NCAR

Introduction to R and extRemes

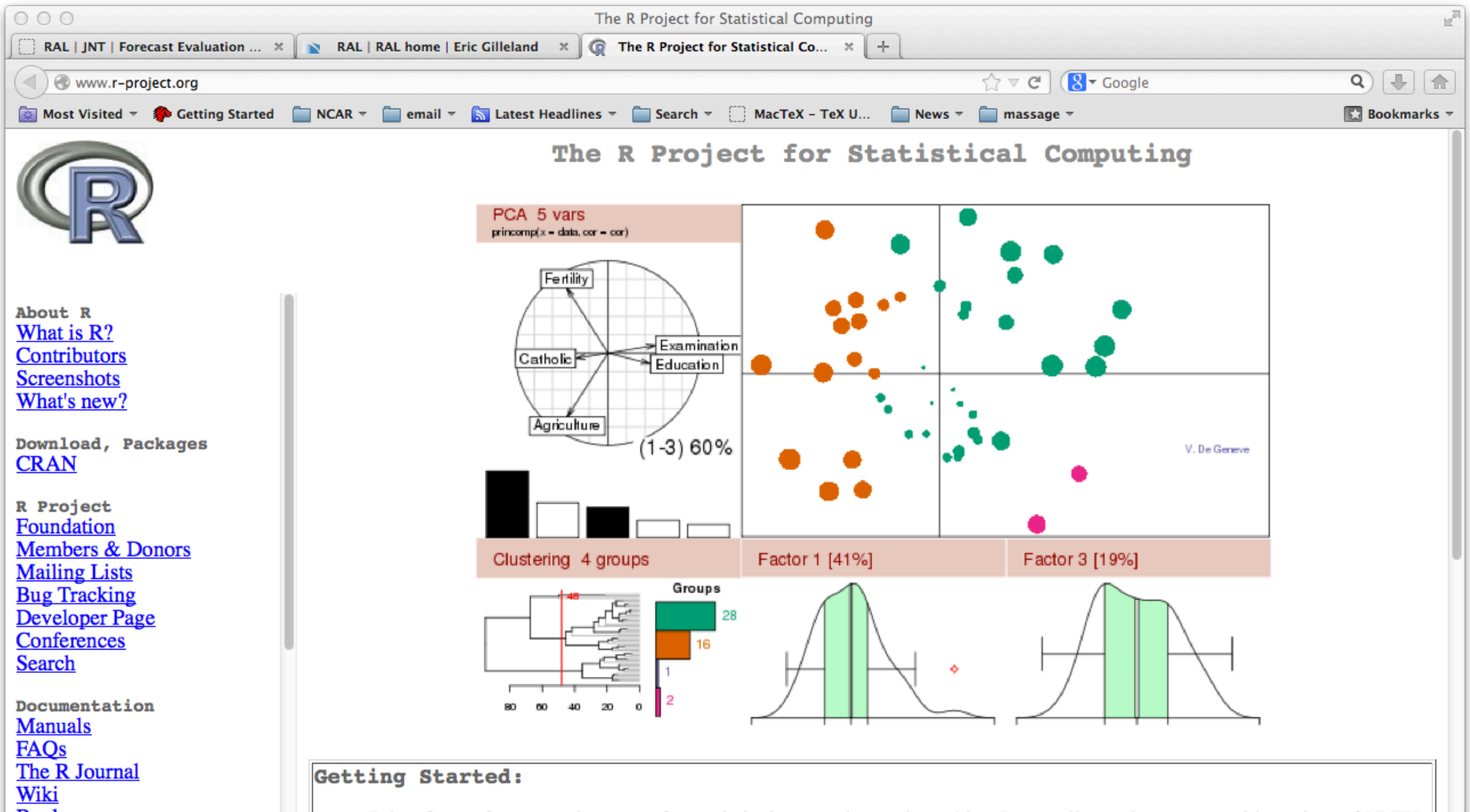
Eric Gilleland

Research Applications Laboratory

21 July 2014, Trieste, Italy

National Center for Atmospheric Research

Part I: Intro to R



R-Project website

- Documentation
- Access to packages via CRAN
- Search engines (including one specific to R)
- other
- http://www.R-project.org

Advanced

- Reading and Writing Net CDF files
  http://www.image.ucar.edu/Software/Netcdf/
- A climate related precipitation example
  http://www.image.ucar.edu/~nychka/FrontrangePrecip/

.RData

- All work conducted within a "dot" file called .RData, which is referred to as the workspace and exists in the working directory
- save.image()
- getwd()
- setwd()
- citation()
- q()
- Functions exist to export results outside of the workspace

Accessing Help Files

- ?save.image
- ?getwd
- ?setwd
- ?citation
- ?q
- To write to a file outside of .Rdata working directory
  - ?write (text, csv, etc.)
  - ?write.table (text, csv, etc.)

Object Oriented

- plot
- methods(plot)
- methods(class="lm")
- print
- summary
- predict
- length

Logical Operators

- &, && (and)
- |, || (or)
- ==
- >=, >, <, <=
- is.na, is.finite, is.numeric, is.logical, is.element

Logical Operators: Accessing Help Files

- ?”&”, ?”&&”
- ?”|”
- ?”==“
- ?is.na

Arithmetic Operators

- +
- -
- *
- /
- ^, **
- %*% (matrix multiplication)

Assignment Operators

NCAR

- <-
  - x <- 1:10
- ->
  - 1:10 -> x
- =
  - x = 1:10
  - 1:10 = x
- ?assign

Functions

- help, ?
- plot
- Sys.time
- Sys.time()
- x <- Sys.time()
  - x is the result of the Sys.time call
- y <- Sys.time
  - y is a copy of the function Sys.time
- args(plot)

Some of my terminology

- x “gets” 1 to 10
  - x <- 1:10
  - x <- seq(1,10,1)
  - x <- seq(1, 10, , 10)
- plot x
  - plot(x)
- x “at” 3
  - x[3]

Some of my terminology

NCAR

- open paren
  - (
- close paren
  -)
- square brackets
  - open [
  - close]
- curly brackets / braces
  - open {
  - close }

Packages

- Thousands of users have written packages. Most are freely available in the same place, and accessible (installation) from within an R session (provided you are connected to the internet).
- Must first install the package you want (need only do once), and load the library for each new R session.
- ?install.packages
- ?update.packages
- ?library
- citation(“pkgname”)

Installing / Updating a package

- Need be done only once (per update)
- Install extRemes
  - install.packages("extRemes")
  - Select a mirror
  - Installs extRemes and dependencies
- Updating already installed packages (all at once)
  - update.packages()
  - will prompt you for a mirror and may ask for confirmation to update each package.

