UNSW SYDNEY

MATH5805 – A Modern Introduction to Extreme Value Theory

Week 10 - Multivariate Extremes: a Point Process Representation (II)

Dr Boris Béranger
School of Mathematics and Statistics
UNSW Sydney

CRICOS Provider Code 00098G

Lecture outline

1. Introduction & Recap
2. The Poisson Process Representation
3. Extremal Dependence Structure
4. Quantifying and Representing Extremal Dependence
5. Parametric Models for Extremal Dependence
6. Extremal Dependence Estimation
7. Appplication to Pollution Data
8. Application to Wind Speed Data



CRICOS Provider Code 00098G

6. Extremal Dependence Estimation



Extremal dependence estimation

We have used the **composite likelihood** approach to estimate the parameters of extremal dependence models.

Rather than relying on the **distribution function** (and thus the exponent function), we consider a second approach based on the **angular density** which we call **approximate likelihood**

From slides 22-24, we know that
if $\mathbf{X}_1, \ldots, \mathbf{X}_n$ are i.i.d. copies of $\mathbf{X}$ on $\mathbb{R}_+^d$ with a distribution in the domain of attraction of an MEVD,
then the sequence $\{R_i/n, \mathbf{W}_i, i = 1, \ldots, n\}$ where $R_i = X_{i,1} + \cdots + X_{i,d}$ and $\mathbf{W}_i = \mathbf{X}_i/R_i$, converges as $n \to \infty$ to the distribution of a PPP with density $\mathrm{d}\Lambda(r, \mathbf{w}) = r^{-2}\mathrm{d}r \times \mathrm{d}H(\mathbf{w})$.

Approximate likelihood estimation

Construction

- Assume that $\mathbf{x}_1, \ldots \mathbf{x}_n \overset{i.i.d.}{\sim} F$.
- **Transform** the data into the sample $\mathbf{z}_1, \ldots, \mathbf{z}_n$ with **unit Fréchet margins**.
- Change the coordinates of the datapoints from Euclidean to **pseudo-polar** by the transformation

$$r_i = z_{i,1} + \ldots + z_{i,d}, \quad \boldsymbol{\omega}_i = \mathbf{z}_i / r_i, \quad i = 1, \ldots, n.$$

- The sequence $\{(r_i, \boldsymbol{\omega}_i), i = 1, \ldots, n : r_i > r_0\}$, where $r_0 > 0$ is a large threshold, comes approximately from a Poisson point process with intensity measure Λ
- Let $A_{r_0} = \{(r, \boldsymbol{\omega}) : r > r_0\}$ be the set of points with radial component larger than r_0, then $N(A_{r_0}) \sim \text{Pois}\,(\Lambda(A_{r_0}))$.

Construction (continued)

Conditionally on $N(A_{r_0}) = m$, the points $\{(r_{(i)}, \boldsymbol{\omega}_{(i)}), i = 1, \ldots, m\}$ are i.i.d. with common density $f(r, \boldsymbol{\omega}) = \mathrm{d}\Lambda(r, \boldsymbol{\omega})/\Lambda(A_{r_0})$.

Assuming that the angular measure H is known apart from a parameter vector $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^p$, we can therefore write down the likelihood as

$$\begin{aligned} L\left(\boldsymbol{\theta}; \left(r_{(i)}, \boldsymbol{\omega}_{(i)}\right), i = 1, \ldots, m\right) &= \mathbb{P}\left(N(A_{r_0}) = m\right) \times \prod_{i=1}^{m} f\left(r_{(i)}, \boldsymbol{\omega}_{(i)}\right) \\ &= \frac{e^{-\Lambda(A_{r_0})} \Lambda(A_{r_0})^m}{m!} \prod_{i=1}^{m} \frac{\mathrm{d}\Lambda(r, \boldsymbol{\omega})}{\Lambda(A_{r_0})} \\ &\propto \prod_{i=1}^{m} h(\boldsymbol{\omega}_{(i)}; \theta). \end{aligned}$$

Consequently we can maximise the log-likelihood $\ell(\boldsymbol{\theta}) \propto \sum_{i=1}^{m} \log h(\boldsymbol{\omega}_{(i)}; \theta)$.

