High Resolution Simulation of Nonstationary Random Fields

Will Kleiber

Department of Applied Mathematics
University of Colorado
Boulder, CO

Workshop on Stochastic Weather Generators, Avignon, France, September 18, 2014

Research supported by NSF DMS-1417724

Stochastic Weather Generators

- Agricultural, ecological, hydrologic models often require daily weather (e.g. precipitation, minimum/maximum temperature, solar radiation)
  - On grid
  - In the future
- Stochastic Weather Generators (SWGs) can be used to produce infinitely long series of synthetic weather, for observation network infilling, or climate model downscaling
- SWGs are statistical models whose simulated values “look like” observed weather
  - Daily statistics
  - Interannual statistics
  - Spatial statistics
- Approaches: empirical, model-based, (combination?)

Spatial Models

Modern SWGs require spatial simulations that honor observed spatial correlations.

Difficulties:

- Simulation over complex terrain
- Simulation over large heterogeneous areas

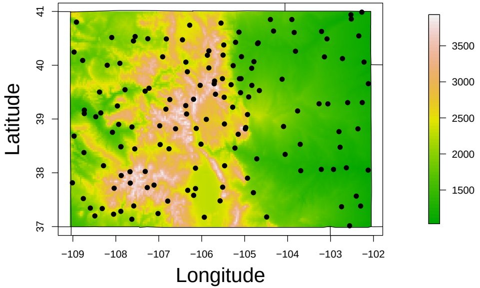
Spatial Models

Modern SWGs require spatial simulations that honor observed spatial correlations.

Difficulties:

- Simulation over complex terrain
- Simulation over large heterogeneous areas

Data: 145 stations from the Global Historical Climatology Network. Daily minimum temperature during 1893-2011.



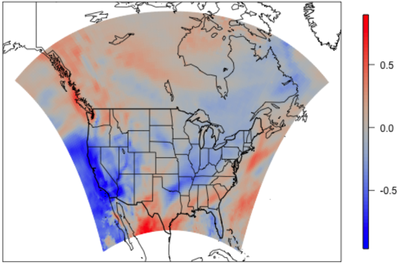
Spatial Models

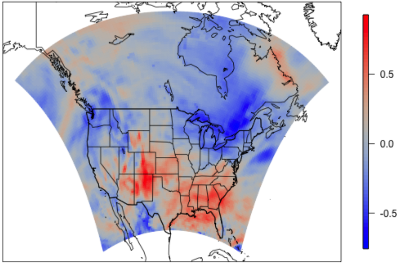
Modern SWGs require spatial simulations that honor observed spatial correlations.

Difficulties:

- Simulation over complex terrain
- Simulation over large heterogeneous areas

More data: Gridded average winter precipitation anomalies from a regional climate model output (11,760 grid cells).





Properties of $Z(\mathbf{s})$

If $Z(\mathbf{s}), \mathbf{s} \in \mathbb{R}^d$ is a Gaussian process, then we need only specify

- The mean function $\mathbb{E}Z(\mathbf{s})$ (suppose $= 0$)
- The covariance function $C(\mathbf{s}_1, \mathbf{s}_2) = \mathrm{Cov}(Z(\mathbf{s}_1), Z(\mathbf{s}_2))$

Properties of $Z(\mathbf{s})$

If $Z(\mathbf{s}), \mathbf{s} \in \mathbb{R}^d$ is a Gaussian process, then we need only specify

- The mean function $\mathbb{E}Z(\mathbf{s})$ (suppose $= 0$)
- The covariance function $C(\mathbf{s}_1, \mathbf{s}_2) = \mathrm{Cov}(Z(\mathbf{s}_1), Z(\mathbf{s}_2))$

If $C(\mathbf{s}_1, \mathbf{s}_2) = C(\mathbf{s}_1 - \mathbf{s}_2)$ then $Z(\mathbf{s})$ is a stationary process. Otherwise, $Z(\mathbf{s})$ is nonstationary.

Properties of $Z(\mathbf{s})$

If $Z(\mathbf{s}), \mathbf{s} \in \mathbb{R}^d$ is a Gaussian process, then we need only specify

- The mean function $\mathbb{E}Z(\mathbf{s})$ (suppose $= 0$)
- The covariance function $C(\mathbf{s}_1, \mathbf{s}_2) = \text{Cov}(Z(\mathbf{s}_1), Z(\mathbf{s}_2))$

If $C(\mathbf{s}_1, \mathbf{s}_2) = C(\mathbf{s}_1 - \mathbf{s}_2)$ then $Z(\mathbf{s})$ is a stationary process.
Otherwise, $Z(\mathbf{s})$ is nonstationary.

