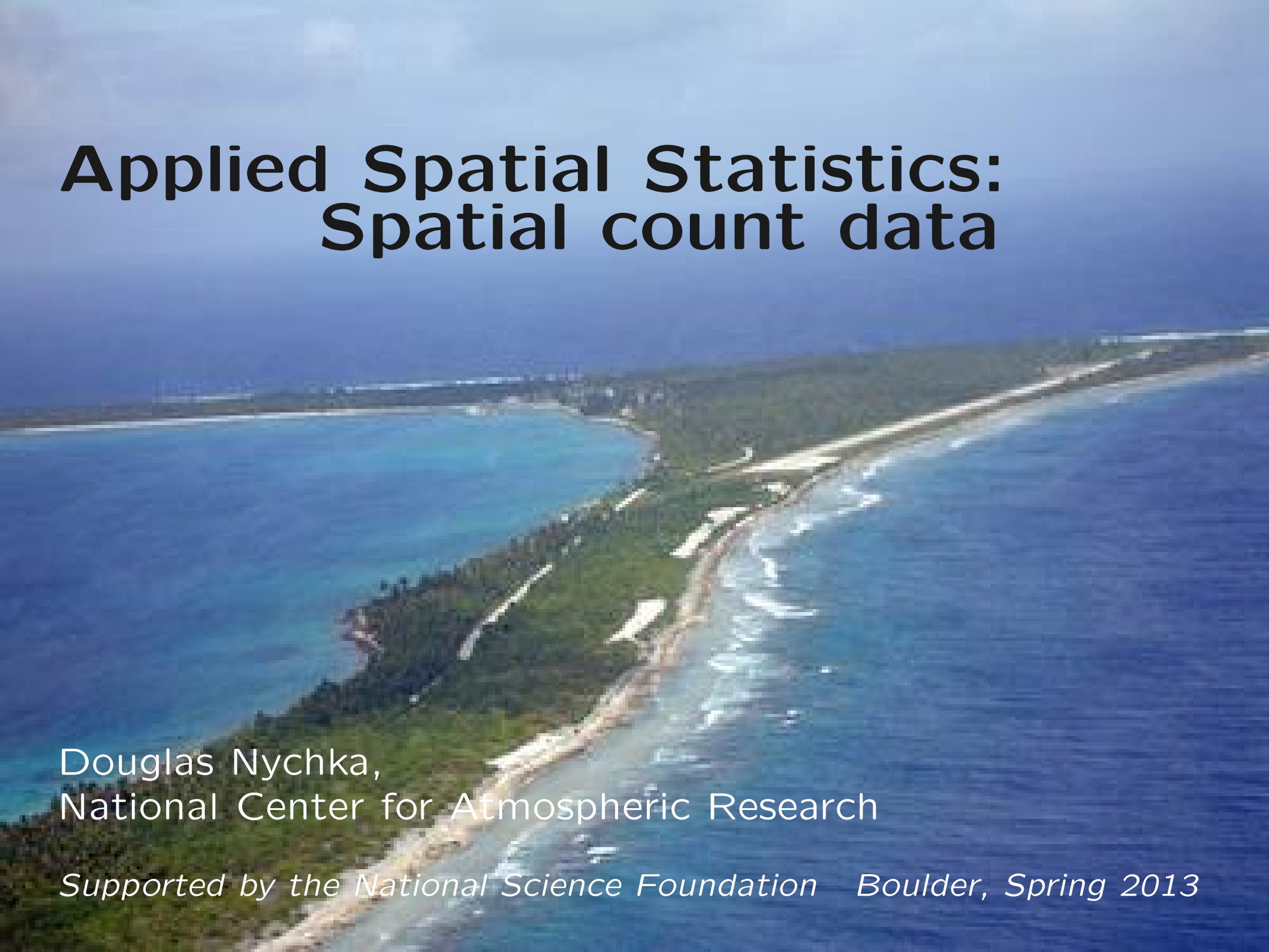
Applied Spatial Statistics: Spatial count data



Douglas Nychka,
National Center for Atmospheric Research

Supported by the National Science Foundation *Boulder, Spring 2013*

Outline

- Poisson distribution
- A hierarchical model
- Gaussian approximations and pseudo data
- Rongelap island

Combine a simple model for counts with the dependence on a spatial field that controls the parameters

Estimating a curve or surface.