Approximate likelihood estimation

Here we present the asymptotics of the approximate likelihood estimator $\hat{\boldsymbol{\theta}}$.

Let $\ell'(\boldsymbol{\theta}) = \nabla_{\boldsymbol{\theta}} \ell(\boldsymbol{\theta})$ denote the score function.

Since we use an approximation of the true likelihood, from the theory on model misspecification, it follows that

$$\sqrt{n}\left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right) \xrightarrow{d} \mathcal{N}_p\left(\mathbf{0}, \mathbf{J}(\boldsymbol{\theta})^{-1}\mathbf{K}(\boldsymbol{\theta})\mathbf{J}(\boldsymbol{\theta})^{-1}\right), \quad n \to \infty$$

where $\mathcal{N}_p(\boldsymbol{\mu}, \Sigma)$ is the p-dim normal distribution with mean $\boldsymbol{\mu}$ and covariance Σ, and

$$\mathbf{J}(\boldsymbol{\theta}) = -\mathbb{E}\left[\nabla_{\boldsymbol{\theta}} \ell'(\boldsymbol{\theta})\right] \ \textbf{(sensitivity)} \quad \text{and} \quad \mathbf{K}(\boldsymbol{\theta}) = -\mathbb{V}\left[\ell'(\boldsymbol{\theta})\right] \ \textbf{(variability)}.$$

Note that $J(\boldsymbol{\theta})K(\boldsymbol{\theta})^{-1}J(\boldsymbol{\theta})$ is called the *Godambe* or *Sandwich* information matrix. For model selection, we select the model with the lowest value of the Takeuchi information criterion (TIC):

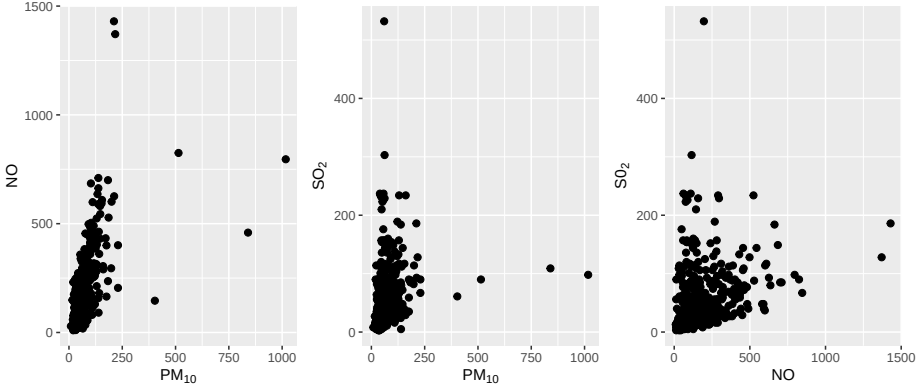
$$\text{TIC} = -2\left[\ell(\hat{\boldsymbol{\theta}}) - \text{tr}\left(K(\hat{\boldsymbol{\theta}})J^{-1}(\hat{\boldsymbol{\theta}})\right)\right]$$

7. Appplication to Pollution Data



Application to pollution data

We are going back to the example of the pollution levels in Leeds, UK with a focus on PM_{10}, NO and SO_2. Let's recall that the unit-Fréchet standardised dataset looks like



Application to pollution data

Pairwise analyses

Using the approximate likelihood procedure we fit the Hüsler-Reiss and Extremal-t models:

Pollutants	Hüsler-Reiss : $\hat{\lambda}$	$\widehat{\vartheta}_{\mathbf{HR}}$	TIC	Extremal-t : $(\hat{\rho}, \hat{\nu})$	$\widehat{\vartheta}_{\mathbf{ET}}$	TIC
PM_{10} & NO	0.64(0.05)	**1.48**	−127.4	0.91(0.01), 6.53(1.90)	**1.43**	−142.0
PM_{10} & SO_2	0.88(0.04)	**1.62**	−151.9	0.95(0.01), 28.84(24.20)	**1.61**	−151.602
NO & SO_2	0.95(0.04)	**1.66**	−180.7	0.98(0.00), 70.97(16.71)	**1.66**	−179.83

Application to pollution data



How did we get those results?