Goal: Simulate $Z(\mathbf{s})$ on a fine grid.

Properties of $Z(\mathbf{s})$

If $Z(\mathbf{s}), \mathbf{s} \in \mathbb{R}^d$ is a Gaussian process, then we need only specify

- The mean function $\mathbb{E}Z(\mathbf{s})$ (suppose $= 0$)
- The covariance function $C(\mathbf{s}_1, \mathbf{s}_2) = \text{Cov}(Z(\mathbf{s}_1), Z(\mathbf{s}_2))$

If $C(\mathbf{s}_1, \mathbf{s}_2) = C(\mathbf{s}_1 - \mathbf{s}_2)$ then $Z(\mathbf{s})$ is a stationary process. Otherwise, $Z(\mathbf{s})$ is nonstationary.

Goal: Simulate $Z(\mathbf{s})$ on a fine grid.

Usual approach: Simulate independent normals ε and form $L\varepsilon$, where L is a Cholesky factor.

Properties of $Z(\mathbf{s})$

If $Z(\mathbf{s}), \mathbf{s} \in \mathbb{R}^d$ is a Gaussian process, then we need only specify

- The mean function $\mathbb{E}Z(\mathbf{s})$ (suppose $= 0$)
- The covariance function $C(\mathbf{s}_1, \mathbf{s}_2) = \mathrm{Cov}(Z(\mathbf{s}_1), Z(\mathbf{s}_2))$

If $C(\mathbf{s}_1, \mathbf{s}_2) = C(\mathbf{s}_1 - \mathbf{s}_2)$ then $Z(\mathbf{s})$ is a stationary process. Otherwise, $Z(\mathbf{s})$ is nonstationary.

Goal: Simulate $Z(\mathbf{s})$ on a fine grid.

Usual approach: Simulate independent normals ε and form $L\varepsilon$, where L is a Cholesky factor.

Main problem: $Z(\mathbf{s})$ is (often) a nonstationary process, computing L is difficult.

Nonstationary simulation

For nonstationary processes there are very few fast simulation algorithms (spectral method for adapted spectra, e.g.).

Nonstationary simulation

For nonstationary processes there are very few fast simulation algorithms (spectral method for adapted spectra, e.g.).

Suppose $Z(\mathbf{s})$ is a nonstationary process that can be written

$$Z(\mathbf{s}) = D(\varphi(\mathbf{s}))$$

where

- $D(\mathbf{s})$ is a stationary process with covariance $C_D(\cdot)$
- $\varphi : \mathbb{R}^d \to \mathbb{R}^d$ is an invertible deformation function.

Then,

$$\begin{aligned} \text{Cov}(Z(\mathbf{s}_1), Z(\mathbf{s}_2)) &= C(\mathbf{s}_1, \mathbf{s}_2) \\ &= C_D\big(\varphi(\mathbf{s}_1) - \varphi(\mathbf{s}_2)\big). \end{aligned}$$

Nonstationary simulation

For nonstationary processes there are very few fast simulation algorithms (spectral method for adapted spectra, e.g.).

Suppose $Z(\mathbf{s})$ is a nonstationary process that can be written

$$Z(\mathbf{s}) = D(\varphi(\mathbf{s}))$$

where

- $D(\mathbf{s})$ is a stationary process with covariance $C_D(\cdot)$
- $\varphi : \mathbb{R}^d \to \mathbb{R}^d$ is an invertible deformation function.

Then,

$$\begin{aligned}\text{Cov}(Z(\mathbf{s}_1), Z(\mathbf{s}_2)) &= C(\mathbf{s}_1, \mathbf{s}_2) \\ &= C_D\big(\varphi(\mathbf{s}_1) - \varphi(\mathbf{s}_2)\big).\end{aligned}$$

Key idea: To simulate a vector $(Z(\mathbf{s}_1), \ldots, Z(\mathbf{s}_n))'$, simulate $(D(\varphi(\mathbf{s}_1)), \ldots, D(\varphi(\mathbf{s}_n)))'$ via stationary simulation algorithm.

