Estimating the Distribution of an Extremum using a Peaks-Over-Threshold Model and Monte Carlo Simulation

Adam L. Pintar

April 5, 2017
Acknowledgements

- Emil Simiu
- Dat Duthinh
Examples of Extreme Value Analyses

- Smith & Davidson 1990 – Nuclear power plant sitings
- Smith 2004 – Insurance claims
- Mannshardt-Shamseldin et al. 2010 – Rainfall
- Pintar et al. 2015 – Wind speeds
- Goal in each example was different
- Used same underlying probability model
Wind Tunnel Tests
- Distribution of largest (or smallest) is the goal
- Data are stationary in time (but can relax that assumption)
- Similarities to previous examples
Fit Generalized Extreme Value Distribution
Classical Extrapolation in Time

Order statistics (raise CDF to appropriate power)
How Many Partitions?

Number of Partitions
Mean Peak Pressure

<table>
<thead>
<tr>
<th>5</th>
<th>10</th>
<th>15</th>
<th>20</th>
<th>25</th>
<th>30</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.05</td>
<td>3.15</td>
<td>3.25</td>
<td>3.35</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Peaks Over Threshold (POT) Model

Two dimensional Poisson process
Intensity Function

\[ \lambda(y, t) = \begin{cases} 
\frac{1}{\sigma} \exp \left[ \frac{-(y-\mu)}{\sigma} \right] & \zeta = 0 \\
\frac{1}{\sigma} \left(1 + \frac{\zeta(y-\mu)}{\sigma} \right)^{-1/\zeta-1} & \zeta \neq 0 
\end{cases} \]
Declustering

![Graphs showing time series and autocorrelation functions (ACF)]
Choosing the Threshold

\[ W = \begin{cases} \frac{1}{\zeta} \log \left[ 1 + \frac{\zeta(y-B)}{\sigma + \zeta(B-\mu)} \right] & \zeta \neq 0 \\ \frac{y-B}{\sigma} & \zeta = 0 \end{cases} \]

- For high thresholds the \( W \)-Statistic follows a standard exponential distribution
- Balance bias-variance trade off
- Maximum likelihood for fitting

![W-Statistic Plot](High Threshold)
![W-Statistic Plot](Low Threshold)
![W-Statistic Plot](Optimal Threshold)
Simulated Data

Original Data Set

Fake Data Set #1

Fake Data Set #2
Distribution of the Peak Value

Peak Value

Density

3.0 3.5 4.0 4.5

0.0 0.5 1.0 1.5

Mean

Distribution of the Peak Value

Peak Value
Uncertainty in the Parameters

- Resample $\mu$, $\sigma$ and $k$ directly from multivariate Gaussian
- $k = 0$ below
Uncertainty in the Distribution of the Peak Value

Distribution of the Peak Value

- Mean

Bootstrap Replicates

80% CI for the Mean
Does it work?
Probability of Non-exceedance

Distribution of Peak Value

Estimate: 0.948
Uncertainty in the Probability of Non-exceedance

Distribution of Peak Value

- 80% confidence interval: [0.817, 0.989]
Parameters Varying with Time

- Smith 2004 – wind speed data

\[ \lambda(y, t) = \frac{1}{4.18} \left[ 1 + \left( \frac{-0.12(y - \mu(t))}{4.18} \right) \right]^{-1/(-0.12)-1} \]

\[ \mu(t) = 40.9 + 0.9 \sin(2\pi t) + 5.29 \cos(2\pi t) \]

- Simulated data below
Parameters Varying with Time

- Smith 2004 – wind speed data
- \( \lambda(y, t) = \frac{1}{4.18} \left[ 1 + \frac{(-0.12)(y-\mu(t))}{4.18} \right]^{-1/(-0.12)-1} \)
- \( \mu(t) = 40.9 + 0.9 \sin(2\pi t) + 5.29 \cos(2\pi t) \)
- Simulated data below
R Package

- Available from the NIST Git Hub site
  - [https://github.com/usnistgov/potMax](https://github.com/usnistgov/potMax)

- Installation

```r
install.packages(c('Rcpp', 'numDeriv',
                    'devtools', 'knitr'),
                    repos = 'https://cloud.r-project.org/')
devtools::install_github('usnistgov/potMax',
                        build_vignettes = TRUE)
```

- Must have Rtools installed on Windows systems
  - I will provide a Windows binary on request

- Must have LaTeX installed to build the vignette

- Computationally intense parts are in C++ to boost speed

- Does not include everything discussed today, e.g., parameters varying with time
Summary

- Distribution of time series peak value using a POT model
- Leverages Monte Carlo simulation to construct the distribution and quantify uncertainty
- Inspired by wind tunnel testing, but applicable more broadly
- Efficient software implementation using R and C++
Illustrate with built-in data set

```r
library(potMax)
orig_x <- -jp1tap1715wind270$value
```
Declustering

declust_x <- decluster(complete_series = orig_x, obs_times = NULL)
Optimal Threshold

- **$k = 0$**

```r
thresh_x_g <- gumbelEstThreshold(x = declust_x,
                           lt = 100,
                           n_min = 10,
                           n_max = 100)
```

- **$k \neq 0$**

```r
thresh_x_f <- fullEstThreshold(x = declust_x,
                           lt = 100,
                           n_min = 10,
                           n_max = 100,
                           n_starts = 20)
```
Parameter Estimates

- $k = 0$

```r
gumbel_fit <- gumbelMLE(x = thresh_x_g, 
                        hessian_tf = TRUE)
```

- $k \neq 0$

```r
full_fit <- fullMLE(x = thresh_x_f, 
                     hessian_tf = TRUE, 
                     n_starts = 20)
```
Distribution of the Peak

- Estimate distribution

```r
gumbel_md <- gumbelMaxDist(x = gumbel_fit, 
lt_gen = 200,
n_mc = 1000)
```

- Uncertainty

```r
gumbel_uncert <- gumbelMaxDistUncert(x = gumbel_fit, 
lt_gen = 200, 
n_mc = 1000, 
n_boot = 1000)
```

- S3 methods for the generic functions plot, summary, and mean