Building a High-Resolution Earthquake Catalog from Raw Waveforms: A Step-by-Step Guide
Template Matching

Eric Beaucé[1]

[1]Columbia University, Lamont-Doherty Earth Observatory, New York

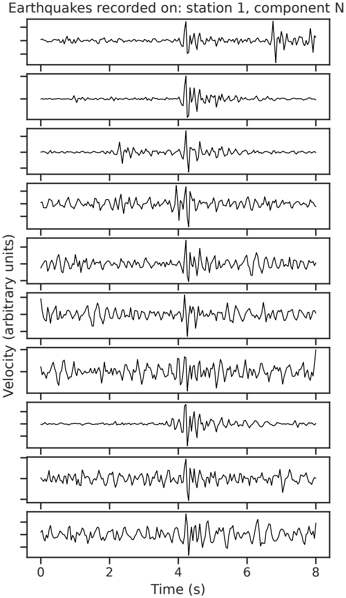
April 14, 2025

Outline for section 1

1. Introduction

2. Demonstration
   - Notebook 1
   - Notebook 2
   - Notebook 3 (bonus)

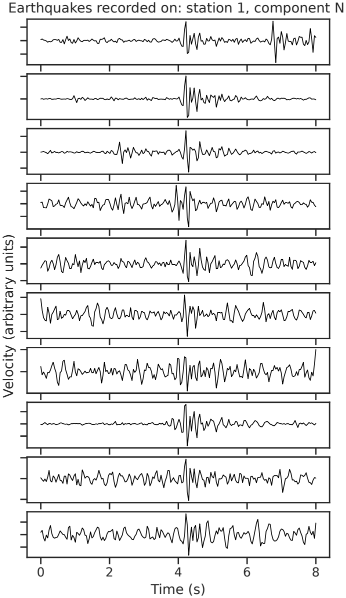
Earthquake signals often come in "families"



Family = near-repeats of the same waveform.

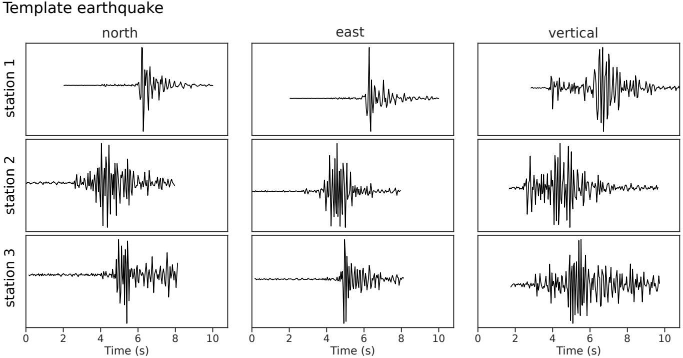
Can we use the similarity of these signals to our advantage to design an earthquake detection algorithm?

Earthquake signals often come in "families"



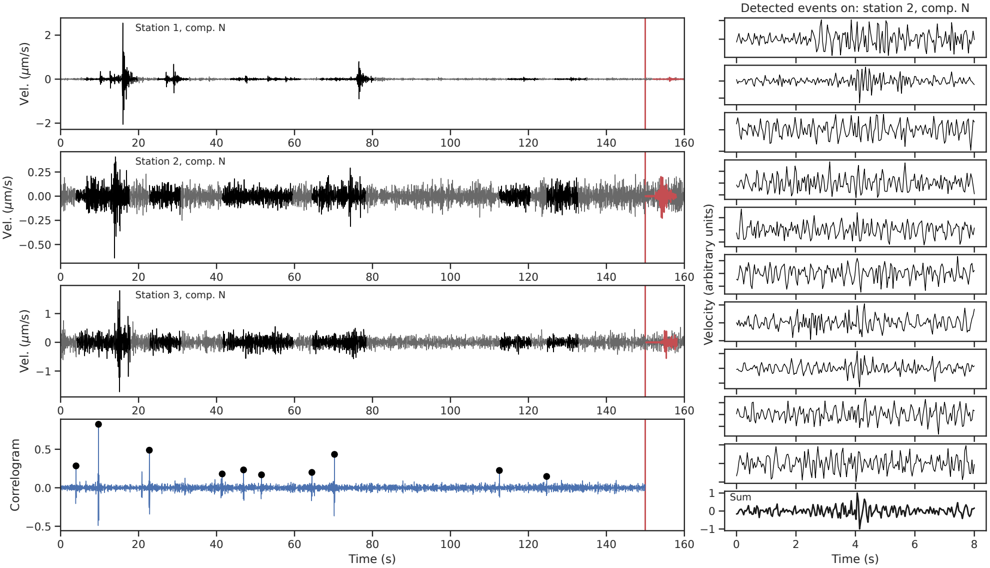
Family = near-repeats of the same waveform.

