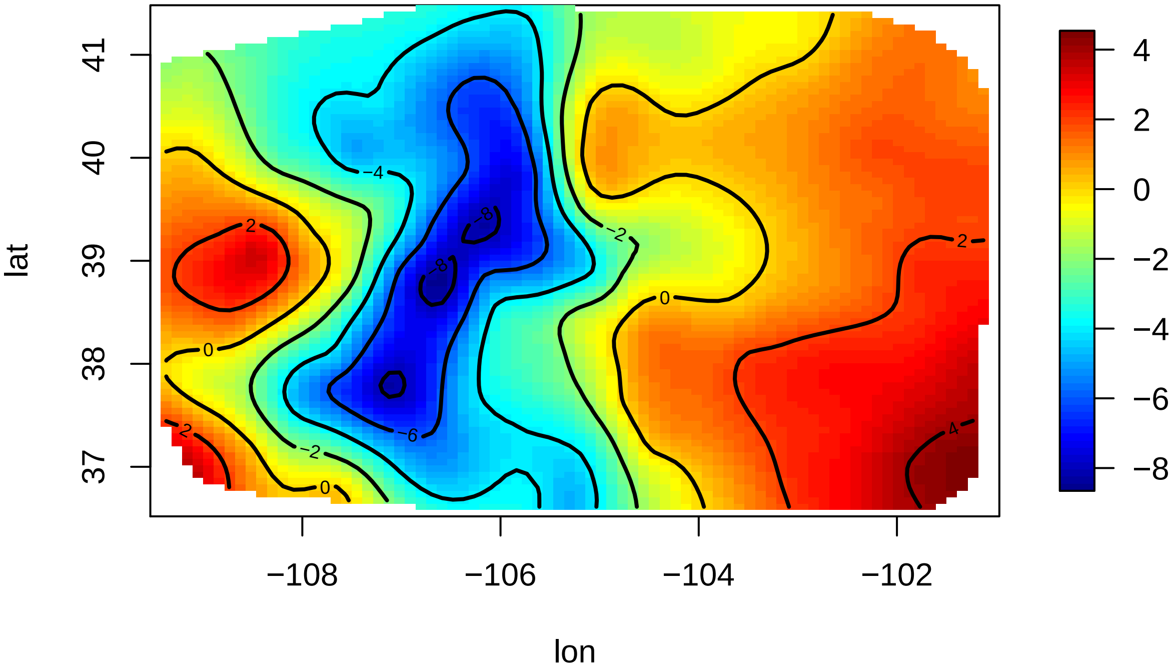
ColoradoExample.R

nychka

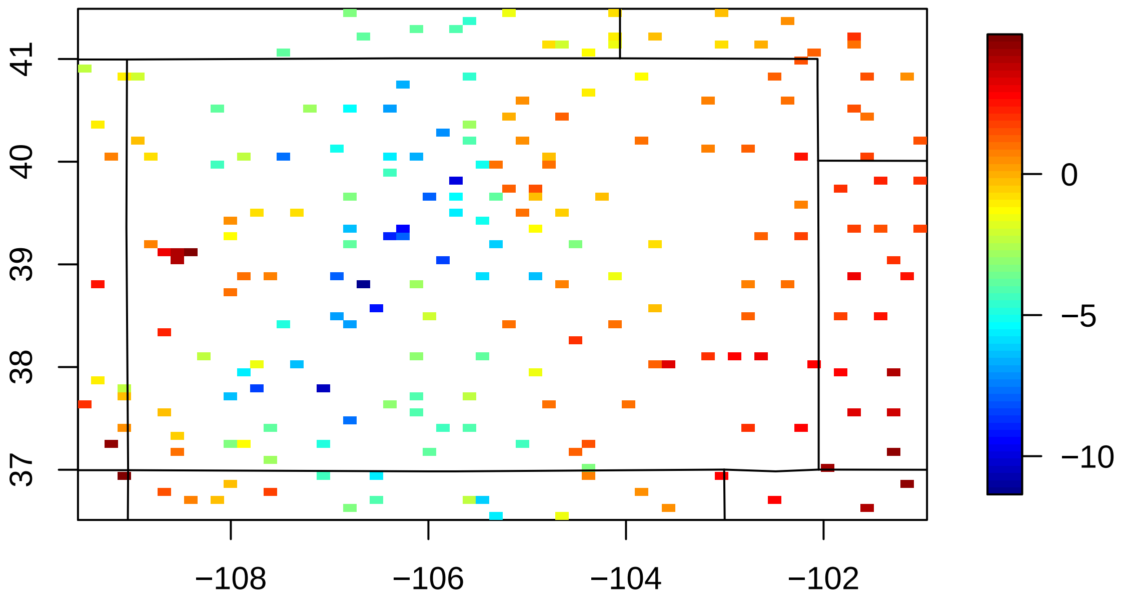
2019-09-13

```
suppressMessages(library( fields))

#  colorado climate data
data(COmonthlyMet)

x<- CO.loc
y<- CO.tmin.MAM.climate
elev<- CO.elev
good<- !is.na( y)
x<- x[good,]
y<- y[good]
elev<- elev[good]

### A quick spatial analysis -- lots of defaults and w/o elevation
obj<- spatialProcess( x,y)
surface( obj)
```

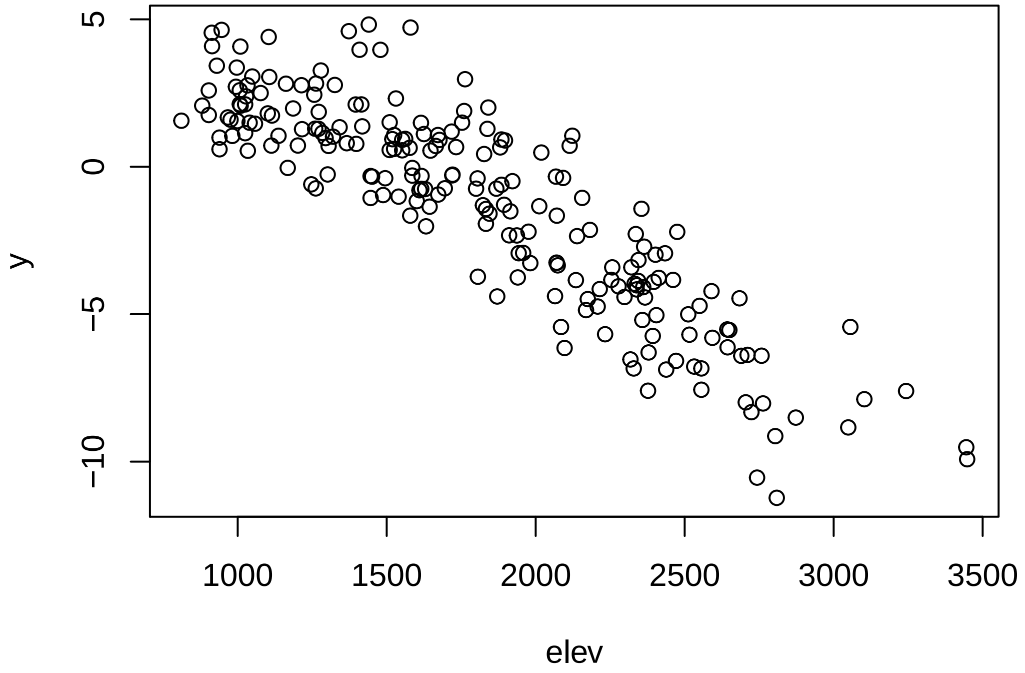


```
NGRID <- 50
# get elevations on a grid (will use these later)
COGrid<- fields.x.to.grid( x, nx=NGRID, ny=NGRID)
COGridPoints<- make.surface.grid( COGrid)
data( RMelevation)

