Bootstrapping extreme value series

Antoine Doizé

supervisor : Olivier Wintenberger

September 13, 2023

Context

The idea of this work is to analyse variance estimations calculated by bootstrapping an estimator, in different contexts.
This way we can use the variance estimates to get confidence estimates of the estimation.
Mainly we will focus on the Hill estimator and the Max likelihood estimator on BM analysis.

Outline

1. Section 1 : Bootstrapping the Hill estimator
   - Hill estimator
   - Different form of variance estimator
   - Estimations on dependent data
   - Unbiaised Hill estimators for dependant data
2. Section 2 : Bootstrapping Block Maxima samples
   - Bootstrap variance dependance on the input sample
   - Bootstrap variance dependance on subsampling
   - Try variations of the bootstrap scheme

Hill estimator

We will study a series denoted : $X_1, ..., X_n$; which can be ordered in decreasing order : $X_{1,n} \geq X_{2,n} ... \geq X_{n,n}$. This first part will focus on the Hill estimator, which expression is

Definition

Hill estimator :

$$\hat{\alpha}^H_{k,n} = \frac{1}{k} \sum_{j=1}^{k} [ln(X_{j,n}) - ln(X_{k,n})]$$

Remark

This estimator is dependent on k the order statistic. In general we will use $k = \frac{n}{10}$

First method to estimate variance : moment 2 Hill estimator

In [3], it is suggested to use the "second-order" Hill moment estimator as estimator of the variance. By defining :

$$M_{k,n} := \frac{1}{k}\sum_{j=1}^{k}[ln(X_{j,n}) - ln(X_{k,n})]^2$$

they state that $\frac{M_{k,n}}{2\hat{\alpha}^{H}_{k,n}} - \hat{\alpha}^{H}_{k,n}$ and $\hat{\alpha}^{H}_{k,n} - \alpha$ have similar asymptotic behavior. That's why, by denoting E^* a bootstrapped estimator of E, we can use

Definition

$$var_{mom2} := \mathbb{E}[(\frac{M^*_{k,n}}{2\hat{\alpha}^{H*}_{k,n}} - \hat{\alpha}^{H*}_{k,n})^2]$$

as the fist variance estimator of the Hill estimator.

Second method to estimator variance of estimator : Kulik bootstrapping scheme

In [7] they suggest another bootstrapping scheme relying on a bloc-weighted estimations of the Hill estimator.
For a given n we chose blocs of size r_n
By denoting for any m : $\xi_m = (\xi_{j,m}, j \in \mathcal{Z})$ a sequence of i.i.d. random variables with $\mathbb{E}[\xi_{1,1}] = 0$ and $\mathbb{E}[\xi_{1,1}^2] = 1$, we have the bootstrap estimate of α :

Definition

$$\alpha_{kulik} = \frac{\sum_{i=1}^{m_n} (1+\xi_i) \sum_{j=(i-1)r_n+1}^{ir_n} \log \left(|\boldsymbol{X}_j| / |\boldsymbol{X}|_{(n:n-k)} \right) \mathbb{1} \left\{ |\boldsymbol{X}_j| > |\boldsymbol{X}|_{(n:n-k)} \right\}}{\sum_{i=1}^{m_n} (1+\xi_i) \sum_{j=(i-1)r_n+1}^{ir_n} \mathbb{1} \left\{ |\boldsymbol{X}_j| > |\boldsymbol{X}|_{(n:n-k)} \right\}}$$

By getting several values of this bootstrap estimate $\alpha_{kulik,1} \ldots \alpha_{kulik,m}$, we can have an estimate of its variance, that we will call var_{kulik}.

Dependence of the results on the input sample

The issue is that the results are dependent on the input sample : see 1 for variance calculation with Kulik and Hill second moment estimator (Note that variance is normalized by being multiplied by $\sqrt{k}$). The sample are following a Frechet distribution with $\alpha = 1/3$, and variance is calculated over 100 bootstrap iterations. To solve this, we are going to smooth the result by averaging the bootstrap calculated variance on several samples.

