

The primitive method assumes the m_{max} is greater than maximum observed magnitude by 0.5.

$$m_{max} = m^{obs}_{max} + 0.5$$
$$\sigma_{m_{max}} = \sigma_{m^{obs}_{max}}$$

Parameters
`configuration` (*dict*) – General configuration container, which is the dictionary of all parameters required for Ha3Py modules and results of all computations.

Returns
m_max - estimated maximum magnitude sd_m_max - standard deviation of maximum magnitude

Return type
(float, float)

9.2.1.5 The m_{max} support procedures

Maximum possible earthquake magnitude estimation

copyright
Jan Wiszniowski <jwisz@igf.edu.pl>, Andrzej Kijko <andrzej.kijko@up.ac.za>

license
GNU Lesser General Public License, Version 3 (https://www.gnu.org/copyleft/lesser.html)

version 0.0.1
2025-01-01

`compute_m_max.`**`m_max_estimation`**(*configuration*)

The estimation of $\hat{m}_{max}$ and $\sigma_{m_{max}}$ by the one of all available methods, both bayesian and non-bayesian. The non-bayesian method is defined in the configuration "m_max_assessment" parameter. The bayesian method is defined in the configuration "bayesian_m_max_assessment" parameter. If the "bayesian_m_max_assessment" parameter is missing or empty, the $\hat{m}_{max}$ is estimated by the non-bayesian method.

Parameters
`configuration` (*`dict`*) – General configuration container, which is the dictionary of all parameters required for Ha3Py modules and results of all computations.

Returns
Estimated by non bayesian method maximum magnitude and standard deviation of the maximum magnitude.

Return type
(float, float)

Non-bayesian Maximum possible earthquake magnitude estimation

copyright
Jan Wiszniowski <jwisz@igf.edu.pl>, Andrzej Kijko <andrzej.kijko@up.ac.za>

license
GNU Lesser General Public License, Version 3 (https://www.gnu.org/copyleft/lesser.html)

version 0.0.1
2025-01-01

`m_max_utils.`**`non_bayesian_m_max_estimation`**(*configuration*, *magnitude_distribution=None*)

The estimation of $\hat{m}_{max}$ and $\sigma_{m_{max}}$ by the non-bayesian method defined in the configuration *m_max_assessment* parameter. The function chooses the method.

Parameters

- **`magnitude_distribution`**
- **`configuration`** (*`dict`*) – General configuration container, which is the dictionary of all parameters required for Ha3Py modules and results of all computations.

Returns
Estimated by non bayesian method maximum magnitude and standard deviation of the maximum magnitude.

Return type
(float, float)

9.2.1.6 Uncertainty assessment methods

9.2.1.7 Correction procedures

`corrections.`**`lambda_correction`**`(`*configuration*`)`

Required parameters:
paleo_catalog historic_catalog complete_catalogs

9.2.1.8 Result visualization

Return period calculation

copyright
Jan Wiszniowski <jwisz@igf.edu.pl>, Andrzej Kijko <andrzej.kijko@up.ac.za>

license
GNU Lesser General Public License, Version 3 (https://www.gnu.org/copyleft/lesser.html)

version 0.0.1
2025-01-01

`return_period.`**`grad_return_period`**`(`*m*, *lamb*, *mag_distr*`)`

Procedure calculates the returns gradient of return period parameters

Example

Gradient of return period for $\lambda\left(m\right)=\lambda_0\left(1-F_M\left(m|m_0\right)\right)$:

$$\frac{\partial T_R\left(m\right)}{\partial\lambda_0}=\frac{\partial\left(\frac{1}{\lambda_0\cdot\left(1-F_M\left(m|m_0\right)\right)}\right)}{\partial\lambda_0}=\frac{-1}{\lambda_0^2\cdot\left(1-F_M\left(m|m_0\right)\right)}$$

Gradients of return period for typical parameters of a magnitude distribution.

β:

$$\frac{\partial T_R\left(m\right)}{\partial\beta}=\frac{\partial\left(\frac{1}{\lambda_0\cdot\left(1-F_M\left(m|m_0\right)\right)}\right)}{\partial\beta}=\frac{-1}{\lambda\cdot\left(1-F_M\left(m|m_0\right)\right)^2}\frac{\partial\left(1-F_M\left(m|m_0\right)\right)}{\partial\beta}$$

m_{max}:

$$\frac{\partial T_R\left(m\right)}{\partial m_{max}}=\frac{\partial\left(\frac{1}{\lambda_0\cdot\left(1-F_M\left(m|m_0\right)\right)}\right)}{\partial m_{max}}=\frac{-1}{\lambda\cdot\left(1-F_M\left(m|m_0\right)\right)^2}\frac{\partial\left(1-F_M\left(m|m_0\right)\right)}{\partial m_{max}}$$

Parameters

- **`m`** – magnitude for the return period
- **`lamb`** – lambda for the minimum magnitude
- **`mag_distr`** – the magnitude distribution

Returns
gradient of the return priod versus lambad and magnitude distribution paramters

`return_period.`**`return_period`**`(`*m*, *lamb*, *mag_dist*`)`

Procedure calculates the return period as the inverse of the annual probability of not occurrence, which is the survive function magnitude not occurrence distribution.

$$T_R\left(m,\lambda_0,F_M\right)=\frac{1}{S_M^{max}\left(m\right)}$$

Example

Let

$$\lambda(m) = \lambda_0 (1 - F_M(m|m_0)),$$

then return period is

$$T_R(m, \lambda_0, F_M) = \frac{1}{\lambda(m)}$$

Parameters

- **m** – magnitude for the return period
- **lamb** – lambda for the minimum magnitude
- **mag_dist** – the magnitude distribution

Returns
the return period

Ha3Py (c) Andrzej Kijko, Jan Wiszniowski ver. 2024-01

Plotting results procedures

copyright
Jan Wiszniowski <jwisz@igf.edu.pl>, Andrzej Kijko <andrzej.kijko@up.ac.za>

license
GNU Lesser General Public License, Version 3 (https://www.gnu.org/copyleft/lesser.html)

version 0.0.1
2025-01-01

The module contain procedure for plotting seismic hazard diagrams resulted from estimation of earthquake hazard parameters by Ha3Py. Most functions plot diagrams curves with uncertainty margins.