The additive statistical model:

Given n pairs of observations (x_i, y_i), $i = 1, \ldots, n$ *Distribution of* $\boldsymbol{y}_i$ *depends on* $f(\boldsymbol{x}_i)$

$$[\boldsymbol{y}_i | f(x_i)]$$

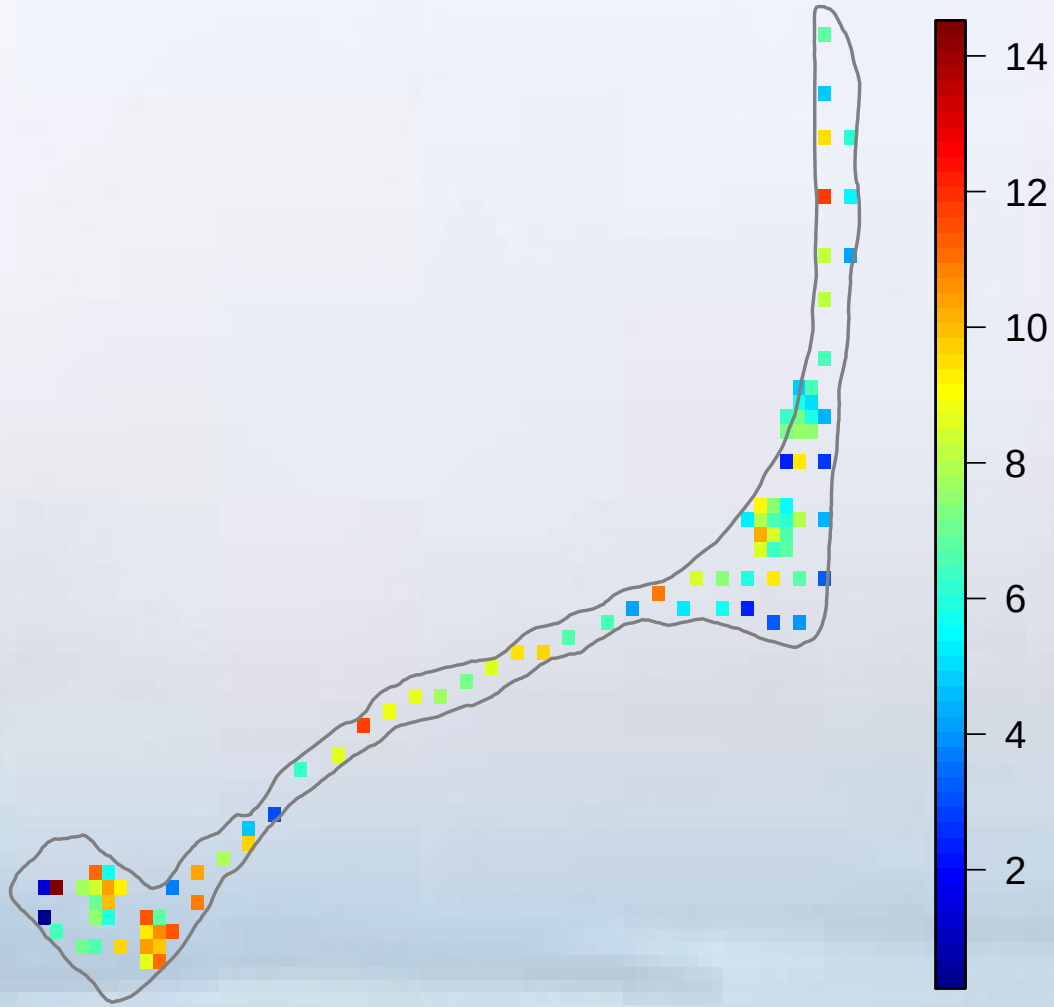
f is an unknown smooth function.

$f(\boldsymbol{x})$ is a Gaussian process but $\boldsymbol{y}_i$ may not be normally distributed.