COElev<- interp.surface( RMelevation, COGridPoints )
COElevGrid<- as.surface( COGridPoints, COElev)
```

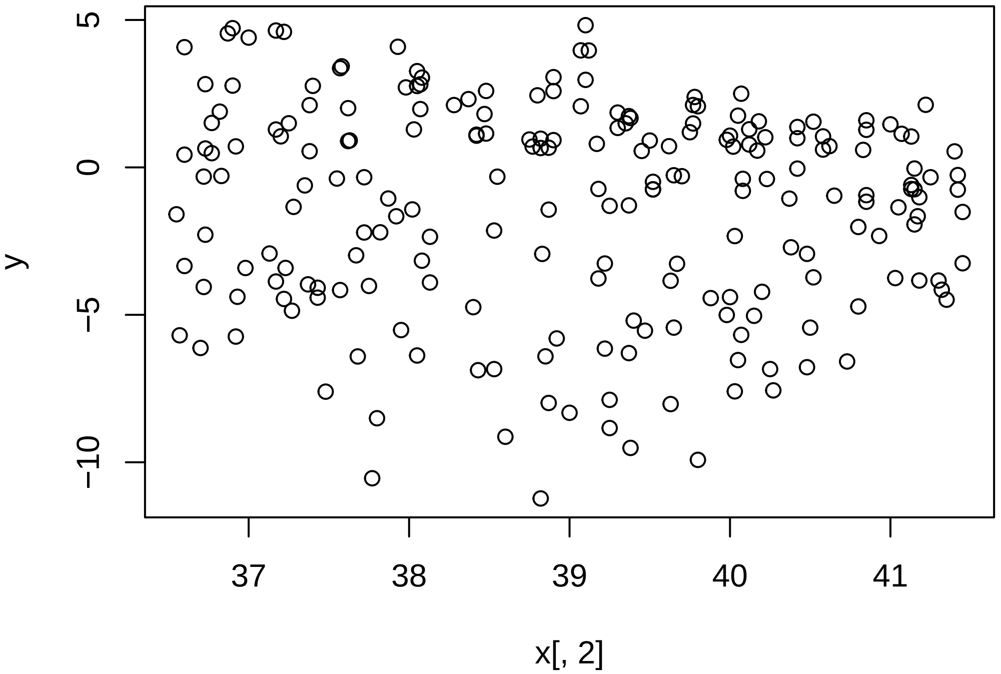
```
# take a look at the data
quilt.plot( x, y)
US( add=TRUE)
```



```
plot( elev, y)
```



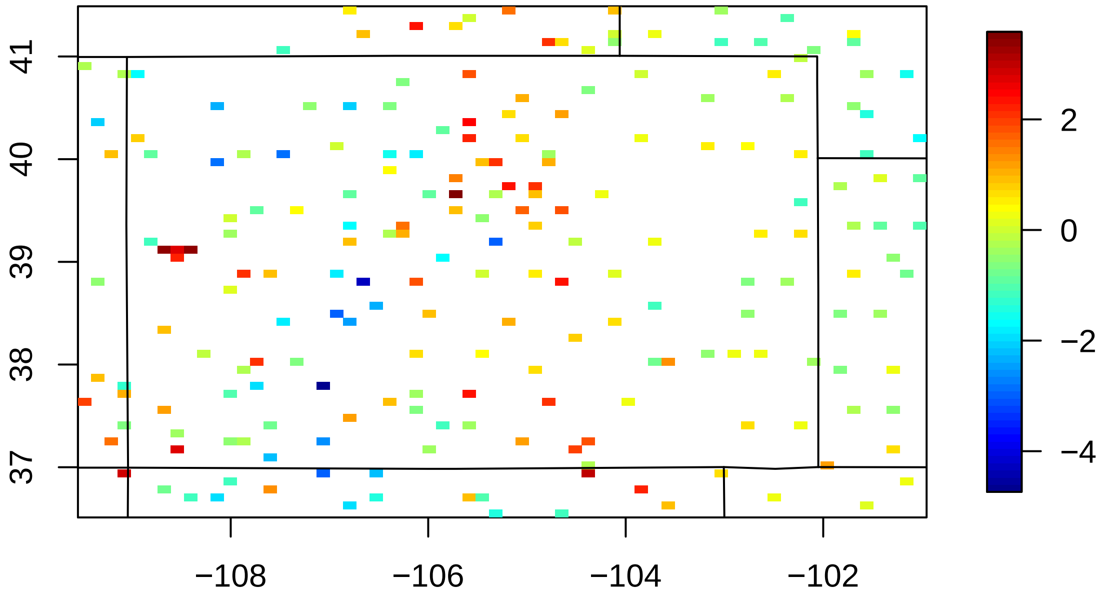
```
plot( x[,2], y)
```



```
X<- cbind( x, elev)
#
lmObj<- lm( y ~ lon+lat +elev, data=X )
summary( lmObj)
```

```
##
## Call:
## lm(formula = y ~ lon + lat + elev, data = X)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.6706 -0.8541 -0.1308  0.9088  3.5183
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.9372343  5.9584509   1.332    0.184