Loading a package

NCAR

- Must be done for every new R session (if you want to use the package)
- Load extRemes
  - library(“extRemes”)
  - loads all dependent packages as well

Citing R packages

- citation(“pkgname”)
- Example: to cite extRemes:
  - citation(“extRemes”)

Some Atmospheric Science Oriented Packages

NCAR

- Spatial Statistics
  - fields
  - spatstat,
  - sp
  - maps
  - Many More! (See the Spatial Data Task View on the R-Project web site)
- Extreme Value Analysis
  - extRemes
  - SpatialExtremes
  - texmex
  - Many More! (http://www.ral.ucar.edu/staff/ericg/softextreme.php)
- Forecast Verification / Model Evaluation
  - verification
  - SpatialVx
- RadioSonde
- ncdf, netCDF
- smoothie

Formulas in R

Left as an exercise. Should get somewhat familiar with these because they are used for modeling nonstationary EV models in extRemes ($\geq$ 2.0).

?formula

Types of R objects

NCAR

- functions
- vectors, matrices, arrays
  - numeric, integer, etc.
  - character
  - factor
- data frames
- lists
- other

vectors, matrices, arrays

```
x <- cbind(c(1,2,3), c(4,5,6))
y <- matrix(1:6, ncol = 2)
x
y
x == y
any(x != y)
x[, 1]
x[2, ] <- NA
is.na(x)
x – y
x[1:2, 3]
x[, -1]
```

vectors, matrices, arrays

```
x <- rnorm(100)

class(x)

is.vector(x)

is.matrix(x)

x[ 1:10 ]

plot(x)
```

vectors, matrices, arrays

```
x <- array(1:24, dim = c(2, 3, 4))

x

x[, , 1] <- 0

x

apply(x, 1:2, sum)
```

data frames

```
x <- data.frame(obs = 1:10, x = runif(10),
    y = rnorm(10))

x

x[, “x”]

x[, 2]

plot(x)

class(x)

is.list(x)
```

lists

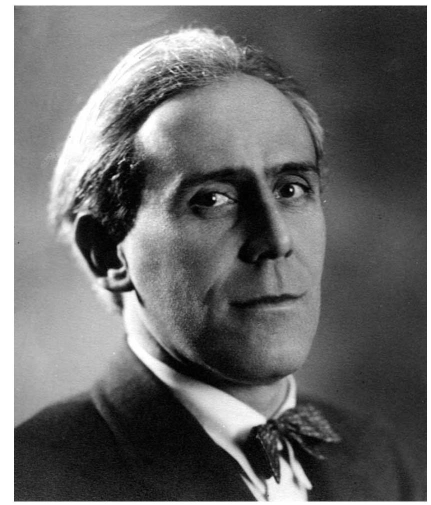
```
x <- list(f = function(x) return(x ^2),
    x = rnorm(10), y = 1:6)

x

x$f(3)

x$x

class(x)
```

Conclusion of Part I: Intro to R

Part II: Extreme Value Analysis with extRemes ≥ 2.0

Block Maxima

Midwest flood 1993 (NCAR Digital Image Library, DI00578)



"Il est impossible que l'improbable n'arrive jamais"
--Emil Gumbel

Tutorial

http://www.ral.ucar.edu/staff/ericg/extRemes/extRemes2.pdf

Generalized Extreme Value Distribution

Simulate a sample of size 100 of maxima of standard normal distributed samples

```
library(extRemes)

Zmax <- matrix(rnorm(100 * 1000),
    1000, 100)

dim(Zmax)

Zmax <- apply(Zmax, 2, max)

dim(Zmax)
```

Generalized Extreme Value Distribution

Simulate a sample of size 100 of maxima of standard normal distributed samples

```
length(Zmax)

plot(Zmax, type = “h”, col = “darkblue”)
```

Generalized Extreme Value Distribution

Fit a GEV distribution to the simulated sample

```
fit <- fevd(Zmax)

fit

plot(fit)

ci(fit, type = "parameter")

distill(fit)
```

Generalized Extreme Value Distribution

Plot maximum winter temperature (°C) in Sept-Iles, Québec

```
data(SEPTsp)

?SEPTsp

par(mfrow = c(2, 2))
```

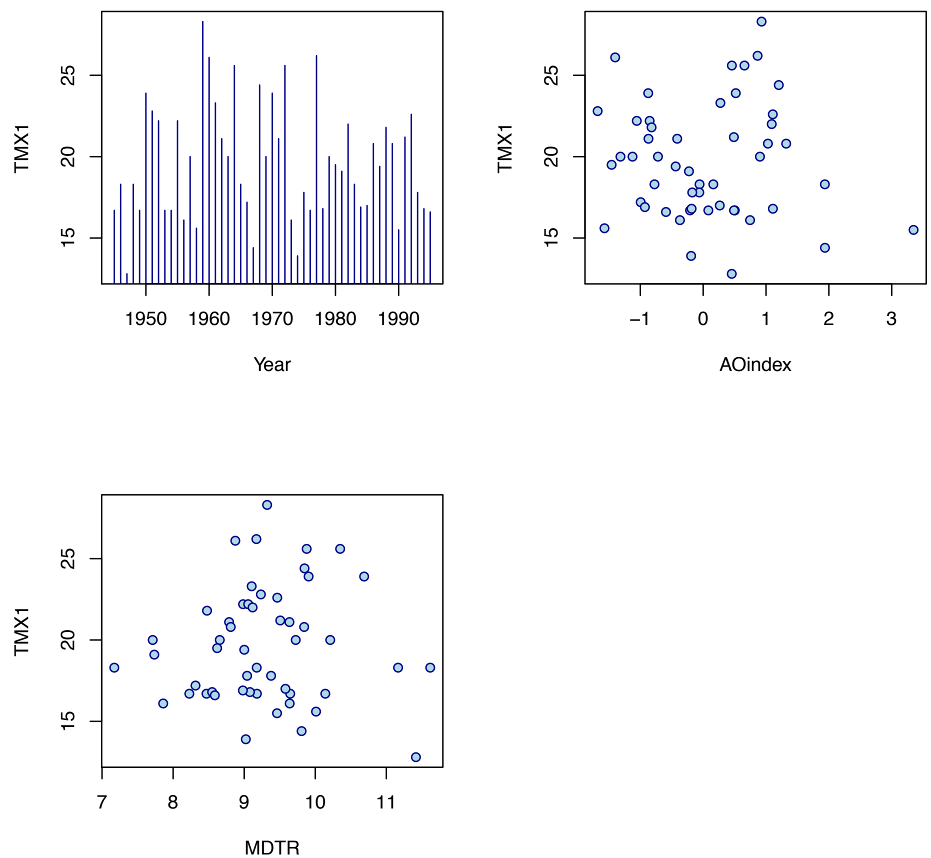
Generalized Extreme Value Distribution

Plot maximum winter temperature (°C) in Sept-Iles, Québec

```
plot(TMX1~ Year, data = SEPTsp,
    type = "h", col = "darkblue")

plot(TMX1~ AOindex, data = SEPTsp,
    pch = 21, col = "darkblue",
    bg = "lightblue")

plot(TMX1~ MDTR, data = SEPTsp,
    pch = 21, col = "darkblue",
    bg = "lightblue")
```

