Computational Resources for Extremes

T. Kleinow[1] M. Thomas[2]

November 24, 1999

[1] Institut für Statistik und Ökonometrie, Wirtschaftswissenschaftliche Fakultät, Humboldt-Universität zu Berlin

[2] Fachbereich Mathematik, Universität-Gesamthochschule Siegen

Abstract

The necessity to quantify the risk caused by the high volatility of asset prices, large insurance claims or floods has lead to an increasing interest in extreme value analysis. Generalized Pareto and extreme value distributions are well suited to model data which are exceedances above a threshold or maxima. We describe two statistical software systems - XploRe and Xtremes - that support a user in performing an extreme value analysis.

Within both systems, various estimators for the above distributions are provided. We give an overview of their application and mention visual tools to check the adequacy of a parametric modeling by means of non-parametric procedures.

Both systems utilize a client/server architecture to provide access to their resources across a network. While the server version of XploRe supports an interactive Java client which can be used from a web browser, the Xtremes system implements a CORBA interface that exports statistical objects to a client program.

1 Introduction

In extreme value analysis one is interested in parametric models for the distribution of maxima and exceedances. Suitable models are obtained by using

limiting distributions. In the following lines, we cite some basic results from extreme value theory. The reader is referred to [4] and [16] for a theoretical and to [14] for an applied introduction. A more detailed review is given in [11].

A classical result for the distribution of maxima was given by Fisher and Tippett in 1928 [6]. Assume that $X, X_1, X_2, \ldots$ are i.i.d. with common distribution function F. If for suitable constants a_n and b_n the standardized distribution of the maximum

$$P\left\{\frac{\max\{X_1,\ldots,X_n\}-b_n}{a_n}\leq x\right\}=F^n(a_nx+b_n)$$

converges to a continuous limiting distribution function G, then G is equal to one of the following types of extreme value (EV) distribution functions.

(i)	Gumbel (EV0)	$G_0(x)=\exp(-e^{-x})$,	$x\in\mathbb{R}$,
(ii)	Fréchet (EV1)	$G_{1,\alpha}(x)=\exp(-x^{-\alpha})$,	$x\geq 0$, $\alpha>0$,
(iii)	Weibull (EV2)	$G_{2,\alpha}(x)=\exp(-(-x)^{-\alpha})$,	$x\leq 0$, $\alpha<0$.

By employing the reparametrization $\gamma=1/\alpha$, these models can be unified using the von Mises parametrization

$$G_\gamma(x)=\begin{cases}\exp(-(1+\gamma x)^{-1/\gamma}), & 1+\gamma x>0,\ \gamma\neq 0,\\ \exp(-e^{-x}), & x\in\mathbb{R},\ \gamma=0.\end{cases}$$

One says that the distribution function F belongs to the domain of attraction of the extreme value distribution G, in short $F\in\mathcal{D}(G)$. The Gnedenko-De Haan theorem as well as the von Mises conditions provide sufficient conditions for $F\in\mathcal{D}(G)$ (see, e.g., [5] for details). Moreover, the assumption of independence can be weakened (see, e.g., [10]).

One may also consider the distribution function $F^{[t]}:=P(X<\cdot|X>t)$ of exceedances above a threshold t, where F lies in the domain of attraction of the extreme value distribution G_γ. Balkema and de Haan [1] as well as Pickands [13] showed that for suitable a_u and b_u the truncated distribution $F^{[u]}(b_u+a_ux)$ converges to a generalized Pareto (GP) distribution W_γ as $u\to\omega(F):=\sup\{x:F(x)<1\}$, with

$$W_\gamma(x)=\begin{cases}1-(1+\gamma x)^{-1/\gamma} & x>0,\ \gamma>0\\ & 0<x<-1/\gamma,\ \gamma<0\\ 1-e^{-x} & x\geq 0,\ \gamma=0.\end{cases}$$

Again, by using the parametrization $\alpha=1/\gamma$, one obtains the three submodels

(i)	Exponential (GP0)	$W_0(x) = 1 - e^{-x}$,	$x \geq 0$,
(ii)	Pareto (GP1)	$W_{1,\alpha}(x) = 1 - x^{-\alpha}$,	$x \geq 1$, $\alpha > 0$,
(iii	Beta (GP2)	$W_{2,\alpha}(x) = 1 - (-x)^{-\alpha}$,	$-1 \leq x \leq 0$, $\alpha < 0$.