We use routines from the package **ExtremalDep** as follows.

```
library(ExtremalDep)
data(pollution)

Leeds.frechet <- Leeds.frechet[apply(Leeds.frechet, 1,
                                     function(x) !any(is.na(x))),]
# Thresholds on Frechet scale
thresh <- c(quantile(Leeds.frechet[,1],0.92),
            quantile(Leeds.frechet[,2],0.90),
            quantile(Leeds.frechet[,5],0.93))

# Starting values
p0.hr <- 1
p0.et <- c(0.5, 1)
```

Application to pollution data

```r
# Focusing on PM10 and NO
PN.dat <- Leeds.frechet[,c(1,2)]
r.PN <- rowSums(PN.dat)
PN.w <- PN.dat/r.PN
PN.w <- PN.w[order(r.PN, decreasing=TRUE)[1:100],]

# Fitting the Husler-Reiss model
fit.hr.PN <- fExtDep(method="PPP", data=PN.w, model="HR", par.start=p0.hr)
```

```r
fit.hr.PN$par
```

```
## [1] 0.645
```

```r
fit.hr.PN$SE
```

```
## [1] 0.052
```

```r
fit.hr.PN$TIC
```

```
## [1] -127.389
```

```r
index.ExtDep(object="extremal", model="HR", par=fit.hr.PN$par)
```

```
## [1] 1.481073
```

Application to pollution data

```
# Fitting the Extremal-t model
fit.et.PN <- fExtDep(method="PPP", data=PN.w, model="ET", par.start=p0.et)
```

```
fit.et.PN$par
```

```
## [1] 0.911 6.535
```

```
fit.et.PN$SE
```

```
## [1] 0.012 1.908
```

```
fit.et.PN$TIC
```

```
## [1] -141.976
```

```
index.ExtDep(object="extremal", model="ET", par=fit.et.PN$par)
```

```
## [1] 1.429066
```

Application to pollution data

```r
# Focusing on PM10 and SO2
PS.dat <- Leeds.frechet[,c(1,5)]
r.PS <- rowSums(PS.dat)
PS.w <- PS.dat/r.PS
PS.w <- PS.w[order(r.PS, decreasing=TRUE)[1:100],]

# Fitting the Husler-Reiss model
fit.hr.PS <- fExtDep(method="PPP", data=PS.w, model="HR", par.start=p0.hr)
```

```r
fit.hr.PS$par
```

```
## [1] 0.884
```

```r
fit.hr.PS$SE
```

```
## [1] 0.043
```

```r
fit.hr.PS$TIC
```

```
## [1] -151.862
```

```r
index.ExtDep(object="extremal", model="HR", par=fit.hr.PS$par)
```

```
## [1] 1.623304
```

Application to pollution data

```
# Fitting the Extremal-t model
fit.et.PS <- fExtDep(method="PPP", data=PS.w, model="ET", par.start=p0.et)
```

```
fit.et.PS$par
```

```
## [1]  0.950 28.835
```

```
fit.et.PS$SE
```

```
## [1]  0.005 24.202
```

```
fit.et.PS$TIC
```

```
## [1] -151.604
```

```
index.ExtDep(object="extremal", model="ET", par=fit.et.PS$par)
```

```
## [1] 1.611245
```

Application to pollution data

```
# Focusing on NO and SO2
NS.dat <- Leeds.frechet[,c(2,5)]
r.NS <- rowSums(NS.dat)
NS.w <- NS.dat/r.NS
NS.w <- NS.w[order(r.NS, decreasing=TRUE)[1:100],]

# Fitting the Husler-Reiss model
fit.hr.NS <- fExtDep(method="PPP", data=NS.w, model="HR", par.start=p0.hr)
```

```
fit.hr.NS$par
```

```
## [1] 0.949
```

```
fit.hr.NS$SE
```

```
## [1] 0.037
```

```
fit.hr.NS$TIC
```

```
## [1] -180.738
```

```
index.ExtDep(object="extremal", model="HR", par=fit.hr.NS$par)
```

```
## [1] 1.657379
```

Application to pollution data

```
# Fitting the Extremal-t model
fit.et.NS <- fExtDep(method="PPP", data=NS.w, model="ET", par.start=p0.et)
```

```
fit.et.NS$par
```

```
## [1]  0.975 70.972
```

```
fit.et.NS$SE
```

```
## [1]  0.001 16.711
```

```
fit.et.NS$TIC
```

