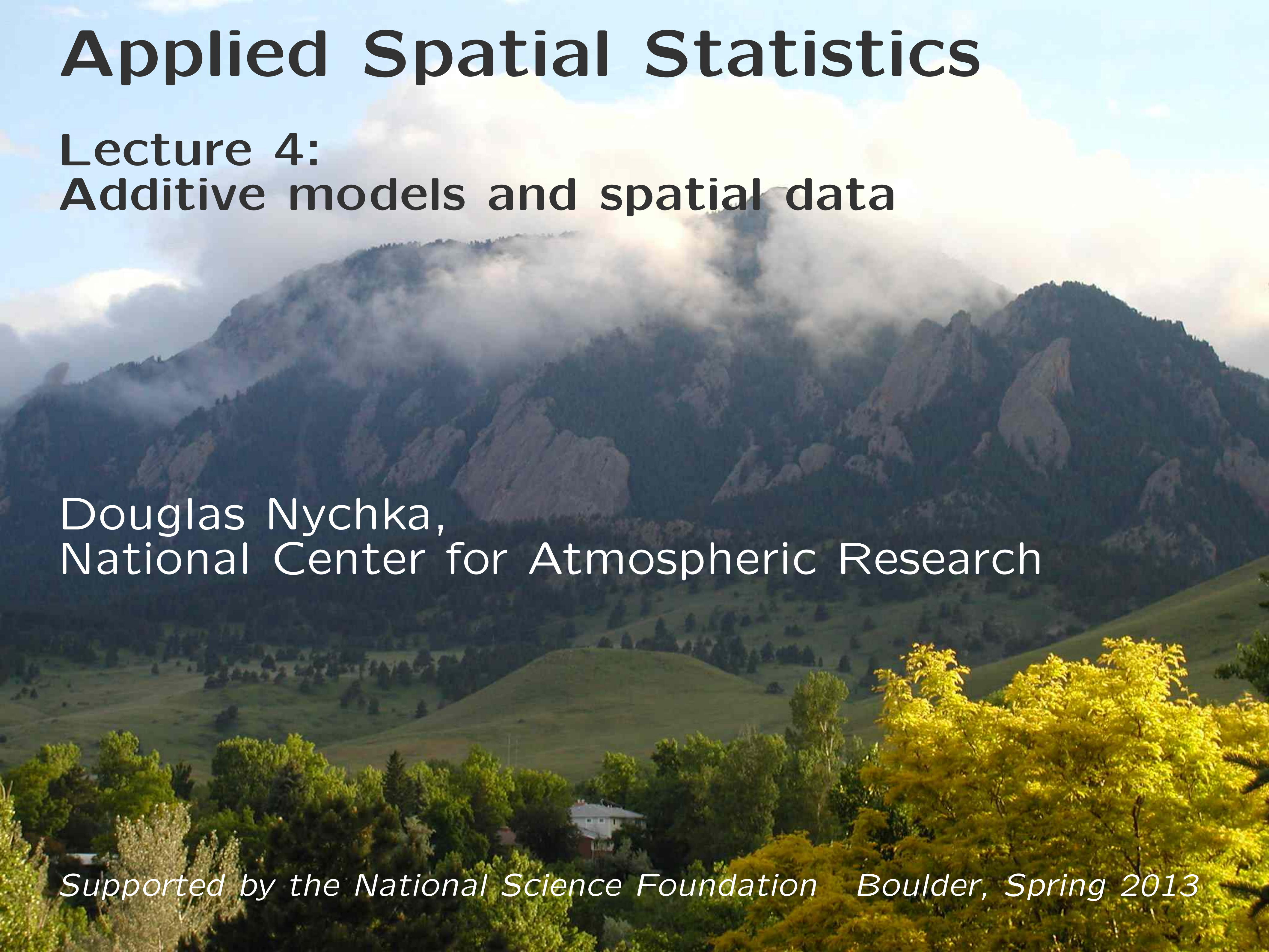


Applied Spatial Statistics

Lecture 4:
Additive models and spatial data

Douglas Nychka,
National Center for Atmospheric Research

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

Outline

- λ the smoothing parameter
- Additive models and fixed effects
- Universal Kriging
- Climate example

