

A User's Guide to the evd Package (Version 2.2)

Alec Stephenson

Copyright ©2006

Department of Statistics and Applied Probability,
National University of Singapore,
Singapore 117546.

E-mail: alec_stephenson@hotmail.com
25th March 2006

1 Introduction

1.1 What is the evd package?

The evd (extreme value distributions) package is an add-on package for the R (Ihaka and Gentleman, 1996) statistical computing system. The package contains the following (user-level) functions. It also contains the demo `soe9`, giving examples from Chapter Nine of Beirlant *et al.* (2004).

Univariate Distributions. Density, distribution, simulation and quantile (inverse distribution) functions for univariate parametric distributions.

```
dgev dgpd dgumbel drweibull dfrechet dextreme dorder  
pgev pgpd pgumbel prweibull pfrechet pextreme porder  
rgev rgpd rgumbel rrweibull rfrechet rextreme rorder  
qgev qgpd qgumbel qrweibull qfrechet qextreme
```

Multivariate Distributions. Density, distribution, simulation and dependence functions for multivariate parametric extreme value models.

```
dbvevd dmvevd pbvevd pmvevd rbvevd rmvevd abvevd amvevd
```

Non-parametric Estimation. Calculate and plot non-parametric estimates of dependence functions and quantile curves.

```
abvnonpar amvnonpar qcbvnonpar
```

Stochastic Processes. Generate stochastic processes associated with extreme value theory, identify extreme clusters and estimate the extremal index.

```
evmc marma mar mma clusters exi
```

Fitting Models. Obtain maximum likelihood estimates and standard errors for univariate and bivariate models used in extreme value theory.

```
fbvevd fgev fpot forder fextreme
```

Pre-model Diagnostics. Threshold identification and dependence summaries.

```
mrlplot tcplot chiplot
```

Model Diagnostics. Model diagnostics for fitted models; diagnostic plots and analysis of deviance.

```
plot.uvevd plot.bvevd anova.evd
```

Profile likelihoods. Obtain profile traces, plot profile log-likelihoods and obtain profile confidence intervals from fitted models.

```
profile.evd plot.profile.evd profile2d.evd plot.profile2d.evd
```

The following datasets are also included in the package.

```
failure fox lisbon ocmulgee oldage oxford lossalae  
portpirie sask sealevel uccle venice sealevel2
```

1.2 Obtaining the package/guide

The `evd` package can be downloaded from CRAN (The Comprehensive R Archive Network) at <http://cran.r-project.org/>. This guide (in pdf) will be in the directory `evd/doc/` underneath wherever the package is installed.

1.3 Contents

This guide contains examples* on the use of the `evd` package. The examples do not include any theoretical justification. See Coles (2001) for an introduction to the statistics of extreme values, and Beirlant *et al.* (2004) for a more detailed treatment.

Section 2 covers the standard (non-fitting) functions for univariate distributions. Sections 3 and 4 do the same for bivariate and multivariate extreme value models. Dependence functions of extreme value distributions are discussed in Section 5. Stochastic processes are discussed in Section 6. Maximum likelihood fitting of univariate models, peaks over threshold models and bivariate extreme value models is discussed in Sections 7, 8 and 9 respectively. Three practical examples using the data sets `oxford`, `rain` and `sealevel` are given in Sections 10, 11 and 12 respectively.

This guide should not be viewed as an alternative to the documentation files included within the package. These remain the definitive source of information. A reference manual containing all the documentation files can be downloaded from CRAN.

1.4 Citing the package

Volume 2/2 of R-News (the newsletter of the R-project) contains an article that describes an earlier version of the `evd` package. To cite the package in publications please cite the R-News article. The article and the corresponding citation can be downloaded from <http://www.cran.r-project.org/doc/Rnews/>.

1.5 Caveat

I have checked these functions as best I can but, as ever, they may contain bugs. If you find a bug or suspected bug in the code or the documentation please report it to me at alec_stephenson@hotmail.com. Please include an appropriate subject line.

1.6 Legalese

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but without any warranty; without even the implied warranty of merchantability or fitness for a particular purpose. See the GNU General Public License for more details.

A copy of the GNU General Public License can be obtained from <http://www.gnu.org/copyleft/gpl.html>. You can also obtain it by writing to the Free Software Foundation, Inc., 59 Temple Place – Suite 330, Boston, MA 02111-1307, USA.

*All of the examples presented in this guide are called with `options(digits = 4)`, and with the option `show.signif.stars` set to `FALSE`.

2 Univariate Distributions

The Gumbel, Fréchet and (reversed) Weibull distribution functions are respectively given by

$$G(z) = \exp \left\{ -\exp \left[-\left(\frac{z-a}{b} \right) \right] \right\}, \quad -\infty < z < \infty \quad (1)$$

$$G(z) = \begin{cases} 0, & z \leq a, \\ \exp \left\{ -\left(\frac{z-a}{b} \right)^{-\alpha} \right\}, & z > a, \end{cases} \quad (2)$$

$$G(z) = \begin{cases} \exp \left\{ -\left[-\left(\frac{z-a}{b} \right) \right]^\alpha \right\}, & z < a, \\ 1, & z \geq a, \end{cases} \quad (3)$$

where a is a location parameter, $b > 0$ is a scale parameter and $\alpha > 0$ is a shape parameter. The distribution (3) is often referred to as the Weibull distribution. To avoid confusion I will call this the reversed Weibull, since it is related by a change of sign to the three parameter Weibull distribution used in survival analysis.

The Generalised Extreme Value (GEV) distribution function is given by

$$G(z) = \exp \left\{ -[1 + \xi(z - \mu)/\sigma]_+^{-1/\xi} \right\}, \quad (4)$$

where (μ, σ, ξ) are the location, scale and shape parameters respectively, $\sigma > 0$ and $h_+ = \max(h, 0)$. When $\xi > 0$ the GEV distribution has a finite lower end point, given by $\mu - \sigma/\xi$. When $\xi < 0$ the GEV distribution has a finite upper end point, also given by $\mu - \sigma/\xi$. The parametric form of the GEV encompasses that of the Gumbel, Fréchet and reversed Weibull distributions. The Gumbel distribution is obtained in the limit as $\xi \rightarrow 0$. The Fréchet and Weibull distributions are obtained when $\xi > 0$ and $\xi < 0$ respectively. To recover the parameterisation of the Fréchet distribution (2) set $\xi = 1/\alpha > 0$, $\sigma = b/\alpha > 0$ and $\mu = a + b$. To recover the parameterisation of the reversed Weibull distribution (3) set $\xi = -1/\alpha < 0$, $\sigma = b/\alpha > 0$ and $\mu = a - b$.

The generalised Pareto distribution (GPD) function is given by

$$G(z) = 1 - [1 + \xi(z - \mu)/\sigma]_+^{-1/\xi},$$

for $z > \mu$, where (μ, σ, ξ) are the location, scale and shape parameters respectively, $\sigma > 0$ and $h_+ = \max(h, 0)$. The GPD has a finite lower end point, given by μ . When $\xi < 0$ the GPD also has a finite upper end point, given by $\mu - \sigma/\xi$. A shifted exponential distribution is obtained in the limit as $\xi \rightarrow 0$.

It is standard practice within R to concatenate the letters r, p, q and d with an abbreviated distribution name to yield the names of the corresponding simulation, distribution, quantile (inverse distribution) and density functions respectively. The evd package follows this convention. Each of the five distributions defined above has an associated set of functions, as given in Section 1.1. Some examples are given below. They should be familiar to those who have had previous experience with R.

```
> rgev(6, loc = c(20,1), scale = .5, shape = 1)
[1] 23.7290 1.2492 19.6680 0.8662 19.7939 2.6512
> rgpd(3, loc = 2)
[1] 2.483681 3.666805 2.837809

> qrweibull(seq(0.1, 0.4, 0.1), 2, 0.5, 1, lower.tail = FALSE)
> qrweibull(seq(0.9, 0.6, -0.1), loc = 2, scale = 0.5, shape = 1)
```

```
# Both give
[1] 1.947 1.888 1.822 1.745

> pfrechet(2:6, 2, 0.5, 1)
[1] 0.0000 0.6065 0.7788 0.8465 0.8825
> pfrechet(2:6, 2, 0.5, 1, low = FALSE)
[1] 1.0000 0.3935 0.2212 0.1535 0.1175

> drweibull(-1:3, 2, 0.5, log = TRUE)
[1] -5.307 -3.307 -1.307 -Inf -Inf
> dgumbel(-1:3, 0, 1)
[1] 0.17937 0.36788 0.25465 0.11820 0.04737
```

Let F be an arbitrary distribution function, and let X_1, \dots, X_m be a random sample from F . Define $U_m = \max\{X_1, \dots, X_m\}$ and $L_m = \min\{X_1, \dots, X_m\}$. The distributions of U_m and L_m are given by

$$\Pr(U_m \leq x) = [F(x)]^m \quad (5)$$

$$\Pr(L_m \leq x) = 1 - [1 - F(x)]^m. \quad (6)$$

Simulation, distribution, quantile and density functions for the distribution of U_m , given an integer m and an arbitrary distribution function F , are provided by **rextreme**, **pextreme**, **qextreme** and **dextreme** respectively. The integer m should be given to the argument **m**len. The distribution F is most easily specified by passing an abbreviated distribution name to the argument **distn**. If **largest** is **FALSE** the distribution of L_m is used. Some examples are given below.

```
> rextreme(1, distn = "norm", sd = 2, mlen = 20, largest = FALSE)
> min(rnorm(20, mean = 0, sd = 2))
# Both simulate from the same distribution
[1] -2.612

> rextreme(4, distn = "exp", rate = 1, mlen = 5)
> rextreme(4, distn = "exp", mlen = 5)
# Both simulate from the same distribution
[1] 2.2001 0.8584 4.5595 3.9397

> pextreme(c(.4, .5), distn = "norm", mean = 0.5, sd = c(1, 2), mlen = 4)
[1] 0.04484 0.06250
> dextreme(c(1, 4), distn = "gamma", shape = 1, scale = 0.3, mlen = 100)
[1] 0.3261328 0.0005398
```

Let $X_{(1)} \geq X_{(2)} \geq \dots \geq X_{(m)}$ be the order statistics of the random sample X_1, \dots, X_m . The distribution of the j th largest order statistic, for $j = 1, \dots, m$, is

$$\Pr(X_{(j)} \leq x) = \sum_{k=0}^{j-1} \binom{m}{k} [F(x)]^{m-k} [1 - F(x)]^k. \quad (7)$$

The distribution of the j th smallest order statistic is obtained by setting $j = m+1-j$. Simulation, distribution and density functions for the distribution of $X_{(j)}$, for given integers m and $j \in \{1, \dots, m\}$, and for an arbitrary distribution function F , are provided by **rorder**, **porder** and **dorder** respectively. The integer m should again be given to the argument **m**len. If **largest** is **FALSE** the distribution of the j th smallest order statistic $X_{(m+j-1)}$ is used. Some examples are given below.

```

> rorder(1, distn = "norm", mlen = 20, j = 2)
[1] 2.284
> porder(c(1, 2), distn = "gamma", shape = c(.5, .7), mlen = 10, j = 2)
[1] 0.5177 0.8259
> dorder(c(1, 2), distn = "gamma", shape = c(.5, .7), mlen = 10, j = 2)
[1] 0.7473 0.3081

```

3 Bivariate Extreme Value Distributions

The evd package contains functions associated with nine parametric bivariate extreme value distributions. The univariate marginal distributions in each case are GEV, with marginal parameters (μ_1, σ_1, ξ_1) and (μ_2, σ_2, ξ_2) .