These limit theorems suggest parametric distributions for data which are block maxima or exceedances above a threshold t. In the next section, we describe a computational approach for fitting these distributions to data.

2 Computational Resources

A similar extreme value module is implemented in the two software packages XploRe and Xtremes. We give a short introduction to the systems and provide an overview of the extreme value methods that are implemented.

2.1 XploRe

XploRe is an interactive statistical computing environment. It provides an integrated programming language with a large library of predefined functions and interactive tools for graphical analysis. A program written in the XploRe language is called *quantlet*. These quantlets are collected in libraries. The interactive tools include displays, with one or more plots, and low level GUI elements for user interaction during quantlet execution. To use XploRe without writing quantlets, it is possible to execute simple instructions on the command line, such as reading data, loading libraries or applying quantlets from a library to data.

There are two fundamental versions of XploRe. The first is a standalone statistical software available on several computer platforms, while the second one is a client/server system (http://www.xplore-stat.de). As described in section 3.1.2, the client/server architecture has many advantages. However, due to the early state of development, the XploRe client does not yet provide the same functionality as the standalone application.

2.2 Xtremes

The MS-Windows based statistical software Xtremes offers a menu-driven environment for data analysis. Besides the usual visualization options, there are parametric estimation procedures for Gaussian, extreme value and generalized Pareto distributions. Special menus offer applications of extreme

value analysis to problems arising in actuarial and financial mathematics as well as hydrology. See http://www.xtremes.de for more informations.

2.3 Extreme Value Analysis with XploRe and Xtremes

2.3.1 Estimators for GP and EV

Various estimators for extreme value and generalized Pareto distributions are implemented. We list the estimators available for GP distributions:

- Hill estimator, which is a m.l. estimator for the Pareto (GP1) submodel,
- m.l. estimator for the full GP model,
- Pickands estimator (see [13]),
- Drees-Pickands estimator, which uses a convex combination of Pickands estimators (see [3]),
- moment estimator (see [2]).

Two estimators for the EV distributions are provided:

- M.l. estimator for the full EV model,
- linear combination of ratio of spacings estimator, a construction similar to that of the Pickands estimator.

More details on the estimators are given in the cited literature as well as in [14] and [15]. While the fitting of an extreme value distribution is straight forward, a generalized Pareto distribution is fitted in two steps.

1. Select a threshold t and fit a GP distribution $W_{\gamma,t,\sigma}$ to the exceedances above t, where γ is the shape parameter and t and σ are location and scale parameter.
2. Transform the distribution to $W_{\hat{\gamma},\hat{\mu},\hat{\sigma}}$ which fits to the tail of the original data. The transformation is determined by the conditions $W^{[t]}_{\hat{\gamma},\hat{\mu},\hat{\sigma}} = W_{\gamma,t,\sigma}$ and $W^{[t]}_{\hat{\gamma},\hat{\mu},\hat{\sigma}}(t) = (n-k)/n$, where n is the sample size and k the number of exceedances above t. One obtains $\hat{\gamma} = \gamma$, $\hat{\sigma} = \sigma(k/n)^{\gamma}$ and $\hat{\mu} = t - (\sigma - \hat{\sigma})/\gamma$ as estimates of the tail fit. The latter values are displayed by the software.

In our implementation, we fix the number of upper extremes and use the threshold $t = x_{n-k+1:n}$.

2.3.2 Choosing a Threshold

The selection of an optimal threshold is still an unsolved problem. We employ a visual approach that plots the estimated shape parameter against the number of extremes. Within such a plot, one often recognizes a range where the estimates are stable. A typical diagram of estimates is shown in section 2.3.4.

2.3.3 Checking the Quality of a Fit

A basic idea of our implementation is to provide the ability to check a parametric modeling by means of nonparametric procedures. The software supports QQ-plots and a comparison of parametric and empiric versions of densities, distribution and quantile functions. An important tool for assessing the adequacy of a GP fitting is the mean excess function. It is given by

$$e_F(t) := E(X - t | X > t),$$

where X is a random variable with distribution function F. For a generalized Pareto distribution W_γ, the mean excess function is

$$e_{W_\gamma}(t) = \frac{1 + \gamma t}{1 - \gamma}.$$

We can therefore check if a GP tail is plausible by means of the sample mean excess function. Moreover, by comparing sample and parametric mean excess functions fitted by an estimator, a visual check of an estimation and a choice between different estimators becomes possible. The following section 2.3.4 demonstrates this approach.

