LatticeKrigLab3.R

nychka

2023-07-17

```
###########################################
### Lattice vs spatialProcess demo
###########################################

library(LatticeKrig)
```

```
## Loading required package: spam
```

```
## Spam version 2.9-1 (2022-08-07) is loaded.
## Type 'help( Spam)' or 'demo( spam)' for a short introduction
## and overview of this package.
## Help for individual functions is also obtained by adding the
## suffix '.spam' to the function name, e.g. 'help( chol.spam)'.
```

```
##
## Attaching package: 'spam'
```

```
## The following objects are masked from 'package:base':
##
##     backsolve, forwardsolve
```

```
## Loading required package: fields
```

```
## Loading required package: viridisLite
```

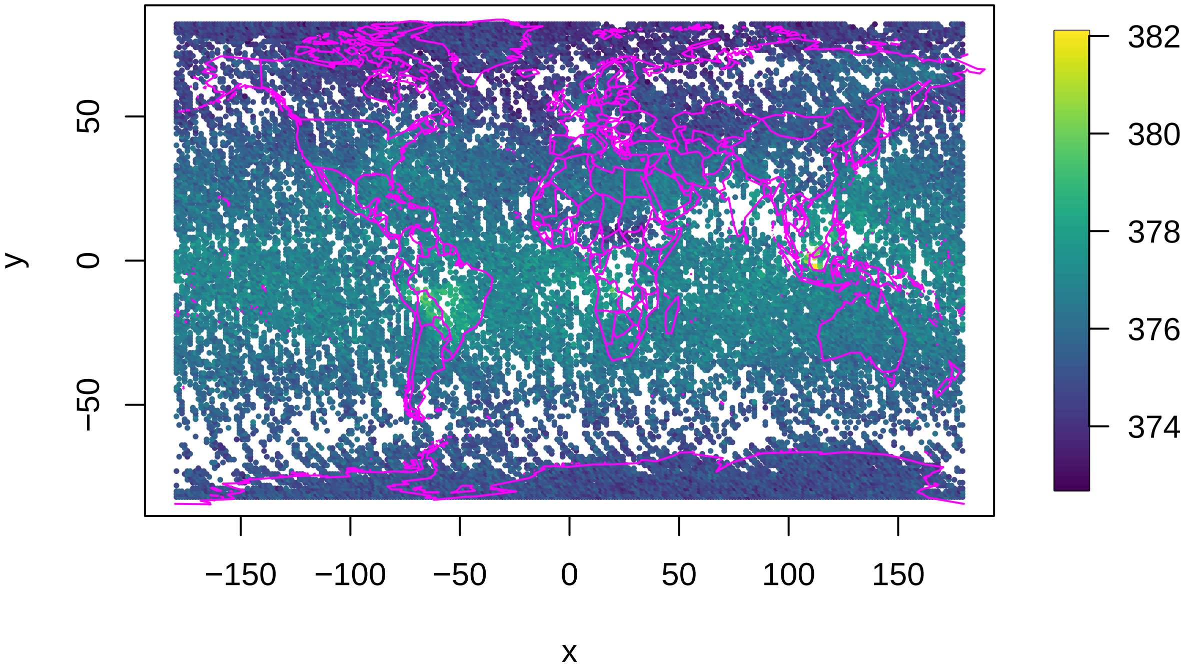
```
##
## Try help(fields) to get started.
```

```
data(CO2)

s<- CO2$lon.lat
z<- CO2$y
dim( s)
```

```
## [1] 26633     2
```

```
bubblePlot( s,z, highlight=FALSE, size=.4)
world( add=TRUE, col="magenta")
```