Nonstationary simulation

For nonstationary processes there are very few fast simulation algorithms (spectral method for adapted spectra, e.g.).

Suppose $Z(\mathbf{s})$ is a nonstationary process that can be written

$$Z(\mathbf{s}) = D(\varphi(\mathbf{s})) = \left(\text{actually } \sigma(\mathbf{s})D(\varphi(\mathbf{s}))\right)$$

where

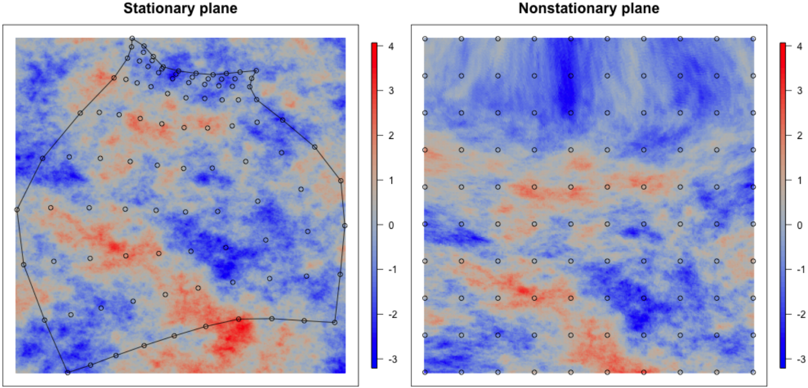
- $D(\mathbf{s})$ is a stationary process with covariance $C_D(\cdot)$
- $\varphi : \mathbb{R}^d \to \mathbb{R}^d$ is an invertible deformation function.

Then,

$$\begin{aligned} \text{Cov}(Z(\mathbf{s}_1), Z(\mathbf{s}_2)) &= C(\mathbf{s}_1, \mathbf{s}_2) \\ &= C_D\left(\varphi(\mathbf{s}_1) - \varphi(\mathbf{s}_2)\right). \end{aligned}$$

Key idea: To simulate a vector $(Z(\mathbf{s}_1), \ldots, Z(\mathbf{s}_n))'$, simulate $(D(\varphi(\mathbf{s}_1)), \ldots, D(\varphi(\mathbf{s}_n)))'$ via stationary simulation algorithm.

Example



Stationary simulation: spectral method

The spectral method (Shinozuka and Jan, 1972) relies on the representation

$$Z(\mathbf{s}) = \mathrm{Re}\left[\int \exp(2\pi\imath\boldsymbol{\omega}'\mathbf{s})\mathrm{d}Y(\boldsymbol{\omega})\right].$$

Simulations follow from a discretization of the integral.

This method is approximate.

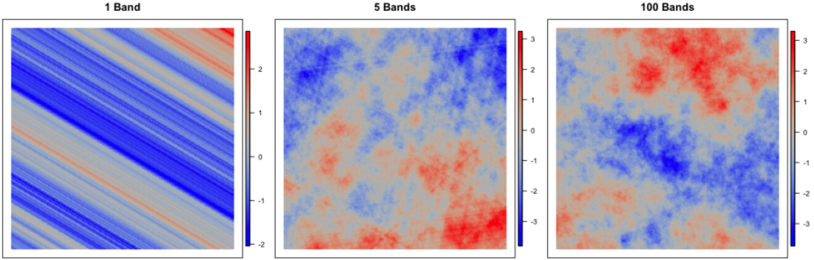
Stationary simulation: turning bands

The turning bands (Matheron, 1973; Mantoglou and Wilson 1982) method uses

$$Z(\mathbf{s}) = \frac{1}{\sqrt{L}} \sum_{i=1}^{L} Z_i(\mathbf{s} \cdot \mathbf{e}_i)$$

where $\{Z_i(\cdot)\}_{i=1}^{L}$ are mutually independent one-dimensional processes, $\{\mathbf{e}_i\}_{i=1}^{L}$ are unit vectors.

This method is approximate.