2.3.4 Example Analysis of a Data Set

To exemplify the computational approach, we analyze a data set with the daily (negative) returns of the Yen related to the U.S. Dollar from Dec. 78 to Jan. 91. Figure 1 (left) shows a scatterplot of the 4444 returns. A fat tail of the distribution is clearly visible. In the following, we fit a generalized Pareto distribution to the tail of the returns by using the moment estimator.

To find a suitable threshold, a diagram of the estimates is plotted in Figure 1 (right). For $50 \leq k \leq 200$ the estimates are quite stable.

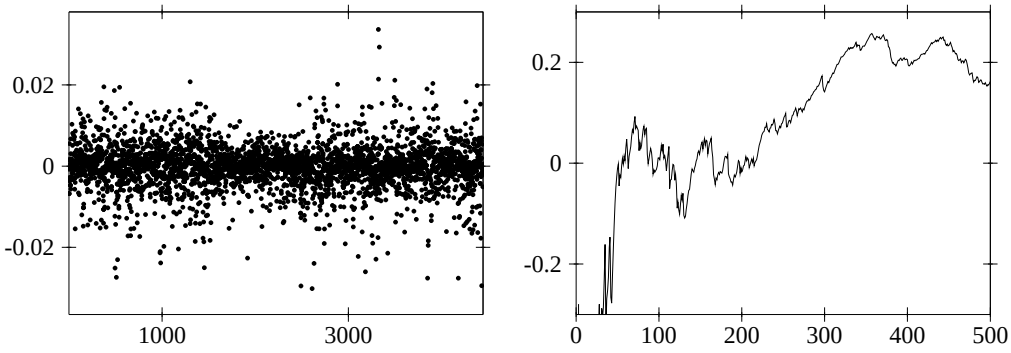

Figure 1: Daily returns of Yen/U.S. Dollar from Dec. 1978 to Jan. 1991 (left) and diagram of estimated shape parameters (right).

We select $k = 160$ extremes, yielding a threshold $t = 0.00966$ and plot a kernel density estimate (solid) as well as the parametric density fitted by the moment estimator (dotted) and the Hill estimator (dashed) for that number of extremes. The resulting picture is shown in Figure 2 (left).

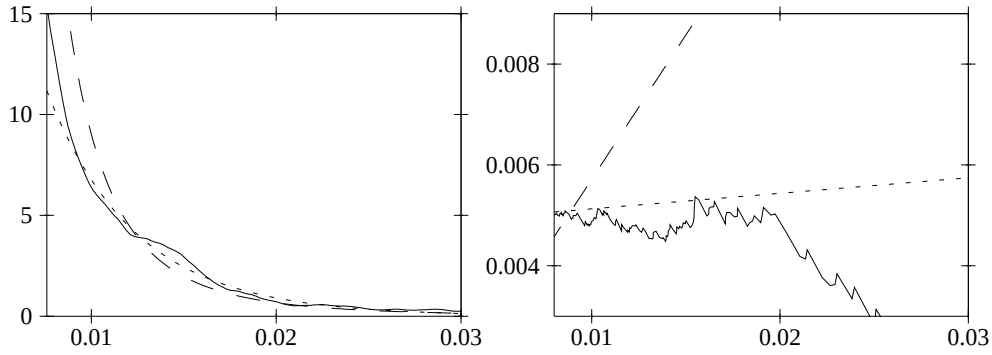

Figure 2: Densities (left) and mean excess functions (right) fitted by moment estimator (dotted) and Hill estimator (dashed).

Although the parameters estimated by the moment estimator seem to fit the kernel density slightly better, it is not easy to justify a parametric model

from the plot of the densities. We therefore also plot the mean excess functions. The right hand picture in Figure 2 shows the empirical mean excess function and the parametric versions, based on the same estimates. While the one fitted by the moment estimator (dotted) is close to the empiric version (solid), the one fitted by the Hill estimator (dashed) shows a strong deviation. This indicates that the parameters obtained by the moment estimator may be more appropriate.