Can we use the similarity of these signals to our advantage to design an earthquake detection algorithm?



Looking for repeating patterns (template matching)

All earthquakes detected by the same template are **co-located**.



Similar earthquakes share same mechanism and location

Seismic signals result from the combination of:
source + **propagation** (including local site effects) + **instrument response**

Similar earthquakes share same mechanism and location

Seismic signals result from the combination of:
source + **propagation** (including local site effects) + **instrument response**

$$\underbrace{u(t;r)}_{\textit{seismogram}} = \underbrace{S(t;\xi,t_0)}_{\textit{source}} * \underbrace{G(t-t_0,r-\xi)}_{\textit{propagation}} * \underbrace{I(t)}_{\textit{instrument}} \tag{1}$$

- t: recording time.
- r: recording location.
- ξ: source location.
- t_0: source origin time.

Similar earthquakes share same mechanism and location

Seismic signals result from the combination of:
source + **propagation** (including local site effects) + **instrument response**

$$\underbrace{u(t; r)}_{seismogram} = \underbrace{S(t; \xi, t_0)}_{source} * \underbrace{G(t - t_0, r - \xi)}_{propagation} * \underbrace{I(t)}_{instrument} \tag{1}$$

- t: recording time.
- r: recording location.
- ξ: source location.
- t_0: source origin time.

$\Rightarrow$ Similar waveforms and moveouts **=** Similar source location and mechanism

Measuring waveform similarity

Template matching uses the Pearson correlation coefficient, CC, to measure the similarity between the template waveforms, T, and the continuous seismograms, u:

Measuring waveform similarity

Template matching uses the Pearson correlation coefficient, CC, to measure the similarity between the template waveforms, T, and the continuous seismograms, u:

$$CC(t) = \sum_{s,c} w_{s,c} \sum_{i=1}^{N} \frac{T^*_{s,c}(n\Delta t) u^*_{s,c}(t + \tilde{\tau}_{s,c} + n\Delta t)}{\sqrt{\sum_{i=1}^{N} {T^*_{s,c}}^2(n\Delta t) \sum_{i=1}^{N} {u^*_{s,c}}^2(t + \tilde{\tau}_{s,c} + n\Delta t)}} \tag{2}$$

- Δt: Sampling time.
- $\tilde{\tau}_{s,c}$: Moveout on station s and channel c.
- $T^*_{s,c}$: Centered template waveform, $T^*_{s,c} = T_{s,c} - \frac{1}{N} \sum_{i=1}^{N} T_{s,c}(n\Delta t)$.
- $u^*_{s,c}(t)$: Centered continuous seismogram, $u^*_{s,c}(t) = u_{s,c}(t) - \frac{1}{N} \sum_{i=1}^{N} u_{s,c}(t + n\Delta t)$.
- $w_{s,c}$: Weight, e.g., $w_{s,c} = \frac{1}{N_s N_c}$.