Generalized Extreme Value Distribution

Maximum winter temperature in Sept-Iles, Québec



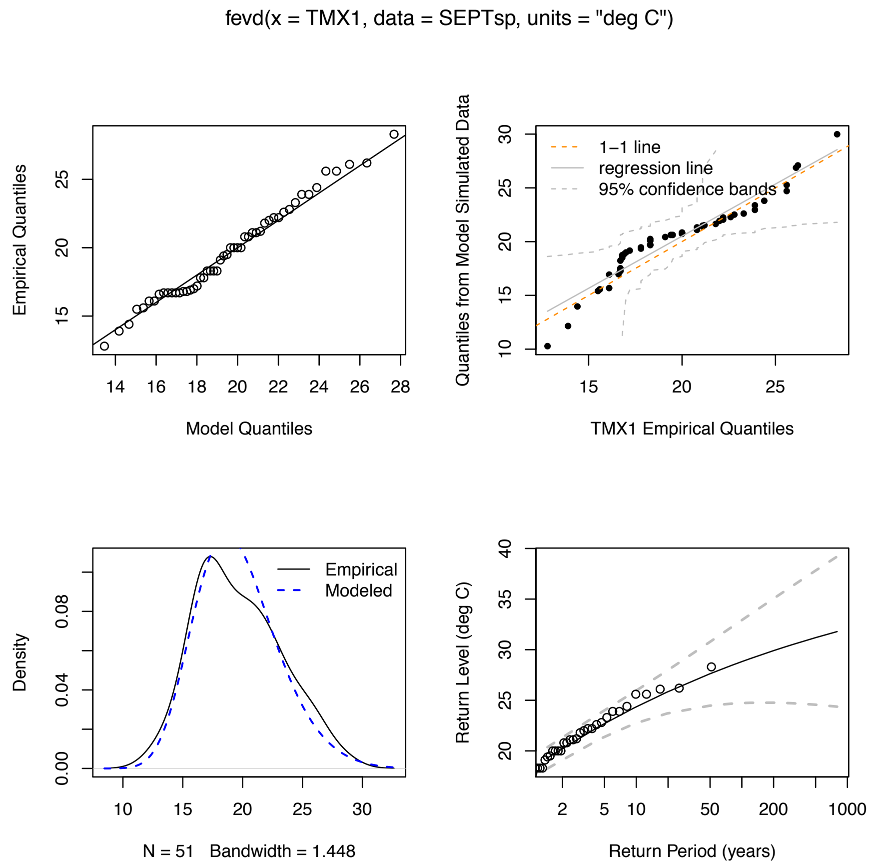
Generalized Extreme Value Distribution

Fit a GEV distribution to maximum winter temperature in Sept-Iles, Québec

```
fit0 <- fevd(TMX1, data = SEPTsp,
    units = "deg C")

fit0

plot(fit0)

ci(fit0, type = "parameter")

ci(fit0)
```

Generalized Extreme Value Distribution

Fit a GEV distribution to maximum winter temperature in Sept-Iles, Québec



Generalized Extreme Value Distribution

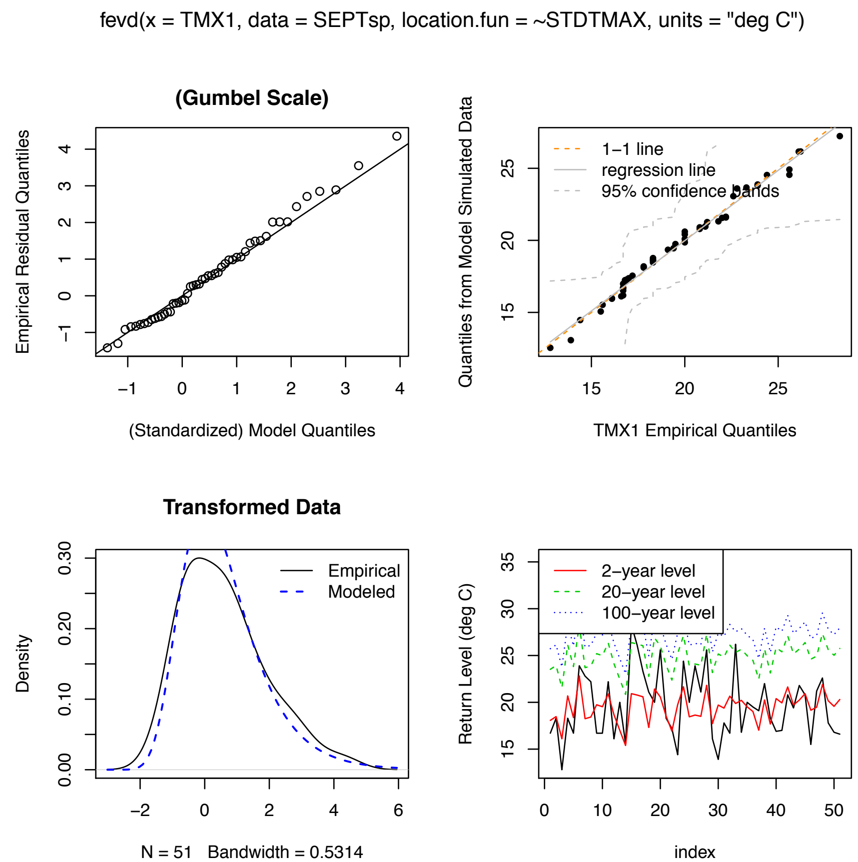
Fit a GEV distribution to maximum winter temperature in Sept-Iles, Québec

```
fit1 <- fevd(TMX1, data = SEPTsp,
    location.fun = ~AOindex,
    units = "deg C")

fit1

plot(fit1)

lr.test(fit0, fit1)
```

Inclusion of AOindex in location parameter is not significant.

Generalized Extreme Value Distribution

Fit a GEV distribution to maximum winter temperature in Sept-Iles, Québec



Generalized Extreme Value Distribution

NCAR

Fit a GEV distribution to maximum winter temperature in Sept-Iles, Québec

```
fit2 <- fevd(TMX1, data = SEPTsp,
      scale.fun = ~STDTMAX,
      use.phi = TRUE, units = "deg C")

fit2

plot(fit2)

lr.test(fit0, fit2)
```

Addition of AOindex to scale parameter is not statistically significant.