Some examples

Rongelap Island

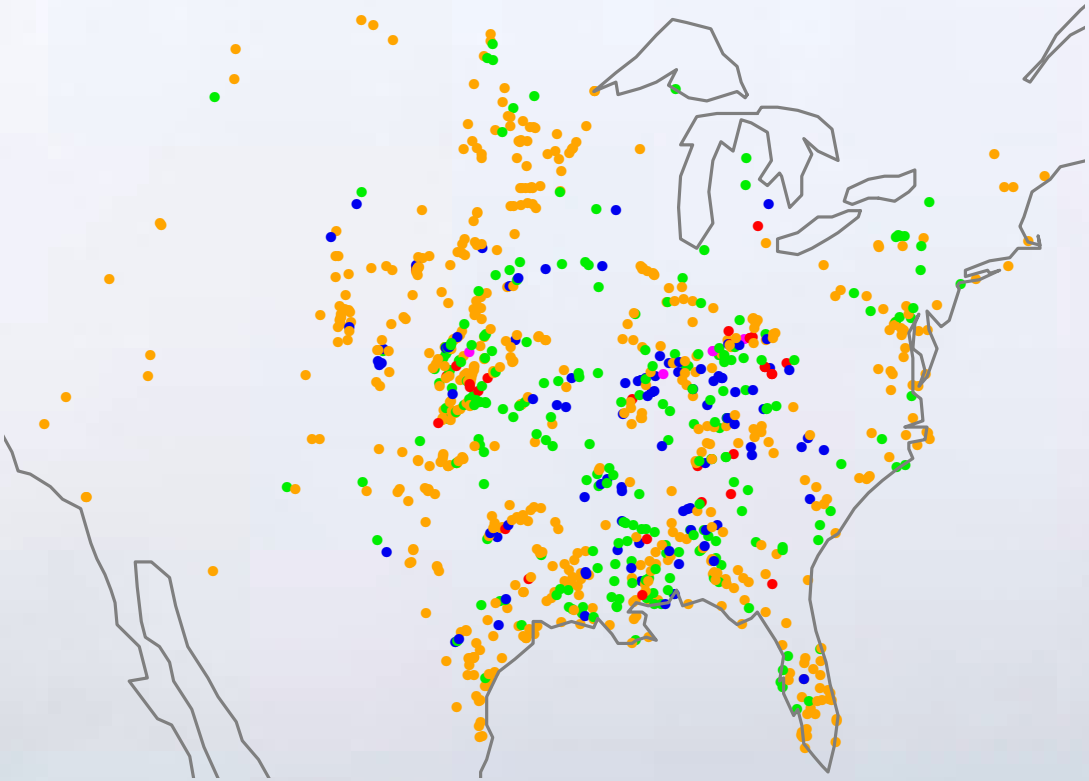
157 γ detector counts measuring residual radiation from nuclear tests



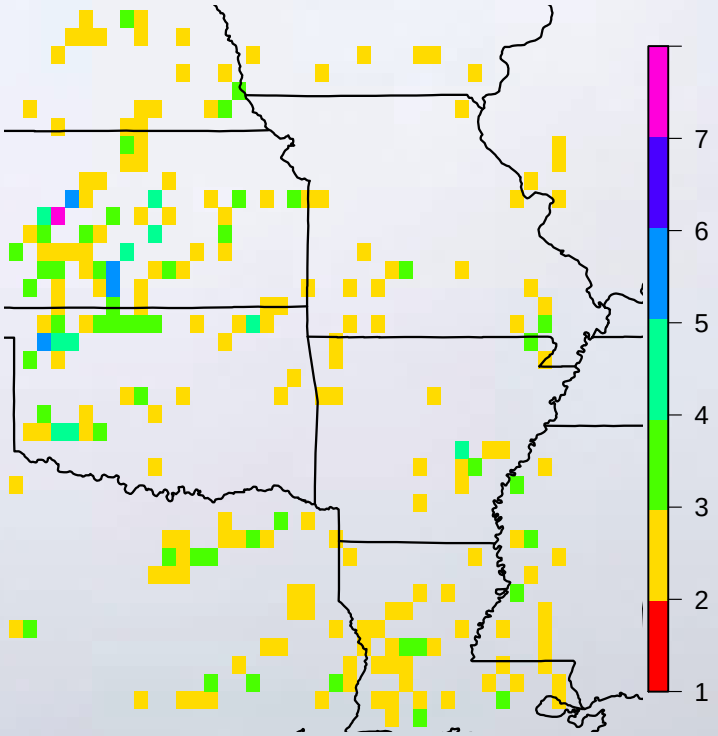
Tornado Alley

Reported starting locations for tornados in 2012.