$CC(t) = +1$
perfect match

$CC(t) = 0$
nothing in commmon

$CC(t) = -1$
perfect match with inverse polarity

Measuring waveform similarity

Template matching uses the Pearson correlation coefficient, CC, to measure the similarity between the template waveforms, T, and the continuous seismograms, u:

$$CC(t) = \sum_{s,c} w_{s,c} \sum_{i=1}^{N} \frac{T^*_{s,c}(n\Delta t)\, u^*_{s,c}(t + \tilde{\tau}_{s,c} + n\Delta t)}{\sqrt{\sum^{N} T^{*\,2}_{s,c}(n\Delta t) \sum^{N} u^{*\,2}_{s,c}(t + \tilde{\tau}_{s,c} + n\Delta t)}} \qquad (2)$$

Template matching is extremely computation intensive!

$\rightarrow$ requires **High Performance Computing** $\leftarrow$

- $u^*_{s,c}(t)$: Centered continuous seismogram, $u^*_{s,c}(t) = u_{s,c}(t) - \frac{1}{N} \sum_{i=1} u_{s,c}(t + n\Delta t)$.
- $w_{s,c}$: Weight, e.g., $w_{s,c} = \frac{1}{N_s N_c}$.

$CC(t) = +1$	$CC(t) = 0$	$CC(t) = -1$
perfect match	nothing in commmon	perfect match with inverse polarity

Why template matching? And when can I use it?

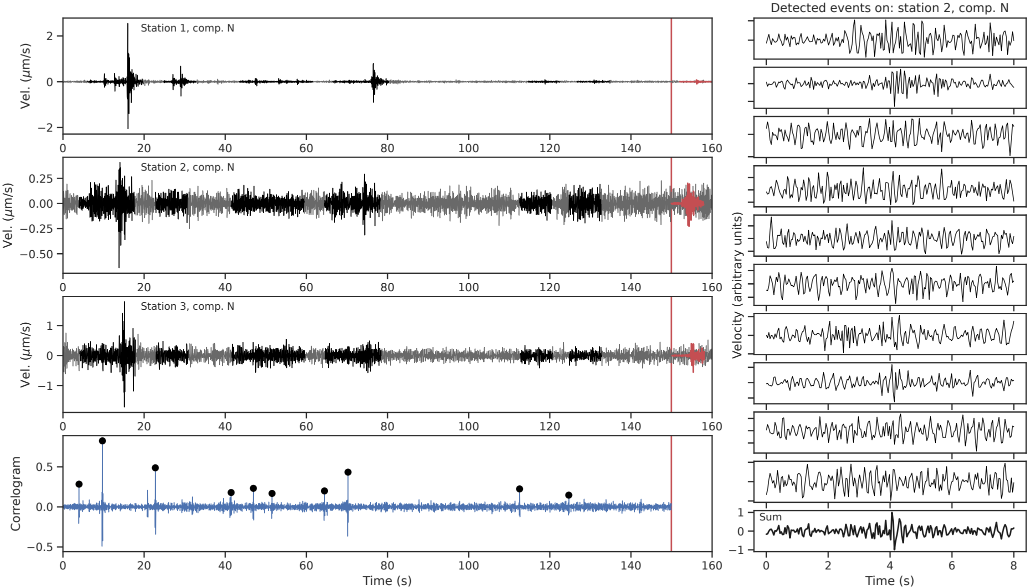
WHY?

- Searching for a *specific* seismic waveform allows to find *very small* events.
- Event location is known approximately, even for very small events.

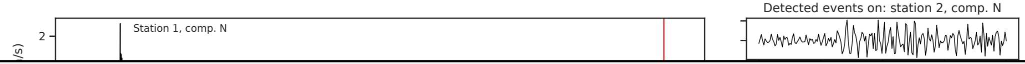
WHEN?

- Requires a starting catalog!
- For example: densify a regional catalog or a deep learning catalog.