There are three symmetric models, with distribution functions

$$G(z_1, z_2) = \exp \left\{ -(y_1^{1/\alpha} + y_2^{1/\alpha})^\alpha \right\}, \quad 0 < \alpha \leq 1, \quad (8)$$

$$G(z_1, z_2) = \exp \left\{ -y_1 - y_2 + (y_1^{-r} + y_2^{-r})^{-1/r} \right\}, \quad r > 0, \quad (9)$$

$$G(z_1, z_2) = \exp \left(-y_1 \Phi \{ \lambda^{-1} + \frac{1}{2} \lambda [\log(y_1/y_2)] \} - y_2 \Phi \{ \lambda^{-1} + \frac{1}{2} \lambda [\log(y_2/y_1)] \} \right), \quad \lambda > 0,$$

known as the logistic (Gumbel, 1960), negative logistic (Galambos, 1975) and Hüsler-Reiss (Hüsler and Reiss, 1989) models respectively, where

$$y_j = y_j(z_j) = \{1 + \xi_j(z_j - \mu_j)/\sigma_j\}_+^{-1/\xi_j} \quad (10)$$

for $j = 1, 2$. Independence* is obtained when $\alpha = 1$, $r \downarrow 0$ or $\lambda \downarrow 0$. Complete dependence† is obtained when $\alpha \downarrow 0$, $r \rightarrow \infty$ or $\lambda \rightarrow \infty$.

The distributions functions (8) and (9) have asymmetric extensions, given by

$$G(z_1, z_2) = \exp \left\{ -(1 - \theta_1)y_1 - (1 - \theta_2)y_2 - [(\theta_1 y_1)^{1/\alpha} + (\theta_2 y_2)^{1/\alpha}]^\alpha \right\}, \quad 0 < \alpha \leq 1,$$

$$G(z_1, z_2) = \exp \left\{ -y_1 - y_2 + [(\theta_1 y_1)^{-r} + (\theta_2 y_2)^{-r}]^{-1/r} \right\}, \quad r > 0,$$

known as the asymmetric logistic (Tawn, 1988) and asymmetric negative logistic (Joe, 1990) models respectively, where the asymmetry parameters $0 \leq \theta_1, \theta_2 \leq 1$. For the asymmetric logistic model independence is obtained when either $\alpha = 1$, $\theta_1 = 0$ or $\theta_2 = 0$. Different limits occur when θ_1 and θ_2 are fixed and $\alpha \downarrow 0$. For the asymmetric negative logistic model independence is obtained when either $r \downarrow 0$, $\theta_1 \downarrow 0$ or $\theta_2 \downarrow 0$. Different limits occur when θ_1 and θ_2 are fixed and $r \rightarrow \infty$.

The remaining five bivariate models are defined in Appendix A. Density, distribution and simulation functions for each of the nine models are provided by **dbvevd**, **pbvevd** and **rbvevd** respectively. The argument **model** denotes the specified model, which must be either "log" (the default), "alog", "hr", "neglog", "aneglog", "bilog", "negbilog", "ct" or "amix" (or any unique partial match). The first argument in **pbvevd** and **dbvevd** should be a vector of length two or a matrix with two columns, so that each row specifies a value for (z_1, z_2) .

The parameters of the specified model can be passed using one or more of the arguments **dep**, **asy**, **alpha** and **beta**. The marginal parameters (μ_1, σ_1, ξ_1) and (μ_2, σ_2, ξ_2) can be passed using the arguments **mar1** and **mar2** respectively. Gumbel marginal distributions are used by default.

*Independence occurs when $G(z_1, z_2) = \exp\{-(y_1 + y_2)\}$.

†Complete dependence occurs when $G(z_1, z_2) = \exp\{-\max(y_1, y_2)\}$.

The arguments `mar1` and `mar2` can also be matrices with three columns, in which case each column represents a vector of values to be passed to the corresponding marginal parameter. Some examples are given below.

```
> rbvevd(3, dep = .8, asy = c(.4, 1), model = "alog")
      [,1] [,2]
[1,] 0.07876 -0.7971
[2,] 0.01091 -0.8113
[3,] -0.10491 -0.8831

> rbvevd(3, alpha = .5, beta = 1.2, model = "negb", mar1 = rep(1, 3))
      [,1] [,2]
[1,] 0.7417 1.085
[2,] 0.8391 1.825
[3,] 2.0142 2.280

> pbvevd(c(1, 1.2), dep = .4, asy = c(.4, .6), model = "an", mar1 = rep(1, 3))
[1] 0.173
> tmp.quant <- matrix(c(1,1.2,1,2), ncol = 2, byrow = TRUE)
> tmp.mar <- matrix(c(1,1,1,1.2,1.2,1.2), ncol = 3, byrow = TRUE)
> pbvevd(tmp.quant, dep = .4, asy = c(.4, .6), model = "an", mar1 = tmp.mar)
[1] 0.173 0.175

> dbvevd(c(1, 1.2), alpha = .2, beta = .6, model = "ct", mar1 = rep(1, 3))
[1] 0.1213
> dbvevd(tmp.quant, alpha = 0.2, beta = 0.6, model = "ct", mar1 = tmp.mar)
[1] 0.1213 0.0586
```

4 Multivariate Extreme Value Distributions

Let $z = (z_1, \dots, z_d)$. The d -dimensional logistic model (Gumbel, 1960) has distribution function

$$G(z) = \exp \left\{ - \left(\sum_{j=1}^d y_j^{1/\alpha} \right)^\alpha \right\}$$

where $\alpha \in (0, 1]$ and (y_1, \dots, y_d) is defined by the transformations (10).

This distribution can be extended to an asymmetric form. Let B be the set of all non-empty subsets of $\{1, \dots, d\}$, let $B_1 = \{b \in B : |b| = 1\}$, where $|b|$ denotes the number of elements in the set b , and let $B_{(i)} = \{b \in B : i \in b\}$. The multivariate asymmetric logistic model (Tawn, 1990) is given by

$$G(z) = \exp \left\{ - \sum_{b \in B} \left[\sum_{i \in b} (\theta_{i,b} y_i)^{1/\alpha_b} \right]^{\alpha_b} \right\}$$

where the dependence parameters $\alpha_b \in (0, 1]$ for all $b \in B \setminus B_1$, and the asymmetry parameters $\theta_{i,b} \in [0, 1]$ for all $b \in B$ and $i \in b$. The constraints $\sum_{b \in B_{(i)}} \theta_{i,b} = 1$ for $i = 1, \dots, d$ ensure that the marginal distributions are GEV. There exists further constraints which arise from the possible redundancy of asymmetry parameters in the expansion of the distributional form. Specifically, if $\alpha_b = 1$ for some $b \in B \setminus B_1$ then $\theta_{i,b} = 0$ for all $i \in b$. Let $b_{-i_0} = \{i \in b : i \neq i_0\}$. If, for some $b \in B \setminus B_1$, $\theta_{i,b} = 0$ for all $i \in b_{-i_0}$, then $\theta_{i_0,b} = 0$.

Density, distribution and simulation functions for these models are provided by `dmvevd`, `pmvevd` and `rmvevd` respectively. The argument `model` denotes the specified model, which must be

either "log" (the default) or "alog" (or any unique partial match). The argument `d` denotes the dimension of the model. By default, `d = 2`. The first argument in `pbvevd` and `dbvevd` should be a vector of length `d` or a matrix with `d` columns, so that each row specifies a value for (z_1, \dots, z_d) . The marginal parameters (μ_i, σ_i, ξ_i) , for $i = 1, \dots, d$, can be passed using the argument `mar`. Gumbel marginal distributions are used by default. For the symmetric logistic model, the argument `dep` represents the parameter α . Some examples are given below.

```
> rmvevd(3, dep = .6, model = "log", d = 5)
      [,1] [,2] [,3] [,4] [,5]
[1,] 0.1335 0.2878 1.07886 1.55515 1.310
[2,] 1.7100 0.9453 1.02070 -0.02553 1.527
[3,] -0.3376 -0.5814 0.07426 0.10906 2.827

> tmp.mar <- matrix(c(1,1,1,1,1,1.5,1,1,2), ncol = 3, byrow = TRUE)
> rmvevd(3, dep = .6, d = 5, mar = tmp.mar)
      [,1] [,2] [,3] [,4] [,5]
[1,] 2.803 4.6415 1.8531 3.5569 8.854
[2,] 0.751 0.9704 2.3328 2.6537 1.233
[3,] 4.641 1.4321 0.5825 0.6041 2.021

> tmp.quant <- matrix(rep(c(1,1.5,2), 5), ncol = 5)
> pmvevd(tmp.quant, dep = .6, d = 5, mar = tmp.mar)
[1] 0.07233 0.16387 0.21949
> dmvevd(tmp.quant, dep = .6, d = 5, mar = tmp.mar, log = TRUE)
[1] -3.564 -6.610 -9.460
```

For the asymmetric logistic model `dep` should be a vector of length $2^d - d - 1$ containing the dependence parameters. For example, when `d = 4`

$$\text{dep} = c(\alpha_{12}, \alpha_{13}, \alpha_{14}, \alpha_{23}, \alpha_{24}, \alpha_{34}, \alpha_{123}, \alpha_{124}, \alpha_{134}, \alpha_{234}, \alpha_{1234}).$$

The asymmetry parameters should be passed to `asy` in a list with $2^d - 1$ elements, where each element is a vector (including vectors of length one) corresponding to a set $b \in B$, containing $\{\theta_{i,b} : i \in b\}$. For example, when `d = 4`

$$\begin{aligned} \text{asy} = \text{list}(\theta_{1,1}, \theta_{2,2}, \theta_{3,3}, \theta_{4,4}, c(\theta_{1,12}, \theta_{2,12}), c(\theta_{1,13}, \theta_{3,13}), c(\theta_{1,14}, \theta_{4,14}), c(\theta_{2,23}, \theta_{3,23}), \\ c(\theta_{2,24}, \theta_{4,24}), c(\theta_{3,34}, \theta_{4,34}), c(\theta_{1,123}, \theta_{2,123}, \theta_{3,123}), c(\theta_{1,124}, \theta_{2,124}, \theta_{4,124}), \\ c(\theta_{1,134}, \theta_{3,134}, \theta_{4,134}), c(\theta_{2,234}, \theta_{3,234}, \theta_{4,234}), c(\theta_{1,1234}, \theta_{2,1234}, \theta_{3,1234}, \theta_{4,1234})). \end{aligned}$$

All the constraints, including $\sum_{b \in B(i)} \theta_{i,b} = 1$ for $i = 1, \dots, d$, must be satisfied or an error will occur. Some examples are given below.