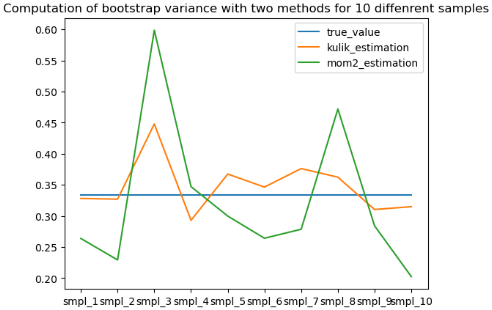


Figure: Bootstrapped calculated variance for different samples

Third method to estimate variance : De Haan and Zhou estimation

When bootstraping from several samples, two sources of randomness appear : one coming from the bootstrap scheme and one from the sample generation. To evaluate the very variance of the bootstrap estimator, [4] suggest to use a new estimator. Its specificity is to delete the source of randomness due to the sample generation.

Definition

$$var_{substracted} := \mathbb{V}[\sum_{l=1}^{d} (\hat{\alpha}_i^* - \hat{\alpha})^2]$$

$\hat{\alpha}_i$ being the bootstrap estimations and $\hat{\alpha}$ being the estimation on the original sample.

Bootstrap estimation of the variance

On 2 we show
the result of the computation of the bootstrap variance averaged on several samples. We bootstrapped 100 times and averaged on 100 samples. The sample is still of Frechet distribution with $\alpha = 1/3$. We see that the second moment estimator is underestimating the variance. The kulik estimator and the substracted estimator are quite similar.

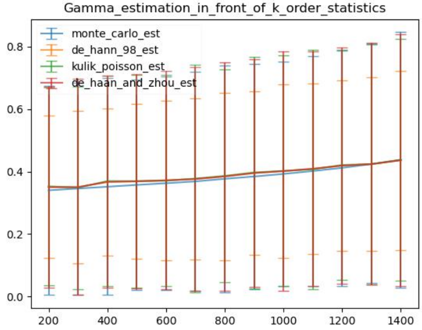


Figure: Bootstrapped calculated variance averaged on several samples

Regular estimators on dependent data 1/2

On 3 we show the result of the computation of the bootstrap variance averaged on several samples of dependent data (moving average of order 2). We bootstrapped 100 times and averaged on 100 samples. The sample is still of Frechet distribution with $\alpha = 1/3$. We see that the true value of *alpha* is underestimated. This effect is even stronger with higher values of order (see next slides).

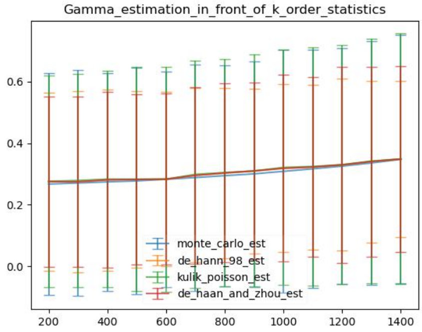
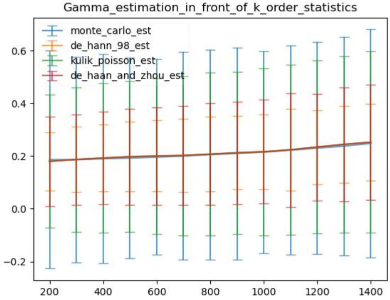


Figure: Bootstrapped calculated estimators averaged on several samples of dependent data

Regular estimators on dependent data 2/2



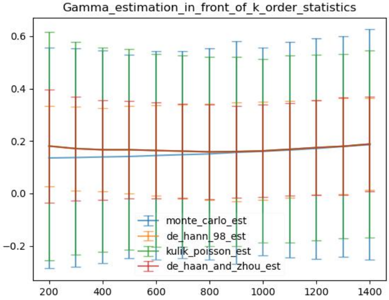


Figure: Caption

We show the estimations of *alpha* for Frechet moving average of parameter $alpha = 1/3$, with :

- $windowsize = 5$ to the left ‘
- $windowsize = 10$ to the right

showing that for dependent data, we need other unbiaised estimators.