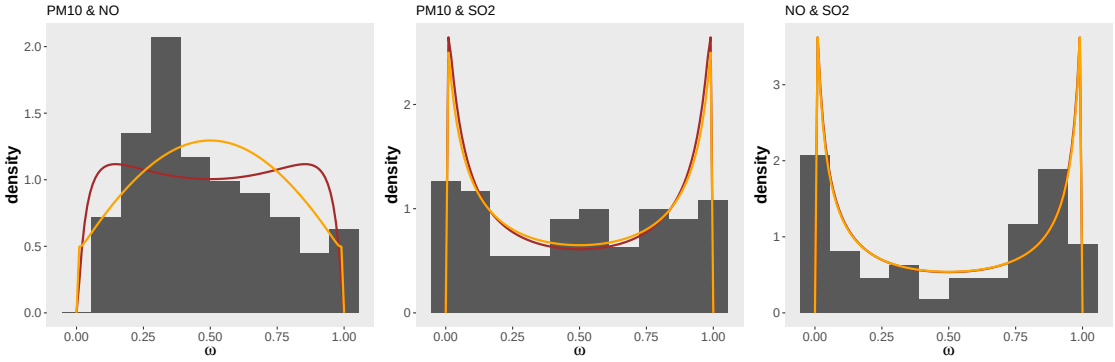
```
## [1] -179.857
```

```
index.ExtDep(object="extremal", model="ET", par=fit.et.NS$par)
```

```
## [1] 1.656966
```

Application to pollution data

We represent the fitted densities



Application to pollution data

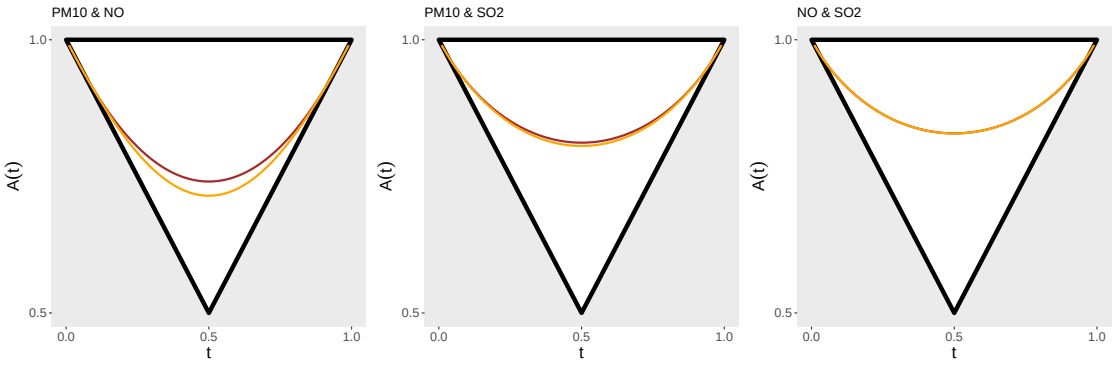
```
vs <- simplex(d=2, n=100)
h.dat <- data.frame(
  t = vs[,1],
  PN_ang_hr = apply(vs, 1, dExtDep, model="HR", par=fit.hr.PN$par)/2,
  PN_ang_et = apply(vs, 1, dExtDep, model="ET", par=fit.et.PN$par, c=0)/2,
  PS_ang_hr = apply(vs, 1, dExtDep, model="HR", par=fit.hr.PS$par)/2,
  PS_ang_et = apply(vs, 1, dExtDep, model="ET", par=fit.et.PS$par, c=0)/2,
  NS_ang_hr = apply(vs, 1, dExtDep, model="HR", par=fit.hr.NS$par)/2,
  NS_ang_et = apply(vs, 1, dExtDep, model="ET", par=fit.et.NS$par, c=0)/2)
```