`plot_results.`**`comp_mean_return_period`**(*configuration*)

Procedure compute the mean return period

Parameters
configuration (*dict*) – General configuration container, which is the dictionary of all parameters required for Ha3Py modules and results of all computations.

Returns

Return type

9.3 Classes

Below are definition of all classes used in Ha3Py

9.3.1 Magnitude distribution

The magnitude distribution define the probability of exceeding the magnitude. The magnitude distribution realises an object-oriented approach to this issue. All algorithms work on an abstract magnitude probability class *MagnitudeDistribution*. It allows the assessment of various magnitude distribution, which depend on various parameters.

9.3.1.1 Base magnitude distribution class

BaseDelta class of magnitude distributions

copyright
Jan Wiszniowski <jwisz@igf.edu.pl>, Andrzej Kijko <andrzej.kijko@up.ac.za>

license
GNU Lesser General Public License, Version 3 (https://www.gnu.org/copyleft/lesser.html)

version 0.0.1
2025-01-01

`class MagnitudeDistribution.BaseMagnitudeDistribution(parameters, name, long_name=None, m_min=None, m_max=None)`

BaseDelta magnitude distribution class manages m_{min} and m_{max}. The base magnitude distribution is a derived class from the SciPy generic continuous random variable class.

Parameters

- **parameters** – dictionary of Ha3Py params params Required items in the params dictionary are unless they are define in the constructor:
  - m_min
  - m_max_current
  - m_max (required if m_max_current is missing in the params dictionary)
- **name** – magnitude distribution prompt
- **long_name** – magnitude distribution long prompt
- **m_min** – minimum value of the magnitude distribution
- **m_max** – maximum value of the magnitude distribution

Classes derived from the MagnitudeDistribution classes define exact magnitude distribution, e.g. Gutenberg-Richter magnitude distribution. They must define methods:

Required params if they are not define in the contractor:

- m_min
- m_max_current
- m_max (required if 'm_max_current' is missing in the params dictionary)
- _prepare - preparation of probability distribution computation
- _parameters - return the list of all probability distribution Parameters
- **_grad_sf - return the survive function gradients of all probability distribution Parameters**
  (see grad_sf)

`property coefficient_names`
They are magnitude distribution Parameters

`property coefficients`
They are magnitude distribution Parameters

`property const_coefficients`
They are magnitude distribution Parameters

grad_sf(*m*, *coefficient_name=None*)

Compute gradients of magnitude distribution survive function Parameters

$$\frac{\partial S_M(m)}{\partial x_i}, i = 1, \ldots$$

where a survive function $S_M(m) = 1 - F_M(m)$ and $x_i, i = 1, \ldots$ are the magnitude distribution Parameters .

Parameters

- **coefficient_name**
- **m** – magnitude distribution of the survive function

Returns

dictionary of magnitude distribution Parameters names and their gradients. The magnitude distribution Parameters depend on the magnitude distribution

Users can use one from three predefined magnitude distribution classes:

- classic Gutenberg-Richter magnitude distribution,
- compound Gutenberg-Richter magnitude distribution,
- non-parametric magnitude distribution.

The non-parametric magnitude distribution is sensitive to the completeness of the magnitudes. Therefore, applying this to complete and extreme catalogues should be done with caution, as additional conditions must be fulfill, e.g., there must be no gaps in the magnitude ranges.

9.3.1.2 Predefined magnitude distribution classes

Gutenberg-Richter magnitude distribution

copyright

Jan Wiszniowski <jwisz@igf.edu.pl>, Andrzej Kijko <andrzej.kijko@up.ac.za>

license

GNU Lesser General Public License, Version 3 (https://www.gnu.org/copyleft/lesser.html)

version 0.0.1

2025-01-01

class GutenbergRichter.**GutenbergRichter**(*parameters*, *beta=None*, *m_min=None*, *m_max=None*)

The continuous random variable.

The Gutenberg-Richter probability density function is:

$$f_M(m) = \begin{cases} 0 \\ : \text{for } m < m_{min} \\ \frac{\beta \exp[-\beta(m-m_{min})]}{1-\exp[-\beta(m_{max}-m_{min})]} \\ : \text{for } m_{min} \leqslant m \leqslant m_{max} \\ 0 \\ : \text{for } m > m_{max} \end{cases}$$

The Gutenberg-Richter cumulate density function is

$$F_M(m) = \begin{cases} 0 & : \text{for } m < m_{min} \\ \dfrac{1 - \exp\left[-\beta\left(m - m_{min}\right)\right]}{1 - \exp\left[-\beta\left(m_{max} - m_{min}\right)\right]} & : \text{for } m_{min} \leqslant m \leqslant m_{max} \\ 1 & : \text{for } m > m_{max} \end{cases}$$

The Gutenberg-Richter gradients for m_{max} and β are:

$$\frac{\partial S_M(m)}{\partial m_{max}} = \frac{N\beta \exp\left[-\beta\left(m_{max}-m_{min}\right)\right]}{D^2}$$

and

$$\frac{\partial S_M(m)}{\partial \beta} = \frac{N\left(m_{max}-m_{min}\right)\exp\left[-\beta\left(m_{max}-m_{min}\right)\right]-D\left(m-m_{min}\right)\exp\left[-\beta\left(m-m_{min}\right)\right]}{D^2}$$

where

$$D = 1-\exp\left[-\beta\left(m_{max}-m_{min}\right)\right]$$

end

$$N = 1-\exp\left[-\beta\left(m-m_{min}\right)\right]$$

Module Compound Gutenberg-Richter magnitude distribution

class `CompoundGutenbergRichter.`**`CompoundGutenbergRichter`**(*parameters*, *beta=None*, *q_beta=None*, *m_min=None*, *m_max=None*)

The continuous random variable.