Unbiaised Hill estimators

An unbiaised estimator is suggested in [5] :

Definition

$$\hat{\alpha}_{k_n,k_\rho,\alpha} := \hat{\alpha}_{k_n} - \frac{M_{k_n}^{(2)} - 2\hat{\alpha}_{k_n}^2}{2\hat{\gamma}_{k_n}\hat{\rho}_{k_\rho}^{(\beta)}\left(1-\hat{\rho}_{k_\rho}^{(\beta)}\right)^{-1}},$$

but relies on the evaluation of the second order parameter ρ. For this purpose, we have two ways of evaluating it.

First, still in [5], they suggest to define $M_k^{(\beta)} := \frac{1}{k}\sum_{i=1}^{k}\left(\log X_{n-i+1,n} - \log X_{n-k,n}\right)^{\beta}$, and then

$$S_k^{(2)} = \frac{3}{4}\frac{\left(M_k^{(4)} - 24\left(M_k^{(1)}\right)^4\right)\left(M_k^{(2)} - 2\left(M_k^{(1)}\right)^2\right)}{M_k^{(3)} - 6\left(M_k^{(1)}\right)^3}.$$

and compute

$$\hat{\rho}_k = \frac{-4 + 6S_k^{(2)} + \sqrt{3S_k^{(2)} - 2}}{4S_k^{(2)} - 3}.$$

However, the quantity is not always defined. That's why we have another way of estimating *rho* suggested in [6]. They suggest to compute

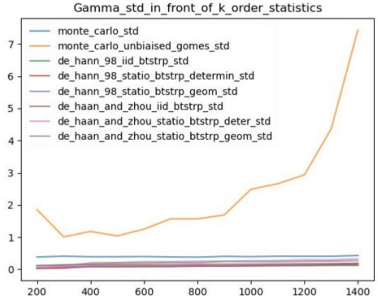
$$T_n^{\tau}(k) := \frac{ln(M_n^{(1)}(k) - 1/2ln(M_n^{(2)}(k)/2)}{1/2ln(M_n^{(2)}(k)/2) - 1/3ln(M_n^{(3)}(k)/6)}$$

and then, compute

$$\widehat{\rho}(k;\tau) := -\left|3\left(T_n^{(\tau)}(k) - 1\right) / \left(T_n^{(\tau)}(k) - 3\right)\right.$$

Instability of the estimator

The main issue of the unbiaised estimator is its high standard deviation. This variance is even higher for higher window sizes. This instability holds us from using it for a bootstrap scheme. Here are examples with window sizes of 5 (left) and 10 (right) :



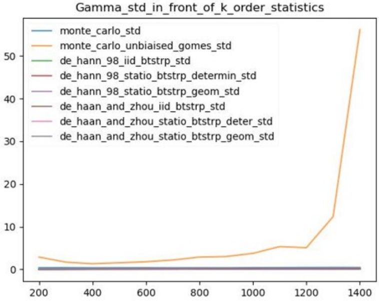


Figure: Unbiaised Hill estimator standard deviation

That's why we are going to move to another framework of bootstrapping

Block Maxima series

Extreme value analysis can rely on two different approaches, and Block Maxima method is one of them. The blocks from which we extract the maxima can be sliding ones (overlapping blocks) on disjoint ones, with similar behaviour (juste a little more stable with sliding ones). More details can be found in [2].
The BM method has been well studied, and bootstrapping in the case of i.i.d. series has already been studied, for example see [8].
We are going to try and see how bootstrapping is performing on dependent data.

Bootstrapping on one fixed sample

To begin with, we try to generate a sample a MA(10) of Frechet innovations of parameter $alpha = 1/3$. We extract the sliding and disjoint block maxima series and fit a GEV on those series. For one fixed sample we try 6 different bootstrap schemes. The results are quite stable. Let's see how it behaves with different samples.

