Statistical Modeling of Spatial Extremes II
Computing & Applications

Manuele Leonelli

First LARS-IASC School on Computational Statistics and Data Science
Federal University of Bahia, Salvador, Brazil
November 17th, 2018

University of Glasgow

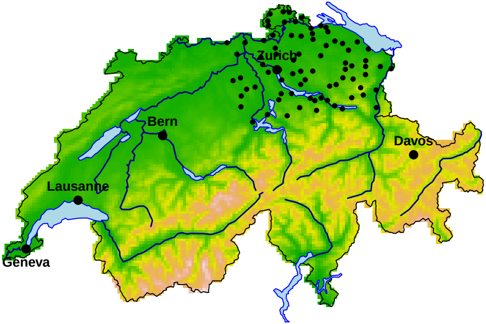
Plan for this session

1. Exploratory analysis for spatial extremes
2. Fitting models using the `spatialExtremes` package
    - Spatial GEVs
    - Copulae
    - Extreme Copulae
    - Max-stable processes
3. Diagnostics and extreme inference
4. Concurrent probabilities

Swiss precipitation data set

The dataset includes the summer annual maxima precipitation in the north-east of Switzerland at 79 weather stations between 1962-2008.

```
library(spatialExtremes)
data(rainfall) #rainfall dataset
data(swissalt) #elevation in Switzerland (for map)
```



Swiss precipitation data set

The command `data(rainfall)` creates two matrices:

- a 47×57 matrix `rain` where each row refers to one year and each column to a weather station.
- a 57×3 matrix `coord` where each row refers to one year and the columns are as follows:
  1. Latitude
  2. Longitude
  3. Altitude

Spatial dependence

Intuitively stations closer to each other will have a stronger extreme dependence.

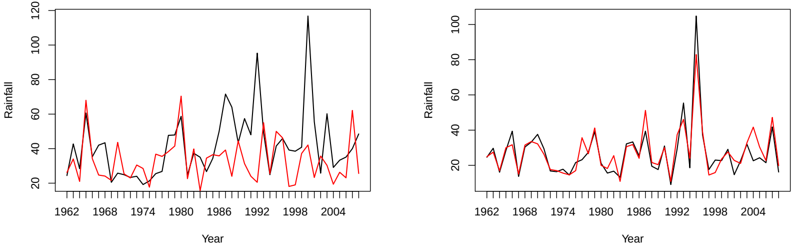


Figure: Annual maxima of two stations at 112km (left plot) and 5km (right plot) distance

Modelling spatial data

The modelling can be split into two tasks:

1. Characterizing how the marginal distribution varies over a study region;
2. Capturing the residual spatial dependence assuming that the marginal effects have been accounted for.

A simplistic approach for part 1 would be to model individually the marginal distributions at each location. This is often done in multivariate applications, but for spatial ones this becomes critical:

- No straightforward interpolation of the results to make inference at unobserved locations.
- Because marginals are often dependent, the results are more robust and less uncertain with multivariate modelling.

Spatial GEV

The function `fitspatGEV` fits the spatial GEV model where location $\mu(\boldsymbol{x})$, scale $\sigma(\boldsymbol{x})$ and shape $\xi(\boldsymbol{x})$ are allowed to vary with covariate levels.

```
fitspatgev(data, covariables, loc.form,
scale.form, shape.form,...)
```

In the spatial setting natural covariables are latitude and longitude, but others can be included.

The function `symbolplot` can be used to choose a starting model.