## lon         -0.1982295  0.0499149  -3.971 9.83e-05 ***
## lat         -0.4951096  0.0691068  -7.164 1.32e-11 ***
## elev        -0.0059590  0.0002029 -29.374  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.412 on 209 degrees of freedom
## Multiple R-squared:  0.8415, Adjusted R-squared:  0.8393
## F-statistic:   370 on 3 and 209 DF,  p-value: < 2.2e-16
```

```
quilt.plot( x, lmObj$residuals)
US( add=TRUE)
```



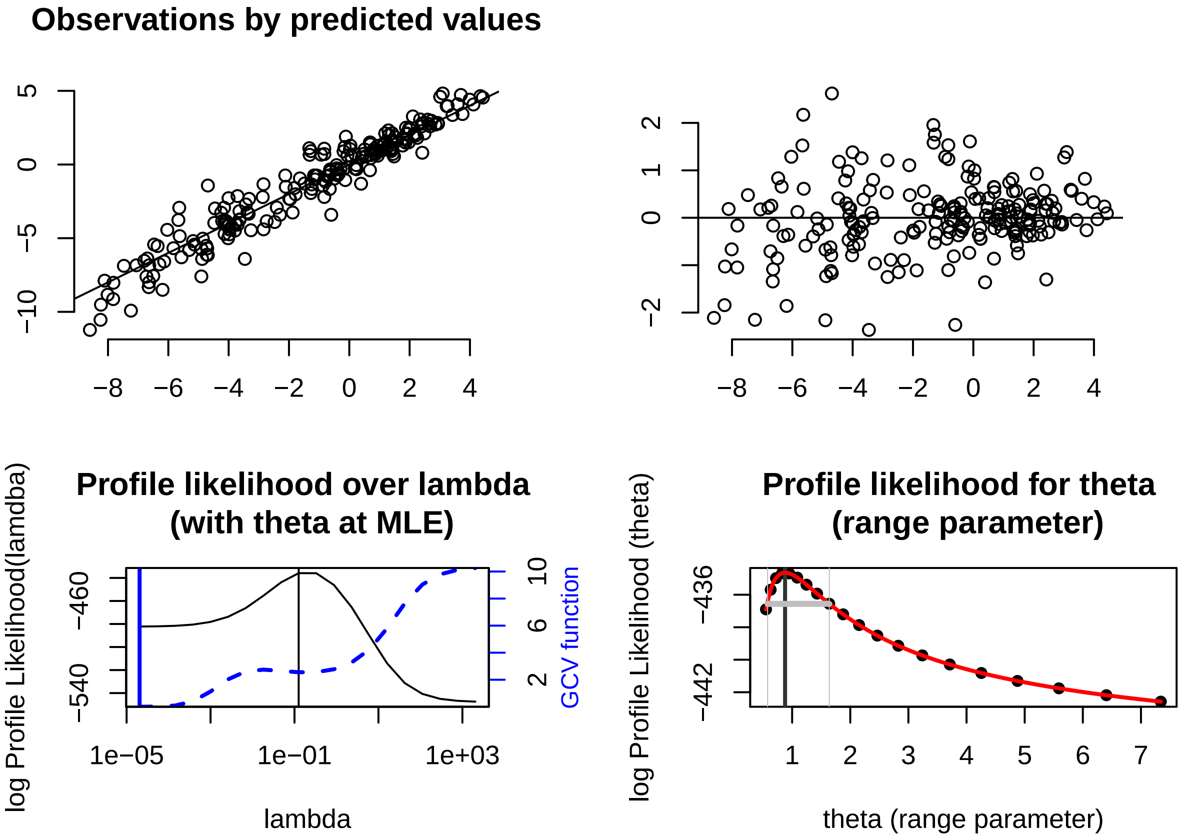
```r
fit0E<- Tps( x,y, Z=elev- mean(elev))


# fit a Kriging estimator Matern covariance smoothness =1.0
# range and nugget estimated by maxmimum likelihood
fit1<- spatialProcess( x,y)
# summary of the fit
print( fit1)
```

```