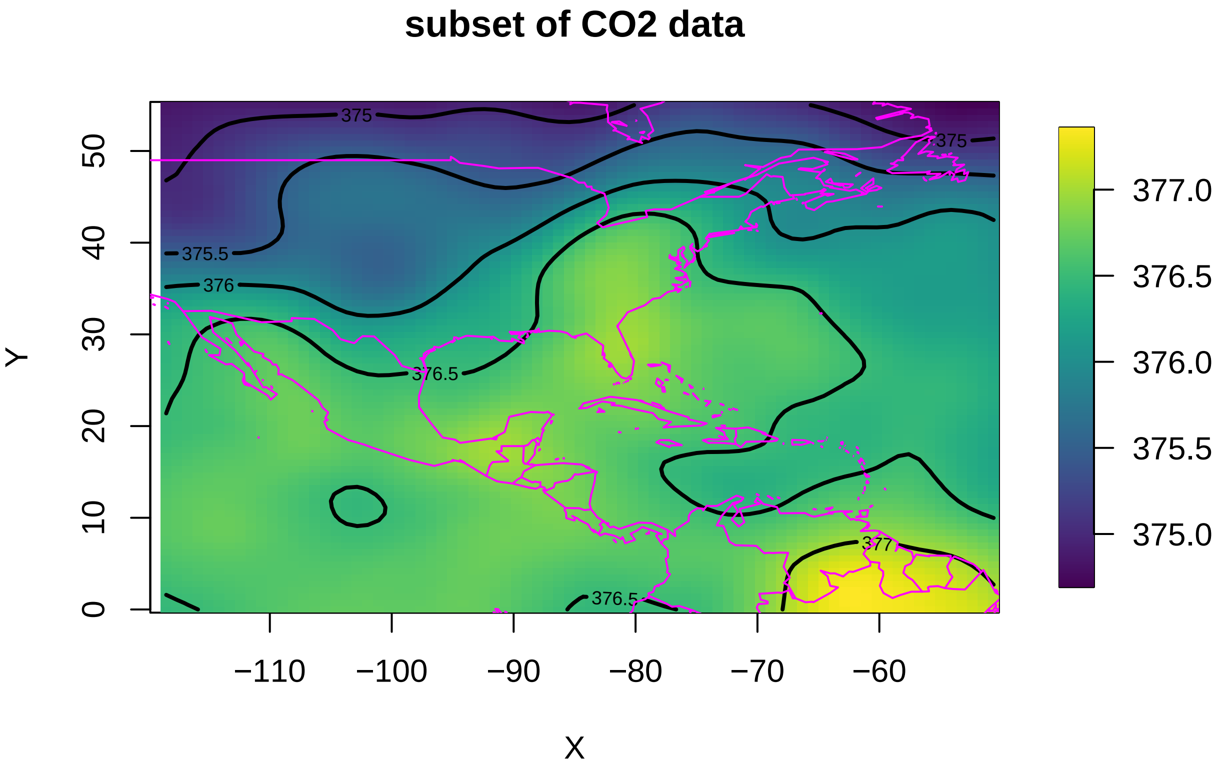
```
# 1884 locations
ind2<- which(
       s[,1]>= -120 & s[,1] <= -50 &
       s[,2]>=  0 & s[,2] <=  55
        )

system.time(
  fit2<- spatialProcess(s[ind2,], z[ind2], cov.function="Tps.cov" )
)
```

```
##    user  system elapsed
##   6.076   0.728   6.807
```

```
# for n=1884 get about 7 seconds
# a conservative  lower bound   for the full data set is
# 3 hours ...

surface( fit2)
world( add=TRUE, col="magenta")
title("subset of CO2 data")
```



```
# approx thin plate spline fit using fixed rank Kriging
system.time(
  fit4<- LatticeKrig(s, z, a.wght = 4.01 )
)
```