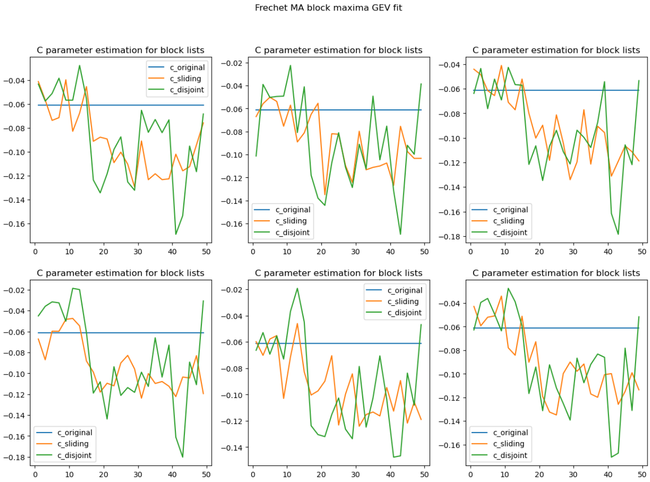


Figure: Bootstrapping the same sample

Bootstrapping different samples

Here we generate a different sample of a MA(10) of Frechet innovations of parameter $alpha = 1/3$. Each time we extract a bootstrap series and fit a GEV on those series. Here we see that depending on the samples, the bootstrap results are very variable, showing that the variance of the bootstrap scheme is highly dependent on the original sample.

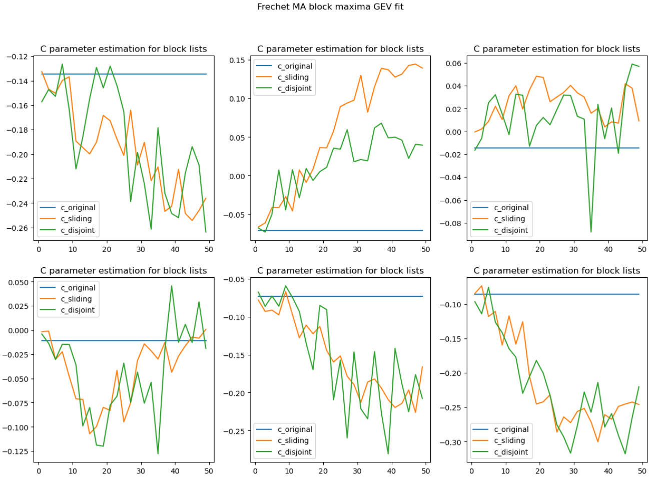


Figure: Bootstrapping from different sample

High variance of bootstrap method

The fact that the BM method is highly variant can explain why it has not been prolifically been dealt with in the litterature. With only one original sample, bootstrap computed variance won't be representative of the true variance of the estimator, because it will depend highly on the original sample data. This can explain why in articles such as [1], they must smooth the bootstrap schemes on 3000 repetitions to get smooth curves. To try and stabilize the results, we can try to downsample during the bootstrap estimation, as it can reduce variance.

Influence of downsampling on variance

On next figure we check the effect of downsampling on the variance.
For different downsampling rates, we extract sliding and disjoint block maxima for 50 different samples.
For block sizes (ranging from 1 to 50) we show the variance of the estimation on the different sample.
There don't seem to be any strong link between downsampling rate and variance of the estimations.
Downsampling don't seem to have any strong effect on the stabilization of the process.