The dependence parameters used in the following trivariate asymmetric logistic model are $(\alpha_{12}, \alpha_{13}, \alpha_{23}, \alpha_{123}) = (.6, .5, .8, .3)$. The asymmetry parameters are $\theta_{1,1} = .4$, $\theta_{2,2} = 0$, $\theta_{3,3} = .6$, $(\theta_{1,12}, \theta_{2,12}) = (.3, .2)$, $(\theta_{1,13}, \theta_{3,13}) = (.1, .1)$, $(\theta_{2,23}, \theta_{3,23}) = (.4, .1)$ and finally $(\theta_{1,123}, \theta_{2,123}, \theta_{3,123}) = (.2, .4, .2)$.

```
> asy <- list(.4, 0, .6, c(.3,.2), c(.1,.1), c(.4,.1), c(.2,.4,.2))
> rmvevd(3, dep = c(.6,.5,.8,.3), asy = asy, model = "alog", d = 3)
      [,1] [,2] [,3]
[1,] 0.52375 -0.8844 1.4898
[2,] 1.16174 -0.4368 -0.7404
```

```
[3,] -0.03737 1.5139 -0.5996
```

```
> dmvevd(c(2, 2, 2), dep = c(.6,.5,.8,.3), asy = asy, model = "a", d = 3)
[1] 0.006636
```

```
> tmp.quant <- matrix(rep(c(1,1.5,2), 3), ncol = 3)
> pmvevd(tmp.quant, dep = c(.6,.5,.8,.3), asy = asy, model = "a", d = 3)
[1] 0.4131 0.5849 0.7223
```

The dependence parameters used in the following four dimensional asymmetric logistic model are $\alpha_b = 1$ for $|b| = 2^*$ and $(\alpha_{123}, \alpha_{124}, \alpha_{134}, \alpha_{234}, \alpha_{1234}) = (.7, .3, .8, .7, .5)$. The asymmetry parameters are $\theta_{i,b} = 0$ for all $i \in b$ when $|b| \leq 2$, $(\theta_{1,123}, \theta_{2,123}, \theta_{3,123}) = (.2, .1, .2)$, $(\theta_{1,124}, \theta_{2,124}, \theta_{4,124}) = (.1, .1, .2)$, $(\theta_{1,134}, \theta_{3,134}, \theta_{4,134}) = (.3, .4, .1)$, $(\theta_{2,234}, \theta_{3,234}, \theta_{4,234}) = (.2, .2, .2)$ and finally $(\theta_{1,1234}, \theta_{2,1234}, \theta_{3,1234}, \theta_{4,1234}) = (.4, .6, .2, .5)$.

```
> asy <- list(0, 0, 0, 0, c(0,0), c(0,0), c(0,0), c(0,0), c(0,0), c(0,0),
  c(.2,.1,.2), c(.1,.1,.2), c(.3,.4,.1), c(.2,.2,.2), c(.4,.6,.2,.5))
> rmvevd(3, dep = c(rep(1,6),.7,.3,.8,.7,.5), asy = asy, model = "alog", d = 4)
      [,1]      [,2]      [,3]      [,4]
[1,] -0.5930 -0.1916 1.0211 0.6113
[2,] 4.3522 -1.0050 2.3618 -0.1875
[3,] 0.5805 0.4443 -0.5958 0.9717
```

5 Dependence Functions

Let $z = (z_1, \dots, z_d)$ and $\omega = (\omega_1, \dots, \omega_d)$. Any d -dimensional extreme value distribution function can be represented in the form

$$G(z) = \exp \left\{ - \left\{ \sum_{j=1}^d y_j \right\} A \left(\frac{y_1}{\sum_{j=1}^d y_j}, \dots, \frac{y_d}{\sum_{j=1}^d y_j} \right) \right\},$$

where (y_1, \dots, y_d) is defined by the transformations (10). It follows that $A(\omega) = -\log\{G(y_1^{-1}(\omega_1), \dots, y_d^{-1}(\omega_d))\}$, defined on the simplex $S_d = \{\omega \in \mathbb{R}_+^d : \sum_{j=1}^d \omega_j = 1\}$. $A(\cdot)$ is known as the dependence function. The dependence function characterises the dependence structure of G . It can be shown that $A(\omega) = 1$ when ω is one of the d vertices of S_d (i.e. when one component of ω is equal to one, and all remaining components are equal to zero), and that A is a convex function with $\max(\omega_1, \dots, \omega_d) \leq A(\omega) \leq 1$ for all $\omega \in S_d$. The lower and upper bounds are obtained at complete dependence and mutual independence respectively. In particular, $A(1/d, \dots, 1/d)$ is equal to $1/d$ at complete dependence, and 1 at mutual independence.

The dependence function of a *bivariate* extreme value distribution is a special case (because the sets S_2 and $[0,1]$ are equivalent), and is typically defined as follows. Any bivariate extreme value distribution function can be represented in the form

$$G(z_1, z_2) = \exp \left\{ -(y_1 + y_2) A \left(\frac{y_1}{y_1 + y_2} \right) \right\}, \quad (11)$$

so that $A(\omega) = -\log\{G(y_1^{-1}(\omega), y_2^{-1}(1 - \omega))\}$, defined on $0 \leq \omega \leq 1$.^{*} It follows that $A(0) = A(1) = 1$, and that $A(\cdot)$ is a convex function with $\max(\omega, 1 - \omega) \leq A(\omega) \leq 1$ for all $0 \leq \omega \leq 1$. At independence $A(1/2) = 1$. At complete dependence $A(1/2) = 0.5$.

^{*}The values taken by α_b when $|b| = 2$ are irrelevant here because $\theta_{i,b} = 0$ for all $i \in b$ when $|b| = 2$.

^{*}Some authors (e.g. Pickands, 1981) use $A(\omega) = -\log\{G(y_1^{-1}(1 - \omega), y_2^{-1}(\omega))\}$.

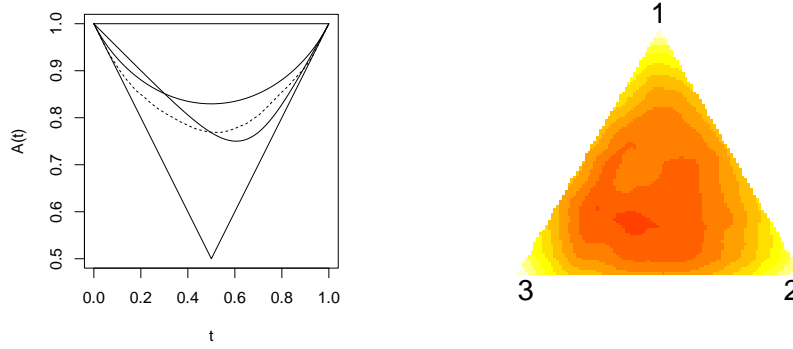


Figure 1: Left: Parametric (solid lines) and non-parametric (dashed line) dependence functions for bivariate distributions. The triangular border represents the constraint $\max(\omega, 1 - \omega) \leq A(\omega) \leq 1$ for all $\omega \in [0, 1]$. Right: non-parametric dependence function for a trivariate distribution. Darker colours depict smaller values, and hence stronger dependence.

Dependence functions for parametric bivariate and trivariate extreme value models can be calculated and plotted, at given parameter values, using the functions `abvevd` and `amvevd`. Non-parametric estimators of dependence functions can also be calculated and plotted, using the functions `abvnonpar` and `amvnonpar`. Some examples are given below. The last lines of code produce Figure 1.

```
> bvlsim <- rmvevd(100, dep = 0.6, model = "log", d = 2)
> tvlsim <- rmvevd(100, dep = 0.6, model = "log", d = 3)

> abvevd(seq(0,1,0.25), dep = 0.3, asy = c(.7,.9), model = "alog")
[1] 1.0000 0.8272 0.7013 0.7842 1.0000
> abvnonpar(seq(0,1,0.25), data = bvlsim)
[1] 1.0000 0.8634 0.8158 0.8392 1.0000

> abvnonpar(data = bvlsim, plot = TRUE, bty = 1, lty = 2)
> abvevd(dep = .3, asy = c(.5, .9), model = "al", add = TRUE)
> abvevd(dep = 1.05, model = "hr", add = TRUE)
> amvnonpar(data = tvlsim, plot = TRUE, lower = 0.6)
```

6 Stochastic Processes

The `evd` package contains four functions that simulate from stochastic processes associated with extreme value theory. The functions `marma`, `mar` and `mma` generate max autoregressive moving average processes, and the function `evmc` generates Markov chains with extreme value dependence structures. The function `clusters` identifies extreme clusters of a stochastic process, and `exi` estimates a quantity known as the Extremal Index.

A max autoregressive moving average process $\{X_k\}$, denoted by $\text{MARMA}(p, q)$, satisfies

$$X_k = \max\{\phi_1 X_{k-1}, \dots, \phi_p X_{k-p}, \epsilon_k, \theta_1 \epsilon_{k-1}, \dots, \theta_q \epsilon_{k-q}\}$$

where (ϕ_1, \dots, ϕ_p) and $(\theta_1, \dots, \theta_p)$ are vectors of non-negative parameters, and $\{\epsilon_k\}$ is a series of *iid* random variables with a common distribution defined by the argument `rand.gen`. The standard Fréchet distribution is used by default. A max autoregressive process $\{X_k\}$, denoted by $\text{MAR}(p)$, is equivalent to a $\text{MARMA}(p, 0)$ process, so that

$$X_k = \max\{\phi_1 X_{k-1}, \dots, \phi_p X_{k-p}, \epsilon_k\}.$$

A max moving average process $\{X_k\}$, denoted by $\text{MMA}(q)$, is equivalent to a $\text{MARMA}(0, q)$ process, so that

$$X_k = \max\{\epsilon_k, \theta_1 \epsilon_{k-1}, \dots, \theta_q \epsilon_{k-q}\}.$$

The functions `mar`, `mma` and `marma` generate $\text{MAR}(p)$, $\text{MMA}(q)$ and $\text{MARMA}(p, q)$ processes respectively. Examples of calls to these functions are given below. The `n.start` argument denotes the burn-in period, which can be specified so that the output series is not unduly influenced by the p starting values, which are all zero by default.

```
> marma(100, p = 1, q = 1, psi = 0.75, theta = 0.65)
> mar(100, psi = 0.85, n.start = 20)
> mma(100, q = 2, theta = c(0.75, 0.8))
```

The function `evmc` generates first order Markov chains. Informally, a first order Markov chain X_1, \dots, X_n is a stochastic process such that at any given time t the probability distribution of X_{t+1} is independent the past X_1, \dots, X_{t-1} , given the current state X_t . The `evmc` function generates a first order Markov chain such that each pair of consecutive values has the dependence structure of a parametric bivariate extreme value model. The main arguments of `evmc` are the same as those of `rbvevd`. The function `evmc` also has the argument `margin`, which denotes the marginal distribution of each value. This must be either `"uniform"` (the default), `"rweibull"`, `"frechet"` or `"gumbel"` (or any unique partial match), for the uniform, standard reversed Weibull, standard Gumbel and standard Fréchet distributions respectively. Examples of calls to `evmc` are given below.

```
> evmc(100, alpha = 0.1, beta = 0.1, model = "bilog")
> evmc(100, dep = 10, model = "hr", margins = "gum")
```

The function `clusters` identifies extreme clusters within (stationary) stochastic processes. A simple way of determining clusters is to specify a threshold u and define consecutive exceedances of u to belong to the same cluster. It is more common though to consider a cluster to be active until r consecutive values fall below (or are equal to) u , for some given clustering interval length r . The following code uses `clusters` to generate the plots depicted in Figure 2. These plots identify clusters graphically. If the argument `plot` is `FALSE` (the default), then `clusters` returns a list of extreme clusters.

```
> set.seed(150)
> x <- evmc(50, dep = 0.55, model = "log")
> clusters(x, 0.8, plot = TRUE)
> clusters(x, 0.8, 4, plot = TRUE)
```

The function `exi` returns estimates of the Extremal Index of a (stationary) stochastic process. The Extremal Index is defined in Chapter 3 of Leadbetter *et al.* (1983). A more informal treatment is given in Chapter 5 of Coles (2001). The extremal index can be estimated using the inverse of the average size of extreme clusters, where the cluster size is defined as the number of exceedances that it contains.