Next: Implementing template matching with `Fast Matched Filter`



Next: Implementing template matching with `Fast Matched Filter`



`Fast Matched Filter`

RESEARCH ARTICLE | DECEMBER 20, 2017

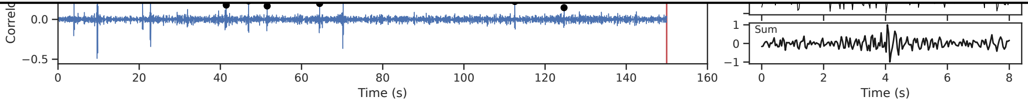
Fast Matched Filter (FMF): An Efficient Seismic Matched-Filter Search for Both CPU and GPU Architectures

Eric Beaucé; William B. Frank; Alexey Romanenko

+ Author and Article Information

Seismological Research Letters (2018) 89 (1): 165–172.
https://doi.org/10.1785/0220170181 | Article history

Other software: `EQCorrScan`, `Obspy`, `etc.`



Outline for section 2

1. Introduction

2. Demonstration
   - Notebook 1
   - Notebook 2
   - Notebook 3 (bonus)

Hands-on demonstration with Jupyter notebooks

- Notebook 1: Introduction to template matching with a single template.
- Notebook 2: Application with multiple templates over a single day.
- Notebook 3 (bonus): Common issue with detection threshold.

Github repository:
`https://github.com/AI4EPS/Earthquake_Catalog_Workshop`

Example of template matching with a single template

`https://ai4eps.github.io/Earthquake_Catalog_Workshop/notebooks/tm_one_template/`

Github repository:

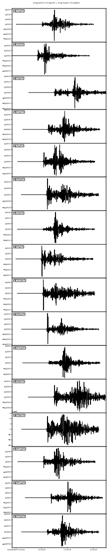
`https://github.com/AI4EPS/Earthquake_Catalog_Workshop/blob/main/notebooks/tm_one_template.ipynb`

Outline of `tm_one_template.ipynb`:

- Step 1: Build the template event from catalog data.
- Step 2: Compute the time series of correlation coefficients with `Fast Matched Filter`.
- Step 3: Choose and apply a detection threshold.
- Step 4: Extract detected events, build catalog and plot waveforms.
- Step 5: Compute relative magnitudes.

Step 1: Build template from catalog

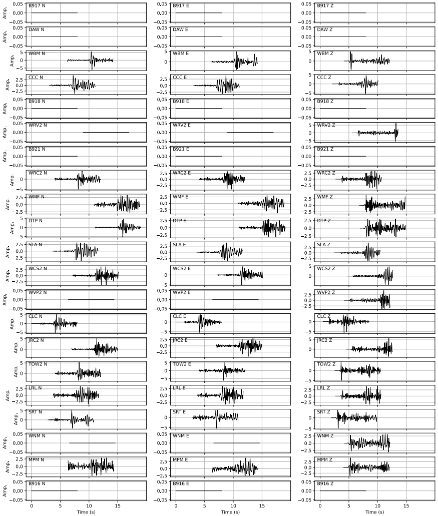
https://ai4eps.github.io/Earthquake_Catalog_Workshop/notebooks/tm_one_template/#build-template



Slice out most relevant part

⟶

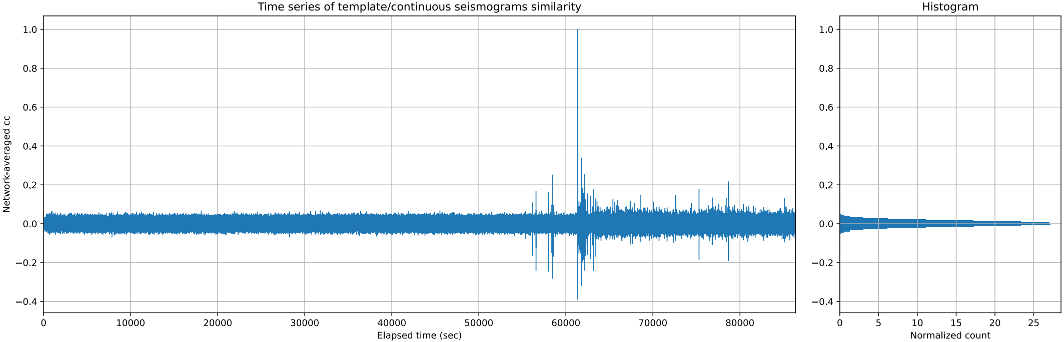
Template = Collection of waveforms and moveouts