The compound Gutenberg-Richter-Bayes probability density function is:

$$f_M(m) = \begin{cases} 0 & : \text{for } m < m_{min} \\ \overline{\beta} C_\beta \left[\dfrac{q_\beta}{q_\beta+\overline{\beta}\left(m-m_{min}\right)}\right]^{q_\beta+1} & : \text{for } m_{min} \leqslant m \leqslant m_{max} \\ 0 & : \text{for } m > m_{max} \end{cases}$$

where:

$$C_\beta = \frac{1}{1-\left[\frac{q_\beta}{q_\beta+\overline{\beta}\left(m_{max}-m_{min}\right)}\right]^{q_\beta}}$$

and $q_\beta = \left(\overline{\beta}/\sigma_\beta\right)^2 = \text{constant}$.

The compound Gutenberg-Richter cumulate density function is:

$$F_M(m) = \begin{cases} 0 & : \text{for } m < m_{min} \\ C_\beta \left[1-\left(\dfrac{q_\beta}{q_\beta+\overline{\beta}\left(m-m_{min}\right)}\right)^{q_\beta}\right] & : \text{for } m_{min} \leqslant m \leqslant m_{max} \\ 1 & : \text{for } m > m_{max} \end{cases}$$

Gradient of the survive function for m_{max} is:

$$\frac{\partial(1-F_M(m))}{\partial m_{max}} = -\frac{N}{D^2}\overline{\beta}\left(\frac{q_\beta}{q_\beta+\overline{\beta}\left(m_{max}-m_{min}\right)}\right)^{q_\beta+1}$$

Gradient of the survive function for β is

$$\frac{\partial(1-F_M(m))}{\partial\overline{\beta}} = \frac{N\left(m_{max}-m_{min}\right)\left(\frac{q_\beta}{q_\beta+\overline{\beta}\left(m_{max}-m_{min}\right)}\right)^{q_\beta+1}-D\left(m-m_{min}\right)\left(\frac{q_\beta}{q_\beta+\overline{\beta}\left(m-m_{min}\right)}\right)^{q_\beta+1}}{D^2}$$

Non-parametric magnitude distribution

copyright
Jan Wiszniowski <jwisz@igf.edu.pl>, Andrzej Kijko <andrzej.kijko@up.ac.za>

license
GNU Lesser General Public License, Version 3 (https://www.gnu.org/copyleft/lesser.html)

version 0.0.1
2025-01-01

class `NonParametricGaussian.`**`NonparametricGaussianKernel`**(*configuration*, *no_largest=None*, *m_min=None*, *m_max=None*)

The continuous random variable.

The non-parametric magnitude distribution with Gaussian kernel probability density function is:

$$f_M(m) = \begin{cases} 0 & : \text{for } m < m_{min} \\ \dfrac{\left(h\sqrt{2\pi}\right)^{-1} \sum_{i=1}^{M} \frac{1}{T_i} \exp\left[-0.5\left(\frac{m-m_i}{h}\right)^2\right]}{\sum_{i=1}^{M} \frac{1}{T_i}\left[\Phi\left(\frac{m_{max}-m_i}{h}\right) - \Phi\left(\frac{m_{min}-m_i}{h}\right)\right]} & : \text{for } m_{min} \leqslant m \leqslant m_{max} \\ 0 & : \text{for } m > m_{max} \end{cases}$$

where h is the smoothing parameter and T_i is the sum of time periods of catalogues, where $m_i \geq m_x$. Values $1/T_i$ can be replaced by predefined weights if they are defined in catalogs. The smoothing parameter h is estimated by the smooth_parameter_estimation function unless it is predefined.

The non-parametric magnitude distribution with Gaussian kernel cumulate density functions:

$$F_M(m) = \begin{cases} 0 & : \text{for } m < m_{min} \\ \dfrac{\sum_{i=1}^{M} \frac{1}{T_i}\left[\Phi\left(\frac{m-m_i}{h}\right) - \Phi\left(\frac{m_{min}-m_i}{h}\right)\right]}{\sum_{i=1}^{M} \frac{1}{T_i}\left[\Phi\left(\frac{m_{max}-m_i}{h}\right) - \Phi\left(\frac{m_{min}-m_i}{h}\right)\right]} & : \text{for } m_{min} \leqslant m \leqslant m_{max} \\ 1 & : \text{for } m > m_{max} \end{cases}$$

where $\Phi(m)$ is the normal cdf.

The only variable parameter is m_{max}. The survive function gradient for m_{max} is

$$\frac{\partial S_M(m)}{\partial m_{max}} = \frac{F_M(m) f_M\left(m_{max}\right)}{\sum_{i=1}^{M} \frac{1}{T_i}\left[\Phi\left(\frac{m_{max}-m_i}{h}\right) - \Phi\left(\frac{m_{min}-m_i}{h}\right)\right]}$$

class `NonParametricGaussian.`**`NonparametricPseudoGaussianKernel`**(*configuration*, *no_largest=None*, *m_min=None*, *m_max=None*)

The continuous random variable.

The non-parametric magnitude distribution with pseudo Gaussian kernel probability density function is:

$$f_M(m) = \begin{cases} 0 & : \text{for } m < m_{min} \\ \dfrac{\sum_{i=1}^{M} \frac{1}{T_i} \psi\left(\frac{m-m_i}{h}\right)}{\sum_{i=1}^{M} \frac{1}{T_i}\left[\Psi\left(\frac{m_{max}-m_i}{h}\right) - \Psi\left(\frac{m_{min}-m_i}{h}\right)\right]} & : \text{for } m_{min} \leqslant m \leqslant m_{max} \\ 0 & : \text{for } m > m_{max} \end{cases}$$

where

$$\psi(x) = 2\frac{(a_1 + 2a_2x + 3a_3x^2 + 4a_4x^3)}{\sigma(1 + a_1x + a_2x^2 + a_3x^3 + a_4x^4)^5},$$
$$\Psi(x) = 1.0 - 0.5(1 + a_1x + a_2x^2 + a_3x^3 + a_4x^4)^{-4}$$
$$a_1 = 0.196854, a_2 = 0.115194, a_3 = 0.000344, a_4 = 0.019527,$$

h is the smoothing parameter and T_i is the sum of time periods of catalogues, where $m_i \geq m_x$. Values $1/T_i$ can be replaced by predefined weights if they are defined in catalogs. The smoothing parameter h is estimated by the smooth_parameter_estimation function unless it is predefined.