2.4 Differences between XploRe and Xtremes

The XploRe system provides the user with an immediate language. Typical features of such a language (according to Huber [9]) are the omission of declarations and the ability to implement macros using the same constructs as in an immediate analysis.

Xtremes implements a menu interface for interactions and a compiled language for user written routines, whereby the user is required to declare all objects used within a program. That approach results in longer and more complex programs which are typically less flexible than interpreted ones with runtime type checking. However, a higher execution speed can be achieved as syntactic and semantic checks are performed at compile time.

3 Client/Server Architectures

Client/server architectures are becoming increasingly important in statistical computing. We discuss two of their advantages which are employed in XploRe and Xtremes: the separation of computational part and user interface and the provision of servers for special, user-written clients.

3.1 Client/Server Architecture of XploRe

The client/server version of the XploRe software package consists of three parts. The XploRe server is a batch program which provides several methods for statistical computing. The client is a GUI written in Java providing an interface for the user to interact with the server. Between these two resides a middleware program which manages the communication between client and server. Figure 3 shows the structure of this architecture.

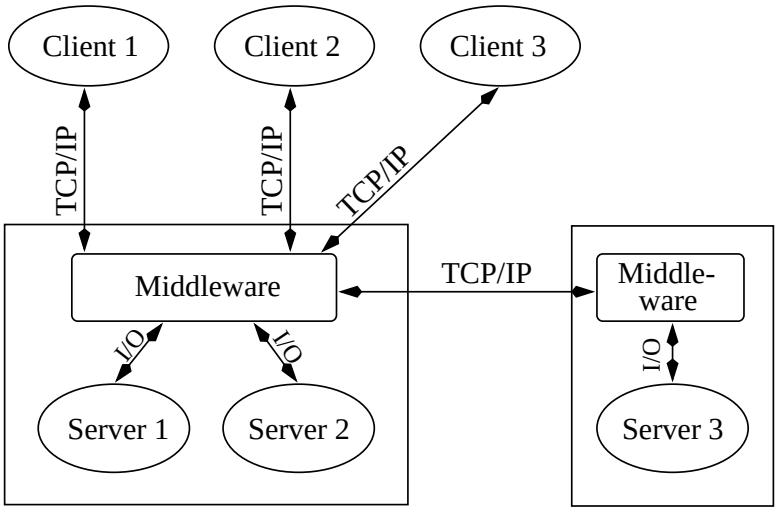

Figure 3: The Client/Server architecture of XploRe

3.1.1 Details of the Architecture

The main task of the XploRe server is to provide a statistical programming language and a variety of numerical methods for statistical analysis. To ensure high flexibility, it is possible to add methods (shared libraries, dynamically linked libraries) to the server dynamically. The xtremes library uses this mechanism. The server executes quantlets (programs written in the XploRe language) and writes the output to the standard output stream. Graphical output is encoded in a special protocol which is interpreted by the client.

The client provides the user with a GUI that lets him write and execute programs on a server, show numerical results and display the graphical output of an analysis. The platform independent client runs on every machine where a Java runtime environment is available. The server is written in C and C++, providing the numerical power needed for fast statistical computations.

The central part of this software package is the middleware. It is written in Java and resides on the same host as the server does. Its most important task is the management of the communication between server and client.

3.1.2 Advantages of the Architecture

One of the main advantages of the client/server architecture that is implemented within XploRe is the separation of the computational part and the user interface. It enables the user to use one or more servers without requiring high computational power on the host where the client is running. Instead, he has remote access to statistical methods as well as to computational resources.

In earlier versions of XploRe, the client/server communication has been managed by the server itself. The advantage of the separation of the managing part and the calculation part is a higher stability of the system as well as the opportunity to use different servers with one middleware. These servers could be Gauss, shazam or any other batch program.

3.1.3 Plans for Future Developments

In the future, the middleware should act as a distribution server; i.e., when an old client logs into the middleware, the user is offered an automatic update of the client. The old client downloads a new version from the middleware and installs it on the client host without user interaction. Another task of the middleware will be load average. This means when a middleware is contacted by a client it asks other middleware programs for the load of the hosts where they are running. The requests will then be sent to the host with the smallest load.