Application to pollution data

```
ggplot() +
  geom_histogram(aes(x=PN.w[,1], y=after_stat(density)),bins=10) +
  geom_line(data=h.dat, aes(x=t, y=PN_ang_hr), col="brown", lwd=1, lty=1) +
  geom_line(data=h.dat, aes(x=t, y=PN_ang_et), col="orange", lwd=1, lty=1) +
  scale_x_continuous(name=expression(omega))+
  ggtitle("PM10 & NO") +
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
        axis.text=element_text(size=12),
        axis.title=element_text(size=16,face="bold"))
```

Application to pollution data

We now display the estimated Pickands dependence function for both models (Hüsler-Reiss and **Extremal-t**)



Application to pollution data

These plots are obtained by first creating a data.frame with evaluations of the Pickands function.

```
library(patchwork)
Pick.dat <- data.frame(
  t = vs[,1],
  PN_hr = apply(vs, 1, index.ExtDep, object="pickands", model="HR", par=fit.hr.PN$par),
  PN_et = apply(vs, 1, index.ExtDep, object="pickands", model="ET", par=fit.et.PN$par),
  PS_hr = apply(vs, 1, index.ExtDep, object="pickands", model="HR", par=fit.hr.PS$par),
  PS_et = apply(vs, 1, index.ExtDep, object="pickands", model="ET", par=fit.et.PS$par),
  NS_hr = apply(vs, 1, index.ExtDep, object="pickands", model="HR", par=fit.hr.NS$par),
  NS_et = apply(vs, 1, index.ExtDep, object="pickands", model="ET", par=fit.et.NS$par))

Pick.dat <- Pick.dat[-c(1, 100),]
```

Application to pollution data

To generate the plot for PM10 and NO:

```
df <- data.frame(x=c(0,0.5,1), y=c(1,0.5,1))

p1 <- ggplot(data=Pick.dat, aes(x=t, y=PN_hr)) +
  geom_polygon(data = df, aes(x=x,y=y), fill="white", linewidth=2, col="black") +
  geom_line(data=Pick.dat, aes(x=t, y=PN_hr), col="brown", lwd=1, lty=1) +
  geom_line(data=Pick.dat, aes(x=t, y=PN_et), col="orange", lwd=1, lty=1) +
  scale_x_continuous(name=expression(t), limits=c(0, 1), breaks=c(0, 0.5, 1)) +
  scale_y_continuous(name=expression(A(t)), limits=c(0.5, 1), breaks=c(0.5, 1)) +
  ggtitle("PM10 & NO") +
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
        axis.text=element_text(size=12),
        axis.title=element_text(size=16,face="bold"))
```

Application to pollution data

We are interested in combinations of the events $PM_{10} > 125$, $NO > 359$ and $SO_2 > 125$, which correspond to the 92, 90 and 93% upper quantiles.

Event	Excess / n	Empirical Est.	HR Est.	ET Est.
$PM_{10} > 125$ and $NO > 359$	27/528	$0.051(0.032, 0.070)$	0.046	0.050
$PM_{10} > 125$ and $SO_2 > 125$	9/528	$0.017(0.008, 0.032)$	0.029	0.028
$NO > 359$ and $SO_2 > 125$	6/528	$0.011(0.004, 0.025)$	0.030	0.027

Application to pollution data

```
## Focusing on PM10 and NO
excess_PN <- sum((Leeds.frechet[,1]>thresh[1]) *
                   (Leeds.frechet[,2]>thresh[2]), na.rm=T)
conf.int(excess_PN, nrow(PN.dat))
```

```
##          est  CIinf  CIsup
## 1 0.05113636 0.0323 0.0699
```

```
pExtDep(thresh[1:2], model="HR", par=fit.hr.PN$par, type="upper")
```

```
## [1] 0.04648099
```

```
pExtDep(thresh[1:2], model="ET", par=fit.et.PN$par, type="upper")
```

```
## [1] 0.05038621
```

Application to pollution data

```
## Focusing on PM10 and SO2
excess_PS <- sum((Leeds.frechet[,1]>thresh[1]) *
                   (Leeds.frechet[,5]>thresh[3]), na.rm=T)
conf.int(excess_PS, nrow(PS.dat))
```

```
##          est       CIinf      CIsup
## 1 0.01136364 0.004181356 0.02456881
```

```
pExtDep(thresh[c(1,3)], model="HR", par=fit.hr.PS$par, type="upper")
```

```
## [1] 0.02915558
```

```
pExtDep(thresh[c(1,3)], model="ET", par=fit.et.PS$par, type="upper")
```

```
## [1] 0.02847267
```