Exploratory graph - Location

Figure: Location parameter departure from mean

Exploratory graph - Scale

Figure: Scale parameter departure from mean

Exploratory graph - Shape

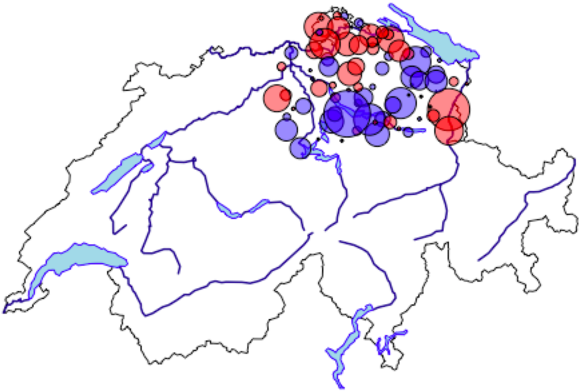


Figure: Shape parameter departure from mean

Fitting spatial GEVs

Let’s fit our fist model!

```
loc.form <- scale.form <- y~lon+lat; shape.form <- y~1
fit <-fitspatgev(rain,coord,loc.form,scale.form,shape.form)

      Model: Spatial GEV model
   Deviance: 29338.55
        TIC: 29513.81

    Location Parameters:
locCoeff1  locCoeff2  locCoeff3
 20.65574    0.06577   -0.15918
       Scale Parameters:
scaleCoeff1  scaleCoeff2  scaleCoeff3
    8.96763      0.01818     -0.04757
       Shape Parameters:
shapeCoeff1
     0.1876
```

Model selection

Let's also fit the following two models

```
loc.form <- y~lon*lat; scale.form <- y~lon+lat; shape.form
fit1<-fitspatgev(rain,coord,loc.form,scale.form,shape.form)

loc.form <- scale.form <- shape.form <- y~1
fit2<-fitspatgev(rain,coord,loc.form,scale.form,shape.form)
```

We can use the TIC criterion to select a model:

```
TIC(fit,fit1,fit2)

   fit1      fit     fit2
29507.36 29513.81 29962.12
```

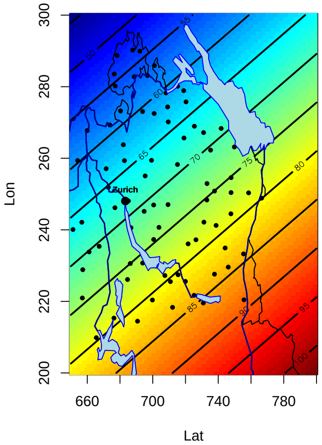
Model selection

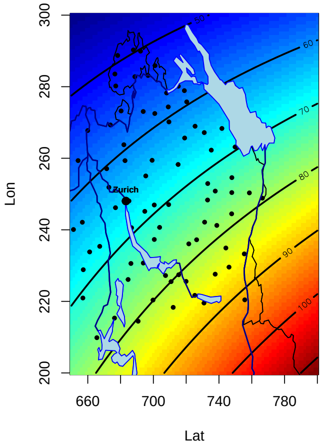
Furthermore, since the models are nested, we can compare them using standard anova tests.

```
anova(fit2,fit)

Analysis of Variance Table
     MDf Deviance Df  Chisq Pr(> sum lambda Chisq)
fit    7    29338
fit2   9    29201  2 137.98              < 2.2e-16 ***
---

anova(fit,fit1)

Analysis of Variance Table
     MDf Deviance Df  Chisq Pr(> sum lambda Chisq)
fit    7    29338
fit1   8    29323  1 15.978                 0.0235 *
---
```

25-years return levels

The function `predict` can be used to produce return levels over the region of interest.





Latent modelling

Spatial extreme dependence can also be modelled by letting the GEV parameters vary smoothly according to a stochastic latent process.
For instance

$$\mu(\boldsymbol{x}) = f_\mu(\boldsymbol{x}, \boldsymbol{\beta}_\mu) + S_\mu(\boldsymbol{x}, \boldsymbol{\theta}_\mu)$$

where f_μ is a deterministic function depending on regression parameters $\boldsymbol{\beta}_\mu$ and S_μ is a zero mean stationary Gaussian process with covariance function parametrised by $\boldsymbol{\theta}_\mu$. Similarly for $\sigma(\boldsymbol{x})$ and $\xi(\boldsymbol{x})$.

Then the maxima at sites $(x_1, \dots, x_D)$ are assumed independent with

$$Y_i(x_d)|(\mu(x_d), \sigma(x_d), \xi(x_d)) \sim GEV(\mu(x_d), \sigma(x_d), \xi(x_d))$$

Here we use the exponential covariance function $\alpha \exp(-||h||/\lambda)$ with sill parameter α and range λ.