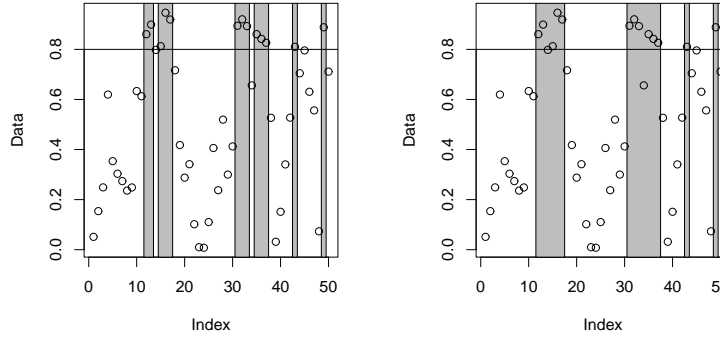


Figure 2: The identification of extreme clusters in a stochastic process. The clustering interval lengths are $r = 1$ (left) and $r = 4$ (right). The threshold in each case is $u = 0.8$.

7 Fitting Univariate Distributions

This section presents functions that produce maximum likelihood estimates for some of the distributions introduced in Section 2. Peaks over threshold models are discussed in Section 8. Maximum likelihood estimates for bivariate extreme value distributions are discussed in Section 9. For illustrative purposes Sections 7, 8 and 9 use only simulated data. Three practical examples using the data sets `oxford`, `rain` and `sealevel` are given in Sections 10, 11 and 12 respectively.

The function `fgev` produces maximum likelihood estimates for the GEV distribution (4). The first argument should be a numeric vector containing data to be fitted. Missing values are allowed. If the argument `start` is given it should be a named list containing starting values, the names of which should be the parameters over which the likelihood is to be maximised. If `start` is omitted the routine attempts to find good starting values for the optimisation using moment estimators.

If any of the parameters are to be set to fixed values, they can be given as separate arguments. For example, the Gumbel distribution (1) can be fitted using `shape = 0`. Arguments of the optimisation function `optim` can also be specified. This includes the optimisation method, which can be passed using the argument `method`. Two examples of the `fgev` function are given below.

```
> data1 <- rgev(1000, loc = 0.13, scale = 1.1, shape = 0.2)
```

```
> m1 <- fgev(data1)
> m1
```

```
Call: fgev(x = data1)
Deviance: 3650
```

```
Estimates
   loc  scale  shape
0.127  1.125  0.224
```

```
Standard Errors
   loc  scale  shape
0.0400 0.0321 0.0248
```

```

Optimization Information
  Convergence: successful
  Function Evaluations: 51
  Gradient Evaluations: 12

```

```

> m2 <- fgev(data1, loc = 0, scale = 1)
> fitted(m2)
shape
0.236

```

In the first example the likelihood is maximised over (**loc**, **scale**, **shape**). In the second example the likelihood is maximised over **shape**, with the location and scale parameters fixed at zero and one respectively.

The maximum likelihood estimators do not necessarily have the usual asymptotic properties, since the end points of the GEV distribution depend on the model parameters. Smith (1985) shows that the usual asymptotic properties hold when $\xi > -0.5$. When $-1 < \xi \leq -0.5$ the maximum likelihood estimators do not have the standard asymptotic properties, but generally exist. When $\xi \leq -1$ maximum likelihood estimators do not often exist. This occurs because of the large mass near the upper end point. The likelihood increases without bound as the upper end point is estimated to be closer and closer to the largest observed value. In terms of the reversed Weibull shape parameter α , the usual asymptotic properties hold when $\alpha > 2$, the asymptotic properties are not standard for $1 < \alpha \leq 2$, and maximum likelihood estimators do not often exist for $\alpha < 1$.

When the usual asymptotic properties hold (as here) the standard errors of the maximum likelihood estimates, approximated using the inverse of the observed information matrix, can be extracted from the fitted object using

```

> std.errors(m1)
      loc      scale      shape
0.03999 0.03214 0.02479

```

Likelihood ratio tests can be performed using the function **anova**. We can compare the two models **m1** and **m2** to test the null hypothesis that the location parameter is zero and the scale parameter is one.

```

> anova(m1, m2)
Analysis of Deviance Table

    M.Df Deviance Df  Chisq Pr(>chisq)
m1      3      3650
m2      1      3669  2  18.8    8.2e-05

```

The deviance difference, **deviance(m2)** minus **deviance(m1)**, is about 18.8, which yields a p-value of 8.2×10^{-5} when compared with a chi-squared distribution on two degrees of freedom. Diagnostic plots and profile traces for fitted models can be constructed using the functions **plot**, **profile** and **profile2d** (see Section 10).

By default the maximum likelihood estimates are calculated under the assumption that the data to be fitted are the observed values of independent random variables Z_1, \dots, Z_n , where $Z_i \sim \text{GEV}(\mu, \sigma, \xi)$ for each $i = 1, \dots, n$. The **nsloc** argument allows non-stationary models of the form $Z_i \sim \text{GEV}(\mu_i, \sigma, \xi)$, where

$$\mu_i = \beta_0 + \beta_1 x_{i1} + \dots + \beta_k x_{ik}.$$

The parameters $(\beta_0, \dots, \beta_k)$ are to be estimated. In matrix notation $\boldsymbol{\mu} = \boldsymbol{\beta}_0 + X\boldsymbol{\beta}$, where $\boldsymbol{\mu} = (\mu_1, \dots, \mu_n)^T$, $\boldsymbol{\beta}_0 = (\beta_0, \dots, \beta_0)^T$, $\boldsymbol{\beta} = (\beta_1, \dots, \beta_k)^T$ and X is the $n \times k$ covariate matrix (excluding the intercept) with ij th element x_{ij} .

The `nsloc` argument must be a data frame containing the matrix X , or a numeric vector which is converted into a single column data frame with column name “trend”. The column names of the data frame are used to derive names for the estimated parameters. This allows any of the $k + 3$ parameters $(\beta_0, \dots, \beta_k, \sigma, \xi)$ to be set to fixed values within the optimisation. The covariates must be (at least approximately) *centred and scaled*, not only for numerical reasons, but also because the starting value (if `start` is not given) for each corresponding coefficient is taken to be zero. When a linear trend is present in the data, the location parameter is often modelled as

$$\mu_i = \beta_0 + \beta_1 t_i,$$

where t_i is some centred and scaled version of the time of the i th observation. More complex changes in μ may also be appropriate. For example, a change-point model

$$\mu_i = \beta_0 + \beta_1 x_i \quad \text{where} \quad x_i = \begin{cases} 0 & i \leq i_0 \\ 1 & i > i_0 \end{cases},$$

or a quadratic trend

$$\mu_i = \beta_0 + \beta_1 t_i + \beta_2 t_i^2.$$

See Sections 10 and 12 for examples of non-stationary modelling.

The function `fgev` also has an argument called `prob`. If `prob = p` is passed a value in the interval $[0,1]$, `fgev` again produces maximum likelihood estimates for the GEV distribution, but the model is re-parameterised from (μ, σ, ξ) to (z_p, σ, ξ) , where z_p is the quantile corresponding to the upper tail probability p . This argument can be used to calculate and plot profile log-likelihoods of extreme quantiles (see Section 10). If `prob` is zero/one, then z_p is defined as the upper/lower end point $\mu - \sigma/\xi$, and ξ is restricted to the negative/positive axis. Under non-stationarity the model is re-parameterised from $(\beta_0, \beta_1, \dots, \beta_k, \sigma, \xi)$ to $(z_p, \beta_1, \dots, \beta_k, \sigma, \xi)$, so that z_p is the quantile corresponding to the upper tail probability p for the distribution obtained when all covariates are zero.

The `fextreme` function produces maximum likelihood estimates for the distributions (5) and (6) given an integer m and an arbitrary distribution function F . The first argument should be a numeric vector containing the data to be fitted, which should represent maxima (if the argument `largest` is `TRUE`, the default) or minima (if `largest` is `FALSE`). The argument `start` (which cannot be missing) should be a named list containing starting values, the names of which should be the parameters over which the likelihood is to be maximised. If any of the parameters are to be set to fixed values, they can be given as separate arguments. Arguments of the optimisation function `optim` can also be specified. The example given below produces maximum likelihood estimates for the distribution (5), where $m = 365$ and F is the normal distribution.

```
> d2 <- rextreme(100, distn = "norm", mean = 0.56, mlen = 365)
# Simulate yearly maxima using normal distribution

> sv <- list(mean = 0, sd = 1)
> nm <- fextreme(d2, start = sv, distn = "norm", mlen = 365)
> fitted(nm)
  mean    sd
0.685 0.959
```

The **forder** function yields maximum likelihood estimates for the distribution (7) given integers m and $j \in \{1, \dots, m\}$, and an arbitrary distribution function F . An example is given below, where $m = 365$, $j = 2$ and F is the normal distribution.

```
> d3 <- rorder(100, distn = "norm", mean = 0.56, mlen = 365, j = 2)
> sv <- list(mean = 0, sd = 1)
> nm2 <- forder(d3, sv, distn = "norm", mlen = 365, j = 2)
> fitted(nm2)
      mean      sd
0.483 1.042
```

8 Fitting Peaks Over Threshold Models

Suppose X_1, \dots, X_n is a sequence of independent and identically distributed random variables, with $M_n = \{X_1, \dots, X_n\}$. Suppose that n is large, so that (assuming certain regularity conditions) the distribution of M_n is approximately GEV. Then for large enough u , the exceedances of the threshold u are approximately distributed as generalised Pareto, with location parameter u . The function **fpot** fits this distribution to the exceedances, and hence produces maximum likelihood estimates for the shape and scale parameters. The value of the threshold u must be specified by the user. It is typically chosen to be as small as possible, subject to the limit model providing a reasonable approximation.