Generalized Extreme Value Distribution

NCAR

Addition from previous slide is a function of the following form.

$$\log(\sigma(\text{AO index})) = \varphi_0 + \varphi_1 \times \text{AO index}$$

Because: `use.phi = TRUE`

Generalized Extreme Value Distribution

NCAR

Plot minimum winter temperature (°C) in Sept-Iles, Québec

```
par(mfrow = c(2, 2))

plot(TMN0~ Year, data = SEPTsp,
    type = "h", col = "darkblue")

plot(TMN0~ AOindex, data = SEPTsp,
    pch = 21, col = "darkblue",
    bg = "lightblue")

plot(TMN0~ MDTR, data = SEPTsp,
    pch = 21, col = "darkblue",
    bg = "lightblue")
```

Generalized Extreme Value Distribution

Fit GEV to (negative) minimum winter temperature (°C) in Sept-Iles, Québec

```
fit0 <- fevd(-TMN0 ~ 1, data = SEPTsp,
    units = “neg. deg. C”)

fit0

plot(fit0)
```

The rest of the analysis of (negative) minimum temperature is left as an exercise.

Part III: Extreme Value Analysis

Frequency of extremes

photo from Wikipedia: http://en.wikipedia.org/wiki/Coligny_calendar

Frequency of Extremes

Number of days that maximum daily temperature (°F) in Fort Collins, Colorado exceeds 95 °F.

```
data(FCwx)

?FCwx

tempGT95 <- aggregate(FCwx$MxT,
    by = list(FCwx$Year),
    function(x) sum(x > 95,
        na.rm = TRUE))

 class(tempGT95)
```

Frequency of Extremes

Number of days that maximum daily temperature (°F) in Fort Collins, Colorado exceeds 95 °F.

```
names(tempGT95)

names(tempGT95) <- c(“Year”, “MxT”)

tempGT95
```

Frequency of Extremes

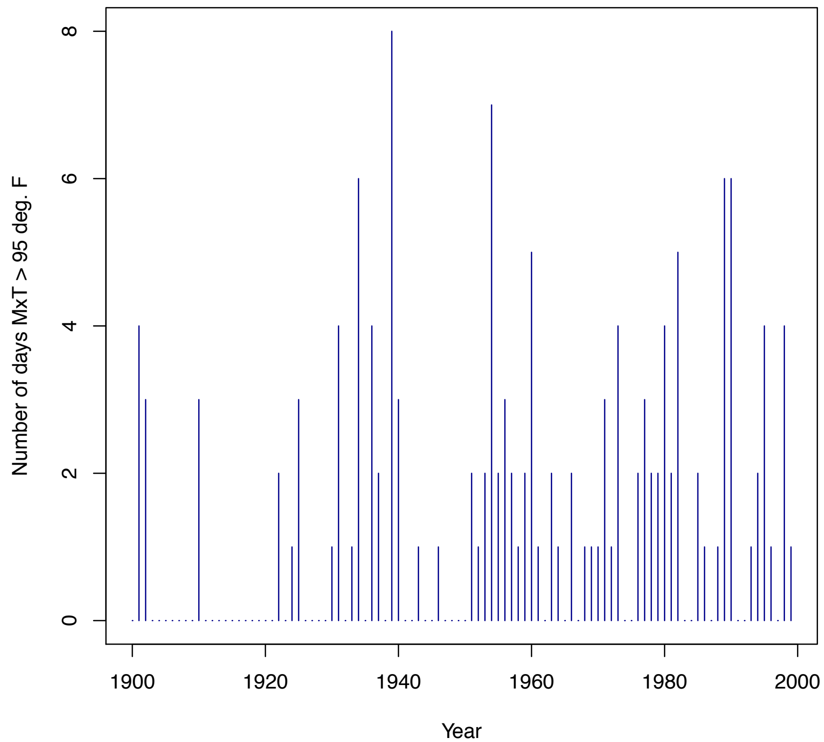
Fit Poisson distribution to number of days that maximum daily temperature (°F) in Fort Collins, Colorado exceeds 95 °F.

```
plot(MxT ~ Year, data = tempGT95,
    type = "h", col = "darkblue",
    ylab =
"Number of days MxT > 95 deg. F")

fpois(tempGT95$MxT)
```

Test for equality of mean and variance says that the two are statistically significantly different.

Frequency of Extremes

Plot of number of days that maximum daily temperature (°F) in Fort Collins, Colorado exceeds 95 °F.



Frequency of Extremes

Fit Poisson to number of days that maximum daily temperature (°F) in Fort Collins, Colorado exceeds 95 degrees F.

```
fit <- glm(tempGT95~yr, family = poisson())

summary(fit)
```

Part IV: Extreme Value Analysis

Threshold Excesses

V.F.D. Pareto

Generalized Pareto

NCAR

Example: Denver, Colorado July hourly Precipitation (mm)

```
par(mfrow = c(2, 2), oma = c(0,0,2,0))

plot(Prec ~ Year, data = Denversp,
    type = "h", col = "darkblue",
    lwd = 2, cex.lab = 1.5,
    cex.axis = 1.5, ylab = "")

plot(Prec ~ Day, data = Denversp,
    type = "h", col = "darkblue",
    lwd = 2, cex.lab = 1.5,
    cex.axis = 1.5, ylab = "")
```

