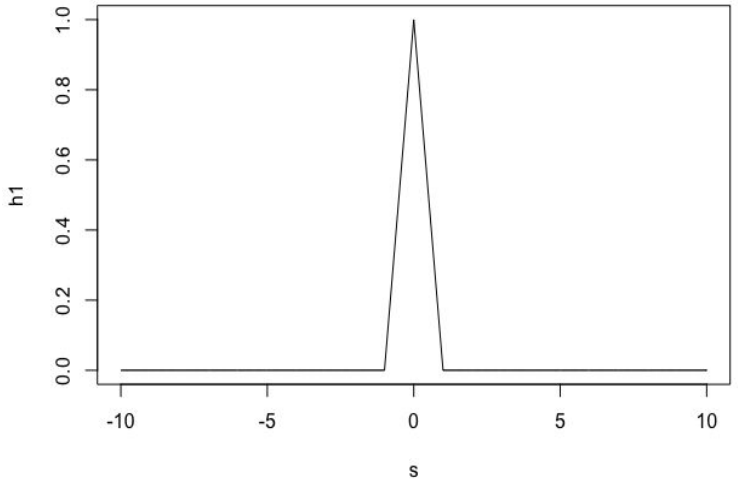
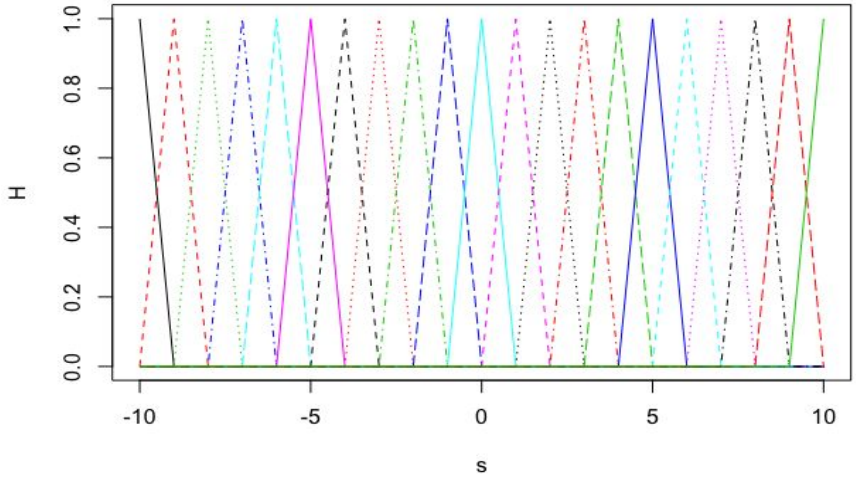
Lattice Kriging

Rebekah Colonnese, Andrew Freedman, Megan Tabor

Visual of LatticeKrig (LK) methods

- Uses **radial basis functions** at multiple spatial scales as covariates to obtain sparse positive definite matrices, lessening computational time dramatically
- What is a basis function? → Below are triangular basis functions in 1-D
- Radial basis functions used in LK methods: 2-D and decay exponentially





Theoretical motivation

- At each resolution **r**, the marginal variance at that resolution **h** is given by the 2-D rectangular grid of basis functions **u**, whose overlap is controlled by **λ**

$$h_{rk}(\boldsymbol{s}) = \psi(\|\boldsymbol{s} - \boldsymbol{u}_{rk}\|/\lambda_r)$$

- Where

$$\psi(d) \propto \begin{cases} \frac{1}{3}(1-d)^6(35d^2+18d+3) & \text{if } d \le 1 \\ 0 & \text{otherwise.} \end{cases}$$

- called **Wendland polynomials** which by being used for these basis functions allows for the sparse matrix given by **h** to be symmetric with all eigenvalues being positive---allowing for a **Cholesky decomposition** to be used to further reduce computational time

Equations from: Heaton, M. J., Datta, A., Finley, A. O., Furrer, R., Guinness, J., Guhaniyogi, R., ... & Lindgren, F. (2019). A case study competition among methods for analyzing large spatial data. Journal of Agricultural, Biological and Environmental Statistics, 24(3), 398-425.

Theoretical motivation

- According to Press et al 1995, Cholesky decompositions are **roughly twice as efficient at solving systems of linear equations** than the typical decomposition used by computer languages based in *C* (which *R* is)
- From *Wikipedia*:

Here is the Cholesky decomposition of a symmetric real matrix:

$$\begin{pmatrix} 4 & 12 & -16 \\ 12 & 37 & -43 \\ -16 & -43 & 98 \end{pmatrix} = \begin{pmatrix} 2 & 0 & 0 \\ 6 & 1 & 0 \\ -8 & 5 & 3 \end{pmatrix} \begin{pmatrix} 2 & 6 & -8 \\ 0 & 1 & 5 \\ 0 & 0 & 3 \end{pmatrix}.$$