Application to pollution data

```
## Focusing on NO and SO2
excess_NS <- sum((Leeds.frechet[,2]>thresh[2]) *
                   (Leeds.frechet[,5]>thresh[3]), na.rm=T)
conf.int(excess_NS, nrow(NS.dat))
```

```
##           est       CIinf      CIsup
## 1 0.005681818 0.001173264 0.01651421
```

```
pExtDep(thresh[2:3], model="HR", par=fit.hr.NS$par, type="upper")
```

```
## [1] 0.0296386
```

```
pExtDep(thresh[2:3], model="ET", par=fit.et.NS$par, type="upper")
```

```
## [1] 0.02747804
```

Application to pollution data

Note that some of the above steps have required loading functions/objects from the analysis performed in the Week 8 Lecture.

Note also that return levels plots can be produced using the same code as in the Week 8 Lecture notes. These are therefore moitted here.

8. Application to Wind Speed Data



Application to wind speed data

Data

- Type: Weekly maximum wind speed (km/h)
- Locations: 4 monitoring stations across Oklahoma, USA.
- Location names: CLOU, CLAY, PAUL & SALL.
- Period: March-May period during 1996-2012.

The observations are **marginally pre-transformed to unit Fréchet distributions**.

Application to wind speed data

Use approximate likelihood to fit the Hüsler-Reiss and Extremal-t models using the 50 observations with largest radial component.

	Hüsler-Reiss Estimates		Extremal-t	
Stations	Estimates	TIC	estimates	TIC
CLOU CLAY SALL	$(0.84, 0.91, 0.87)$	-204.72	$(0.95, 0.94, 0.95, 25.75)$	$\mathbf{-206.266}$
CLOU CLAY PAUL	$(0.84, 0.99, 0.94)$	$\mathbf{-231.03}$	$(0.98, 0.97, 0.98, 69.78)$	-229.91
CLAY SALL PAUL	$(0.81, 0.88, 0.87)$	-199.74	$(0.94, 0.93, 0.93, 19.50)$	$\mathbf{-201.63}$
CLOU SALL PAUL	$(0.91, 0.95, 0.94)$	$\mathbf{-225.69}$	$(0.96, 0.95, 0.96, 36.35)$	-225.19

Application to wind speed data - Model fits

```
data(Wind)

# Starting values
p0.hr <- c(rep(1,3))
p0.et <- c(rep(0.5,3),1)

# CLOU CLAY SALL Stations
r.CCS <- rowSums(CLOU.CLAY.SALL)
CCS.w <- CLOU.CLAY.SALL/r.CCS
CCS.w <- CCS.w[order(r.CCS, decreasing=TRUE)[1:50],]
```

Application to wind speed data - Model fits

```
hr1 <- fExtDep(method="PPP", CCS.w, model="HR", p0.hr)
c(hr1$par, hr1$TIC)
```

```
## [1]    0.844    0.914    0.871 -204.724
```

```
et1 <- fExtDep(method="PPP", CCS.w, model="ET", p0.et)
c(et1$par, et1$TIC)
```

```
## [1]    0.950    0.942    0.948   25.753 -206.266
```

Application to wind speed data - Model fits

```
# CLOU CLAY PAUL Stations
r.CCP <- rowSums(CLOU.CLAY.PAUL)
CCP.w <- CLOU.CLAY.PAUL/r.CCP
CCP.w <- CCP.w[order(r.CCP, decreasing=TRUE)[1:50],]

hr2 <- fExtDep(method="PPP", CCP.w, model="HR", p0.hr)
c(hr2$par, hr2$TIC)
```

```
## [1]    0.839    0.993    0.937 -231.027
```

```
et2 <- fExtDep(method="PPP", CCP.w, model="ET", p0.et)
c(et2$par, et2$TIC)
```

```
## [1]    0.981    0.973    0.976   69.775 -229.913
```