The non-parametric magnitude distribution with Gaussian kernel cumulate density functions:

$$F_M(m) = \begin{cases} 0 & : \text{for } m < m_{min} \\ \dfrac{\sum_{i=1}^{M} \frac{1}{T_i} \left[\Psi\left(\frac{m-m_i}{h}\right) - \Psi\left(\frac{m_{min}-m_i}{h}\right) \right]}{\sum_{i=1}^{M} \frac{1}{T_i} \left[\Psi\left(\frac{m_{max}-m_i}{h}\right) - \Psi\left(\frac{m_{min}-m_i}{h}\right) \right]} & : \text{for } m_{min} \leqslant m \leqslant m_{max} \\ 1 & : \text{for } m > m_{max} \end{cases}$$

The only variable parameter is m_{max}. The survive function gradient for m_{max} is

$$\frac{\partial S_M(m)}{\partial m_{max}} = \frac{F_M(m) f_M\left(m_{max}\right)}{\sum_{i=1}^{M} \frac{1}{T_i} \left[\Psi\left(\frac{m_{max}-m_i}{h}\right) - \Psi\left(\frac{m_{min}-m_i}{h}\right) \right]}$$

`NonParametricGaussian.`**`smooth_parameter_estimation`**(*earthquakes*)

Calculation of optimal smoothing parameter h in the gaussian kernel function.

Criterion of h selection is *Adaptive Estimate of Spread*.

$$h = 0.9 * \min\left[sd, \frac{Q\left(p = 0.75\right) - Q\left(p = 0.25\right)}{1.34} \right] n^{-1/5},$$

where sd is the standard deviation of magnitude values, $Q\left(p = k\right)$ is k quantile, and n is the number of considered events

For more details, see Silverman (1986), pp. 48.

Parameters

earthquakes (*list(dict)*) – column vector of earthquakes dictionaries

Returns

optimal value of smoothing parameter in the non-parametric (Gaussian kernel) function

Return type

(float)

9.3.2 Seismic event occurrence probability

The seismic event occurrence probability define the probability of occurrence of N events in the defined time duration. All algorithms work on an abstract event occurrence probability class *BaseNEventsOccurrenceProb*. It allows the assessment of various event occurrence probability, which depend on various parameters.

The special abstract subclass of the *BaseNEventsOccurrenceProb* is the *LambdaNEventsOccurrence*, which should be the base class of all event occurrence probability classes that have only one λ variable parameter.

9.3.3 Base event occurrence probability classes

9.3.3.1 Base classes of events occurrence probabilities

`class BaseOccurrence.`**`OccurrenceBase`**(*configuration*, *name*, ***kwargs*)

The base class of event occurrence. It defines two probabilities:

- The discrete probability $p_n(n|t)$ of occurrence of n event equal or greater than the minimum magnitude in the period t

  methods: d_pmf, d_cdf, ln_d_pmf, d_grad_sf,

- The continues probability that in the period $time$ magnitude of none event exceed the value m

  methods: pdf, cdf, grad_sf, and other methods of the SciPy continues probability (rv_continuous).

  Parameters

  - **configuration** (*dict*) – General configuration container, which is the dictionary of all parameters required for Ha3Py modules and results of all computations.
  - **name** (*str*) – The name of the derived class object
  - **kwargs** (*dict*) – Optional parameters are: magnitude_distribution (magnitude distribution object), m_min (float), m_max (float), theta (list or np.array). They are used for the object creation. If they are missing, appropriate objects are created based on the configuration

Methods:

***_cdf*:**
Local definition of the cumulative distribution function $F_M^{max}\left(m|t,\mathbf{\Theta}\right)$ of continues probability that in the period t magnitude of none event exceed the value m, where coefficients $\mathbf{\Theta}$, which depend on the specific realisation of the distribution, must be defined before the method call. E.g, If you wanted to compute occurrence for magnitudes m or greater, you must set $m_{min} = m$ earlier.

It modifies the *rv_continuous.pdf* and others rv_continuous's methods that requre the pdf. Derived classes shouldn't overwrite this method but rather define the abstract *_l_cdf*.

***_pdf*:**
Local definition of the probability distribution function $f_M^{max}\left(m|t,\mathbf{\Theta}\right)$ of continues probability that in the period t magnitude of none event exceed the value m, where coefficients $\mathbf{\Theta}$, which depend on the specific realisation of the distribution, must be defined before the method call. E.g, If you wanted to compute occurrence for magnitudes m or greater, you must set $m_{min} = m$ earlier. It modifies the *rv_continuous.pdf* and others rv_continuous's methods that requre the pdf. Derived classes shouldn't overwrite this method but rather define the abstract *_l_pdf*.

***d_cdf*:**
Definition of the cumulative distribution function $F_n\left(n|t,\mathbf{\Theta}\right)$ of discrete probability of occurrence of n events equal or greater than the minimum magnitude in the period t, where coefficients $\mathbf{\Theta}$, which depend on the specific realisation of the distribution, must be defined before the method call. E.g, If you wanted to compute occurrence for magnitudes m or greater, you must set $m_{min} = m$ earlier.

Derived classes shouldn't overwrite this method but rather define the abstract *_d_cdf*.

***d_pmf*:**
Definition of the discrete probability mass function $p_n(n|t,\mathbf{\Theta})$ of occurrence of n events equal or greater than the minimum magnitude in the period t where coefficients $\mathbf{\Theta}$, which depend on the specific realisation of the distribution, must be defined before the method call. E.g, If you wanted to compute occurrence for magnitudes m or greater, you must set $m_{min} = m$ earlier. Derived classes shouldn't overwrite this method but rather define the abstract *_d_cdf*.