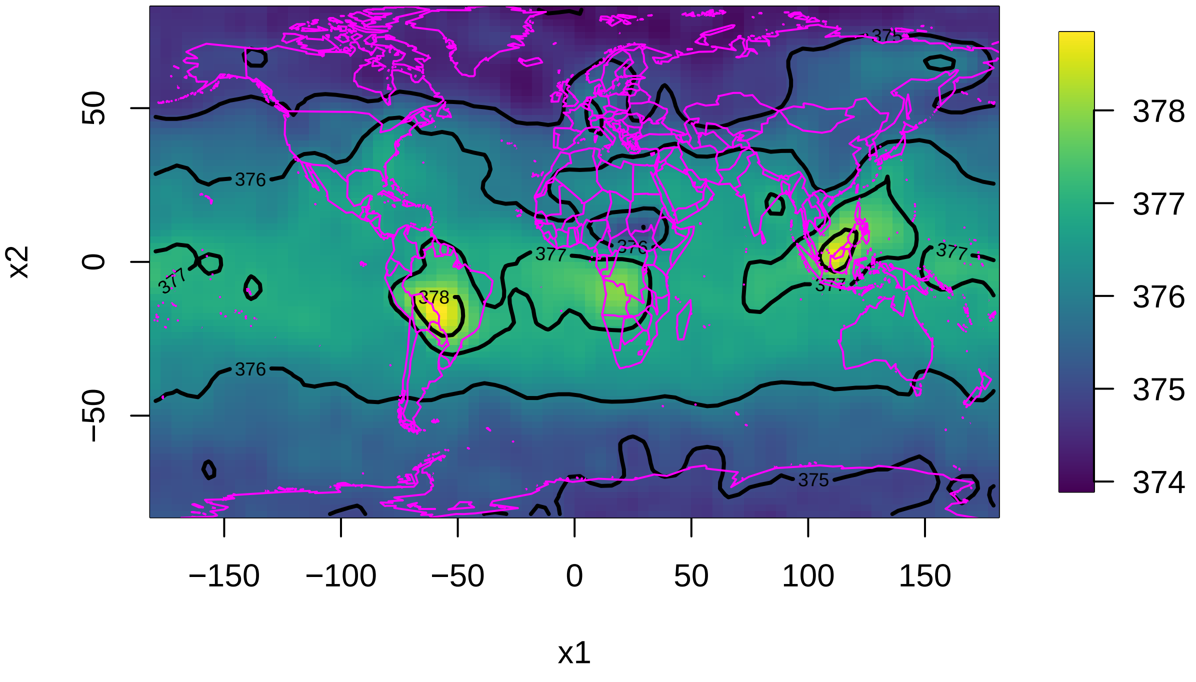
```
##    user  system elapsed
##  34.575   0.733  35.332
```

```
# for ~27K locations get about  35 seconds

# summary of fit
fit4
```

```
## Call:
## LatticeKrig(x = s, y = z, a.wght = 4.01)
##
##
##  Number of Observations:                     26633
##  Number of parameters in the fixed component 3
##   Effective degrees of freedom (EDF)         798.8
##     Standard Error of EDF estimate:          7.584
##  MLE sigma                                   0.5069
##  MLE rho                                     6.296
##  MLE lambda = sigma^2/rho                    0.04081
##
## Fixed part of model is a polynomial of degree 1 (m-1)
##
## Summary of estimated fixed model coefficients
##                  Estimate  Std. Error      t value  Pr(>|t|)
## Intercept  3.749934e+02 1.144268973 327.71436617 0.0000000
## x1        -6.171181e-04 0.006787081  -0.09092541 0.9275525
## x2        -6.771584e-03 0.009510819  -0.71198746 0.4764788
##  Standard errors are  based on generalized LS
##  and for covariance parameters fixed at the estimated values
##
## Basis function :  Radial
## Basis function used:  WendlandFunction
## Distance metric:  Euclidean
##
## Lattice summary:
## 3  Level(s) 16050 basis functions with overlap of  2.5 (lattice units)
##
##  Level Lattice points Spacing
##      1           1242 10.2500
##      2           3483  5.1250
##      3          11325  2.5625
##
## Nonzero entries in Ridge regression matrix 9848348
```

```
surface( fit4)
world( add=TRUE, col="magenta")
```



```
# fit model using cylindrical geometry.



alpha<- fit4$LKinfo$alpha
LKInfo<- LKrigSetup( s, startingLevel=4 , nlevel=3,
                                              a.wght=1.05,
                          alpha=alpha,
                                              LKGeometry="LKSphere" )
# list model specification
LKInfo
```

```
## Classes for this object are:  LKinfo LKSphere
## The second class usually will indicate the geometry
##      e.g.  2-d rectangle is  LKRectangle
##
## Some details on spatial autoregression flags:
## stationary:
## first order (by level):
## isotropic:  TRUE
##
## Ranges of locations in raw scale:
##          [,1] [,2]
## [1,] -179.375  -82
## [2,]  179.375   82
##
## Logical (collapseFixedEffect) if fixed effects will be pooled: FALSE
##
## Number of levels: 3
## delta scalings: 0.176 0.088 0.044
## with an overlap parameter of  2.5
## alpha:  0.7619048 0.1904762 0.04761905
##
## a.wght:  1.05 1.05 1.05
##
## Basis  type: Radial using  WendlandFunction  and GreatCircle  distance.
## Basis functions will be normalized
##
## Total number of basis functions  13205
##  Level Basis size
##      1        625
##      2       2507
##      3      10073
##
## Lambda value:  NA
```

```
# system.time(
#   fit5<- LatticeKrig(s, z, LKinfo=LKinfo)
# )

# this takes abot 2 minutes and is an exact  spherical geometry.
```