Kriging review

Kriging in terms of basis functions

$$\hat{\boldsymbol{c}} = (\rho K + \sigma^2 I)^{-1} \boldsymbol{y}$$

$$\hat{g}(\boldsymbol{x}^*) = \sum_{i=1}^{n} \rho k(\boldsymbol{x}^*, \boldsymbol{x}_i) \hat{\boldsymbol{c}}_i$$

Have added a parameter that scales the covariance function ρ e.g.

$$\rho k(\boldsymbol{x}_1, \boldsymbol{x}_2) = \rho e^{-||\boldsymbol{x}_1 - \boldsymbol{x}_2||/\theta}$$

ρ is usually the variance of g and is called also the *sill*.

$$\hat{\boldsymbol{c}} = (1/\rho)(K + \sigma^2/\rho I)^{-1} \boldsymbol{y} = (1/\rho)(K + \lambda I)^{-1} \boldsymbol{y}$$

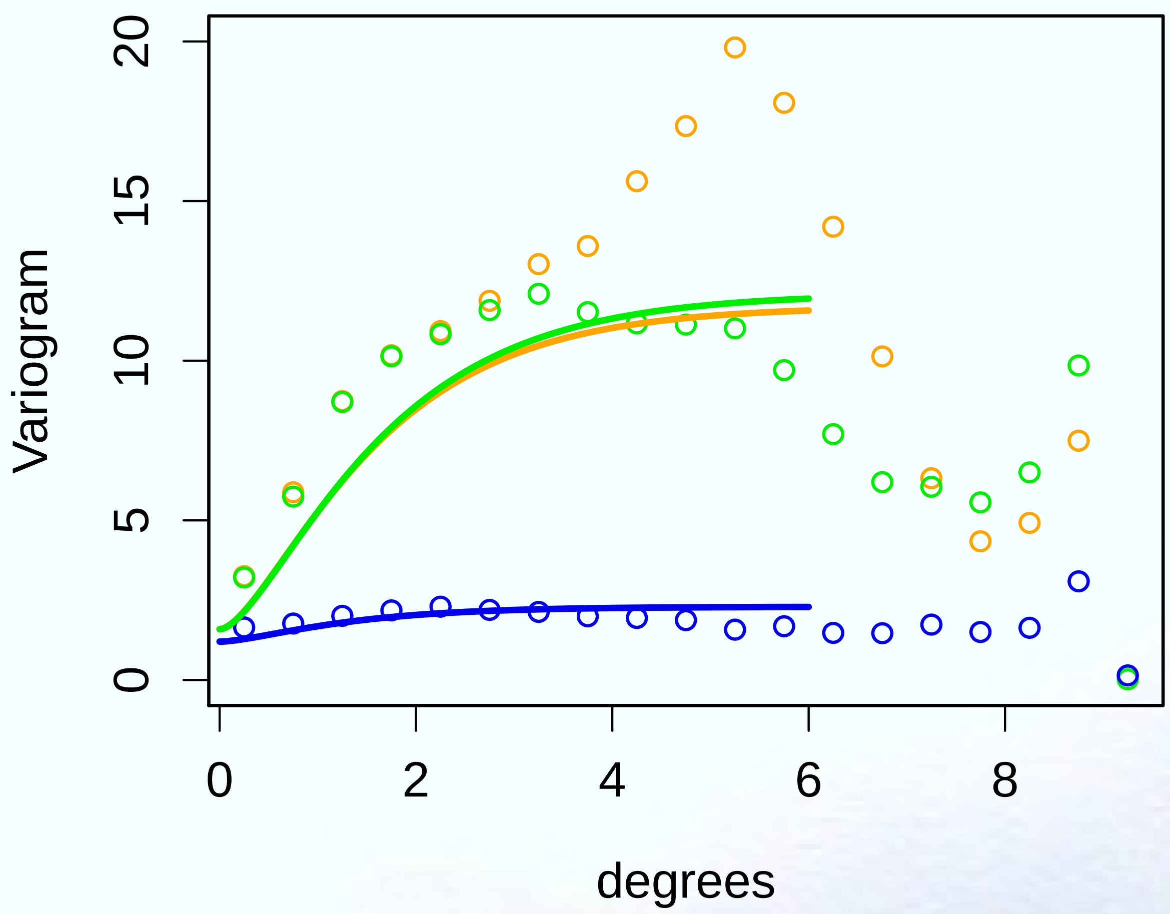
Note the $(1/\rho)$ cancels with ρ in front of the covariance.