Let's set up the model

```
# Functional forms
loc.form <- scale.form <- y~lon+lat; shape.form <- y~1

# Prior distributions
hyper <- list()
hyper$betaMeans<-list(loc=rep(0,3),scale=rep(0,3),shape=0)
hyper$betaIcov<-list(loc= diag(rep(1/1000,3)),
scale= diag(rep(1/1000,3)),shape=1/10)

hyper$sills<-list(loc= c(1,12), scale=c(1,1),
shape = c(1,0.04))
hyper$ranges<-list(loc = c(5,3), scale= c(5,3),
shape = c(5,3))
hyper$smooths<-list(loc = c(1,1), scale= c(1,1),
shape = c(1,1))
```

Let's set up the model

```
# Proposal distributions parameters
prop<-list(gev=c(3,0.1,0.3), ranges=c(1,0.8,1.2),
 smooths=rep(0,3))

# Starting values
start<-list(sills=c(10,10,0.5), ranges=c(20,10,10),
smooths=c(1,1,1),
beta=list(loc=c(25,0,0),scale=c(33,0,0),shape=0.001))

# We can now estimate the model
chain <- latent(rain,coord[,1:2],"powexp",loc.form,
scale.form,shape.form,hyper=hyper,prop=prop,
start=start,n=1000,burn.in=1000,thin=10)
```

Model output

```
  Regression Parameters:
      Location Parameters:
           lm1        lm2        lm3
ci.lower   -10.93121  -0.01029   -0.19006
post.mean   28.37926   0.03945   -0.11757
ci.upper    66.68857   0.09020   -0.03503

         Scale Parameters:
           lm1         lm2          lm3
ci.lower   -4.7572072  -0.0009813  -0.0509904
post.mean   6.8192104   0.0148236  -0.0311692
ci.upper   19.7343389   0.0295477  -0.0100655

         Shape Parameters:
           lm1
ci.lower   0.08389
post.mean  0.16872
ci.upper   0.25172
```

Model output

```
  Latent Parameters:
      Location Parameters:
           sill    range   smooth
ci.lower    5.331  12.570   1.000
post.mean   9.928  23.077   1.000
ci.upper   17.615  38.515   1.000

      Scale Parameters:
           sill     range    smooth
ci.lower    0.1527   7.6265   1.0000
post.mean   0.3943  19.5675   1.0000
ci.upper    0.8403  37.2526   1.0000

      Shape Parameters:
           sill       range       smooth
ci.lower    0.004291   9.354036   1.000000
post.mean   0.008395  22.117331   1.000000
ci.upper    0.015315  37.531265   1.000000
```

25-years return levels (`map.latent`)

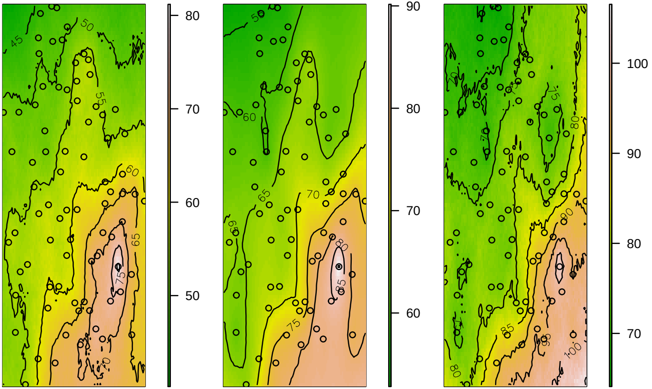


Figure: 25-years return level maps: average pointwise estimate (center) and 95% pointwise credible intervals (left and right)

Simulating from the estimated model

The conditional independence assumption of the maxima given the parameters leads to an unrealistic spatial structure, although pointwise prediction was sensible.

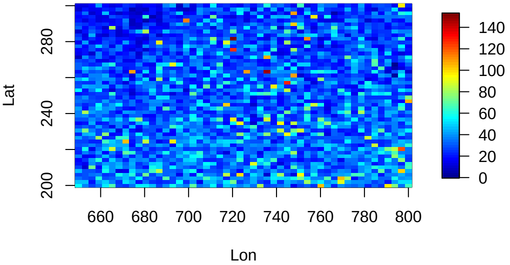


Figure: One simulation from the estimated latent model.