Step 2: Compute time series of correlation coefficients

`https://ai4eps.github.io/Earthquake_Catalog_Workshop/notebooks/tm_one_template/#run-fmf`

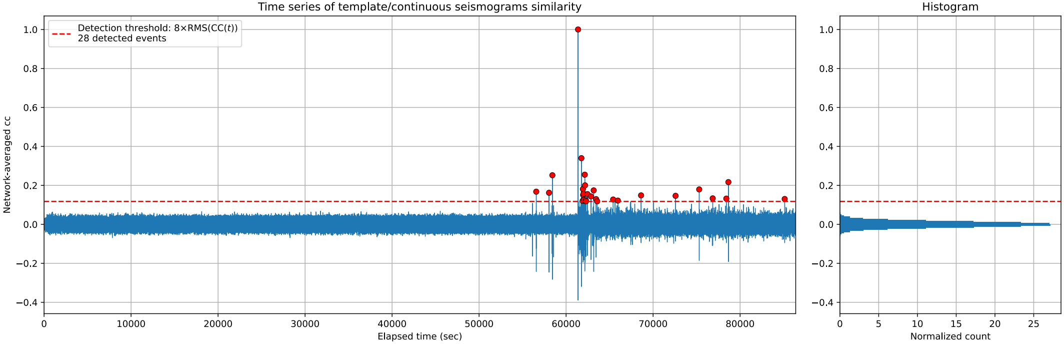
Compute $CC(t)$



Step 3: Design and apply detection threshold

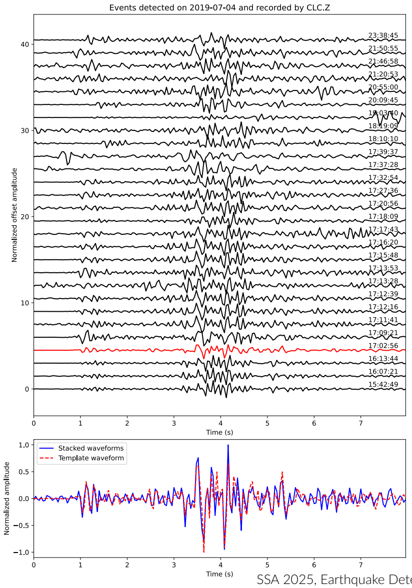
https://ai4eps.github.io/Earthquake_Catalog_Workshop/notebooks/tm_one_template/#set-detection-threshold-and-find-events

Gaussian distribution: 68.2% in $\pm 1\sigma$, 95.4% in $\pm 2\sigma$, ..., $(100 - 1.22 \times 10^{-13})$ in $\pm 8\sigma$



Step 4: Extract information on detected events

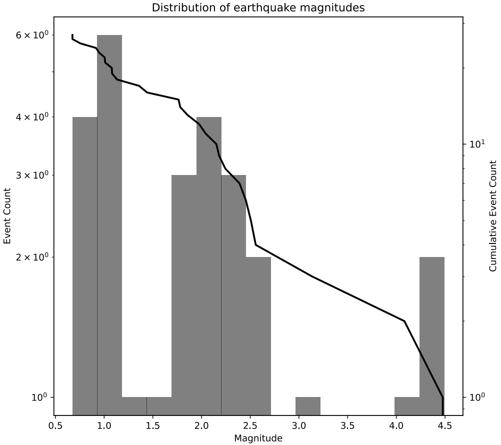
https://ai4eps.github.io/Earthquake_Catalog_Workshop/notebooks/tm_one_template/#set-detection-threshold-and-find-events



- Event timing.
- Peak waveform amplitude.
- Approximately share template location.