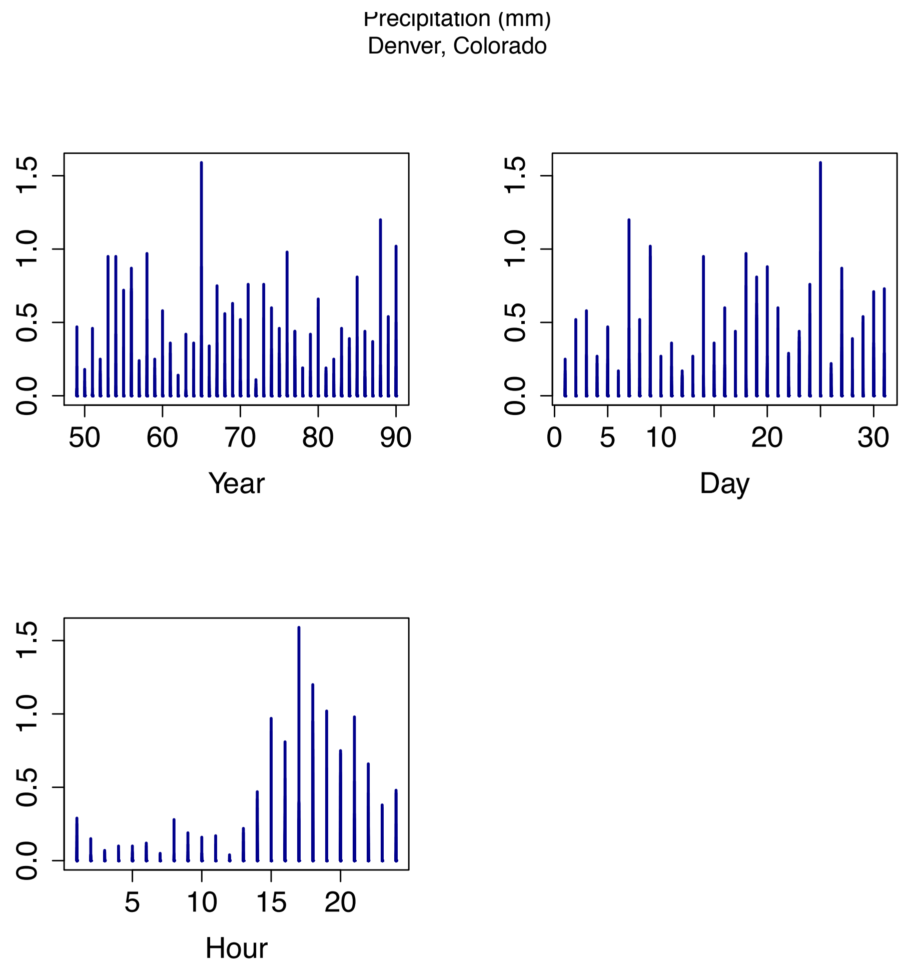
Generalized Pareto

Example: Denver, Colorado July hourly Precipitation (mm)

```
plot(Prec ~ Hour, data = Denversp,
    type = "h", col = "darkblue", lwd = 2,
    cex.lab = 1.5, cex.axis = 1.5,
    ylab = "")

mtext("Precipitation (mm)\nDenver, Colorado",
    side = 3, outer = TRUE)
```

Generalized Pareto

Example: Denver, Colorado July hourly Precipitation (mm)



Generalized Pareto

NCAR

Example: Denver, Colorado July hourly Precipitation (mm)

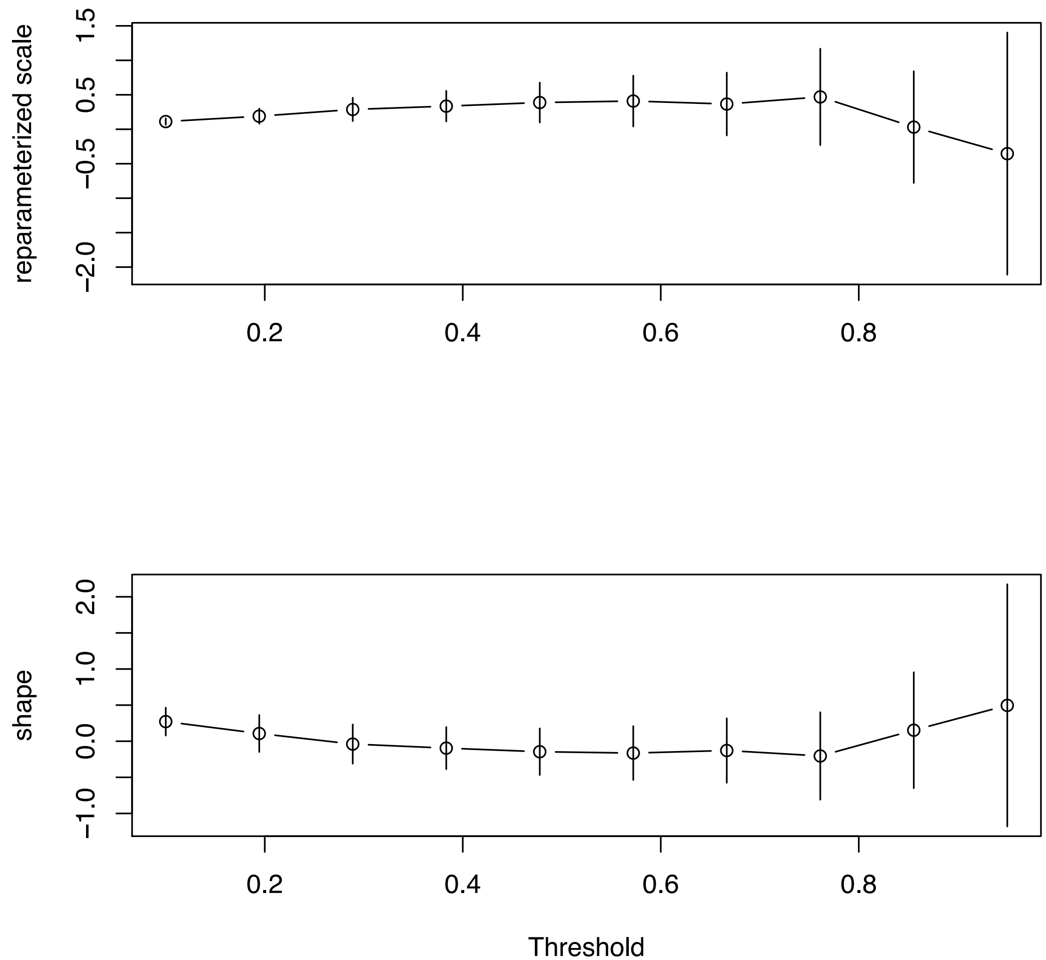
```
threshrange.plot(Denversp$Prec,
    r = c(0.1, 0.95))

extremalindex(Denversp$Prec, threshold=0.5)
```

A threshold of about 0.5 mm seems appropriate for these data. Threshold excesses appear to be independent with this threshold.

Generalized Pareto

Example: Denver, Colorado July hourly Precipitation (mm)

threshrange.plot(x = Denversp$Prec, r = c(0.1, 0.95))