The functions **mrlplot** and **tcplot*** produce diagnostic plots that facilitate the specification of u . The function **mrlplot** produces the empirical mean residual life plot, which is a plot of the empirical mean of the excesses of u (i.e. the exceedances of u minus u), plotted against u . If the exceedances of a threshold u_0 are generalised Pareto, the empirical mean residual life plot should be approximately linear for all $u > u_0$. The function **tcplot** calculates maximum likelihood estimates for the shape and modified scale parameters using a number of different thresholds, and plots these estimates against u . If the exceedances of a threshold u_0 are generalised Pareto, the shape and modified scale parameters should be approximately constant with respect to all thresholds $u > u_0$. Threshold identification plots produced from the example given below are depicted in Figure 3. In this case, the threshold $u = 1$ was chosen.

The following code generates $n = 500$ independent standard normal random variables and fits the (generalised Pareto) peaks over threshold model to the exceedances of the threshold $u = 1$. The function **fpot** performs the fit. Many of the arguments of **fpot** are similar to those of **fgev**. In particular, either of the **scale** or **shape** parameters can be set to fixed values by giving those parameters as arguments. For example, an exponential distribution for the excesses (or equivalently, a shifted exponential distribution for the exceedances) can be fitted using **shape = 0**.

```
> tmp <- rnorm(500)
> mrlplot(tmp, tlim = c(-1,1.5))
> tcplot(tmp, tlim = c(-1,1.5))

> pot1 <- fpot(tmp, 1)
> pot1
```

```
Call: fpot(x = tmp, threshold = 1)
Deviance: 40.5
```

*Both of these functions are heavily based on code by Stuart Coles.

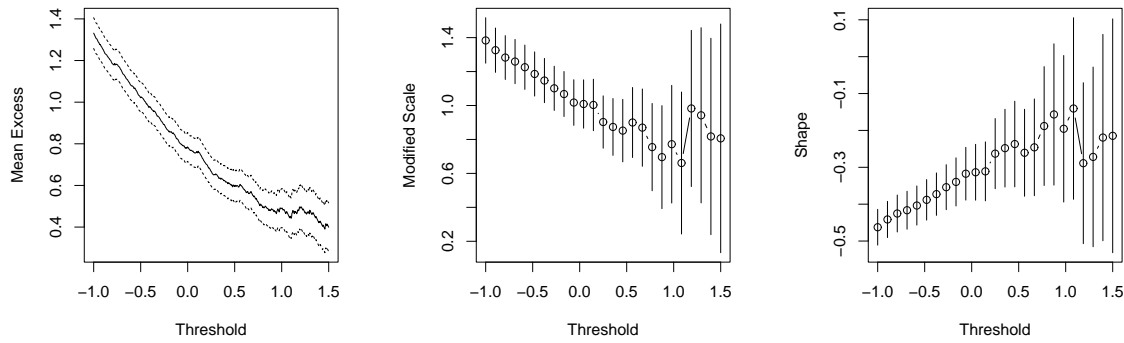


Figure 3: The identification of a threshold for the (generalised Pareto) peaks over threshold model. From left to right; the empirical mean residual life plot, modified scale parameter estimates and shape parameter estimates.

```
Threshold: 1
Number Above: 76
Proportion Above: 0.152
```

```
Estimates
  scale  shape
0.593 -0.211
```

The fitted model `pot1` gives the estimates for the scale and shape parameters of the generalised Pareto distribution fitted to the exceedances. Also given is the proportion of values above the threshold, or equivalently, the maximum likelihood estimate for the probability of an exceedance. Diagnostic plots and profile traces for fitted models can be constructed using the functions `plot` and `profile` (see Section 11).

The peaks over thresholds model is typically extended to stationary series via declustering, which corresponds to a filtering of dependent observations to obtain a set of threshold exceedances which are approximately independent. An empirical rule is used to identify clusters of exceedances, and the generalised Pareto model is then fitted to the cluster maxima, assuming those maxima to be independent. The empirical rule, as given in Section 6, is defined by the function `clusters`. A model of this form can be implemented by setting the logical argument `cmax` to `TRUE`. The clusters are identified using the threshold of the peaks over threshold model. An illustration of this technique is given below. The argument `r` is the clustering interval length.

```
> tmp2 <- evmc(500, dep = 0.8, margins = "gum")
> pot2 <- fpot(tmp2, 1.5, cmax = TRUE, r = 3)
> pot2
```

```
Call: fpot(x = tmp, threshold = 1, cmax = TRUE, r = 3)
Deviance: 101.1
```

```
Threshold: 1.5
Number Above: 92
Proportion Above: 0.184
```

```

Clustering Interval: 3
Number of Clusters: 41
Extremal Index: 0.446

```

Estimates

```

scale  shape
1.657  -0.272

```

The Extremal Index is a quantity briefly discussed in Section 6. The estimate of the Extremal Index is simply the number of clusters divided by the number of exceedances.

The function `fpot` also has an argument called `mper`. If `mper = m` is passed a positive value, `fpot` again produces maximum likelihood estimates for the generalised Pareto model, but the model is re-parameterised from (σ, ξ) to (z_m, ξ) , where z_m is the m -period return level, defined as follows. Let G be the fitted generalised Pareto distribution function, with location parameter equal to the specified threshold u , so that $1 - G(z)$ is the fitted probability of an exceedance over $z > u$ given an exceedance over u . The fitted probability of an exceedance over $z > u$ is therefore $p(1 - G(z))$, where p is the estimated probability of exceeding u , which is given by the empirical proportion of exceedances. The m -period return level z_m satisfies $p(1 - G(z_m)) = 1/(mN\hat{\theta})$, where N is the number of observations per period, and $\hat{\theta}$ is the estimate of the extremal index if cluster maxima are fitted, with $\hat{\theta} = 1$ otherwise. The value N can be specified using the argument `npp`. For example, if observations are recorded weekly and `npp = 52`, then z_m is the m -year return level. If `mper` is `Inf`, then z_m is defined as the upper end point $u - \sigma/\xi$, and ξ is then restricted to be negative. The argument `mper` can be used to calculate and plot profile log-likelihoods of return levels (see Section 11).

9 Fitting Bivariate Extreme Value Distributions

The function `fbvevd` produces maximum likelihood estimates for nine bivariate extreme value models. The first argument should be a numeric matrix (or a data frame) with two columns containing the data to be fitted. Missing values are allowed. If the argument `start` is given it should be a named list containing starting values, the names of which should be the parameters over which the likelihood is to be maximised. If `start` is omitted the routine attempts to find good starting values for the optimisation using maximum likelihood estimators under the assumption of independence. If any of the parameters are to be set to fixed values, they can be given as separate arguments. Common marginal parameters can be fitted using the arguments `cshape`, `cshape` and `cloc`, and the dependence function can be constrained to symmetry using the argument `sym` (see the `fbvevd` help file for details).

The `nsloc1` and `nsloc2` arguments allow non-stationary modelling of the location parameters on the first and second margins respectively. They should be used in the same manner as the `nsloc` argument of `fgev`. Examples of bivariate models with non-stationary margins are given in Section 12.

The first example given below produces maximum likelihood estimates for the (symmetric) logistic model. The second example constrains the model at independence (where `dep = 1`). The estimates produced in the second example are the same as those that would be produced if `fgev` was separately applied to each margin.

```
> bvdata <- rbvevd(100, dep = 0.6, mar1 = c(1.2,1.4,0), mar2 = c(1,1.6,0.1))
```

```

> m1 <- fbvevd(bvdata, model = "log")
> m1

Call: fbvevd(x = bvdata, model = "log")
Deviance: 728.5
AIC: 742.5
Dependence: 0.3526

Estimates
      loc1  scale1  shape1      loc2  scale2  shape2      dep
1.2121  1.3831 -0.1813   0.8404  1.4005  0.0834  0.7202

Standard Errors
      loc1  scale1  shape1      loc2  scale2  shape2      dep
0.1540  0.1091  0.0673   0.1537  0.1144  0.0614  0.0624

Optimization Information
Convergence: successful
Function Evaluations: 47
Gradient Evaluations: 10

> m2 <- fbvevd(bvdata, model = "log", dep = 1)

> fitted(m2)
      loc1  scale1  shape1      loc2  scale2  shape2
1.2231  1.3776 -0.1914   0.8367  1.4083  0.0868

> std.errors(m2)
      loc1  scale1  shape1      loc2  scale2  shape2
0.1543  0.1089  0.0725   0.1565  0.1163  0.0670

> c(logLik(m2), deviance(m2), AIC(m2))
[1] -376  752  764

```

The discussion in Section 7 regarding the properties of maximum likelihood estimators for the GEV distribution also applies to all bivariate models. The usual asymptotic properties will not hold if either of the marginal shape parameters are less than -0.5 . The value in the output labelled **Dependence** is the fitted estimate of $\chi = 2\{1 - A(1/2)\} \in [0, 1]$ (Coles *et al.*, 1999), where $A(\cdot)$ denotes the dependence function (11). At independence $\chi = 0$, and at complete dependence $\chi = 1$.

Diagnostic plots and profile traces for fitted models can be constructed using the functions **plot**, **profile** and **profile2d** (see Section 12). The function **anova** performs likelihood ratio tests. The null hypothesis of the test performed below specifies that the margins are Gumbel distributions (**shape1** = **shape2** = 0). The deviance of the constrained model is compared with the deviance of the unconstrained model, and the p-value is calculated to be 0.78. The hypothesis would not be rejected at any reasonable significance level.

```

> m3 <- fbvevd(bvdata, model = "log", shape1 = 0, shape2 = 0)
> anova(m1, m3)
Analysis of Deviance Table

```

```

      M.Df Deviance Df Chisq Pr(>chisq)
m1      7      708
m3      5      708 2    0.5      0.78

```

In the following example I attempt to fit the asymmetric logistic model to the simulated data set used above, which is known to be distributed as symmetric logistic.

```

> m4 <- fbvevd(bvdata, model = "alog")
> fitted(m4)
      loc1 scale1 shape1 loc2 scale2 shape2 asy1 asy2 dep
1.2097  1.3928 -0.1853 0.8421  1.3831  0.0773  0.8331 0.9996 0.6925

```

A boundary of the parameter space has been reached; the maximum likelihood estimate for the second asymmetry parameter is one. This may cause difficulties for the optimiser. There are two solutions to this problem: the second asymmetry parameter can be fixed at one, or the **L-BFGS-B** method can be used. The **L-BFGS-B** method allows box-constraints using the arguments **lower** and **upper**. The following snippet illustrates both approaches.

```

> mb <- fbvevd(bvdata, model = "alog", asy2 = 1)
> round(fitted(mb), 3)
      loc1 scale1 shape1 loc2 scale2 shape2 asy1 dep
1.212  1.385 -0.176 0.834  1.396  0.086  0.867 0.693

> up <- c(rep(Inf, 6), 1, 1, 1)
> mb <- fbvevd(bvdata, model = "alog", method = "L-BFGS-B", upper = up)
> round(fitted(mb), 3)
      loc1 scale1 shape1 loc2 scale2 shape2 asy1 asy2 dep
1.212  1.385 -0.176 0.834  1.396  0.086  0.867 1.000 0.693

```

10 Example: Oxford Temperature Data

The numeric vector `oxford` contains annual maximum temperatures (in degrees Fahrenheit) at Oxford, England, from 1901 to 1980. It is included in the `evd` package, and can be made available using `data(oxford)`. The data has previously been analysed by Tabony (1983). I begin by plotting the data. The assumptions of stationarity and independence seem sensible, given the plot (not shown) generated using the code below.

```

> data(oxford) ; ox <- oxford
> plot(1901:1980, ox, xlab = "year", ylab = "temperature")

```

The following code fits two models based on the GEV distribution. The first model assumes stationarity. The second model allows for a trend term in the location parameter (even though the plot appears to show that this is unnecessary). The `nsloc` argument is centred and scaled so that the intercept `loc` represents the location parameter in 1950 and the trend `loctrend` represents the increase in the location parameter (or decrease, if negative) over a period of 100 years.