Copulae

- A second approach for spatial data is to use copula functions, as illustrated for the multivariate setting, with the difference now that the covariance needs to be a function of distance between locations.
- Possible choices are the Whittle-Matérn (`whitmat`), the Cauchhy (`cauchy`), the stable (`powexp`) and the exponential (`powexp` with `smooth`=1).
- All these functions are stationary (do not change with location) and isotropic (only depend on distance).
- The function `fitcopula` fits Gaussian and Student-t copulae: these are not in the domain of attraction of max-stable processes.
- The function `fitmaxstab` can be used to fit the extremal-T copula.

```
fit_whitmat<- fitcopula(rain,coord[,1:2],copula="gaussian",
cov.mod = "whitmat",loc.form=loc.form,
shape.form = shape.form,scale.form=scale.form)

         Deviance: 25319.19
              AIC: 25339.19
Covariance Family: Whittle-Matern

    Location Parameters:
locCoeff1  locCoeff2  locCoeff3
 23.43327    0.02435   -0.05788
       Scale Parameters:
scaleCoeff1  scaleCoeff2  scaleCoeff3
    9.99000      0.01366     -0.04041
       Shape Parameters:
shapeCoeff1
    0.06724
  Dependence Parameters:
 nugget    range   smooth
 0.1817  30.0906   0.8664
```

Model choice

The AIC criterion can be used for Gaussian and T, as these do not require composite likelihood methods.

Model	Matern	Cauchy	Stable	Exponential
Gaussian	**25339**	25381	25565	25591
Student-T	25339	25365	25235	**25228**

Table: AIC values for the Gaussian and T-model with different correlation functions.

F-madogram

The F-madogram can be used to estimate the extremal coefficient of the data and the model as well as to investigate the model fit (see function `fmadogram`).

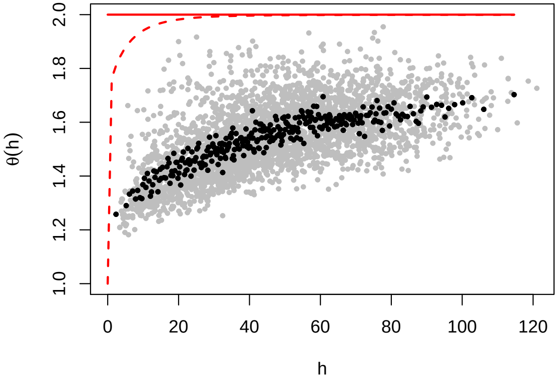


Figure: F-madogram for the rain data. Full red line - estimate from the Gaussian copula; Dashed red line - estimate from the T copula

25-years Return Levels

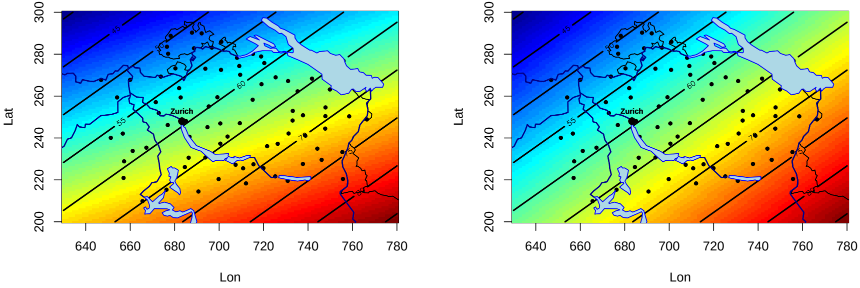


Figure: 25-years returns from the Gaussian (left) and T (right) models.