Generalized Pareto

NCAR

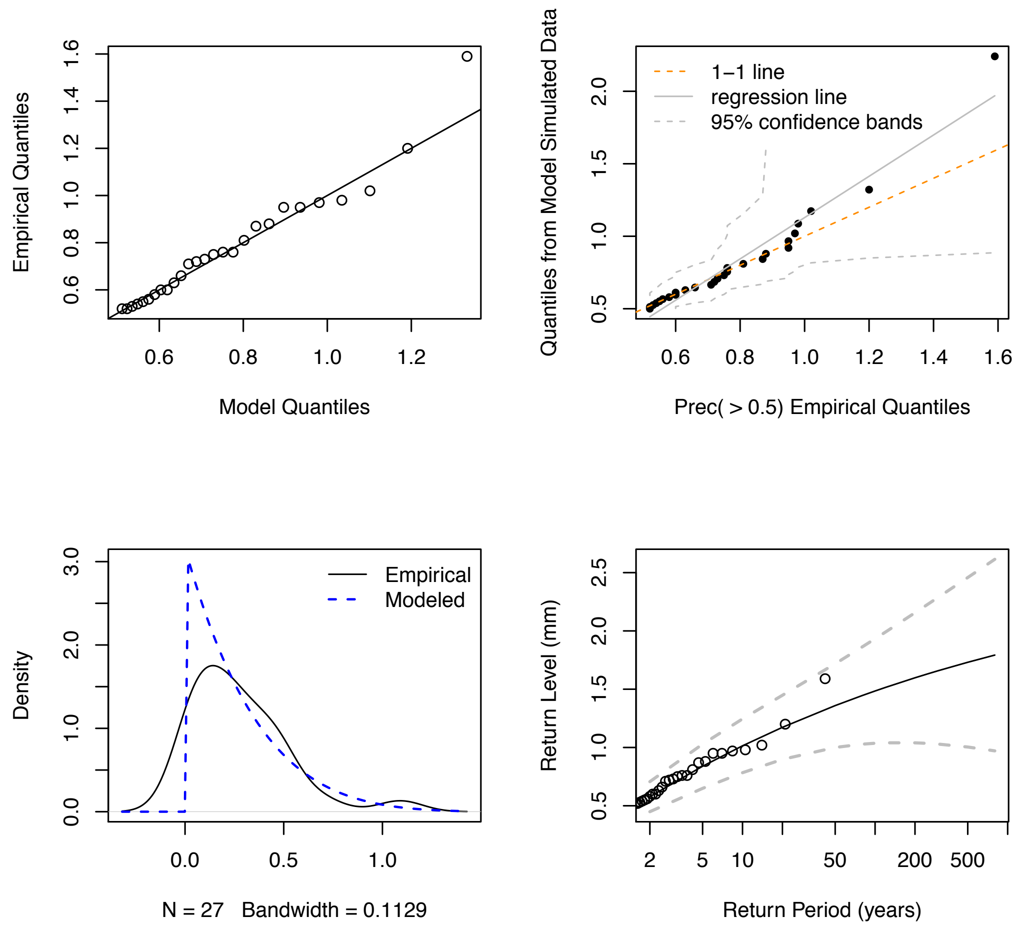
Example: Denver, Colorado July hourly Precipitation (mm)

```
fitGP <- fevd(Prec, Denversp, threshold=0.5,
    type="GP", units="mm",
    time.units="744/year”)

fitGP

plot(fitGP)
```

Generalized Pareto

Example: Denver, Colorado July hourly Precipitation (mm)

fevd(x=Prec, data=Denversp, threshold=0.5, type="GP", units="mm", time.units="744/year", verbose=TRUE)



Generalized Pareto

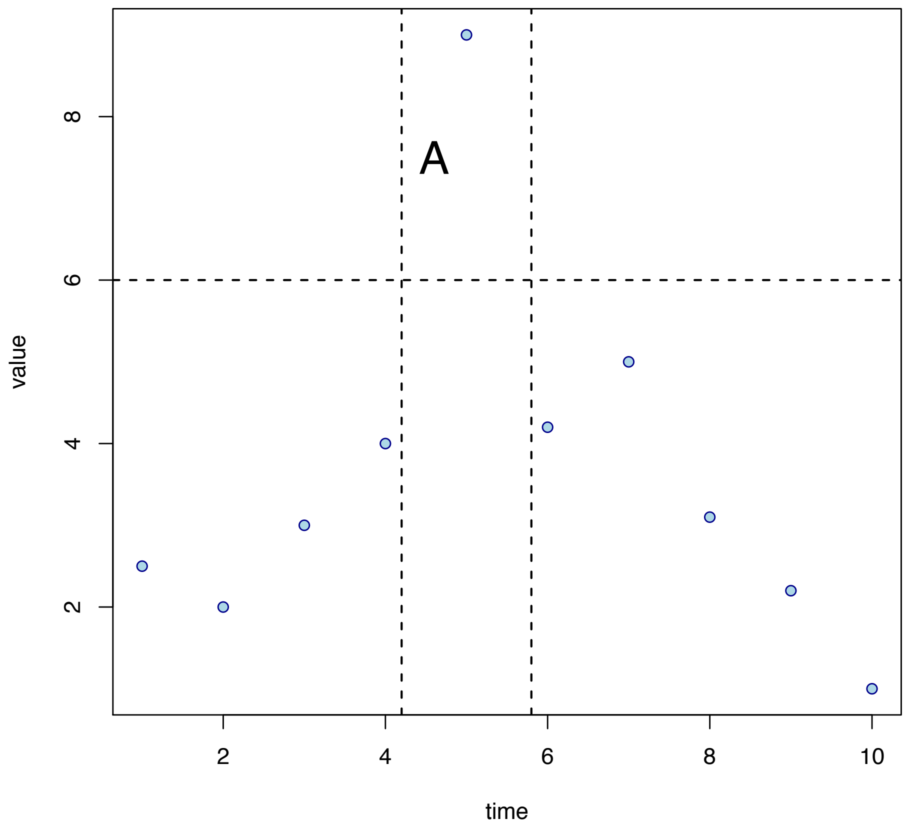
Example: Denver, Colorado July hourly Precipitation (mm)

```
ci(fitGP, type = "parameter")

ci(fitGP)
```

	95% lower CI	Estimate	95% upper CI
$\sigma(u = 0.5$ mm)	0.16	0.32	0.47
ξ	-0.48	-0.15	0.19
100-year return level (mm)	1.04	1.49	1.93