Event locations



bin counts for grid



Severity: 0, 1, 2, 3, 4

Poission distribution

Distribution of counts for rare events, has parameter α

$$P(Y = k) = \frac{\alpha^k e^{-\alpha}}{k!}$$

- $E(Y) = \alpha$ and $VAR(\alpha) = \alpha$
- log Likelihood for a random sample $y_1, y_2, ..., y_n$

$$\sum_{i=1}^{n} [log(\alpha) y_i - \alpha - log(y_i!)]$$

MLE: $\hat{\alpha} = \overline{y}$

Spatial model:

- $\boldsymbol{y}_i$ is Poission with parameter $\alpha(\boldsymbol{x}_i)$
- $\alpha(\boldsymbol{x})$ follows the usual Gaussian spatial model.

Rongelap data ignoring spatial aspect

157 counts of differing time duration: (Y_i, t_i)

Actual data model is $Y_i \sim$ Poisson(αt_i) assuming no spatial variation.

log likelihood:

$$\sum_{i=1}^{n} [log(\alpha t_i) y_i - \alpha t_i - log(y_i!)]$$

Taking derivative and setting equal to zero

$$\sum_{i=1}^{n} [y_i/\alpha - t_i] = 0$$

solving for MLE: $\widehat{\alpha} = \frac{\sum_{i=1}^{n} y_i}{\sum_{i=1}^{n} t_i}$

First a review of Normal Kriging

Observations

$$[\boldsymbol{y}_i|f(\boldsymbol{x}_i)] = \text{Normal}(f(\boldsymbol{x}_i), \sigma^2 w_i)$$

w_i known set of weights.

Process

$$[f(\boldsymbol{x})|\rho, \boldsymbol{d}, \theta]$$

$f(\boldsymbol{x})$ a Gaussian process

$$f(\boldsymbol{x}) = \sum_j \phi(\boldsymbol{x})\boldsymbol{d}_i + g(\boldsymbol{x})$$

- mean $\sum_j \phi(\boldsymbol{x})\boldsymbol{d}_i$ (usual fixed part)
- covariance for g is $\rho k_\theta(.,.)$ (usual random part)

covariance parameters

In the normal case we know how to find estimates using maximum likelihood

Distribution of observations:

$$\boldsymbol{y} \sim MN(X\boldsymbol{d}, \rho K + \sigma^2 I)$$

or

$$\boldsymbol{y} \sim MN(X\boldsymbol{d}, \rho K + \sigma^2 W)$$

if measurements have different weights.

- Likelihood has a closed form

Poisson data

Observations:

$$[\boldsymbol{y}_i|f(\boldsymbol{x}_i)] = \text{Poisson}(f(\boldsymbol{x}_i))$$

implies that

$$E([\boldsymbol{y}_i|f(\boldsymbol{x}_i)]) = f(\boldsymbol{x}_i) \quad VAR([\boldsymbol{y}_i|f(\boldsymbol{x}_i)]) = f(\boldsymbol{x}_i)$$

just the Possion distribution.

Process:

$f(\boldsymbol{x})$ is the same as in Gaussian case

Main idea:

Approximate the distribution of $\boldsymbol{y}$ with a normal assuming the specific form for the mean and variance.

e.g. If we knew $\boldsymbol{f}^*$ then use it in the error part assume

$$\boldsymbol{y} = f(\boldsymbol{x}_i) + \boldsymbol{e}_i$$

where $VAR(\boldsymbol{e}_i) = \sigma^2 \boldsymbol{f}_i^*$

Practical strategy:

We don't really know $\boldsymbol{f}^*$ so use an iterative method where previous estimate of $\boldsymbol{f}$ is used for variances and $\boldsymbol{f}$ is reestimated.

Note: "weights" in fields are the reciprocal of variance. If variance is f_i then specify weight $1/f_i$