Due to the separation of the user interface and the computational part, different clients can be developed. In addition to the usual Java client, a prototype of an MS-Excel add-on exists. Besides clients for special environments (Java/Web, Excel), one could also think of clients for special purposes like finance or time series analysis. A Java-API (Application Program Interface) for the development of clients will be made available in future releases.

3.2 Xtremes CORBA Server

CORBA (see [12] or http://www.omg.org) is a platform and programming language independent standard for distributed, object oriented software systems. It encapsulates all network specific operations. Invoking methods on a remote server object is done by calling methods of a local proxy object. An *interface definition language* (IDL) describes the methods offered by the

server objects.

Xtremes implements a CORBA-compliant server which exports statistical procedures (such as estimators or data generation routines). We list an excerpt of the interface definition.

```
enum TPortType {  PORT_INT, PORT_REAL, ... };
typedef sequence<double> doubleseq;

interface TNode {
    string Name ();
    TNode Clone ();

    long GetNumberOfInports ();
    TPortType GetInportType (in long Nr);
    long GetNumberOfOutports ();
    TPortType GetOutportType (in long Nr);

    void SetInport (in long Nr, in any x);
    void Perform ();
    any GetOutport (in long Nr);
};
```

The server objects (called nodes) are self-describing. Besides a name, they return the number and types of parameters and results. After setting the parameters with `SetInport`, the `Perform` method is invoked, and results are fetched by calling `GetOutport`.

The homogeneous structure of the objects facilitates their creation by means of a factory [7]. On startup, the Xtremes server creates a factory object and publishes its object reference.

4 Conclusion

We have described two software systems that offer statistical methods for extreme value analysis in a distributed environment. Both systems allow the user to invoke statistical operations from a remote client; yet, different approaches are taken. Future effort should be invested in the specification of a general interface allowing the interoperation of different statistical software packages.

References

[1] Balkema, A.A., de Haan, L. (1974). Residual life time at great age. *Ann. Probab.* **2**, 792–804.

[2] Dekkers, A.L.M., Einmahl, J.H.J., de Haan, L. (1989). A moment estimator for the index of an extreme-value distribution. *Ann. Stat.* **17**, 1833–1855.

[3] Drees, H. (1995). Refined Pickands estimators of the extreme value index. *Ann. Stat.* **23**, 2059–2080.

[4] Embrechts, P., Klüppelberg, C., Mikosch, T. (1997). *Modelling Extremal Events.* Springer.

[5] Falk, M., Hüsler, J., Reiss, R.-D. (1994). *Laws of Small Numbers: Extremes and Rare Events.* DMV-Seminar, Birkhäuser, Basel.

[6] Fisher, R.A., Tippett, L.H.C. (1928). Limiting forms of the frequency distribution of the largest and smallest member of a sample. *Proc. Camb. Phil. Soc.* **24**, 180-190.

[7] Gamma, E., Helm, R., Johnson, R., Vlissides, J. (1995). *Design Patterns.* Addison-Wesley, Reading, Massachusetts.

[8] Härdle, W., Klinke, S., Müller, M. (1999). *XploRe -The Statistical Environment.* Springer, New York.

[9] Huber, P.J. (1994). Languages for Statistics and Data Analysis. In: Dirschedl, P., Ostermann, R. (Eds.) (1994). *Computational Statistics.* Physica, Heidelberg.

[10] Leadbetter, M.R., Nandagopalan, S. (1989). On exceedance point processes for stationary sequences under mild oscillation restrictions. In: *Extreme Value Theory.* J. Hüsler and R.-D. Reiss (eds.). *Lect. Notes in Statistics 51*, Springer, New York.

[11] McNeil, A.J. (1997). Estimating the tails of loss severity distributions using extreme value theory. *ASTIN Bulletin* **27**, 117–137.

[12] Object Management Group (1995). *The Common Object Request Broker: Architecture and Specification.*

[13] Pickands, J. (1975). Statistical inference using extreme order statistics. *Ann. Stat.* **3**, 119–131.

[14] Reiss, R.-D., Thomas, M. (1997). *Statistical Analysis of Extreme Values.* Birkhäuser, Basel.

[15] Reiss, R.-D., Thomas, M. (1999). Extreme Value Analysis. In: [8].

[16] Resnick, S.I. (1987). *Extreme Values, Regular Variation, and Point Processes.* Springer, New York.