Part V: Extreme Value Analysis

Point Process

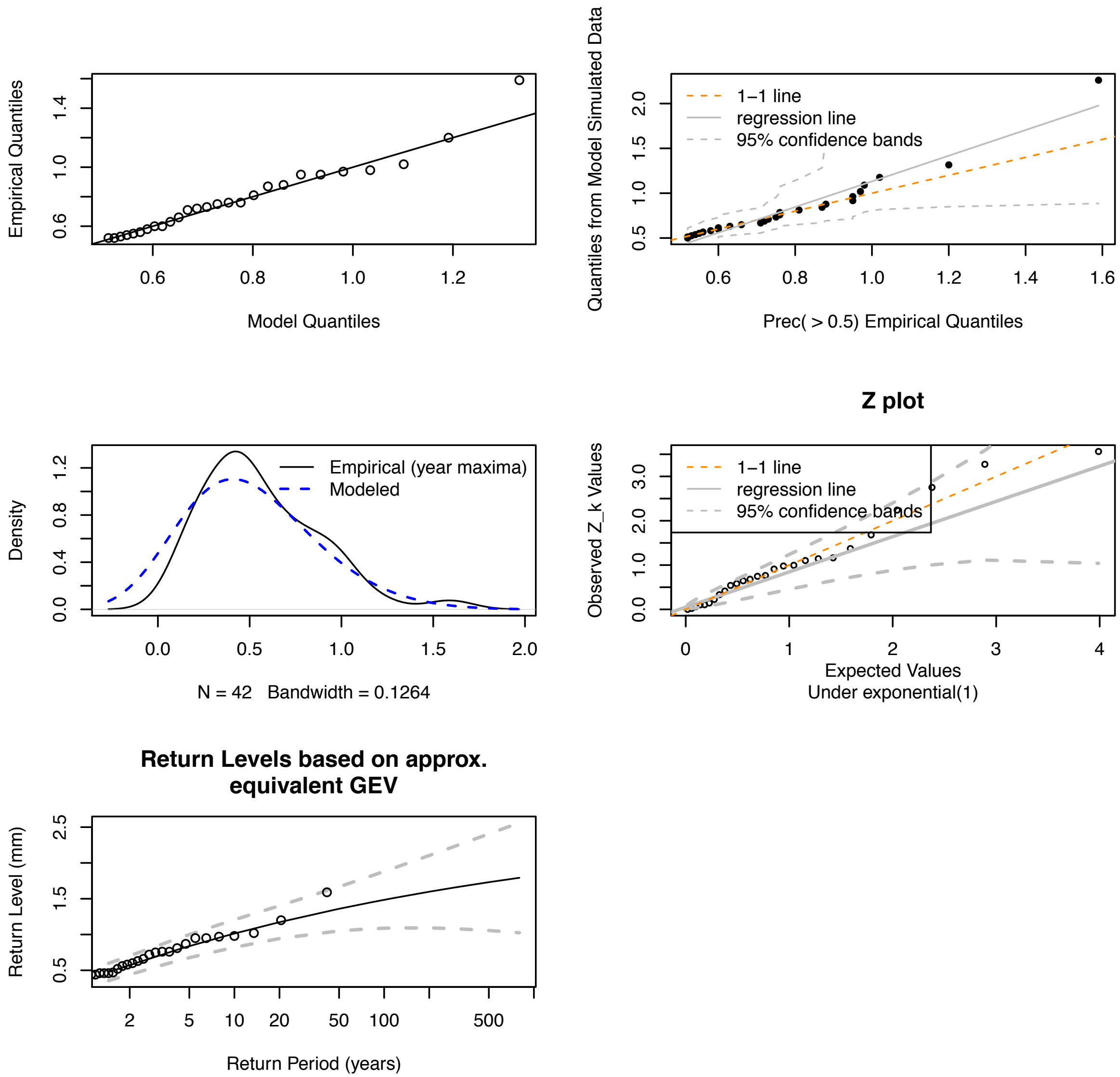


Siméon Denis Poisson

Point Process

```
fitPP <- fevd(Prec, Denversp, threshold=0.5,
    type="PP", units="mm”,
    time.units="744/year")

fitPP

plot(fitPP)
```

Point Process



Point Process

```
ci(fitPP, type = “parameter”)

ci(fitPP)
```

	95% lower CI	Estimate	95% upper CI
μ (mm)	0.21	0.36	0.51
σ (mm)	0.13	0.34	0.55
ξ	-0.48	-0.15	0.19
100-year return level (mm)	1.09	1.48	1.88

Point Process

Also vary the threshold by hour?
And incorporate a diurnal cycle in the location parameter?

```
u <- numeric(dim(Denversp)[1])
u[ Denversp$Hour < 14 ] <- 0.001
u[ Denversp$Hour >= 14 ] <- 0.5

fitPP3 <- fevd(Prec, Denversp,
    threshold = u, location.fun = ~Hour,
    type="PP", units="mm",
    time.units = "744/year")

fitPP3
plot(fitPP3)
```

Ok, maybe not!

Declustering

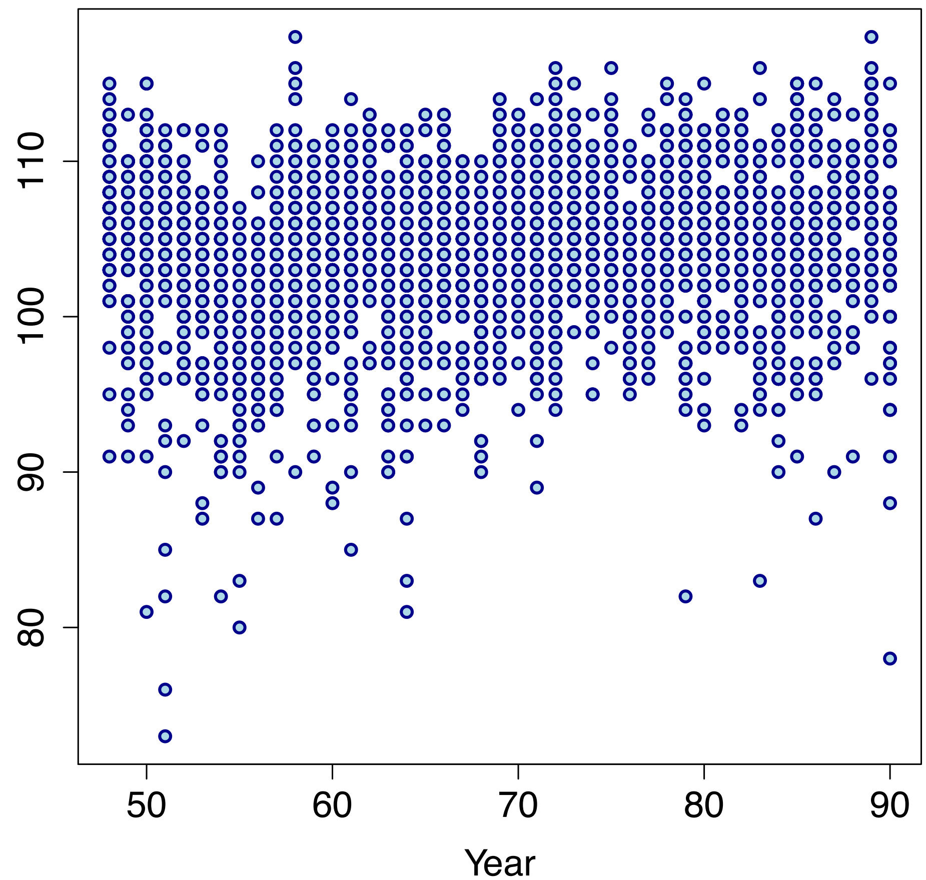
Example: Sky Harbor airport, Phoenix, Arizona July to August maximum temperatures (°F)

```
data(Tphap)

plot(MaxT ~ Year, data = Tphap, pch = 21,
    col = "darkblue", bg = "lightblue",
    lwd = 2, cex.lab = 1.5, cex.axis = 1.5,
    ylab = "")
```