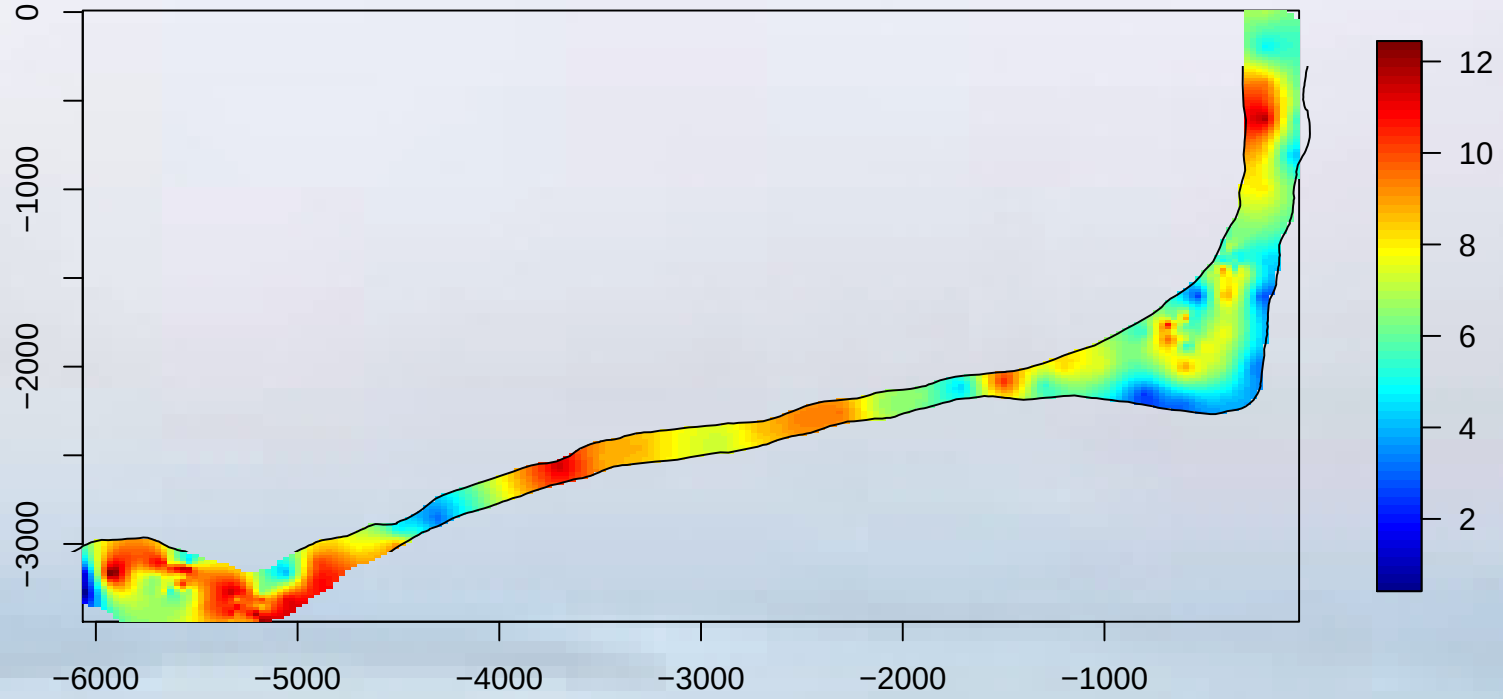
Rongelap data set

Exponential covariance, $\theta = 1500$ and $\lambda = .1$

```
xR<- rongelap$coords
yR<- rongelap$data/rongelap$units.m
wtR<- rongelap$units.m
#
fit.rongelap<- function(theta, lambda, tolerance=1e-6){
  fhat.old<- rep(1, length(yR))
   for( k in 1:50){
     obj<- mKrig( xR, yR, theta= theta, lambda=lambda,
                              weights= wtR/fhat.old , m=1)
     fhat.new<- obj$fitted.values
     test.value<-  mean( abs(fhat.old- fhat.new) )/ mean( abs( fhat.old)
     if( test.value < tolerance){
       break}
     fhat.old<- c(fhat.new)
  }
  return(obj)
 }
```

Take a look

```
R.fit<- fit.rongelap(400,100)
out.p<-predict.surface(R.fit, nx=200, ny=200, extrap=TRUE)
island<- in.poly.grid( out.p, rongelap$border)
out.p$z[!island]<- NA
image.plot( out.p)
lines( rongelap$border)
```