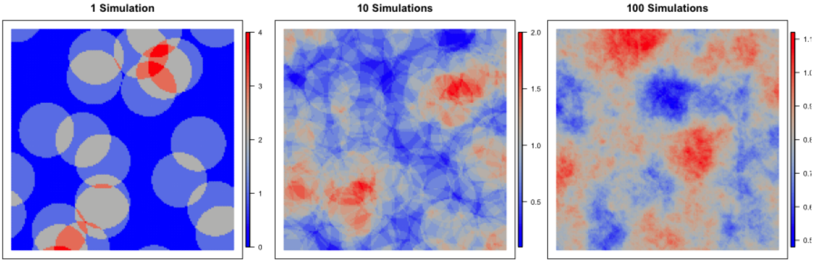
Stationary simulation: random coin

The random coin approach (Chilés and Delfiner 1999; Schlather 2012) requires

$$C(\mathbf{h}) = \int g(\mathbf{s})g(\mathbf{s}+\mathbf{h})\mathrm{d}\mathbf{s}.$$

The approximate simulation is $\sum_{\mathbf{h}\in\Pi} g(\mathbf{s}-\mathbf{h})$ where Π is a stationary Poisson point process with unit intensity.

This method is approximate.



Stationary simulation: Circulant embedding

The circulant embedding method (Dietrich and Newsam 1993, 1997; Wood and Chan 1994) requires simulation locations to be on a regular grid.

In $d = 2$ dimension, if $\{\mathbf{s}_i\}_{i=1}^n$ are regular then

$$\Sigma_0 = \left(C(\mathbf{s}_i - \mathbf{s}_j)\right)_{i,j=1}^n$$

is block Toeplitz. We embed Σ_0 in a circulant matrix

$$\Sigma = \begin{pmatrix} \Sigma_0 & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix} = FDF^*$$

where D is diagonal with eigenvalues, F has complex exponential entries.

This method is exact.

Estimation

Need to estimate:

- $\mathrm{Cov}(Z(\mathbf{s}_1), Z(\mathbf{s}_2)) = C(\mathbf{s}_1, \mathbf{s}_2)$
- $\varphi(\cdot)$

Estimation

Need to estimate:

- $\text{Cov}(Z(\mathbf{s}_1), Z(\mathbf{s}_2)) = C(\mathbf{s}_1, \mathbf{s}_2)$
- $\varphi(\cdot)$

For observations $Y(\mathbf{s}_i)$ at locations $\mathbf{s}_1, \ldots, \mathbf{s}_n$, use

$$\hat{C}(\mathbf{x}, \mathbf{y}) = \frac{\sum_{i=1}^{n} \sum_{j=1}^{n} K_\lambda\left(\|\mathbf{x} - \mathbf{s}_i\|\right) K_\lambda\left(\|\mathbf{y} - \mathbf{s}_j\|\right) Y(\mathbf{s}_i) Y(\mathbf{s}_j)}{\sum_{i=1}^{n} \sum_{j=1}^{n} K_\lambda(\|\mathbf{x} - \mathbf{s}_i\|) K_\lambda(\|\mathbf{y} - \mathbf{s}_j\|)}$$

where K_λ is a kernel function with bandwidth λ.

Estimation

Need to estimate:

- $\text{Cov}(Z(\mathbf{s}_1), Z(\mathbf{s}_2)) = C(\mathbf{s}_1, \mathbf{s}_2)$
- $\varphi(\cdot)$

For observations $Y(\mathbf{s}_i)$ at locations $\mathbf{s}_1, \ldots, \mathbf{s}_n$, use

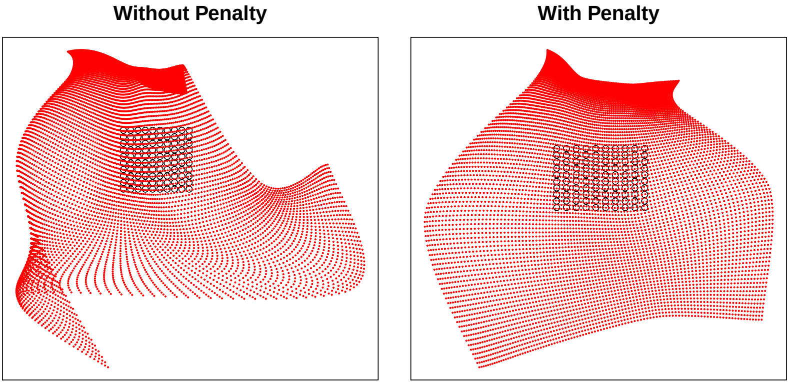
$$\hat{C}(\mathbf{x}, \mathbf{y}) = \frac{\sum_{i=1}^n \sum_{j=1}^n K_\lambda\left(\|\mathbf{x} - \mathbf{s}_i\|\right) K_\lambda\left(\|\mathbf{y} - \mathbf{s}_j\|\right) Y(\mathbf{s}_i) Y(\mathbf{s}_j)}{\sum_{i=1}^n \sum_{j=1}^n K_\lambda(\|\mathbf{x} - \mathbf{s}_i\|) K_\lambda(\|\mathbf{y} - \mathbf{s}_j\|)}$$

where K_λ is a kernel function with bandwidth λ.