***d_expected*:**
Definition of the expected value of discrete probability $p_n(n|t)$ of occurrence of n event in the period t:

***ln_d_pmf*:**
The function compute the natural probability logarithm $\ln\left[p_n(n|t)\right]$ of occurrence of n events in the period t probability mass function.

***grad_sf*:**

Compute gradients of non occurrence of event in the time survive function (see *_cdf*, *_pdf*).

$$\frac{\partial S_M^{max}\left(m|t\right)}{\partial\theta_{\lambda i}}$$

and

$$\frac{\partial S_M^{max}\left(m|t\right)}{\partial\theta_{\beta i}}=\frac{\partial S_M^{max}\left(m|t\right)}{\partial S_M\left(m|\mathbf{\Theta}_\beta\right)}\frac{\partial S_M\left(m|\mathbf{\Theta}_\beta\right)}{\partial\theta_{\beta i}},$$

where $S_M^{max}\left(m|t\right)=1-F_M^{max}\left(m|t\right)$, $\theta_{\beta i}\in\Theta_\beta$ are magnitude distribution coefficients, and $\theta_{\lambda i}\in\Theta_\lambda$ are the occurrence probability own coefficients.

***d_grad_sf*:**
Compute gradients of the discrete probability that, in the time t, n appears with magnitude m or greater survive function (see *d_cdf*, *d_pmf*).

$$\frac{\partial S_n\left(n|t\right)}{\partial\theta_{\lambda i}}$$

and

$$\frac{\partial S_n\left(n|t\right)}{\partial\theta_{\beta i}}=\frac{\partial S_n\left(n|t\right)}{\partial S_M\left(m|\mathbf{\Theta}_\beta\right)}\frac{\partial S_M\left(m|\mathbf{\Theta}_\beta\right)}{\partial\theta_{\beta i}},$$

where $S_n\left(n|t\right)=1-F_n^{max}\left(n|t\right)$, $\theta_{\beta i}\in\Theta_\beta$ are magnitude distribution coefficients, and $\theta_{\lambda i}\in\Theta_\lambda$ are the occurrence probability own coefficients.

Properties:

***coefficients*: (read only)**
Returns the list of all the occurrence probability estimated coefficients values $\mathbf{\Theta}$. Both the continues probability $f_M^{max}\left(m|t,\mathbf{\Theta}\right)$ that in the period t magnitude of none event exceed the value m and the discrete probability $p_n(n|t,\mathbf{\Theta})$ of occurrence of n event in the period t have the same coefficients. First in the list are own occurrence probability coefficients Θ_λ, next are magnitude distribution coefficients Θ_β. The list ends with the maximum magnitude

***coefficient_names*: (read only)**
Returns the list of all the occurrence probability estimated coefficients names. Both the continues probability $f_M^{max}\left(m|t,\mathbf{\Theta}\right)$ that in the period t magnitude of none event exceed the value m and the discrete probability $p_n(n|t,\mathbf{\Theta})$ of occurrence of n event in the period t have the same coefficients. First in the list are own occurrence probability coefficients, next are magnitude distribution coefficients. The list ends with the maximum magnitude

***const_coefficients*: (read only)**
Returns the names of all the occurrence probability constant coefficients. These coefficients must be defined (some values can be default) but are non-estimated

***m_min*:**
The substitution sets the minimum magnitude m_{min} of occurrence probability and modifies the probability.

***m_max*:**
The substitution sets the maximum magnitude m_{max} of occurrence probability and modifies the probability.

Required definition of abstract methods:

All the following methods must be defined in derived classes

***_l_cdf*:**
Returns in the derived class the cumulative distribution function $F_M^{max}\left(m|t\right)$ of continues probability that in the period t magnitude of none event exceed the value m

***_l_pdf*:**
Returns in the derived class the probability distribution function $f_M^{max}(m|t)$ of continues probability that in the period t magnitude of none event exceed the value m,

***_d_cdf*:**
Returns in the derived class the cumulative distribution function $F_n(n|t)$ of discrete probability of occurrence of n event equal or greater than the minimum magnitude in the period t.

***_d_pmf*:**
Returns in the derived class the discrete probability $p_n(n|t)$ of occurrence of n event equal or greater than the minimum magnitude in the period t mass function

***_coefficient_names*:**
Returns the list of own occurrence probability estimated coefficients $\mathbf{\Theta}_\lambda$ names excluding the magnitude distribution coefficients $\mathbf{\Theta}_\beta$.

***_coefficient_values*:**
Returns the list of own occurrence probability estimated coefficients $\mathbf{\Theta}_\lambda`values excluding the values of magnitude distribution coefficients$ : $math$ :mathbf{Theta_{beta}}`.

***_const_coefficients*:**
Returns the names of own occurrence probability constant (non-estimated) coefficients excluding the names of constant magnitude distribution coefficients.

***_grad_sf_magnitude_distribution(m, t)*:**
Returns the gradient of the of non occurrence of event in the time probability survive function with respect to magnitude distribution survive function:

$$\frac{\partial S_M^{max}(m|t)}{\partial S_M(m|\mathbf{\Theta}_\beta)}$$

***_d_grad_sf_magnitude_distribution(n, m, t)*:**
Return the gradient of the discrete probability that, in the time interval t, n appears with magnitude m or greater survive function with respect to magnitude distribution survive function:

$$\frac{\partial S_n(n|m,t)}{\partial S_n(m|\mathbf{\Theta}_\beta)}$$

***_grad_sf(m, t)*:**
Return gradients of the non occurrence of event in the time survive function with respect to the occurrence probability own coefficients

$$\frac{\partial S_M^{max}(n|t)}{\partial \theta}$$

***_d_grad_sf(n, t)*:**
Return gradients of the discrete probability that, in the time t, n appears with magnitude m or greater survive function with respect to the occurrence probability own coefficients

$$\frac{\partial S_n(n|t)}{\partial \theta}$$

d_mean:

property coefficient_names

The coefficient_names function gets the names of variable event occurrence parameters. The names correspond to the coefficient values of the coefficients function. They are given in order: first are own event occurrence parameters names e.g. '*lambda*', next are magnitude distribution parameters names e.g. '*beta*', and at the end is maximum value parameters '*m_max*'.