Application to wind speed data - Model fits

```
# CLAY SALL PAUL Stations
r.CSP <- rowSums(CLAY.SALL.PAUL)
CSP.w <- CLAY.SALL.PAUL/r.CSP
CSP.w <- CSP.w[order(r.CSP, decreasing=TRUE)[1:50],]

hr3 <- fExtDep(method="PPP", CSP.w, model="HR", p0.hr)
c(hr3$par, hr3$TIC)
```

```
## [1]    0.806    0.878    0.869 -199.740
```

```
et3 <- fExtDep(method="PPP", CSP.w, model="ET", p0.et)
c(et3$par, et3$TIC)
```

```
## [1]    0.944    0.933    0.932   19.498 -201.363
```

Application to wind speed data - Model fits

```r
# CLOU SALL PAUL Stations
r.ClSP <- rowSums(CLOU.SALL.PAUL)
ClSP.w <- CLOU.SALL.PAUL/r.ClSP
ClSP.w <- ClSP.w[order(r.ClSP, decreasing=TRUE)[1:50],]

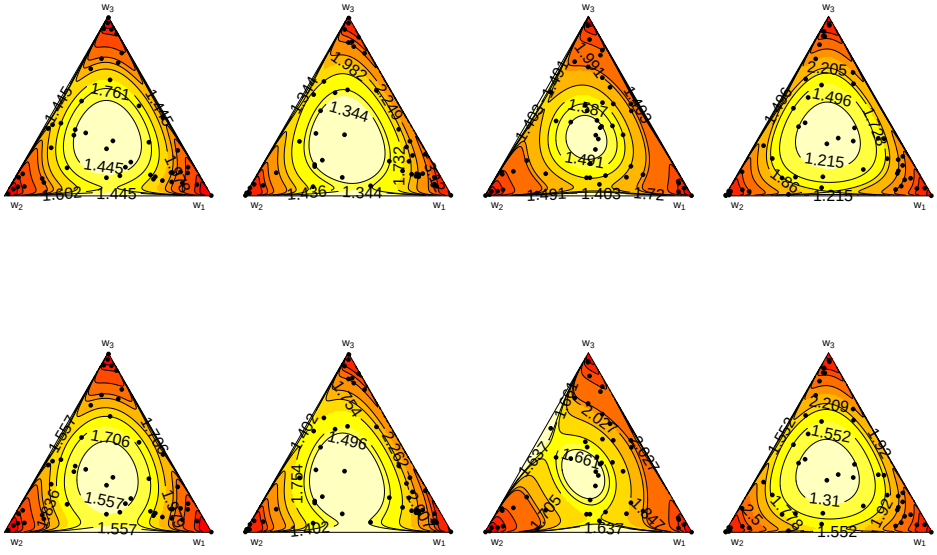
hr4 <- fExtDep(method="PPP", ClSP.w, model="HR", p0.hr)
c(hr4$par, hr4$TIC)
```

```
## [1]    0.909    0.954    0.936 -225.690
```

```r
et4 <- fExtDep(method="PPP", ClSP.w, model="ET", p0.et)
c(et4$par, et4$TIC)
```

```
## [1]    0.958    0.954    0.955   36.349 -225.192
```

Application to wind speed data - Fitted densities



Application to wind speed data - Fitted densities

```
par(mfrow=c(2,4), mar=c(0, 0.4, 0.4, 0.4))
plot_ExtDep(model="HR", par=hr1$par, data=CCS.w)
plot_ExtDep(model="HR", par=hr2$par, data=CCP.w)
plot_ExtDep(model="HR", par=hr3$par, data=CSP.w)
plot_ExtDep(model="HR", par=hr4$par, data=ClSP.w)
plot_ExtDep(model="ET", par=et1$par, data=CCS.w)
plot_ExtDep(model="ET", par=et2$par, data=CCP.w)
plot_ExtDep(model="ET", par=et3$par, data=CSP.w)
plot_ExtDep(model="ET", par=et4$par, data=ClSP.w)
```

Application to wind speed data - Dependence summaries

Calculating **dependence summaries** such as the extremal coefficient and the coefficient of upper tail dependence can be done following the same code as in the Week 8 lecture.

Calculating **(conditional) probabilities of exceeding some large values** can also be performed using the code presented in the Week 8 Lecture.

These quantities should be computing when performing an analysis but are omitted here.