## CALL:
## spatialProcess(x = x, y = y)
##
## SUMMARY OF MODEL FIT:
##
##  Number of Observations:                          213
##  Degree of polynomial in fixed part:              1
##  Total number of parameters in fixed part:        3
##  MLE nugget variance ( sigma^2)                   1.55
##  MLE process variance (rho)                       7.591
##  MLE range parameter (theta, units of distance):  0.8786
##  Approx 95% CI for theta:                         [ 0.5774 , 1.639 ]
##  Approx.  degrees of freedom for curve            84.73
##     Standard Error of df estimate:                2.661
##  Nonzero entries in covariance                    45369
##
##  ESTIMATED COEFFICIENTS FOR FIXED PART:
##
##    estimate      SE pValue
## d1  55.0800 36.6800 0.1332
## d2   0.3434  0.3055 0.2610
## d3  -0.4924  0.4484 0.2722
##
##  COVARIANCE MODEL: stationary.cov
##   Covariance function:  Matern
##    Non-default covariance arguments and their values
##    Argument: Covariance  has the value(s):
## [1] "Matern"
##    Argument: smoothness  has the value(s):
## [1] 1
##    Argument: theta  has the value(s):
##     theta
## 0.8785521
##    Argument: onlyUpper  has the value(s):
## [1] FALSE
##    Argument: distMat  has the value(s):
## [1] NA
```

```
# diagnostic plots
set.panel(2,2)
```

```
## plot window will lay out plots in a 2 by 2 matrix
```