Simulations from the estimated models

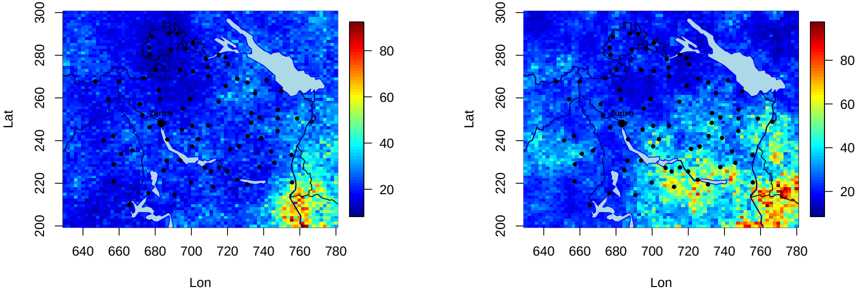


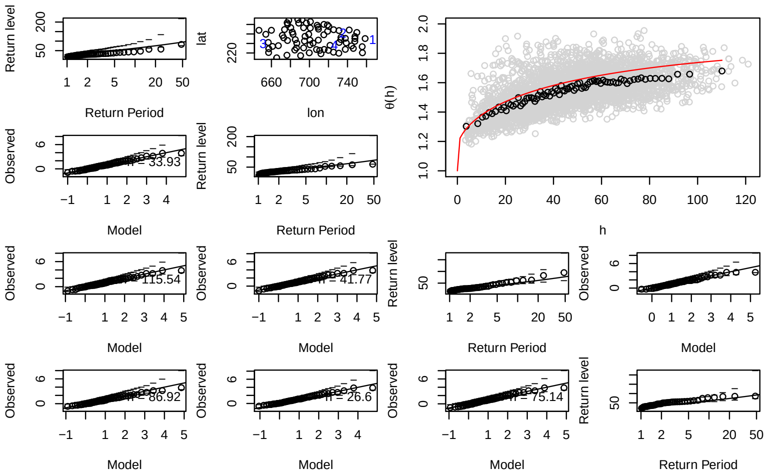
Figure: One simulation from the Gaussian (left) and T (right) models.

Fitting the extreme-T copula

- Extreme T-copulae can be fitted using the `fitmaxstab` function.
- Let's look at the syntax here.
- After fitting extreme-T models with different correlations, model comparison can be carried out using the TIC criterion.

	Matern	Cauchy	Stable	Exponential
AIC	2250784	2250803	**2250721**	2250741

Diagnostics - plot(fitmaxstab)



Additional plots

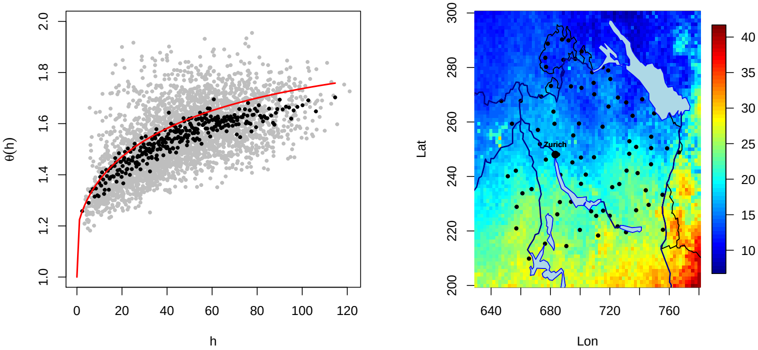


Figure: Left: Madogram with estimated fit from the extremal T model; Right: one simulation from the estimated model

Fitting max-stable models

- Max-stable models can be fitted in R just as we did with the extremal-T copula.
- The `spatialExtreme` package gives the possibility of fitting the Smith model, the Schlather model, the geometric Gaussian model and the Brown-Resnick model.
- The Schlather and geometric Gaussian models can be fitted with the already mentioned correlation structures. Here only the best one (in terms of TIC is reported).
- Let's start comparing the different models

	Smith	Schlather	Geometric Gauss	Brown-Resnick
TIC	2273666	2254345	**2251986**	2252210

Madograms

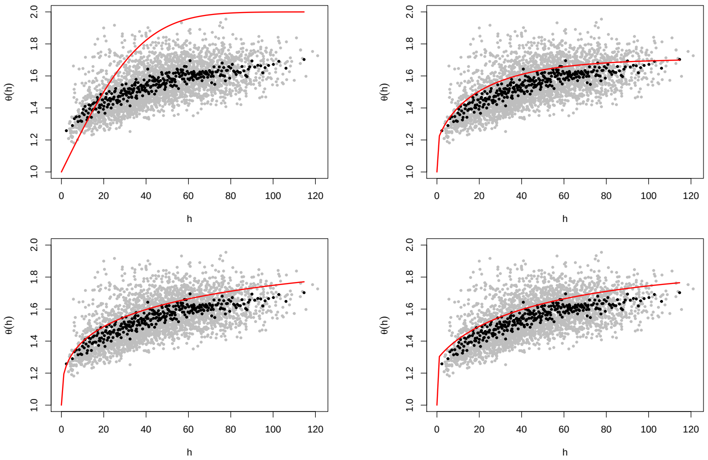


Figure: Smith (top-left); Schlather (top-right); Brown-Resnick (bottom-left); Geometric Gaussian (bottom-right).