Estimate φ via

$$\min_{\varphi} \sum_{i=1}^n \sum_{j=1}^n \left(\hat{C}(\mathbf{s}_i, \mathbf{s}_j) - C_D(\varphi(\mathbf{s}_i) - \varphi(\mathbf{s}_j))\right)^2 + \lambda\langle\varphi, \varphi\rangle.$$

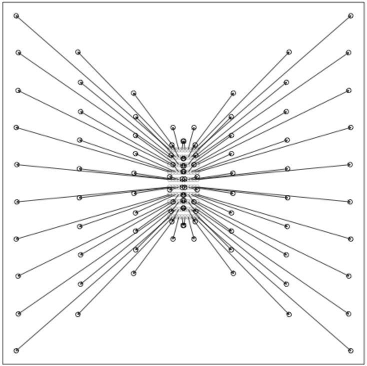
Effect of penalty term



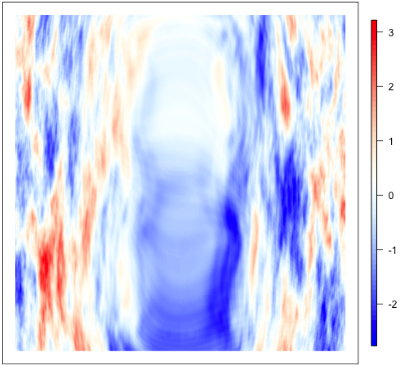
Valleys

A major difficulty is that weather processes often exhibit differing spatial structure in valleys than over higher terrain. Additionally, can allow for continuous simulation between regimes, e.g., land and ocean.

Deformation

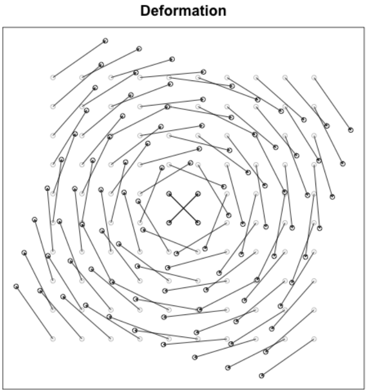


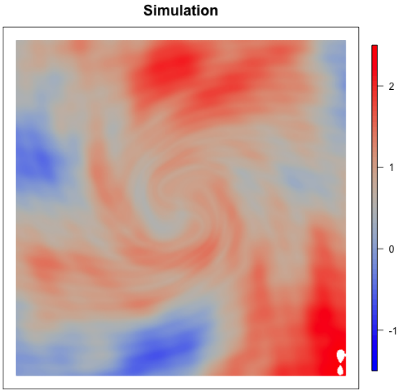
Simulation



Vortices

Vortex-like behavior is crucial at microscales for wind modeling as well as macroscales such as for hurricanes.





Model for observations of minimum temperature $Y(\mathbf{s}), \mathbf{s} \in \mathbb{R}^2$,

$$\begin{aligned} Y(\mathbf{s}) &= \quad \mu(\mathbf{s}) \quad + \quad Z(\mathbf{s}) \quad + \; \varepsilon(\mathbf{s}) \\ &= \text{“Climate”} + \text{“Weather”} + \text{“Error”} \end{aligned}$$

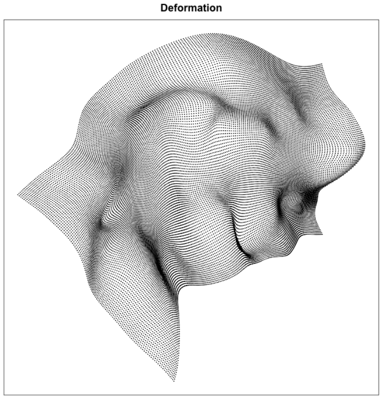
Model for observations of minimum temperature $Y(\mathbf{s}), \mathbf{s} \in \mathbb{R}^2$,