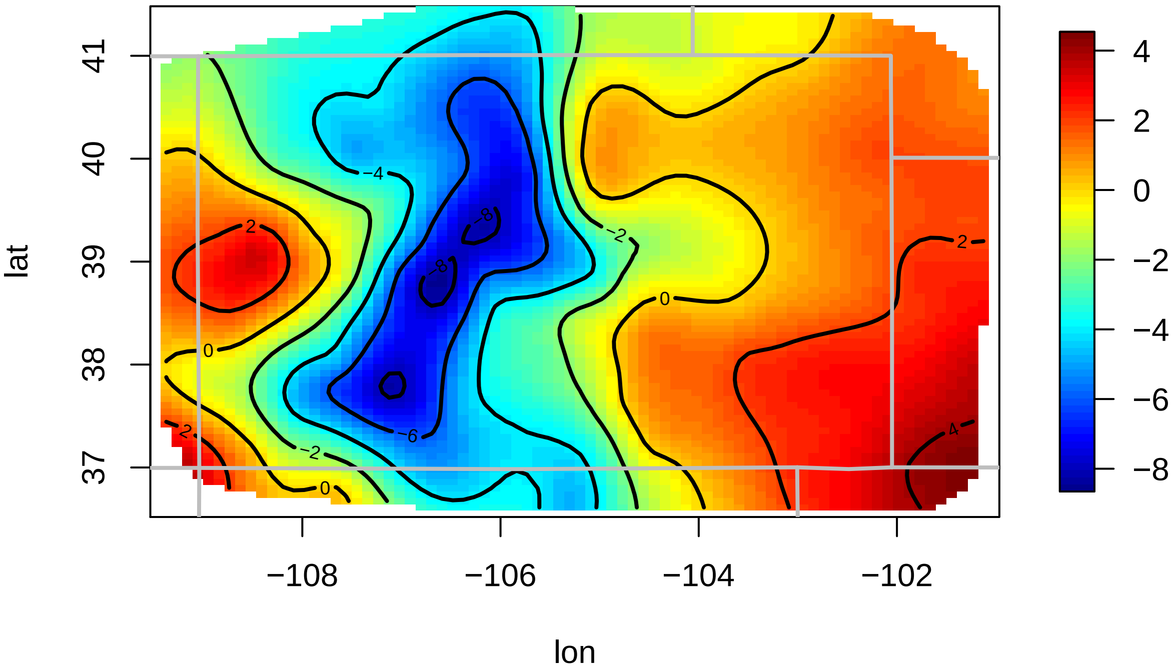
```
plot( fit1)
```



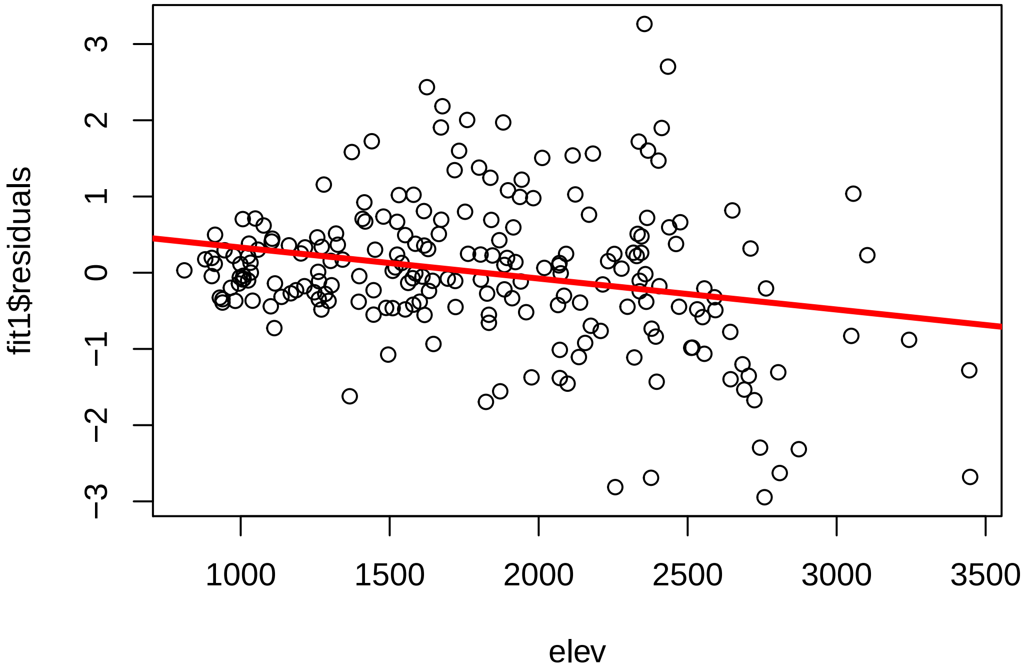
```
set.panel()
```

```
## plot window will lay out plots in a 1 by 1 matrix
```

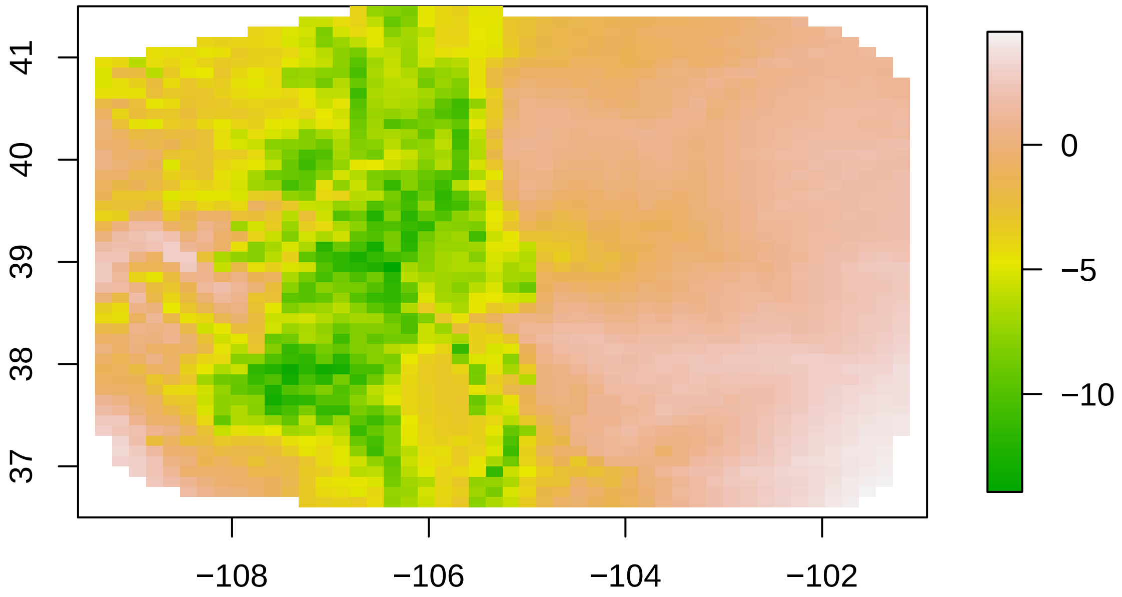
```
surface( fit1)
US( add=TRUE, col="grey", lwd=2)
```



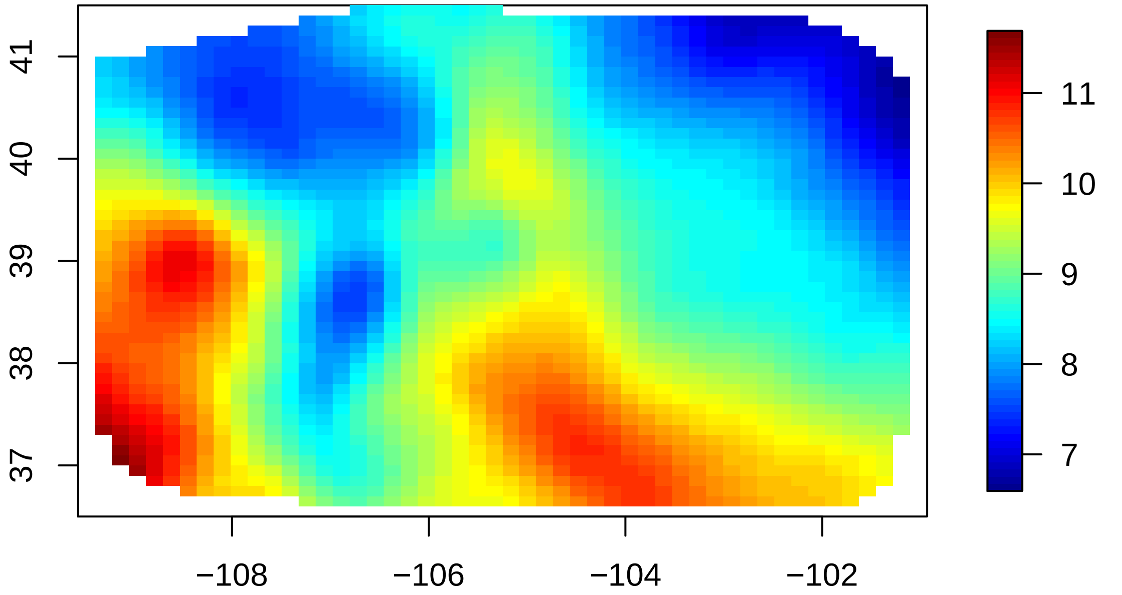
```
# take a look at residuals
plot( elev, fit1$residuals)
lmObj<- lm( fit1$residuals ~ elev)
abline( lmObj, col="red", lwd=3)
```



```
# with elevations
fit1E<- spatialProcess( x,y, Z = elev)

sur0<- predictSurface( fit1E, nx=NGRID, ny=NGRID, Z= COElevGrid)
image.plot( sur0, col=terrain.colors(256))
```



```