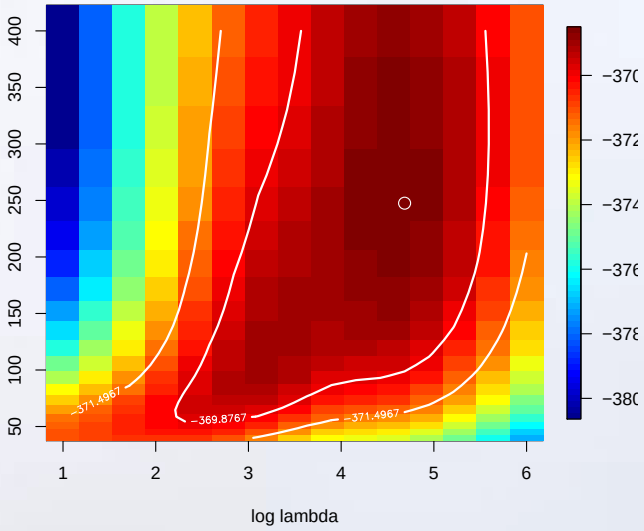


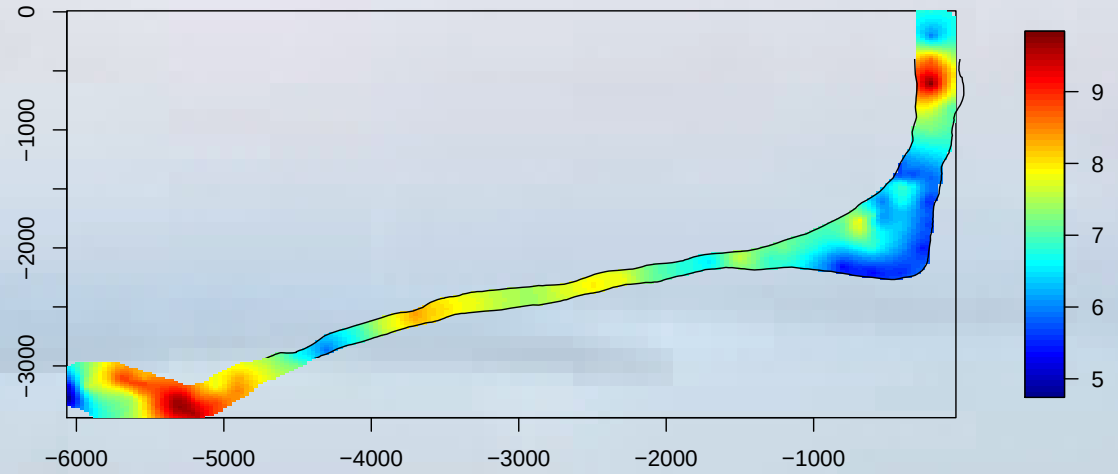
Searching over ρ and θ

```
par.list<- list(
              llambda=seq(1,6,,10),
              theta= exp(seq( log(40),log(400),,15)))
par.grid<- make.surface.grid( par.list)
NG<- nrow( par.grid)
lnLike<- rep( NA, NG)
for( k in 1:NG){
    cat(k," " )
    lnLike[k]<- fit.rongelap(theta=par.grid[k,2],
                             lambda=exp(par.grid[k,1]) )$lnProfileLike
 }
 image.plot( as.surface( par.grid, lnLike))
```

Approximate InProfileLike surface

- Outer contour at 95% level





Some problems

Does it really make sense to have a positive mean (e.g. average counts) follow a Gaussian distribution?

Better model is to allow a link to enforce positivity of the mean

E.g.

$$E([\boldsymbol{y}_i|\boldsymbol{f}_i)]) = \mu(\boldsymbol{f}_i) \quad VAR([\boldsymbol{y}_i|\boldsymbol{f}_i]) = \mu(\boldsymbol{f}_i)$$

where for example $\mu(f) = \exp f$.

Quasi Likelihood solution

In general

$$E([\boldsymbol{y}_i|\boldsymbol{f}_i)]) = \mu(\boldsymbol{f}_i) \quad VAR([\boldsymbol{y}_i|\boldsymbol{f}_i]) = \gamma(\boldsymbol{f}_i)$$

$g(f) = \mu$ the inverse relationship

- Consider the pseudo data

$$\boldsymbol{y}_i^{PS} = \widehat{\boldsymbol{f}}_i + g'(\widehat{\boldsymbol{f}}_i)(\boldsymbol{y}_i - \mu(\widehat{\boldsymbol{f}}_i)$$

- $\widehat{f}$ is a previous estimate or a "pilot" estimate for f.

- Analyze the pseudo data as if it from the spatial model

$$\boldsymbol{y}_i^{PS} = f(\boldsymbol{x}_i) + \boldsymbol{e}_i$$

where $VAR(\boldsymbol{e}_i) = g'(\widehat{\boldsymbol{f}}_i)^2\gamma(\widehat{\boldsymbol{f}}_i)$

- Note: the pilot estimate is kept fixed in this approximate model.