Returns
The list of current event occurrence parameters names

Return type
list(str)

property coefficients

The coefficients function gets the values of current event occurrence parameters. They are given in order: first are own event occurrence parameters e.g. λ, next are magnitude distribution parameters e.g. β, and at the end is maximum value parameters m_{max}.

Returns
The list of current event occurrence parameters

Return type
list(float)

property const_coefficients

The const_coefficients gets the values of constant (not estimated) event occurrence parameters

Returns
The list of constant event occurrence parameters

Return type
list(float)

d_cdf(*n*, **args*)

The function compute the cumulate distribution function $F_n(n|t)$ of occurrence of n event in the period t. If you wanted to compute occurrence for magnitudes m or greater, you must set $m_{min} = m$ earlier.

Parameters

- **n** (*int*) – Number of events
- **args** (*list*) – Optional period value. If no argument missing, one year is assumed.

Returns
The CDF for n events

Return type
float

d_expected(*t*)

The function compute the expected value of occurrence of n events in the period $time$ having magnitudes m or greater. It is realised by :math:`` :param t: :return:

d_grad_sf(*n*, *m*, *t*)

Compute gradients of the probability that, in the time interval $time$, n appears with magnitude m or greater.

$$\frac{\partial S_n\left(n|m, time\right)}{\partial \theta_i}, i = 1, ...$$

where a survive function $S_M\left(m\right) = 1 - F_M\left(m\right)$ and $x_i, i = 1, ...$ are the magnitude distribution Parameters .

Parameters

- **n** (*int*) – The number of events
- **t** (*float*) – The time event does not exceed the magnitude m
- **m** (*float*) – The magnitude that will be not exceeded in the time $time$

Returns
The dictionary of variable events occurrence parameters names and their gradients. The events occurrence parameters depend on the events occurrence and magnitude distribution objects

Return type
dict

d_mean(**args*)

Returns
The

Return type
float

d_pmf(*n*, **args*)

The function compute the probability mass function $p_n(n|t)$ of occurrence of n events in the period t. If you wanted to compute occurrence for magnitudes m or greater, you must set $m_{min} = m$ earlier.

Parameters

- **n** (*int*) – Number of events
- **args** (*list*) – Optional period value. If no argument missing, one year is assumed.

Returns
The probability of n event occurrence in the period $time$

Return type
float

grad_sf(*m*, *t*)

Compute gradients of event non occurrence time survive function for all event non occurrence time probability coefficients including magnitude distribution coefficients and maxim magnitude

$$\frac{\partial S_M\left(time|m\right)}{\partial\theta_i}, i = 1, ...$$

where a survive function $S_M^{max}\left(time|m\right) = 1 - F_M^{max}\left(time|m\right)$ and $\theta_i, i = 1, ...$ are the magnitude non occurrence time probability coefficients.

Parameters

- **t** (*float*) – The time event does not exceed the magnitude m
- **m** (*float*) – The magnitude that will be not exceeded in the time $time$

Returns
The dictionary of variable events occurrence parameters names and their gradients. The events occurrence parameters depend on the events occurrence and magnitude distribution objects

Return type
dict

ln_d_pmf(*n*, **args*)

The function compute the natural probability logarithm $\ln\left[p_n(n|t)\right]$ of occurrence of n events in the period t probability mass function. If you wanted to compute occurrence for magnitudes m or greater, you must set $m_{min} = m$ earlier.

Parameters

- **n**
- **args**

Returns

Ha3Py (c) Jan Wiszniowski, Andrzej Kijko ver. 2024-01

`class LambdaOccurrence.`**`LambdaOccurrence`**(*configuration*, *name*, ***kwargs*)

Users can use one from two event occurrence probability classes: * classic Poisson probability, * gamma compound Poisson probability.

9.3.4 Predefined events event occurrence probability classes

9.3.4.1 Poisson events occurrence probability

`class PoissonOccurrence.`**`PoissonOccurrence`**(*configuration*, ***kwargs*)

The PDF of Poisson distribution of not exceeding magnitude m in time t is

$$f_M^{max}\left(m|\lambda,\mathbf{\Theta}_\beta,t\right) = \lambda t f_m\left(m|\mathbf{\Theta}_\beta\right)\exp\left(\lambda t S_M\left(m|\mathbf{\Theta}_\beta\right)\right).$$

The CDF of Poisson distribution of not exceeding magnitude m in time t is

$$F_M^{max}\left(m|\lambda,\mathbf{\Theta}_\beta,t\right) = 1 - e^{-\lambda t S_M(m|\mathbf{\Theta}_\beta)}.$$

Gradients of the of non occurrence of event in the time probability survive function are:

$$\frac{\partial S_M^{max}\left(m|t\right)}{\partial\lambda} = \exp\left(-\lambda t\left\{S_M\left(m|\mathbf{\Theta}_\beta\right)\right\}\right) t\left\{S_M\left(m|\mathbf{\Theta}_\beta\right)\right\},$$
$$\frac{\partial S_M^{max}\left(m|t\right)}{\partial\theta_i} = \lambda t\exp\left(-\lambda t\left\{S_M\left(m|\mathbf{\Theta}_\beta\right)\right\}\right)\left\{\frac{\partial S_M\left(m\right)}{\partial\theta_i}\right\}.$$

ln_d_pmf(*n*, **args*)

The function compute the natural probability logarithm $\ln\left[p_n(n|t)\right]$ of occurrence of n events in the period t probability mass function. If you wanted to compute occurrence for magnitudes m or greater, you must set $m_{min} = m$ earlier.