surOSmooth<- predictSurface( fit1E, nx=NGRID, ny=NGRID, drop.Z= TRUE)
image.plot( surOSmooth)
```

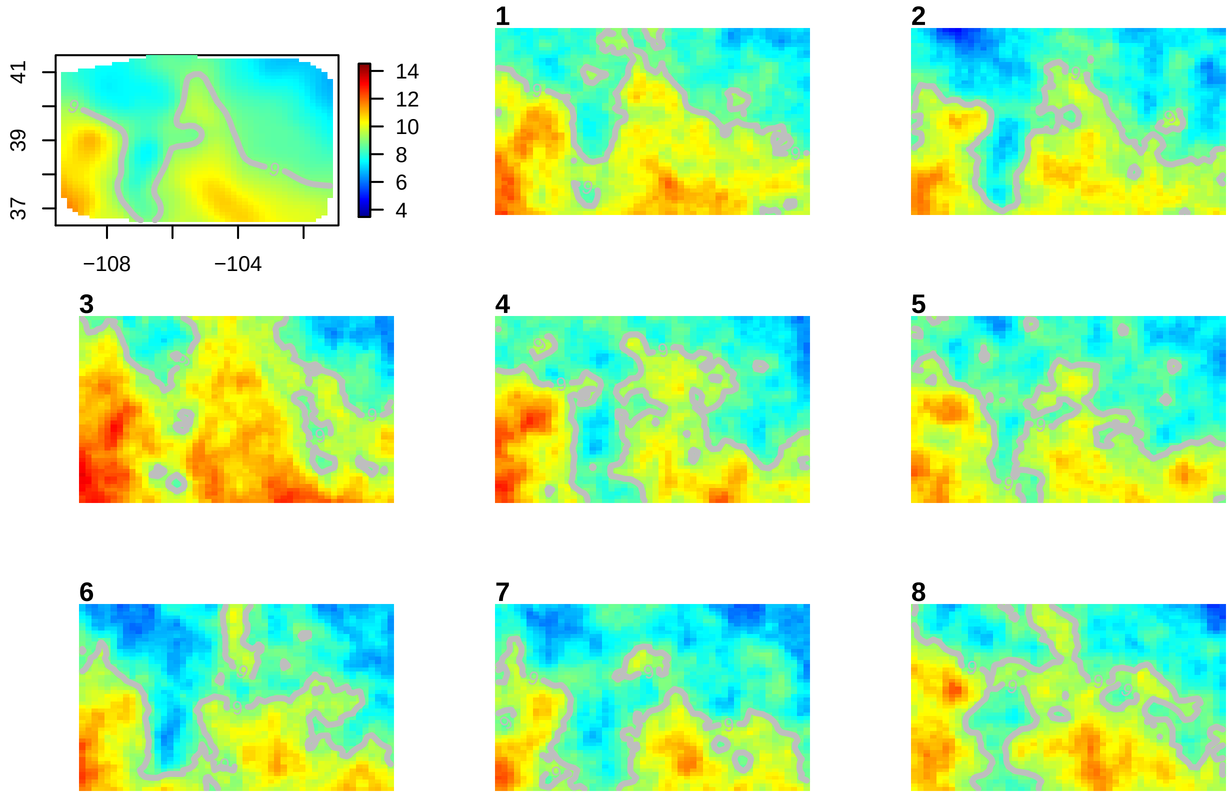


```
# uncertainty 40 draws from posterior distribution
set.seed(123)
# next commmand takes a minute or so
SEout<- sim.spatialProcess( fit1E, xp = COGridPoints,
                           Z = COElevGrid,
                           M = 40,
                           drop.Z = TRUE)

set.panel( 3,3)
```

```
## plot window will lay out plots in a 3 by 3 matrix
```

```
image.plot( surOSmooth,
            zlim=c(3.5,14.5), col=tim.colors(256))
```

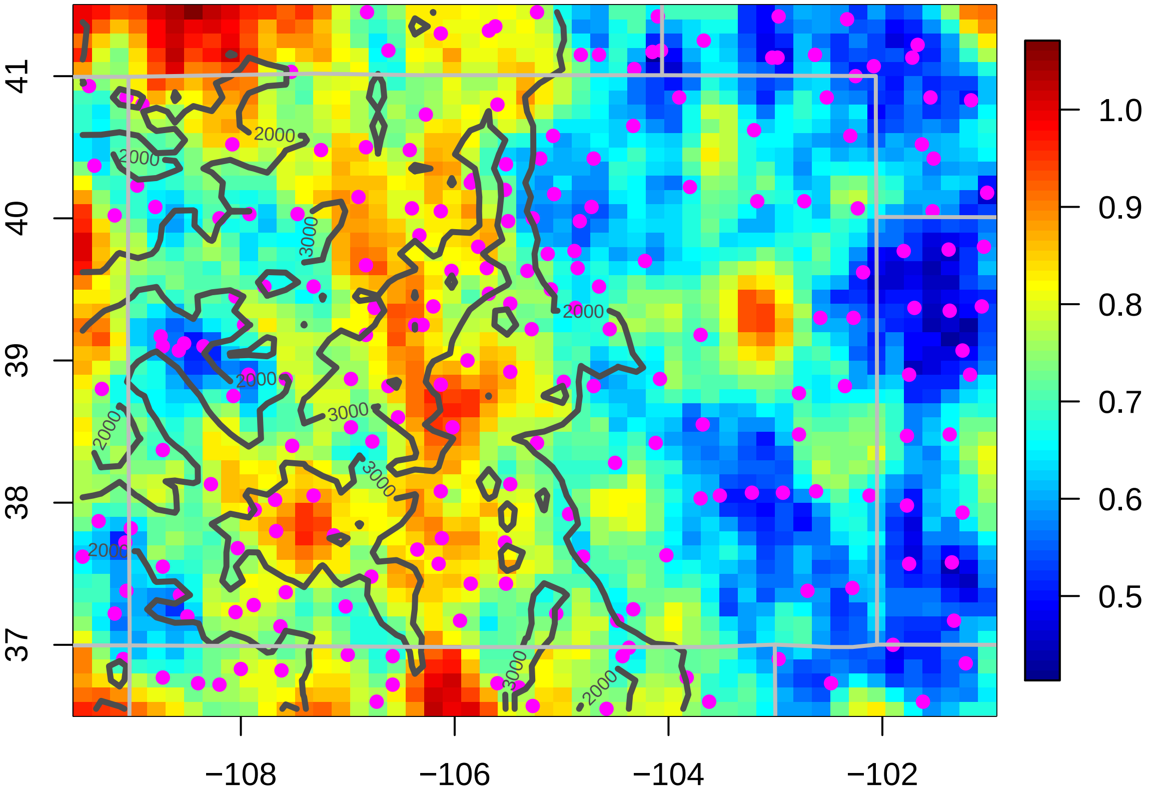
```
contour( surOSmooth, levels= 9, lwd=3, col="grey", add=TRUE)
par( mar=c(3,3,1,1))
for( k in 1:8){
  image( as.surface( COGridPoints, SEout[k,]),
         zlim =c(3.5,14.5),col=tim.colors(256), axes=FALSE)
  contour( as.surface( COGridPoints, SEout[k,]),
           lwd=3, col="grey",level=9, add=TRUE)
  title( k, adj=0, cex=2)
}
```



```
set.panel()
```

```
## plot window will lay out plots in a 1 by 1 matrix
```

```
surSE<- apply( SEout, 2, sd )

image.plot( as.surface( COGridPoints, surSE))
points( x, col="magenta", pch=16)
contour( COElevGrid ,
         level= c(2000, 3000), add= TRUE, col="grey30", lwd=3)
US( add=TRUE, col="grey", lwd=2)
```


41
40
39
38
37
−108
−106
−104
−102
1.0
0.9
0.8
0.7
0.6
0.5
2000
3000