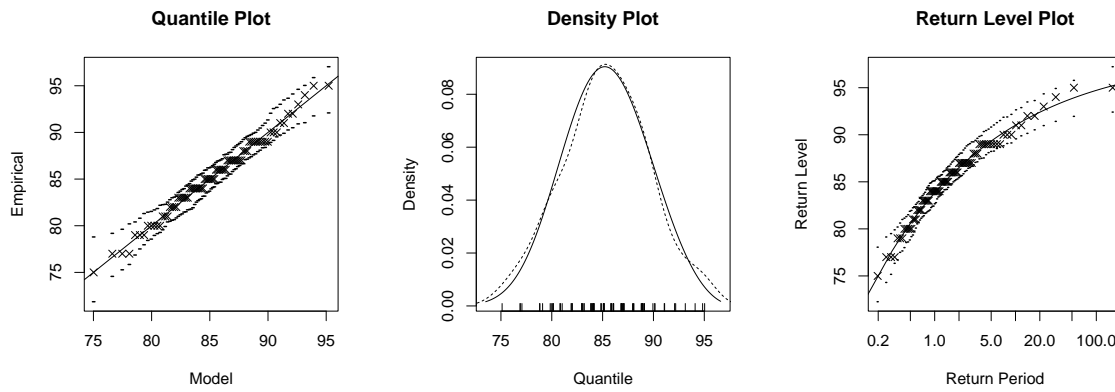


Figure 4: Diagnostic plots for the model `ox.fit`.

```
> ox.fit <- fgev(ox)

> tt <- (1901:1980 - 1950)/100
> ox.fit.trend <- fgev(ox, nsloc = tt)

> fitted(ox.fit.trend)
      loc loctrend   scale   shape
83.6617  -1.8812   4.2233  -0.2841

> std.errors(ox.fit.trend)
      loc loctrend   scale   shape
0.5557   1.9675   0.3650   0.0707
```

The trend term not statistically significant (at any reasonable level). The stationary model `ox.fit` is retained for further analysis.

```
> ox.fit

Call: fgev(x = oxford)
Deviance: 457.8

Estimates
   loc   scale  shape
83.839  4.260 -0.287

Standard Errors
   loc   scale  shape
0.5231 0.3658 0.0683
```

The fitted shape is negative, so the fitted distribution is Weibull. It is often of interest to test the hypothesis that the shape is zero (the Gumbel distribution). A 95% confidence interval for the shape parameter can be constructed using $-0.287 \pm 1.96 \times 0.0683$. The corresponding Wald test can be performed by dividing the maximum likelihood estimate by its standard error. The Wald test would be rejected at significance level 0.05 since the 95% confidence interval does not contain zero. A likelihood ratio test is performed in the following snippet. The hypothesis is rejected at any significance level above 0.00053.

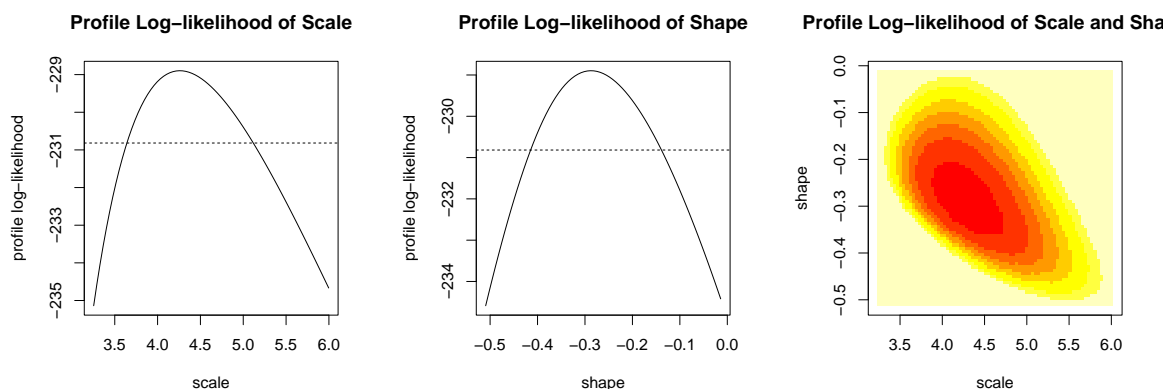


Figure 5: Profile log-likelihoods for the model `ox.fit`.

```
> ox.fit.gum <- fgev(ox, shape = 0)
> anova(ox.fit, ox.fit.gum)
Analysis of Deviance Table
```

	M.Df	Deviance	Df	Chisq	Pr(>chisq)
ox.fit	3	458			
ox.fit.gum	2	470	1	12	0.00053

Diagnostic plots can be produced using `plot(ox.fit)`. The plots produced compare parametric distributions, densities and quantiles to their empirical counterparts (see the `plot.uvevd` help file for details). Selected diagnostics are depicted in Figure 4.

The small bars on the P-P, Q-Q and return level plots represent simulated (pointwise) 95% confidence intervals. The model `ox.prof` is seen to be a good fit. The fitted density is close to the non-parametric estimator, and most points lie within the confidence intervals. Profile log-likelihoods of the parameters can be plotted using

```
> ox.prof <- profile(ox.fit)
> pcis <- plot(ox.prof)
> pcis[["scale"]]
  lower upper
3.643  5.117
```

The profile log-likelihoods for the scale and shape parameters are the first two plots of Figure 5. A horizontal line is (optionally) drawn on each plot so that the intersection of the line with the profile log-likelihood yields a profile confidence interval, with (default) confidence coefficient 0.95. The end points of the intervals can be derived by assigning the expression `plot(ox.prof)` to an object, as above. The profile confidence intervals for the location and shape parameters are approximately the same as the intervals constructed using their standard errors, since the profile log-likelihoods are approximately symmetric. The profile log-likelihood for the scale parameter is asymmetric; both end points of the profile confidence interval (3.64, 5.12) are larger than the corresponding end points of the interval (3.54, 4.98), constructed using the standard error. The joint profile log-likelihood of the scale and shape parameters can be plotted using

```
> ox.prof2d <- profile2d(ox.fit, ox.prof, which = c("scale", "shape"))
> plot(ox.prof2d)
```

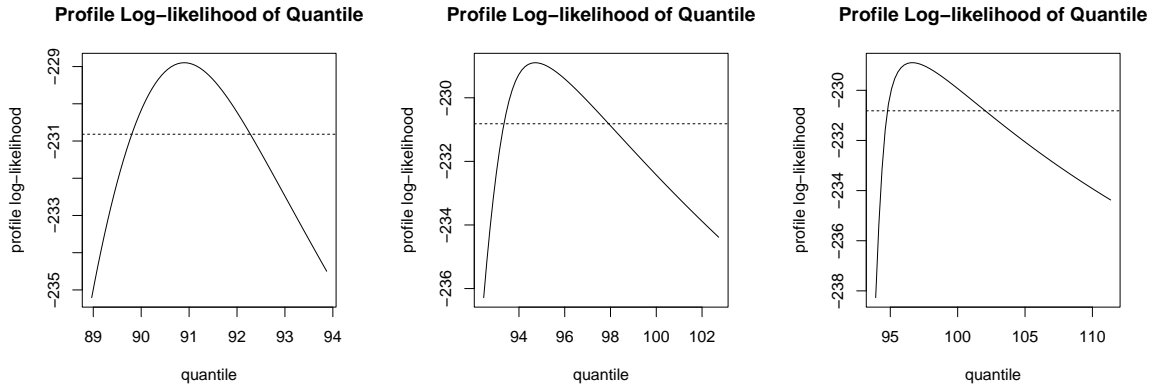


Figure 6: Profile log-likelihoods for $z_{0.1}$, $z_{0.01}$ and $z_{0.001}$.

This produces the image plot in the right panel of Figure 5. The colours of the image plot represent confidence sets with different confidence coefficients. By default, the lightest colour (ignoring the background colour) represents a confidence set with coefficient 0.995; the darkest colour represents a confidence set with coefficient 0.5.

Let G be the GEV distribution function, and let $G(z_p) = 1 - p$, so that

$$z_p = \begin{cases} \mu - \frac{\sigma}{\xi} [1 - \{-\log(1 - p)\}^{-\xi}] & \xi \neq 0 \\ \mu - \sigma \log\{-\log(1 - p)\} & \xi = 0, \end{cases}$$

is the quantile corresponding to the upper tail probability p . The profile log-likelihood for $z_{0.1}$ can be plotted using the following. The argument **prob** = p re-parameterises the GEV distribution so that **fgev** produces maximum likelihood estimates for (z_p, σ, ξ) .

```
> ox.qfit <- fgev(ox, prob = 0.1)
> ox.qprof <- profile(ox.qfit, which = "quantile")
> plot(ox.qprof)
```

Figure 6 shows profile log-likelihoods for $z_{0.1}$, $z_{0.01}$ and $z_{0.001}$. The extent of the asymmetry in the profile log-likelihood increases for decreasing (small) p . This is to be expected, since the data provide increasingly weaker information in the upper tail of the fitted distribution. If **prob** = p is zero, then z_p is the upper end point of the GEV distribution, given by $\mu - \sigma/\xi$ when $\xi < 0$. The profile log-likelihood for z_0 can be plotted using the following code.

```
> ox.qfit <- fgev(ox, prob = 0)
> ox.qprof <- profile(ox.qfit, which = "quantile", conf = 0.99)
> pcis <- plot(ox.qprof, ci = c(0.95, 0.99))
> pcis[["quantile"]]
      lower upper
0.95 95.78 113.0
0.99 95.49 131.8
```

The argument **conf** of the function **profile** controls the range of the profile trace. The profile trace is constructed so that profile confidence intervals with confidence coefficients **conf** or less can be derived from it. By default, **conf** = 0.999, though a smaller value is often appropriate when the profile log-likelihood exhibits strong asymmetry. The 95% and 99% profile confidence intervals for the upper end point z_0 are derived as (95.8,113.0) and (95.5,131.8) respectively.