Simulations

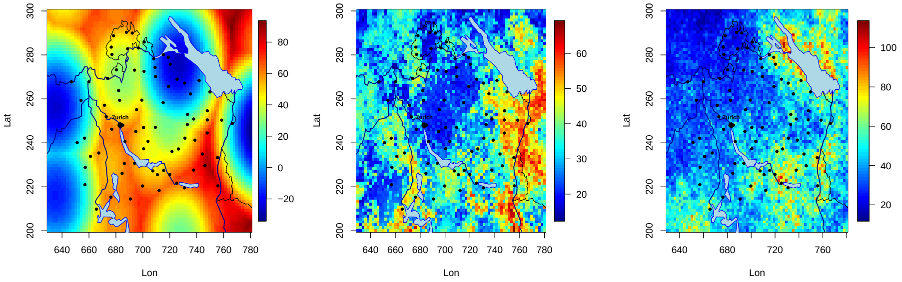
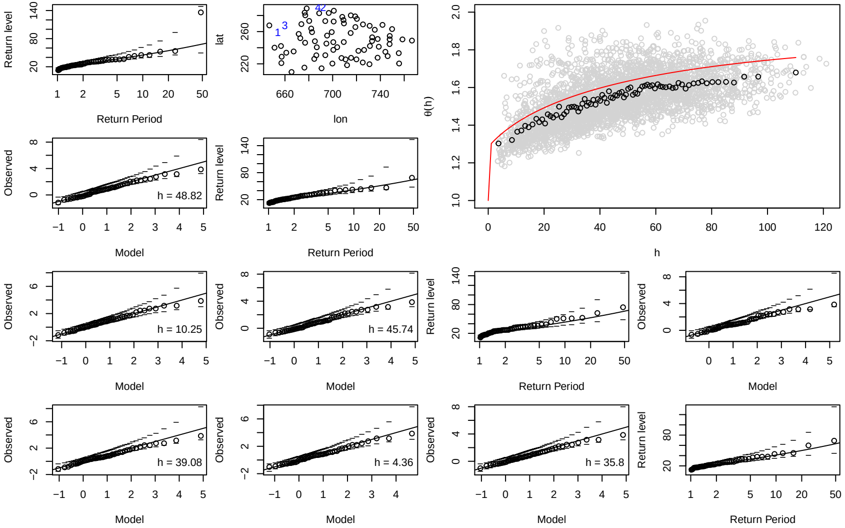


Figure: Smith (left); Schlather (center); Geometric Gauss (right)

Diagnostics



Predictions

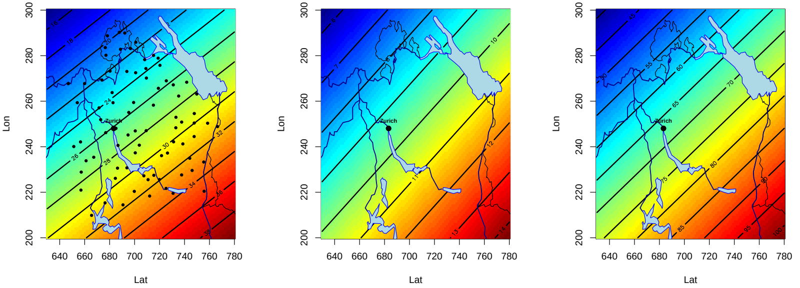
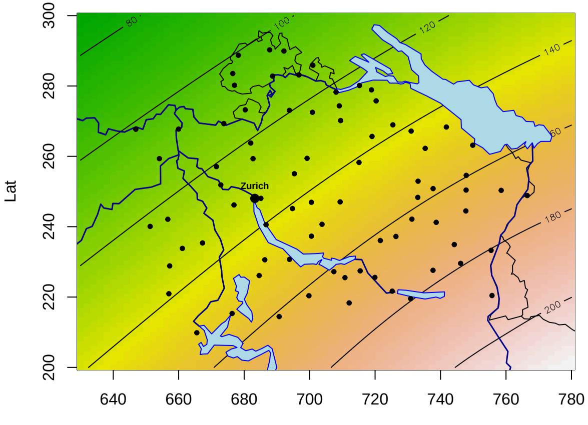