- Iterate to update the pilot until it does not change.

For Possion problem

$\mu(f) = e^f \;\; f = g(\mu) = log(\mu) \;\; g'(\mu) = 1/\mu$

$\gamma(f) = f \;\; g'(\mu)^2 \gamma(\mu) = 1/\mu$

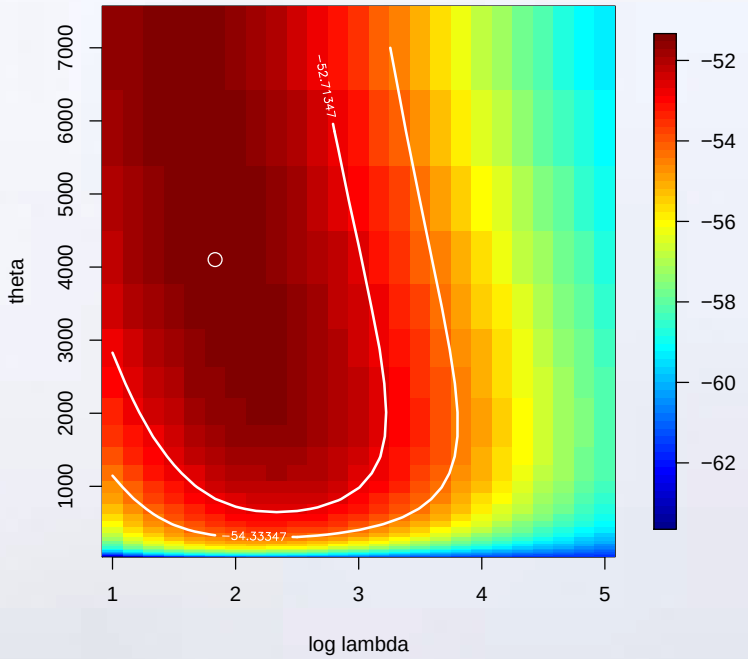
R code for algorithm

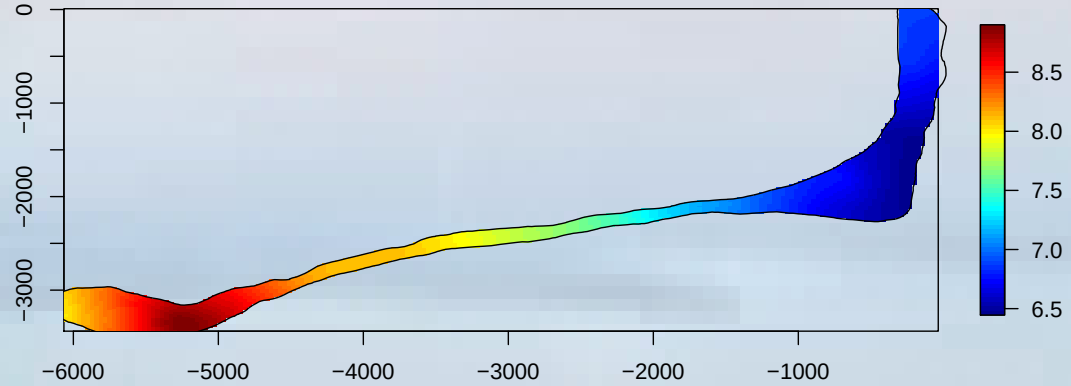
```
fit.rongelap.exp<- function( theta, lambda){
    mu.old<- rep( mean(yR), length(yR))
    fhat.old<- log( mu.old)
    for( k in 1:50){
       yPS<- fhat.old +  (1/mu.old)*( yR- mu.old)
       obj <- mKrig( xR, yPS, weights=wtR*mu.old,
                           lambda=lambda, theta=theta, m=1)
       fhat.new <- obj$fitted.values
       mu.new<- exp( fhat.new)
#add convergence criterion
      fhat.old<- c(fhat.new)
      mu.old<- mu.new
   }
return(obj)
}
```

Interpretation is that at convergence one has fit a Gaussian spatial model – where the weights used are what one gets after fitting the model!

Approximate InProfileLike surface

- log link function Outer contour at 95% level





Summary

- Nongaussian data can be analyzed by relating to a weighted Gaussian model.

- The concept of pseudo data is used to suggest an iterative algorithm to find an estimate.

- Not clear exactly what statistical problem we have solved or what we have approximated.