$\hat{g}(\boldsymbol{x}^*)$ only depends on $\lambda = \sigma^2/\rho$

λ called the smoothing parameter – the "noise to signal" ratio

Variograms for MAM min temps

Some motivation for adding spatial components.



No spatial covariates, removing linear lon/lat function, also elevation.

Additive model with fixed effects

spatial drift

$$y_i = d_1 + \boldsymbol{x}_{i,1} d_2 + \boldsymbol{x}_{i,2} d_3 + g(\boldsymbol{x}_i) + \epsilon_i$$

adding elevation

$$y_i = d_1 + \boldsymbol{x}_{i,1} d_2 + \boldsymbol{x}_{i,2} d_3 + \text{elev}_i d_4 + g(\boldsymbol{x}_i) + \epsilon_i$$

In general

$$y_i = f(\boldsymbol{x}_i) + \epsilon_i = \sum_j \phi(\boldsymbol{x}_i) d_j + g(\boldsymbol{x}_i) + \epsilon_i$$

Interpretation is that the coefficients are fixed values (and g is a random function)

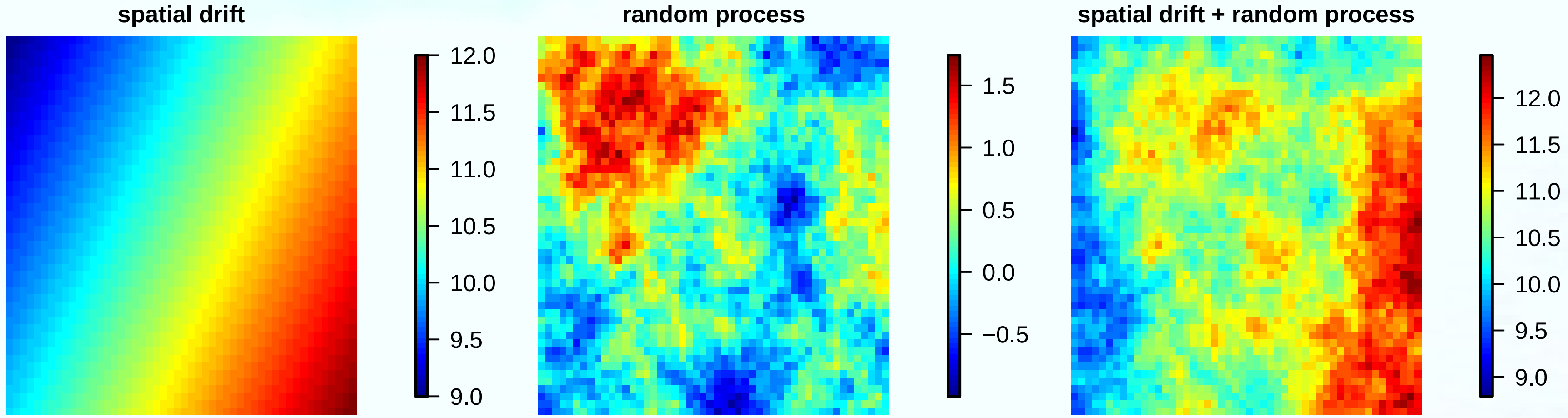
$f(\boldsymbol{x})$ is a random function with

- Fixed effects: $E[f(\boldsymbol{x})] = \sum_j \phi(\boldsymbol{x}_i) d_j$
- Random effects: Covariance function for f same as g.

How do we estimate both the fixed effects and random components simultaneously?

A pitfall ignoring a spatial trend

standard exponential process with linear spatial trend (drift).