Step 5: Compute relative magnitudes

`https://ai4eps.github.io/Earthquake_Catalog_Workshop/notebooks/tm_one_template/#relative-magnitude`



Relative magnitudes between events sharing similar location and focal mechanism:

$$M_r = M_{ref} + \underbrace{\sum_i \log_{10} \frac{A_i}{A_{ref}}}_{\text{multiple seismometers/channels}} \tag{3}$$

Example of template matching with multiple templates

`https://ai4eps.github.io/Earthquake_Catalog_Workshop/notebooks/tm_multiple_templates/`

Github repository:
`https://github.com/AI4EPS/Earthquake_Catalog_Workshop/blob/main/notebooks/tm_multiple_templates.ipynb`

Outline of `tm_multiple_templates.ipynb`:

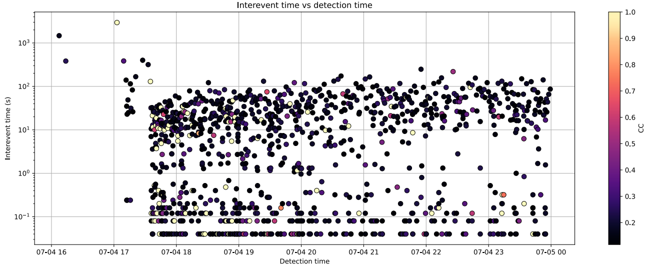
- Build the template events from catalog data.
- Compute the time series of correlation coefficients with `Fast Matched Filter`.
- Choose and apply a detection threshold.
- Extract detected events, build catalog and plot waveforms.
- **Find and remove events detected by multiple templates.**
- Compute relative magnitudes.

Template matching with multiple templates

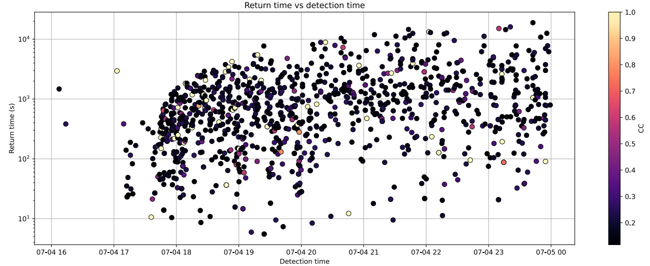
All the steps are similar to the first notebook, except at the end...

Flag and remove redundant detections I (from example with 91 templates)

`https://ai4eps.github.io/Earthquake_Catalog_Workshop/notebooks/tm_multiple_templates/#de-lumping-the-catalog`



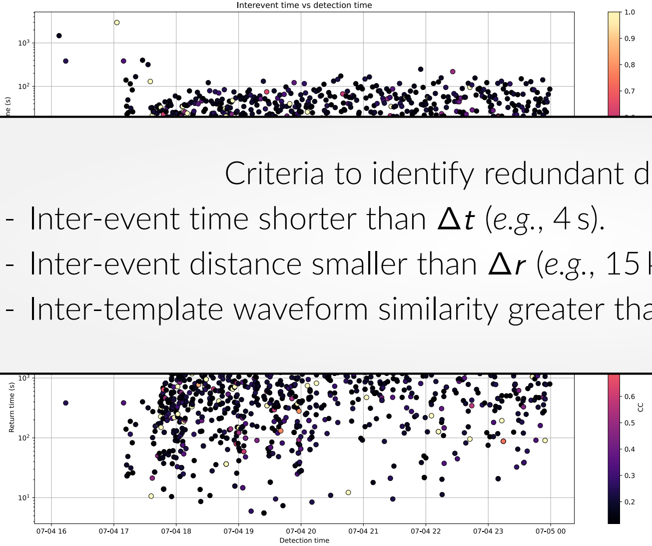
Inter-event time:
Time between *any* two consecutive events.



Return time:
Time between two consecutive events detected by same template.