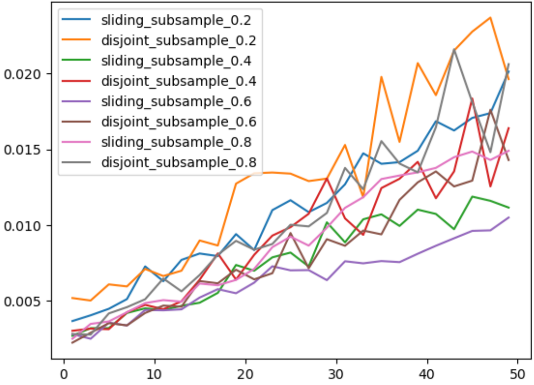


Figure: Dowsampling effect on variance

Bootstrapping scheme

It seems that the bootstrap scheme struggles to get intelligence from one single input sample. We are going to check if this comes from the dependence of the data or from the bootstrap scheme.
Our current bootstrap scheme is :

- Get an input sequence
- Extract Block Maxima from this sequence : this gives a BM sequence
- We bootstrap this BM sequence to get bootstrap samples, and compute a ML estimator on those samples.
- We then get the mean and std of those estimations.

The main advantage of this scheme is that it keeps the dependency structure.

Apply the bootstrap scheme BM-bootstrap on dependent and i.i.d. variables 1/2

Here are the results for an armax distribution, on one sample, bootstrapped 400 times.
We show the results for two different input samples.
It seems that the results have really different behaviours when the input sample is different.

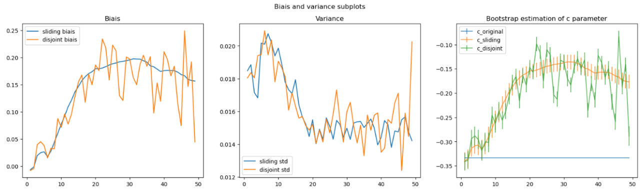


Figure: Bootstrapping the same sample 1

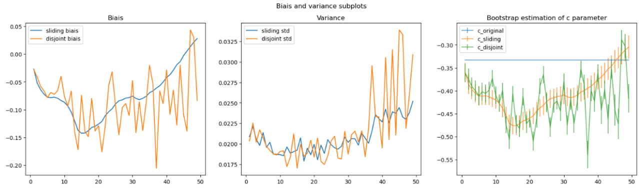


Figure: Bootstrapping the same sample 2

Apply the bootstrap scheme BM-bootstrap on dependent and i.i.d. variables 2/2

Here are the results for an i.i.d. Frechet distribution, on one sample, bootstrapped 400 times.
We show the results for two different input samples.
Similarly to dependent data, it seems that the results have really different behaviours when the input sample is different.

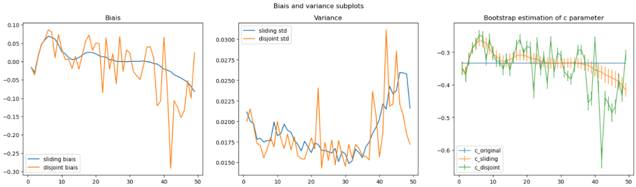


Figure: Bootstrapping the same sample 1

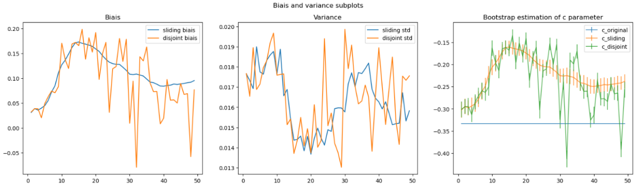


Figure: Bootstrapping the same sample 2

The chaotic behaviour is not due to dependency but to the bootstrap scheme

Hence, we saw that whatever the dependency of the input sample, the bootstrap scheme has the same chaotic behaviour. To verify this asumption, we are going to check the behaviour of another bootstrap scheme on i.i.d. data. Here are the main steps of the next bootstrap scheme :

- Get an input sequence
- First bootstrap this sequence (this drops the dependency structure)
- Then extract Block Maxima of the bootstrapped sample
- We get the mean and std of the bootstrap estimates

Apply the bootstrap scheme bootstrap-BM on i.i.d. variables

Here are the results for an i.i.d. Frechet distribution, on one sample, bootstrapped 100 times.
We show the results for two different input samples.
The result if much more input-sample independent and have nice variance properties.