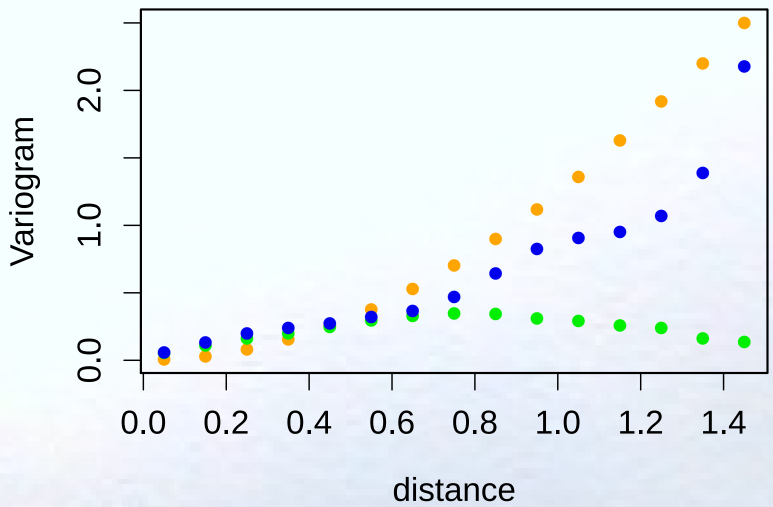


Variograms

spatial drift

$g(x)$

complete field



Generalized least squares

The fixed part of the additive mode can be reexpessed as

$$\boldsymbol{y} = X\boldsymbol{d} + \boldsymbol{E}$$

where $E_i = g(\boldsymbol{x}_i) + \epsilon_i$

Ordinary least squares estimate for $\mathbf{d}$

$$\hat{d} = (X^T X)^{-1} X^T \boldsymbol{y}$$

Only valid (and BLUE) if the errors are uncorrelated and constant variance.

Generalized Least squares

For our case covariance of $\boldsymbol{E}$ is $\rho K + \sigma^2 I$. Better estimate is

$$\hat{d} = (X^T M^{-1} X)^{-1} X^T M^{-1} \boldsymbol{y}$$

$M = \rho K + \sigma^2 I$

Where does this come from?

- $M^{-1/2}$ take as inverse of cholesky decomposition

- Transform observations by $M^{-1/2}$

$$M^{-1/2}\boldsymbol{y} = M^{-1/2}X\boldsymbol{d} + M^{-1/2}\boldsymbol{E}$$

- Transformed errors $\sim (0, I)$!

- Apply ordinary least squares to transformed observations.

$$\boldsymbol{y}^* = X^*\boldsymbol{d} + \boldsymbol{e}^*$$

$\boldsymbol{y}^* = M^{-1/2}\boldsymbol{y}$ and $X^* = M^{-1/2}X$.

- OLS for transformed model = GSL for original model. Since OLS is optimal for estimating $\boldsymbol{d}$ so is the GLS estimator in the original data scale.

Back to Kriging

The idea is simple but harder to prove rigorously.

Estimate linear fixed part of model using GLS.

$$\hat{d} = (X^T M^{-1} X)^{-1} X^T M^{-1} \boldsymbol{y}$$

$M = K + \sigma^2 I$

• Need to know the covariance function! Looking at the OLS residuals is a practical way to identify the spatial process model.

Apply standard Kriging to the GLS residuals

• "Krig" adjusted spatial observations: $\boldsymbol{y} - X\hat{d}$

Using the coefficient/covariance function version:

$$\hat{c} = (\rho K + \sigma^2 I)^{-1}(\boldsymbol{y} - X\hat{d})$$

$$\hat{g}(\boldsymbol{x}) = \sum_i \rho k(\boldsymbol{x}, \boldsymbol{x}_i)\hat{c}_i$$

• More parameters than data points ??

Acronym heuristics

- GLS is BLUE
- Kriging is BLUE

Combining them should be BLUE.

The rigorous approach

Minimize the prediction variance under the constraints that

- Estimate is a linear combination of the data
- Expected value of estimate equal to fixed part of model.

Using `mKrig` in `fields`

Assume through maximum likelihood fitting we have $\rho = 10.1$, $\theta = 1.1$ and $\sigma = 1.26$ for a Matern covariance with smoothness 1.0.