Flag and remove redundant detections I (from example with 91 templates)

https://ai4eps.github.io/Earthquake_Catalog_Workshop/notebooks/tm_multiple_templates/#de-lumping-the-catalog



Inter-event time:
Time between *any* two consecutive events

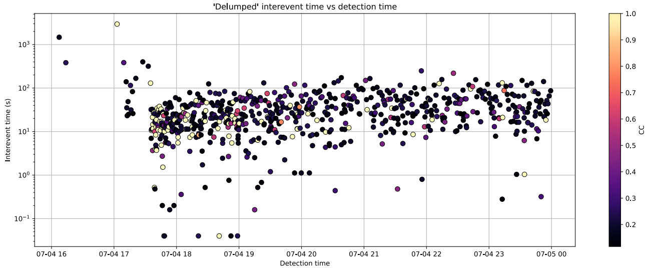
Criteria to identify redundant detections:

- Inter-event time shorter than Δt (*e.g.*, 4 s).
- Inter-event distance smaller than Δr (*e.g.*, 15 km).
- Inter-template waveform similarity greater than S (*e.g.*, 0.1).

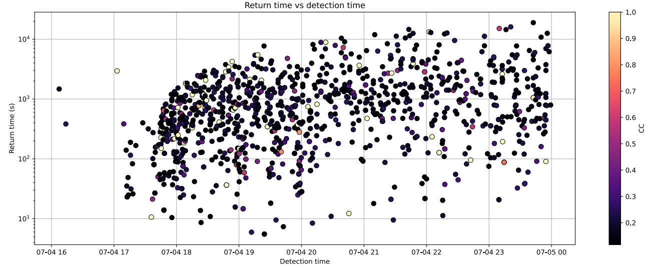
Return time:
Time between two consecutive events detected by same template.

Flag and remove redundant detections II (from example with 91 templates)

`https://ai4eps.github.io/Earthquake_Catalog_Workshop/notebooks/tm_multiple_templates/#de-lumping-the-catalog`



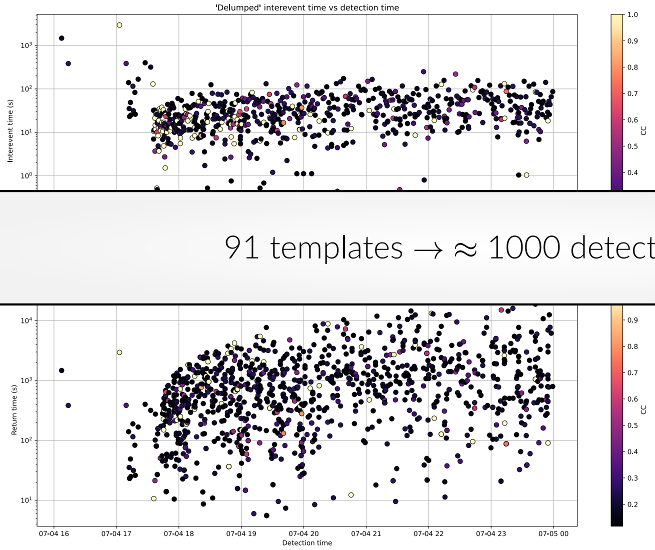
Inter-event time:
Time between *any* two consecutive events.



Return time:
Time between two consecutive events detected by same template.

Flag and remove redundant detections II (from example with 91 templates)

`https://ai4eps.github.io/Earthquake_Catalog_Workshop/notebooks/tm_multiple_templates/#de-lumping-the-catalog`



Inter-event time:
Time between *any* two consecutive events.

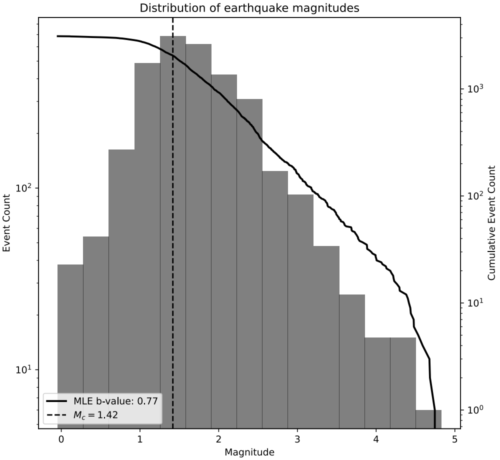
91 templates → ≈ 1000 detected events

Return time:
Time between two consecutive events detected by same template.