Parameters

- **n**
- **args**

Returns

9.3.4.2 Gamma compound Poisson events occurrence probability

copyright

Jan Wiszniowski <jwisz@igf.edu.pl>, Andrzej Kijko <andrzej.kijko@up.ac.za>

license

GNU Lesser General Public License, Version 3 (https://www.gnu.org/copyleft/lesser.html)

version 0.0.1
2025-01-01

class GammaPoissonOccurrence.**PoissonGammaCompoundOccurrence**(*configuration*, ***kwargs*)

It is the combination of Poisson distribution with the gamma distribution to create the Poisson-gamma compound distribution to obtain the probability to observe n seismic events, within a time interval t, for temporal varying seismic activity λ (Benjamin, 1968), as follows

$$p_n(n|\lambda,t) = \frac{\Gamma\left(n+q_\lambda\right)}{n!\Gamma\left(q_\lambda\right)}\left(\frac{p_\lambda}{t+p_\lambda}\right)^{q_\lambda}\left(\frac{t}{t+p_\lambda}\right)^{n}$$

where in which $\Gamma()$ is the gamma function, $q_\lambda = \lambda^2/{\sigma_\lambda}^2$ and $p_\lambda = q_\lambda/\lambda$ are the constant parameters of gamma distribution, and λ means the mean value of the activity rate. This calculation is performed as

$$p_n(n|\lambda,t) = \exp\left(sum\right),$$

where

$$\begin{aligned} sum = {} & \ln\Gamma\left(n+q_\lambda\right) - \ln\Gamma\left(n+1\right) - \ln\Gamma\left(q_\lambda\right) \\ & + q_\lambda \ln\left(\frac{p_\lambda}{t+p_\lambda}\right) + n\ln\left(\frac{t}{t+p_\lambda}\right). \end{aligned}$$

The PDF of Poisson-gamma compound distribution of not exceeding magnitude m in time t is

$$f_M^{max}\left(m|\lambda,t\right) = \frac{\lambda t q_\lambda f_M\left(m\right) F_M^{max}\left(m|\lambda,t\right)}{q_\lambda + \lambda t S_M\left(m\right)}$$

The CDF of Poisson-gamma compound distribution of not exceeding magnitude m in time t is

$$F_M^{max}\left(m|\lambda,t\right)\left[\frac{q_\lambda}{q_\lambda + \lambda t S_M\left(m\right)}\right]^{q_\lambda}$$

Gradients of the of non occurrence of event in the time probability survive function are:

$$\frac{\partial S_M^{max}\left(m|t\right)}{\partial\lambda} = \left(1+\frac{\lambda t\left\{S_M\left(m\right)\right\}}{q_\lambda}\right)^{-q_\lambda-1} t\left\{S_M\left(m\right)\right\}$$

$$\frac{\partial S_M^{max}\left(m|t\right)}{\partial\theta_i} = t\lambda\left(1+\frac{\lambda t\left\{S_M\left(m\right)\right\}}{q_\lambda}\right)^{-q_\lambda-1}\left\{\frac{\partial S_M\left(m\right)}{\partial\theta_i}\right\}$$

Gradients of the n events in the time in the time probability survive function are:

Poisson-gamma compound coefficient are: λ (‘lambda’) and magnitude distribution coefficient. One constant coefficient is q_λ (‘q_lambda’). The class overwrite the OccurrenceBase methods:

_grad_sf_magnitude_distribution:

_d_grad_sf_magnitude_distribution:

_grad_sf:

_d_grad_sf:

d_mean:

ln_d_pmf(*n*, **args*)

The function compute the natural probability logarithm $\ln\left[p_n(n|t)\right]$ of occurrence of n events in the period t probability mass function. If you wanted to compute occurrence for magnitudes m or greater, you must set $m_{min} = m$ earlier.

Parameters

- **n**
- **args**

Returns

9.3.5 Delta (Δ) calculation classes

copyright
Jan Wiszniowski <jwisz@igf.edu.pl>, Andrzej Kijko <andrzej.kijko@up.ac.za>

license
GNU Lesser General Public License, Version 3 (https://www.gnu.org/copyleft/lesser.html)

version 0.0.1
2025-01-01

The delta classes define the Δ calculation methods, They are applied in a few m_{max} estimation algorithms, and realises an object-oriented approach to this issue. It provides a very flexible approach to m_{max} estimation, allowing the assessment of multiple m_{max} estimation algorithms in various combinations. The base Δ is

There exist two formulas for Δ calculation:

- based on Tate-Pisarenko theory,
- based on Kijko-Sellevoll theory.

class Delta.**BaseDelta**(*parameters*, *magnitude_distribution=None*, *m_max=None*, *m_max_obs=None*)

BaseDelta Δ calculation class.

delta(*n=None*, *time=1.0*, *annual_lambda=1.0*)

Calculates the delta value. Instead of using *delta* the object can be called oneself. E.g:

```
delta_object = KijkoSellevollDelta(params)
delta = delta_object(t=123.0, annual_lambda=0.37)
```

Parameters

- **n** – number if events. If n is unset then it is determined as t and λ $n = t\lambda$
- **time** – the time duration in years
- **annual_lambda** – annual occurrence - λ value

Returns
the Δ value - result of the virtual function _method_delta(m, n)

property m_max

It is the minimum magnitude

class Delta.**KijkoSellevoll**(*configuration*, *magnitude_distribution=None*, *m_max=None*, *m_max_obs=None*)

Kijko-Sellevoll Δ calculation class

$$\Delta = \int_{m_{min}}^{m_{max}} F_M\left(m\right)^n dm$$

The integration is performed numerically.

class Delta.**TatePisarenko**(*configuration*, *magnitude_distribution=None*, *m_max=None*, *m_max_obs=None*)

Tate-Pisarenko Δ calculation class

$$\Delta = \frac{1}{nf_M\left(m_{max}^{obs}\right)}$$

9.3.6 Classes for Bayesian maximum magnitude estimation

9.3.6.1 Module-based classes of Bayesian maximum magnitude likelihood

class BayesianMmax.**BayesianBase**(*configuration*, *magnitude_distribution=None*, *m_max_pair=None*)

The base class likelihood probability for most Bayesian methods. It estimates the non-bayesian $\hat{m}_{max}$ and defines the truncated normal distribution ($\pi\left(m_{max}\right)$) of apriori m_{max}. The derived classes must define the *_likelihood* method

Parameters

configuration (*dict*) – General configuration container, which is the dictionary of all parameters required for Ha3Py modules and results of all computations.

pdf(*m*, **args*)

Parameters

m

Returns

Return type

CHAPTER TEN

HOW TO CITE

Wiszniowski J., Kijko A. (2024) Python Package of Computer Codes for Assessment of Earthquake Recurrence Parameters on Incomplete Catalogues, Uncertain Magnitudes, and Complex Model, 39th ESC General Assembly, Corfu, Greece from 22 to 27 September 2024.