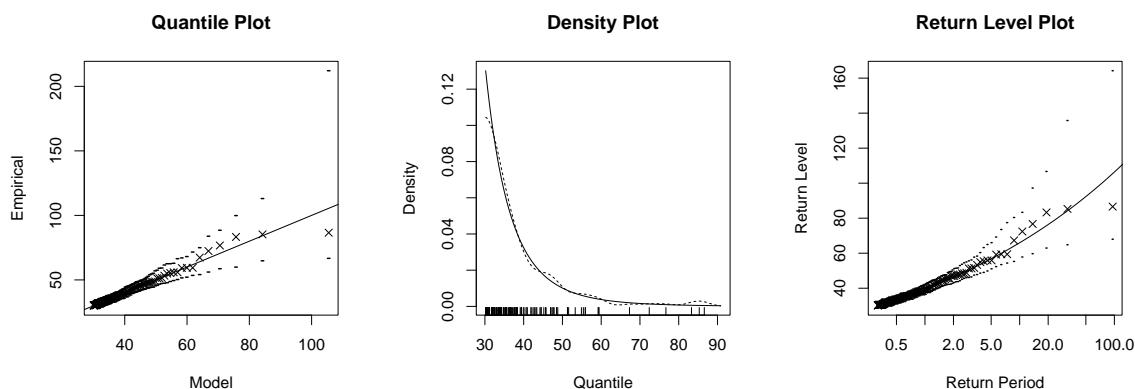


Figure 7: Diagnostic plots for the peaks over threshold model for daily rain data.

11 Example: Rainfall Data

The numeric vector `rain` contains 17531 daily rainfall accumulations at a location in south-west England, recorded over the period 1914 to 1962. The data is not included in the `evd` package, but it is available in the `ismev` package, which can be downloaded from CRAN. As usual, the package can be loaded using `library(ismev)`, and the data can be made available using `data(rain)`. The plot of the data given in Figure 1.7 of Coles (2001) shows that an assumption of stationarity is sensible. The example given here follows Coles (2001), pages 84–86.

```
> mrlplot(rain, tlim = c(0,85), nt = 100)
> par(mfrow = c(2,1))
> tcplot(rain, tlim = c(0,50), nt = 20)
> potgp <- fpot(rain, 30, npp = 365.25)
> potgp2 <- fpot(rain, 30, npp = 365.25, cmax = TRUE, r = 7)
> clusters(rain, 30, r = 7, cmax = TRUE)
```

The first three lines of code produce the threshold diagnostic plots given in pages 80 and 85 of Coles (2001), who subsequently decides to work with the threshold $u = 30$. The model `potgp` reports that 152 observations lie above the threshold, giving an exceedance probability estimate of 0.00867. The estimates and standard errors of the parameters of `potgp` agree with those given page 85 of Coles (2001). In `potgp2` the peaks over threshold model is applied to cluster maxima, where clusters are defined using a clustering interval length of seven. As there is little sign of clustering in the data, this leads to relatively small changes in the parameter estimates, and relatively small increases in the standard errors. The final line of code calls the function `clusters` (see Section 6) in order to produce the cluster maxima that were used for the fitting of model `potgp2`.

Diagnostic plots can be produced using `plot(potgp)`. The plots compare parametric distributions, densities and quantiles to their empirical counterparts (see the `plot.uvevd` help file for details). Selected diagnostics are given in Figure 7. The x-axis of the return level plot gives return periods in units of years, since we specified the number of observations per period as `npp = 365.25`. The small bars on the P-P, Q-Q and return level plots represent simulated (point-wise) 95% confidence intervals. The model `potgp` is seen to be a good fit. The fitted density tail is close to the non-parametric estimator, and most points lie within the confidence intervals.

Profile log-likelihoods of the shape parameter and the 100-year return level (not shown) can be plotted using the following code. The argument `mper = m` re-parameterises the model so that

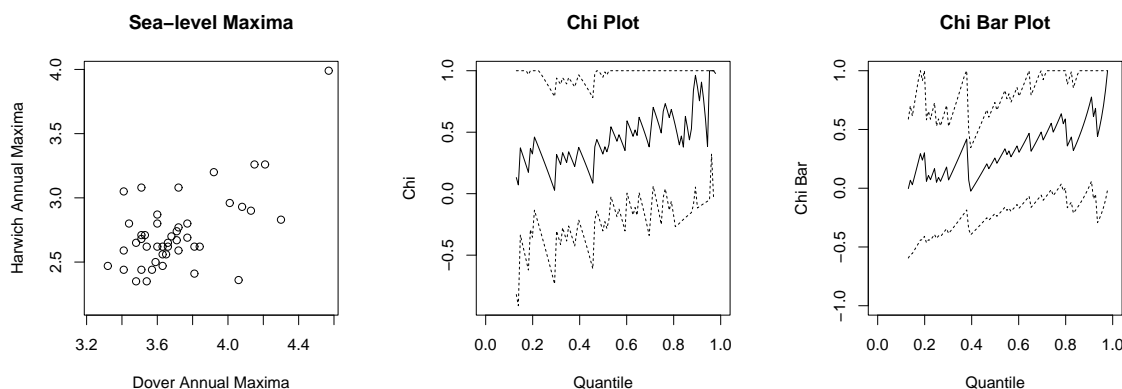


Figure 8: From left to right; Harwich maxima vs Dover maxima, estimated values of $\chi(u)$ vs u , estimated values of $\bar{\chi}(u)$ vs u .

`fpot` produces maximum likelihood estimates for (z_m, ξ) , where z_m is the m period return level, as defined in Section 8. Horizontal lines denoting 95% profile confidence intervals are depicted on each plot. The end points of profile confidence intervals can be derived by assigning the plotting expression to an object (see Section 10).

```
potgp3 <- fpot(rain, 30, npp = 365.25, mper = 100)
prgp3 <- profile(potgp3)
plot(prgp3)
```

12 Example: Sea Level Data

The `sealevel` data frame (Coles and Tawn, 1990) has two columns containing annual sea level maxima from 1912 to 1992 at Dover and Harwich, two sites on the coast of Britain. It contains 39 missing maxima in total; nine at Dover and thirty at Harwich. There are three years for which the annual maximum is not available at either site.

I begin by plotting the data, using the code below. The plot of the Harwich maxima against the Dover maxima, given in the left panel of Figure 8, depicts a reasonable degree of dependence. The outlier corresponds to the 1953 flood resulting from a storm passing over the South-East coast of Britain on 1st February. The marginal plots (not shown) suggest that the Harwich and Dover maxima both increase with time. The last line of code* plots estimates of $\chi(u)$ and $\bar{\chi}(u)$ for $0 < u < 1$ (Coles *et al.*, 1999), as depicted in Figure 8. For bivariate extreme value distributions, $\chi(u) = \chi$ is constant for all $0 < u < 1$, and $\lim_{u \rightarrow 1} \bar{\chi}(u) = 1$. The conditions do not seem unreasonable given the wide confidence intervals in each plot.

```
> data(sealevel) ; sl <- sealevel
> plot(sl, xlab = "Dover Annual Maxima", ylab = "Harwich Annual Maxima")
> plot(1912:1992, sl[,1], xlab = "Year", ylab = "Dover Annual Maxima")
> plot(1912:1992, sl[,2], xlab = "Year", ylab = "Harwich Annual Maxima")
> chiplot(sl)
```

*The function `chiplot` is heavily based on code by Jan Heffernan.

The following three expressions fit (symmetric) logistic models. The first model incorporates linear trend terms on both marginal location parameters. The second model incorporates a linear trend on the Dover margin only. The third model assumes stationarity. The `nsloc1` and `nsloc2` arguments are centred and scaled so that the intercepts `loc1` and `loc2` represent the marginal location parameters in 1950 and the linear trend parameters `loc1trend` and `loc2trend` represent the increase in the marginal location parameters (or decrease, if negative) over a period of 100 years.

```
> tt <- (1912:1992 - 1950)/100
> m1 <- fbvevd(s1, model = "log", nsloc1 = tt, nsloc2 = tt)
> m2 <- fbvevd(s1, model = "log", nsloc1 = tt)
> m3 <- fbvevd(s1, model = "log")
```

The maximum likelihood estimates of the parameters can be compared with their standard errors to perform Wald tests or construct confidence intervals. Likelihood ratio tests are performed in the following snippet. The p-values confirm the statistical significance of the linear trend terms.

```
> anova(m1, m2, m3)
Analysis of Deviance Table

    M.Df Deviance Df Chisq Pr(>chisq)
m1      9    -36.5
m2      8    -29.2  1  7.26      0.007
m3      7     -9.7  1 19.56     9.7e-06
```

Quadratic trends for the location parameter on either or both margins can be incorporated using the following code. Further testing, using the models generated below, suggests that a quadratic trend may be implemented for the location parameter on the Harwich margin. Despite this, I retain the model `m1` for further analysis.

```
> tdframe <- data.frame(trend = tt, quad = tt^2)
> m4 <- fbvevd(s1, model = "log", nsloc1 = tdframe, nsloc2 = tt)
> m5 <- fbvevd(s1, model = "log", nsloc1 = tt, nsloc2 = tdframe)
> m6 <- fbvevd(s1, model = "log", nsloc1 = tdframe, nsloc2 = tdframe)
```

The code given below compares two logistic models that are nested within `m1`. Model `m7` assumes independence. The maximum likelihood estimates are the same as those that would be produced if `fgev` was separately applied to each margin. The asymptotic distribution of the deviance difference between models `m7` and `m1` is non-regular because the dependence parameter in the restricted (independence) model is fixed at the edge of the parameter space. Tawn (1988) discusses non-regular cases, including this case, for which the asymptotic distribution is one-half of a chi-squared random variable on one degree of freedom. In these cases the argument `half` should be set to `TRUE`. The resulting p-value is less than 10^{-6} , and clearly the independence model is rejected.

Model `m8` assumes that both marginal shape parameters are zero (or equivalently, that both marginal distributions are Gumbel). A likelihood ratio test of this hypothesis provides a p-value of 0.72. The hypothesis would not be rejected at any reasonable significance level.