Run with 937 templates

https://ai4eps.github.io/Earthquake_Catalog_Workshop/notebooks/tm_multiple_templates/#relative-magnitudes

937 templates $\rightarrow \approx 3200$ detected events



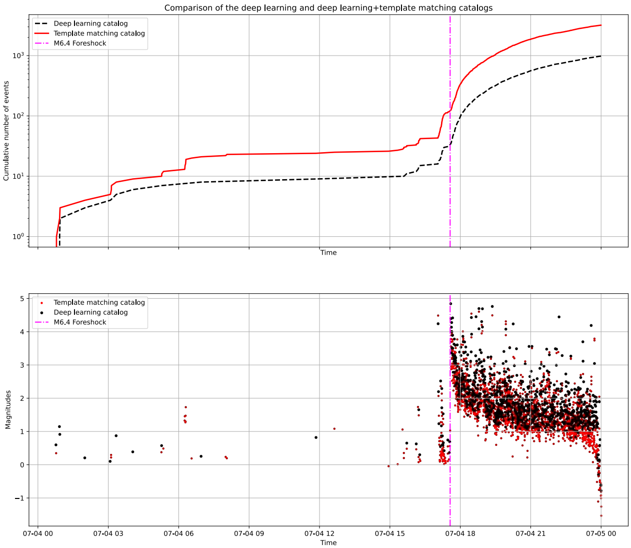
Gutenberg-Richter law:

$$\log_{10} N(m \geq M) = a - bM, \quad (4)$$

for magnitudes above M_c, the magnitude of completeness.

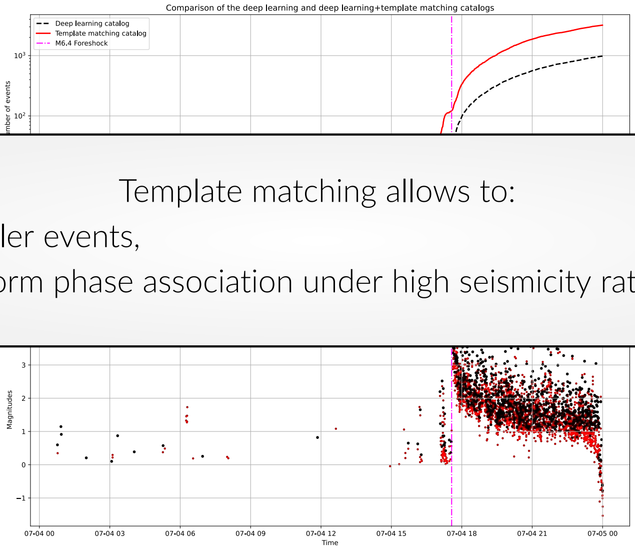
Run with 937 templates: Comparison against deep learning catalog

https://ai4eps.github.io/Earthquake_Catalog_Workshop/notebooks/tm_multiple_templates/#compare-with-deep-learning-catalog



Run with 937 templates: Comparison against deep learning catalog

https://ai4eps.github.io/Earthquake_Catalog_Workshop/notebooks/tm_multiple_templates/#compare-with-deep-learning-catalog



Template matching allows to:

- detect smaller events,
- better perform phase association under high seismicity rates.

Example of deffective detection threshold in template matching (BONUS)

`https://ai4eps.github.io/Earthquake_Catalog_Workshop/notebooks/tm_issues_w_detection_threshold/`

Github repository:
`https://github.com/AI4EPS/Earthquake_Catalog_Workshop/blob/main/notebooks/tm_issues_w_detection_threshold.ipynb`

Outline of `tm_issues_w_detection_threshold.ipynb`:

- Build the template events from catalog data.
- Introduce gaps in the data.
- Compute the time series of correlation coefficients with `Fast Matched Filter`.
- Choose and apply a detection threshold, illustrate gap-induced threshold bias.