CHAPTER

ELEVEN

HA3PY CHANGELOG

Estimation of Earthquake Hazard Parameters from Incomplete Data Files

11.1 v0.0.1 - 2025-01-01

Initial Python package.

11.2 v0.0.2 - 2025-05-16

Program reads dates BC.

Query i printing modification

CHAPTER

TWELVE

CONTRIBUTING

Ha3Py estimation development happens on GitHub.

I'm very open to contributions: if you have new ideas, please open an Issue. Don't hesitate sending me pull requests with new features and/or bugfixes!

CHAPTER THIRTEEN

INDICES AND TABLES

- genindex
- modindex
- search

BIBLIOGRAPHY

[1] Andrzej Kijko, Ansie Smit, and Markvard A. Sellevoll. Estimation of earthquake hazard parameters from incomplete data files. part iii. incorporation of uncertainty of earthquake-occurrence model. *Bulletin of the Seismological Society of America*, 106(3):1210–1222, 05 2016. URL: https://doi.org/10.1785/0120150252, arXiv:https://pubs.geoscienceworld.org/ssa/bssa/article-pdf/106/3/1210/3668777/1210.pdf, doi:10.1785/0120150252.

[2] C. Radhakrishna Rao. *Linear Statistical Inference and its Applications*. Wiley, 1973. doi:10.1002/9780470316436.

[3] A. Smit, A. Stein, and A. Kijko. Bayesian inference in natural hazard analysis for incomplete and uncertain data. *Environmetrics*, 30(6):e2566, 2019. e2566 env-18-0069.R4. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/env.2566, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/env.2566, doi:https://doi.org/10.1002/env.2566.

[4] Andrzej Kijko. Estimation of the maximum earthquake magnitude, mmax. *pure and applied geophysics*, 161(8):1655–1681, Aug 2004. URL: https://doi.org/10.1007/s00024-004-2531-4, doi:10.1007/s00024-004-2531-4.

[5] Andrzej Kijko and Mayshree Singh. Statistical tools for maximum possible earthquake magnitude estimation. *Acta Geophysica*, 59(4):674–700, Aug 2011. URL: https://doi.org/10.2478/s11600-011-0012-6, doi:10.2478/s11600-011-0012-6.

[6] Petrus Vermeulen and Andrzej Kijko. More statistical tools for maximum possible earthquake magnitude estimation. *Acta Geophysica*, 65(4):579–587, Aug 2017. URL: https://doi.org/10.1007/s11600-017-0048-3, doi:10.1007/s11600-017-0048-3.

[7] Andrzej Kijko. Bayesian assessment of the maximum possible earthquake magnitude mmax. *Journal of the Geological Society of India*, 101(6):764–769, 06 2025. URL: https://doi.org/10.17491/jgsi/2025/174157, arXiv:https://pubs.geoscienceworld.org/geosocindia/jour-geosocindia/article-pdf/101/6/764/7220300/174157.pdf, doi:10.17491/jgsi/2025/174157.

[8] D. S. Robson and J. H. Whitlock. Estimation of a truncation point. *Biometrika*, 51(1/2):33–39, 1964. URL: http://www.jstor.org/stable/2334193 (visited on 2024-12-13).

[9] Peter Cooke. Statistical inference for bounds of random variables. *Biometrika*, 66(2):367–374, 08 1979. URL: https://doi.org/10.1093/biomet/66.2.367, arXiv:https://academic.oup.com/biomet/article-pdf/66/2/367/616519/66-2-367.pdf, doi:10.1093/biomet/66.2.367.

[10] B. Gutenberg and C. F. Richter. Earthquake magnitude, intensity, energy, and acceleration: (second paper). *Bulletin of the Seismological Society of America*, 46(2):105–145, 04 1956. URL: https://doi.org/10.1785/BSSA0460020105, arXiv:https://pubs.geoscienceworld.org/ssa/bssa/article-pdf/46/2/105/5302960/bssa0460020105.pdf, doi:10.1785/BSSA0460020105.

[11] A. Kijko and G. Graham. Parametric-historic procedure for probabilistic seismic hazard analysis part i: estimation of maximum regional magnitude mmax. *pure and applied geophysics*, 152(3):413–442, Oct 1998. URL: https://doi.org/10.1007/s000240050161, doi:10.1007/s000240050161.

[12] C.A. Cornell. Statistical analysis of maximum magnitudes in the earthquakes of stable continental regions. In J. Schneider, editor, *The Earthquakes of Stable Continental Regions, Vol. 1. Assessment of Large Earthquake Potential,*, chapter 5, pages 1–27. Electric Power Research Institute, Palo Alto, CA, USA, 1994.

[13] K.J. Coppersmith. Conclusions regarding maximum earthquake assessment. In J. Schneider, editor, *The Earthquakes of Stable Continental Regions, Vol. 1. Assessment of Large Earthquake Potential,*, chapter 6, pages 1–24. Electric Power Research Institute, Palo Alto, CA, USA, 1994.

[14] VF Pisarenko. Statistical estimation of strongest possible earthquakes. *IZVESTIYA AKADEMII NAUK SSSR FIZIKA ZEMLI*, pages 38–46, SEP 1991.

PYTHON MODULE INDEX

INDEX

N

O

P

R

S

T

U