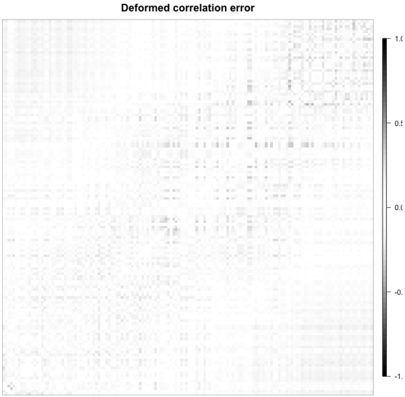
$$\begin{aligned} Y(\mathbf{s}) &= \quad \mu(\mathbf{s}) \quad + \quad Z(\mathbf{s}) \quad + \ \varepsilon(\mathbf{s}) \\ &= \text{“Climate”} + \text{“Weather”} + \text{“Error”} \end{aligned}$$

Steps:

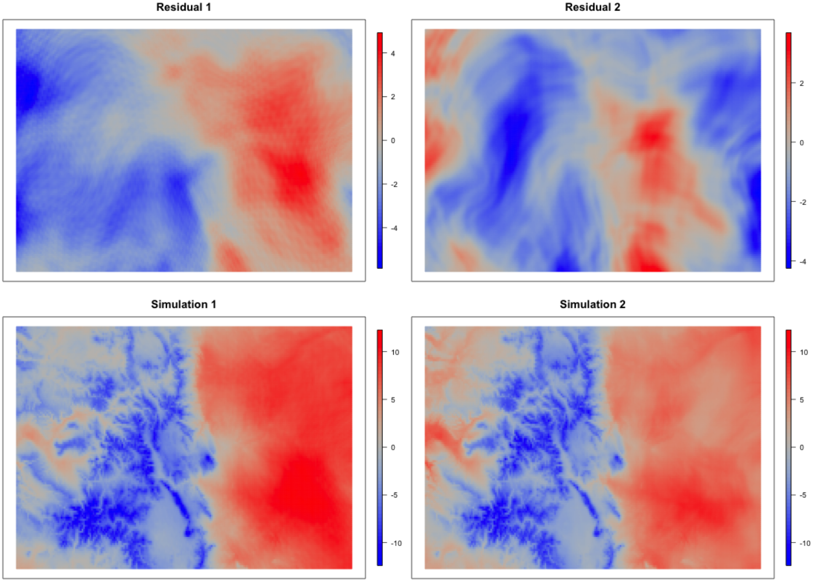
- Extract $\hat{\mu}(\mathbf{s})$ via linear regression with spatially varying coefficients
- Estimate covariance $C(\cdot, \cdot)$ of $Z(\mathbf{s})$ nonparametrically
- Suppose the stationary process $D(\mathbf{s})$ has Matérn covariance with smoothness 1, range estimated from data
- Fit $\hat{\varphi}$
- Simulate $Z(\mathbf{s}) = D(\varphi(\mathbf{s}))$ on a grid of 25,000 locations

Colorado

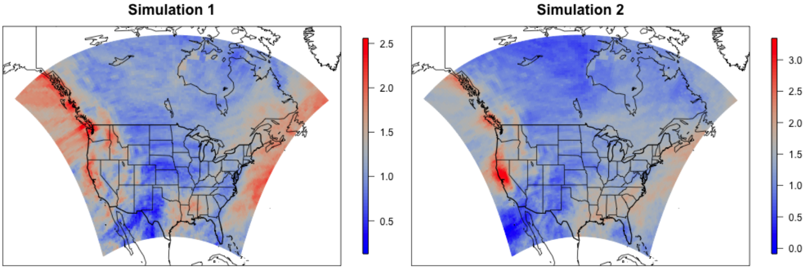




Colorado


Residual 1
4
2
0
-2
-4
Residual 2
2
0
-2
-4
Simulation 1
10
5
0
-5
-10
Simulation 2
10
5
0
-5
-10

Regional Climate Model



Discussion

- High resolution nonstationary simulation via deformation
- Extensions: space-time, multivariate, non-Gaussian fields (precipitation)
- Better deformation functions?

Shameless Plug

Verdin, A., Rajagopalan, B., Kleiber, W. and Katz, R. W. (2014) "Coupled stochastic weather generation using spatial and generalized linear models." *Stochastic Environmental Research and Risk Assessment*, in press.

- Simulation of minimum, maximum temperature and precipitation occurrence
- Temperature coupled to precipitation occurrence in conditional fashion