When fixed part of model is a constant.

```
obj0<- mKrig( xHW1, yHW1, Covariance="Matern", theta=1.1,
                         smoothness=1.0,  lambda= (1.26^2)/10.1, m=1 )
summary( obj0)
```

`lambda`

Noise to signal ratio σ^2/ρ. This can be zero.

`m`

An $m - 1$ degree polynomial used in the fixed part of spatial model. Default is $m = 2$ – a linear function in coordinates.

When fixed part of model is linear in lon/lat.

$\rho = 10.5$, $\theta = 1.13$ and $\sigma = 1.26$

```
obj1<- mKrig( xHW1, yHW1, Covariance="Matern", theta=1.13,
                                smoothness=1.0,  lambda= (1.26^2)/10.5, m=2 )
```

Adding in elevation

$\rho = 1.09$, $\theta = .90$ and $\sigma = 1.10$

```
obj2<- mKrig( xHW1, yHW1, Z= elevHW1, Covariance="Matern", theta=1.13,
                                smoothness=1.0,  lambda= (1.10^2)/1.09, m=2 )
```

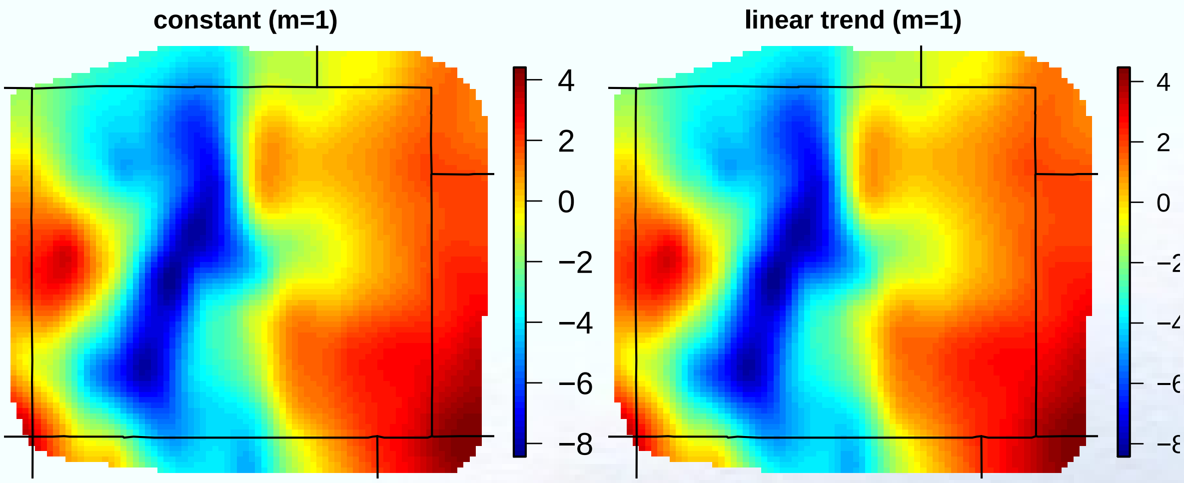
Plotting the spatial estimates

Just the random component and spatial drift

```
surface( obj1)
# or in two steps
out.surface<- predict.surface(obj1)
image.plot( out.surface)
```

For the model with elevation for comparison omit elevation part:

```
out.surface<- predict.surface(obj1, drop.Z=TRUE)
```



Adding elevation

As preliminary recall that images/gridded data in R are organized as a `list` with components
`x`: a vector of x grid locations, `y`: y grid locations
and `z`: a matrix with the actual value of the surface.

```
# smooth surface:
out2<- predict.surface( obj2, drop.Z=TRUE, nx=200, ny=200)
grid.list<- list( x= out2$x, y=out2$y)
# approximate elevations on this grid (this is cheating a bit!)
data(RMelevation)
elev.grid<- interp.surface.grid( RMelevation, grid.list)
elev.coef<- obj2$d[4] # where the coefficients are kept!

# add two surfaces together
full.surface<-  out2$z + elev.grid$z*elev.coef

image.plot( grid.list$x, grid.list$y, full.surface)
```

Surface with elevation