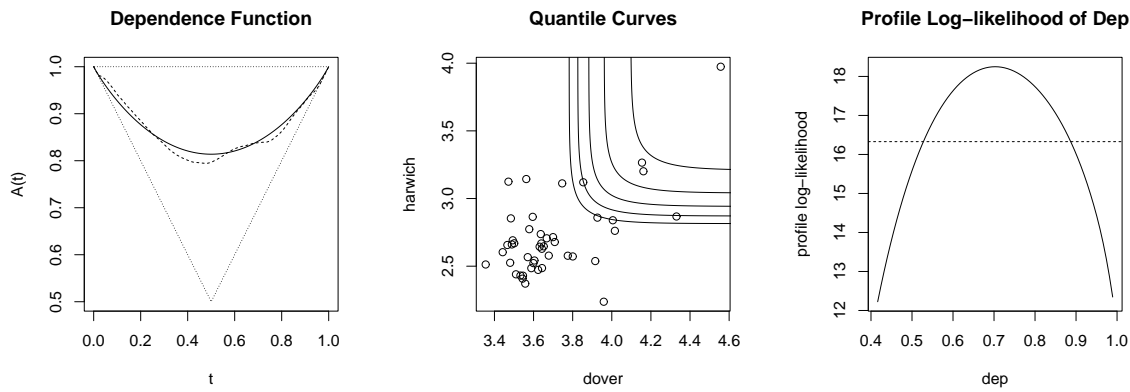


Figure 9: From left to right; dependence function diagnostic plot, quantile curves diagnostic plot, profile log-likelihood of the dependence parameter.

```
> m7 <- fbvevd(s1, model = "log", nsloc1 = tt, nsloc2 = tt, dep = 1)
```

```
> anova(m1, m7, half = TRUE)
```

Analysis of Deviance Table

	M.Df	Deviance	Df	Chisq	Pr(>chisq)
m1	9	-36.5			
m7	8	-22.9	1	27.2	1.9e-07

```
> m8 <- fbvevd(s1, "log", nsloc1 = tt, nsloc2 = tt, shape1 = 0, shape2 = 0)
```

```
> anova(m1, m8)
```

Analysis of Deviance Table

	M.Df	Deviance	Df	Chisq	Pr(>chisq)
m1	9	-36.5			
m8	7	-35.8	2	0.67	0.72

Diagnostic plots for the fitted (generalised extreme value) marginal distributions can be produced using `plot` with `mar = 1` or `mar = 2`. The plots produced are of the same structure as those given in Section 10. Diagnostic plots for the fitted dependence structure can be produced using `plot`. There are five plots available, though only four are plotted by default (see the `plot.bvevd` help file for details). Two diagnostic plots are depicted within Figure 9.

```
> plot(m1, mar = 1)
```

```
> plot(m1, mar = 2)
```

```
> plot(m1, which = 1:5)
```

The model `m1` fits the data reasonably well. There are some minor deviations within the conditional P-P plots (not shown), but they do not represent a serious departure of the empirical estimates from the fitted model. The profile log-likelihood of the dependence parameter `dep`, as given in the right panel of Figure 9, can be plotted using the following. The argument `xmax` denotes the upper bound of the parameter.

```
> m1.prof <- profile(m1, which = "dep", xmax = 1)
```

```
> pcis <- plot(m1.prof)
```

```
> pcis[["dep"]]
      lower  upper
0.5282 0.8865
```

A horizontal line is (optionally) drawn so that the intersection of the line with the profile log-likelihood yields a profile confidence interval, with (default) confidence coefficient 0.95. The end points of the interval can be derived by assigning the expression `plot(m1.prof)` to an object, as above.

Further analysis with models other than the (symmetric) logistic yields the following conclusions. The two models in Section 3 that include three parameters with which to describe the dependence structure (the asymmetric logistic and asymmetric negative logistic) are inappropriate. In both cases, the maximum likelihood estimate for the parameter `dep` is at an artificial boundary, because the fitted model is close to a distribution (obtained in the limit) which contains a singular component. This is clearly illustrated in the density plots of the fitted models, which both depict a ridge of mass extending towards the 1953 outlier. The logistic and the bilogistic models have the lowest deviance of all one and two parameter models respectively. The dependence structure of the fitted bilogistic model is almost symmetric. At symmetry, the bilogistic model reduces to the logistic model, and so the latter would appear to be preferable. A likelihood ratio test between the two (nested) models gives a p-value of 0.93.

Appendix A: Additional Bivariate Parametric Models

It can be shown, using a representation of de Haan (1984), that

$$G(z_1, z_2) = \exp \left\{ - \int_0^1 \max \{ y_1(1 - \alpha)x^{-\alpha}, y_2(1 - \beta)(1 - x)^{-\beta} \} dx \right\}, \quad \alpha, \beta < 1.$$

is a bivariate extreme value distribution function. If we further constrain the parameters to be non-negative we obtain the bivariate bilogistic model proposed by Smith (1990), which can also be expressed as

$$G(z_1, z_2) = \exp \left\{ -y_1\gamma^{1-\alpha} - y_2(1 - \gamma)^{1-\beta} \right\}, \quad 0 < \alpha, \beta < 1,$$

where $\gamma = \gamma(y_1, y_2; \alpha, \beta)$ solves $(1 - \alpha)y_1(1 - \gamma)^\beta = (1 - \beta)y_2\gamma^\alpha$. The logistic model is obtained when $\alpha = \beta$. Independence is obtained as $\alpha = \beta \rightarrow 1$, and when one of α, β is fixed and the other approaches one. Different limits occur when one of α, β is fixed and the other approaches zero.

Alternatively, if we constrain both parameters to be non-positive and set $\alpha_0 = -\alpha > 0$ and $\beta_0 = -\beta > 0$ we obtain the negative bilogistic model (Coles and Tawn, 1994), which has the representation

$$G(z_1, z_2) = \exp \left\{ -y_1 - y_2 + y_1\gamma^{1+\alpha_0} + y_2(1 - \gamma)^{1+\beta_0} \right\}, \quad \alpha_0, \beta_0 > 0,$$

where $\gamma = \gamma(y_1, y_2; -\alpha_0, -\beta_0)$. The negative logistic model is obtained when $\alpha_0 = \beta_0$ (with $r = 1/\alpha_0 = 1/\beta_0$). Independence is obtained as $\alpha_0 = \beta_0 \rightarrow \infty$, and when one of α_0, β_0 is fixed and the other tends to ∞ . Different limits occur when one of α_0, β_0 is fixed and the other approaches zero.

The distribution function of the Coles-Tawn model[†] (Coles and Tawn, 1991) is given by

$$G(z_1, z_2) = \exp \{ -y_1[1 - \text{Be}(u; \alpha + 1, \beta)] - y_2 \text{Be}(u; \alpha, \beta + 1) \}, \quad \alpha, \beta > 0,$$

[†]Coles and Tawn (1991) call this the Dirichelet model.

where $u = \alpha y_2 / (\alpha y_2 + \beta y_1)$ and Be is the incomplete beta function, given by

$$\text{Be}(u; \alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} \int_0^u x^{\alpha-1} (1-x)^{\beta-1} dx.$$

Complete dependence is obtained in the limit as $\alpha = \beta \rightarrow \infty$. Independence is obtained as $\alpha = \beta \rightarrow 0$ and when one of α, β is fixed and the other approaches zero. Different limits occur when one of α, β is fixed and the other tends to ∞ .

The asymmetric mixed model (Tawn, 1988) is typically defined using the corresponding dependence function (11), which is modelled as a cubic polynomial. Specifically, for $0 \leq t \leq 1$ the dependence function of the asymmetric mixed model is

$$A(t) = 1 - (\alpha + \beta)t + \alpha t^2 + \beta t^3,$$

where both α and $\alpha + 3\beta$ are non-negative, and where both $\alpha + \beta$ and $\alpha + 2\beta$ are less than or equal to one. These constraints imply that $\beta \in [-0.5, 0.5]$ and $\alpha \in [0, 1.5]$, though α can only be greater than one if β is negative. The (symmetric) mixed model is obtained when $\beta = 0$. Complete dependence cannot be obtained. Independence is obtained when $\alpha = \beta = 0$.

The asymmetric mixed model is often referred to in the literature because the dependence function has a simple form, and because the $\beta = 0$ case is historically important. However it cannot capture strong dependence, and hence it is of limited use as a statistical model. The extension to an m -degree polynomial can be made, but this is of no statistical interest because the additional parameters add little additional flexibility.

References

- Beirlant, J., Goegebeur, Y., Segers, J. and Teugels, J. (2004) *Statistics of Extremes: Theory and Applications*. Chichester, England: John Wiley & Sons.
- Capérea, P., Fougères, A.-L. and Genest, C. (1997) A non-parametric estimation procedure for bivariate extreme value copulas. *Biometrika*, **84**, 567–577.
- Coles, S. G. (2001) *An Introduction to Statistical Modeling of Extreme Values*. London: Springer-Verlag.
- Coles, S. G., Heffernan, J. and Tawn, J. A. (1999) Dependence measures for extreme value analyses. *Extremes*, **2**, 339–365.
- Coles, S. G. and Tawn, J. A. (1990) Statistics of coastal flood prevention. *Phil. Trans. R. Soc. Lond., A*, **332**, 457–476.
- Coles, S. G. and Tawn, J. A. (1991) Modelling extreme multivariate events. *J. Roy. Statist. Soc., B*, **53**, 377–392.
- Coles, S. G. and Tawn, J. A. (1994) Statistical methods for multivariate extremes: an application to structural design (with discussion). *Appl. Statist.*, **43**, 1–48.
- de Haan, L. (1984) A spectral representation for max-stable processes. *Ann. Probab.*, **12**, 1194–1204.
- Deheuvels, P. (1991) On the limiting behaviour of the pickands estimator for bivariate extreme-value distributions. *Statist. Probab. Letters*, **12**, 429–439.
- Galambos, J. (1975) Order statistics of samples from multivariate distributions. *J. Amer. Statist. Assoc.*, **70**, 674–680.

- Gumbel, E. J. (1960) Distributions des valeurs extrêmes en plusieurs dimensions. *Publ. Inst. Statist. Univ. Paris*, **9**, 171–173.
- Hall, P. and Tajvidi, N. (2000) Distribution and dependence-function estimation for bivariate extreme-value distributions. *Bernoulli*, **6**, 835–844.
- Hüsler, J. and Reiss, R.-D. (1989) Maxima of normal random vectors: between independence and complete dependence. *Statist. Probab. Letters*, **7**, 283–286.
- Ihaka, R. and Gentleman, R. (1996) R: A language for data analysis and graphics. *Journal of Computational and Graphical Statistics*, **5**, 299–314.
- Joe, H. (1990) Families of min-stable multivariate exponential and multivariate extreme value distributions. *Statist. Probab. Letters*, **9**, 75–81.
- Leadbetter, M. R., Lindgren, G. and Rootzén, H. (1983) *Extremes and Related Properties of Random Sequences and Series*. New York: Springer-Verlag.
- Pickands, J. (1981) Multivariate extreme value distributions. *Proc. 43rd Sess. Int. Statist. Inst.*, **49**, 859–878.
- Smith, R. L. (1985) Maximum likelihood estimation in a class of non-regular cases. *Biometrika*, **72**, 67–90.
- Smith, R. L. (1990) Extreme value theory. In *Handbook of Applicable Mathematics* (ed. W. Ledermann), vol. 7. Chichester: John Wiley, pp. 437–471.
- Stephenson, A. G. (2003) *Extreme Value Distributions and their Application*. Ph.D. thesis, Lancaster University, Lancaster, UK.
- Tabony, R. C. (1983) Extreme value analysis in meteorology. *The Meteorological Magazine*, **112**, 77–98.
- Tawn, J. A. (1988) Bivariate extreme value theory: models and estimation. *Biometrika*, **75**, 397–415.
- Tawn, J. A. (1990) Modelling multivariate extreme value distributions. *Biometrika*, **77**, 245–253.
- Tiago de Oliveira, J. (1997) *Statistical Analysis of Extremes*. Pendor.