Declustering

NCAR

Example: Sky Harbor airport, Phoenix, Arizona July to August maximum temperatures (°F)



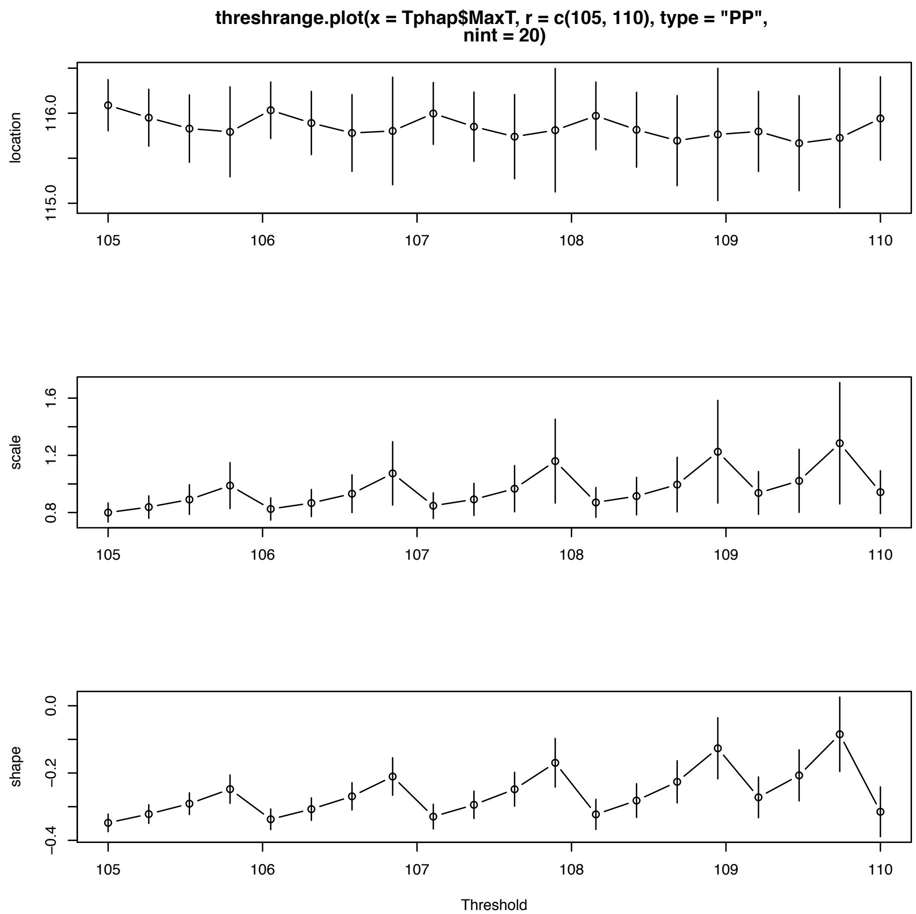
Declustering

Example: Sky Harbor airport, Phoenix, Arizona July to August maximum temperatures (°F)

```
threshrange.plot(Tphap$MaxT,
    r = c(105, 110), type = "PP")
```

Declustering

Example: Sky Harbor airport, Phoenix, Arizona July to August maximum temperatures (°F)



Declustering

Example: Sky Harbor airport, Phoenix, Arizona July to August maximum temperatures (°F)

```
extremalindex(Tphap$MaxT, threshold = 105)
```

θ	Number of Clusters	Run Length
0.21	234	2

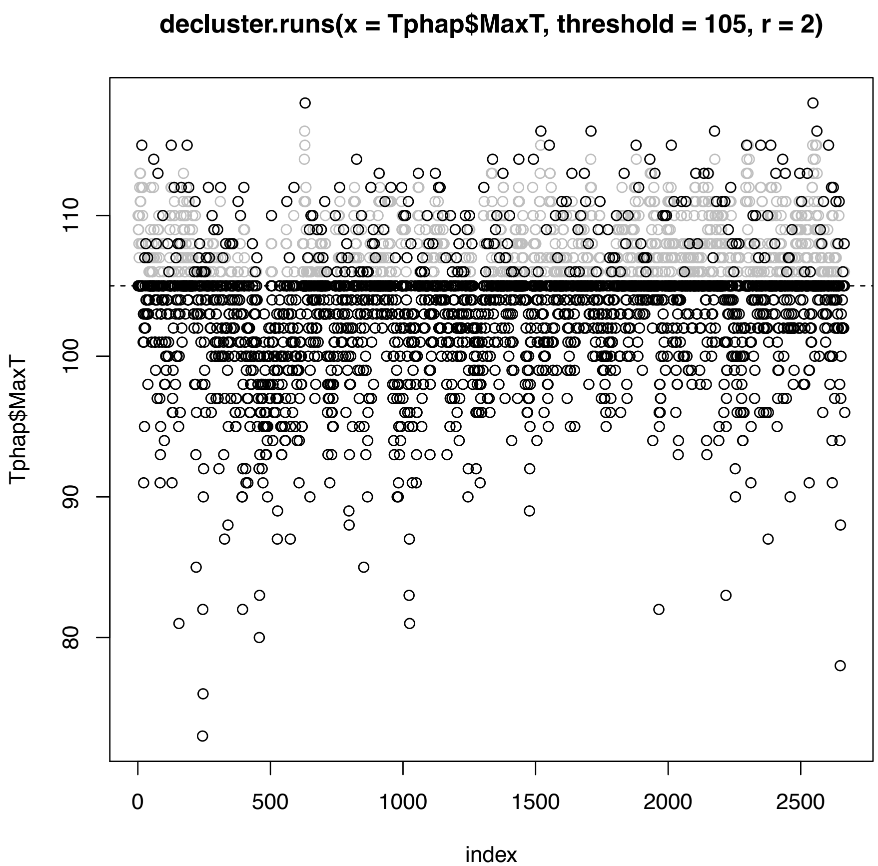
Declustering

Example: Sky Harbor airport, Phoenix, Arizona July to August maximum temperatures (°F)

```
y <- decluster(Tphap$MaxT, threshold = 105,
    r = 2)

y

plot(y)
```

Declustering

Example: Sky Harbor airport, Phoenix, Arizona July to August maximum temperatures (°F)



Declustering

NCAR

Example: Sky Harbor airport, Phoenix, Arizona July to August maximum temperatures (°F)

```
Tphap2 <- Tphap
Tphap2$MaxT.dc <- c(y)

fit <- fevd(MaxT.dc, threshold = 105,
    data = Tphap2, type = "PP",
    time.units = "62/year", units =
    "deg F")

fit

plot(fit)
```

Declustering

Example: Sky Harbor airport, Phoenix, Arizona July to August maximum temperatures (°F)