Figure: Prediction of location (left), scale (center) and 25-years returns (right)

Conditional return levels (`condmap`)

Plot $(x, x_{T_2|T_1}(x))$ where

$$Pr(Z(x) > z_{T_2|T1}(x)|Z(x_*) > z_{T_1}) = 1 - 1/T_2$$



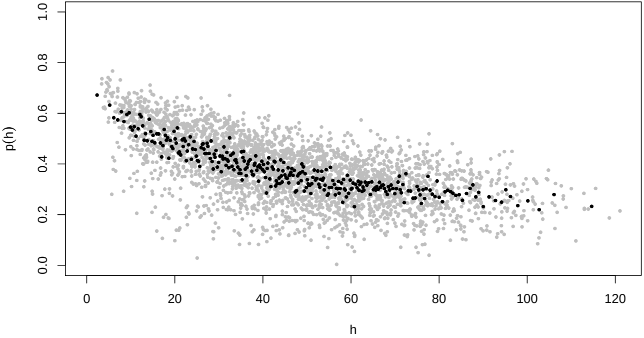
Concurrent probabilities

- The methods investigated so far are mostly concerned with the magnitudes of extreme events. Little attention has been paid in the literature on the timing of extremes.
- Extreme concurrence probabilities quantify the possibility that extremes happen at different locations simultaneously, e.g. in the same year or day.
- We denote with $p(x_1, x_2)$ the extremal concurrence probability between locations x_1 and x_2.
- Currently only empirical estimation is implemented in the package `spatialExtremes`.

Dombry, C., Ribatet, M., & Stoev, S. (2018). Probabilities of concurrent extremes. *Journal of the American Statistical Association* (to appear).

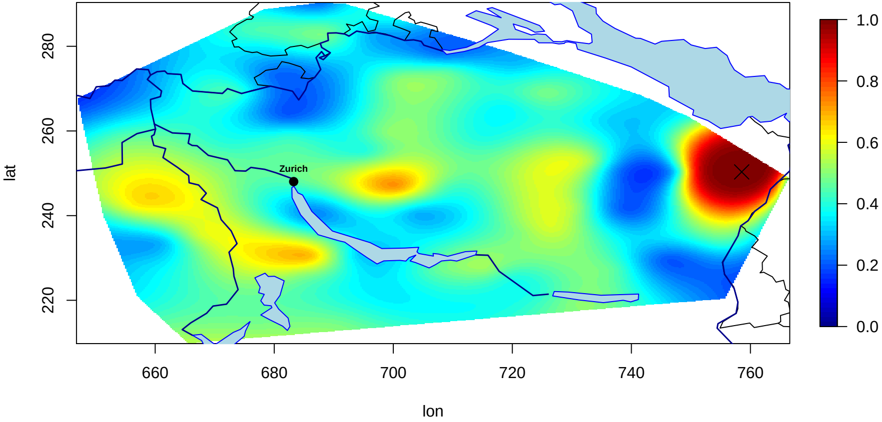
Concurrence probabilities

The function `concprob` computes the extreme concurrence probability between every pair of locations.