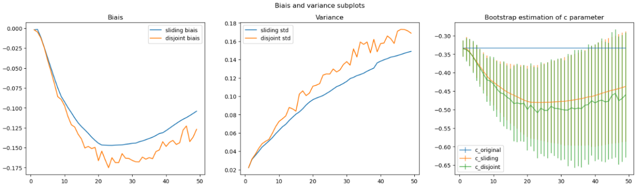


Figure: Bootstrapping the same sample 1

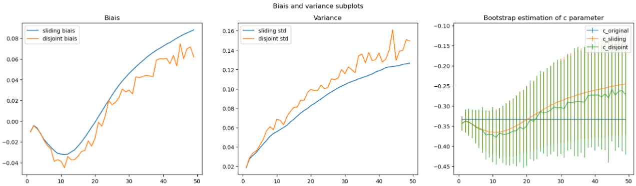


Figure: Bootstrapping the same sample 2

Apply the bootstrap scheme bootstrap-BM on dependent armax. variables

Here are the results for an Armax distribution, on one sample, bootstrapped 100 times. We compare to the monte Carlo estimation The result if much more input-sample independent and have nice variance properties.

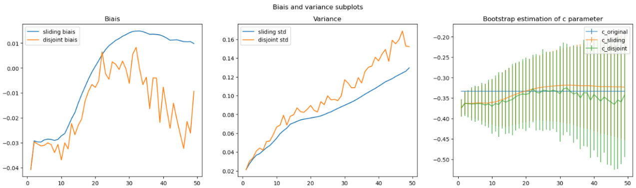


Figure: Bootstrapping the same sample 1

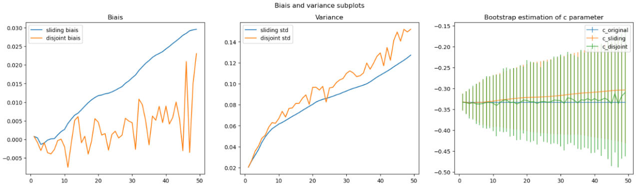


Figure: Monte Carlo BM estimations

Conclusion

Bootstrapping in extreme is a method that could help get an idea of the variance of an estimator, given a fixed sample of data.
This framework is well studied in the i.i.d. case, but when tackling dependent data, they is much less analysis conducted.
This work shows the main hindrance to applying bootstrap to dependent extremes keeping the dependent structure. Indeed, the high variance of the estimators related to the original sample makes the result highly unstable. Finding a way to stabilize this variance by working on the bootstrap scheme could be a real milestones in this area.
Nevertheless, by applying the regular bootstrap-block maxima scheme to bootstrap dependent series, we lose the dependence structure but we still get interpretable variance properties.

Axel Bücher and Johan Segers.
Inference for heavy tailed stationary time series based on sliding blocks.
Electronic Journal of Statistics, 12(1):1098 – 1125, 2018.

Axel Bücher and Chen Zhou.
A horse racing between the block maxima method and the peak-over-threshold approach, 2018.

J. Danielsson, L. de Haan, L. Peng, and C.G. de Vries.
Using a bootstrap method to choose the sample fraction in tail index estimation.
Journal of Multivariate Analysis, 76(2):226–248, 2001.

Laurens de Haan and Chen Zhou.
Bootstrapping extreme value estimators.
Journal of the American Statistical Association, 0(0):1–12, 2022.

Laurens de Haan, Chen Zhou, and Cecile Mercadier.
Adapting extreme value statistics to financial time series: dealing with bias and serial dependence.

Finance and Stochastics, 321(354):0, 2016.

M. Ivette Gomes and Ligia Henriques Rodrigues.
Tail index estimation for heavy tails: accommodation of bias in the excesses over a high threshold.
Extremes, 11(0):0, 2008.

Rafal Kulik and Philippe Soulier.
Heavy-Tailed Time Series.
01 2020.

Aline Mefleh, Romain Biard, Clément Dombry, and Zaher Khraibani.
Permutation bootstrap and the block maxima method.
Communications in Statistics - Simulation and Computation, 50(1):295–311, 2021.