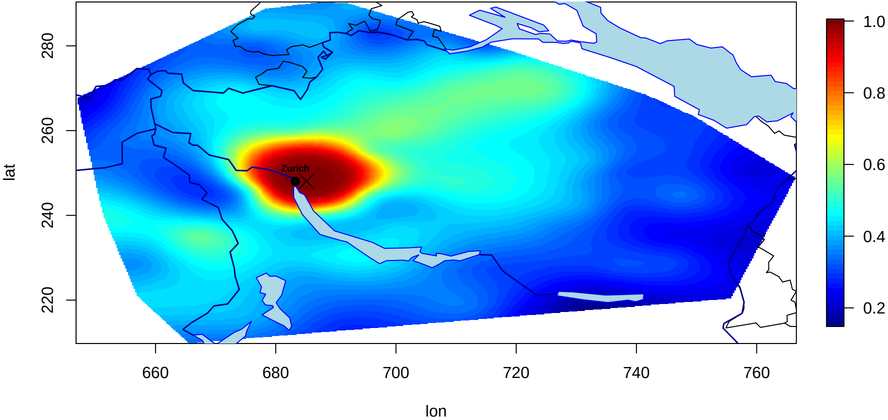


Concurrence maps

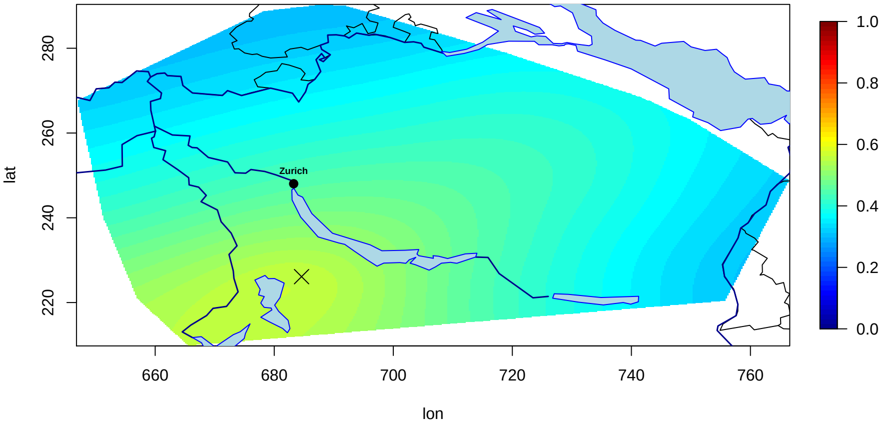
It is also interested to map the concurrence probabilities fixing one of the two locations: i.e. plot $\{(s, p(x, s)) : s \in \mathcal{X}\}$ for a fixed $x \in \mathcal{X}$



Concurrence maps



Concurrence maps



Integrated concurrence

The previous maps, though useful, depend on the choice of an anchor point. To bypass this, consider the integrated concurrence probability

$$I(s) = \int_{x \in \mathcal{X}} p(x, s) ds, \quad x \in \mathcal{X}$$

A measure of how fast the dependence in extremes decreases when moving away from x.

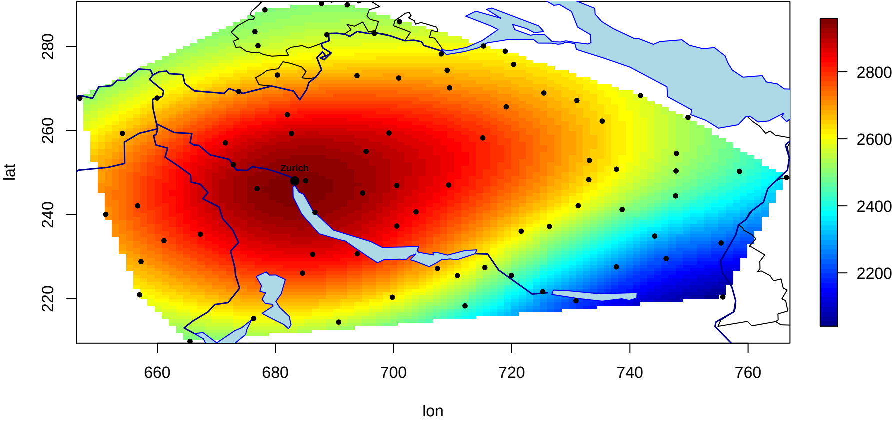
Let

$$C(x) = \{s \in \mathcal{X} : \text{extremes are concurrent at } s \text{ and } x\}$$

and $|C(x)|$ be its volume. Then

$$E(|C(x)|) = I(x).$$

Integrated concurrence



Conclusions

- We have implemented a variety of spatial models to annual rainfall data and discussed their qualities.
- Just as for univariate and multivariate methods, threshold methods could be used, but these have only recently started to appear.
- Models that can also take into account spatial asymptotic independence have been recently proposed.
- The `spatialExtremes` package includes two other datasets. You could try implementing the methods illustrated here to those.
- The above analysis could also be carried out including altitude as a covariate.

Thank you for your attention!

Obrigado!