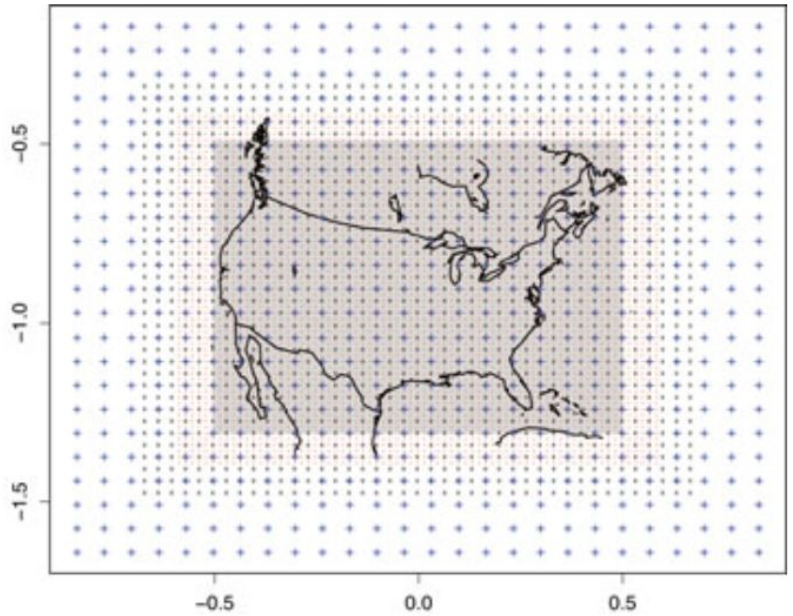
And here is its LDL^T decomposition:

$$\begin{pmatrix} 4 & 12 & -16 \\ 12 & 37 & -43 \\ -16 & -43 & 98 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 3 & 1 & 0 \\ -4 & 5 & 1 \end{pmatrix} \begin{pmatrix} 4 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 9 \end{pmatrix} \begin{pmatrix} 1 & 3 & -4 \\ 0 & 1 & 5 \\ 0 & 0 & 1 \end{pmatrix}$$

Press, William H.; Saul A. Teukolsky; William T. Vetterling; Brian P. Flannery (1992). *Numerical Recipes in C: The Art of Scientific Computing* (second ed.).

Knots (basis functions) from Nychka et al 2015

- Figure to the right shows **multiresolution** basis functions
- Knots given at each point
- **Buffer region** also shown here
- Spatial dependence among coefficients for each resolution is modeled using a multivariate normal distributed **spatial autoregressive (SAR) model**



Douglas Nychka, Soutir Bandyopadhyay, Dorit Hammerling, Finn Lindgren & Stephan Sain (2015) A Multiresolution Gaussian Process Model for the Analysis of Large Spatial Datasets, Journal of Computational and Graphical Statistics, 24:2, 579-599, DOI: 10.1080/10618600.2014.914946

Implementation - coding it up

1) Install **LatticeKrig** package in R
2) Perform LK methods using **obj=LatticeKrig(x,y,...)** which will estimate a full LKinfo object from your data using tuning parameters that you set. Where **x**=locations and **y**=result at those locations…
3) Prediction can be done by using **predict.LKrig** to make spatial predictions at new locations
4) Tune parameters in step 2, and choose “best” tuning parameters based on cross-validation (we used the g=21 data as the test set)

Implementation - LKinfo

Create or update the LatticeKrig model object (LKinfo) for spatial fitting.

Description

This function combines some input arguments with defaults for other to create a list describing the LatticeKrig spatial model. A key to specifying the LatticeKrig spatial model is specifying the geometry, e.g. LKRectangle for a 2-d rectangular domain. Each geometry has some parameters that control the basic model setup and these are included through the ... arguments of this function. See the help for this argument below for some examples.

Usage

```
LKrigSetup(x = NULL, nlevel = NULL, alpha = NA, alphaObject =
        NULL, nu = NULL, a.wght = NA, a.wghtObject = NULL, NC
        = NULL, NC.buffer = NULL, normalize = TRUE, lambda =
        NA, sigma = NA, rho = NA, rho.object = NULL,
        latticeInfo = NULL, basisInfo = NULL, LKGeometry =
        "LKRectangle", distance.type = "Euclidean",
        BasisFunction = "WendlandFunction", overlap = 2.5, V =
        NULL, BasisType = "Radial", fixedFunction =
        "LKrigDefaultFixedFunction", fixedFunctionArgs =
        list(m = 2), collapseFixedEffect = FALSE, max.points =
        NULL, mean.neighbor = 50, choleskyMemory = NULL,
        verbose = FALSE, noCheck = FALSE, returnCall = FALSE,
        dense = FALSE, ...)
```

Implementation - tuning parameters of interest

- **NC** → number of lattice grid points at coursest resolution
- **nlevel** → number of different resolutions used in the multiresolution process
- **alpha** → vector of length nlevel with the relative variances for the different multi-resolution levels.
- **a.wght** → controls the correlation range in the SAR model
- **overlap** → controls the overlap among the radial basis functions
- **nu** → controls for the tail behavior of the basis functions at each resolution

Implementation - choosing tuning parameters

Final Parameters	How to Choose Values	How the tuning parameter effects computation time
NC → 80	Want large NC if dataset is large. An increase in NC increases the number of lattice prediction points and results in higher coverage.	Strong linear relationship- as you increase/decrease NC the computation time also increases/decreases
nlevel → 3	Want large if the goal is to have a lot of multi-resolution levels	Strong linear relationship- as you increase/decrease nlevel the computation time also increases/decreases
a.wght → 50	Want large if goal is to put more emphasis on the weight of the central point of each lattice point.	Weak linear relationship- as you increase or decrease a.wght the computation time increases/decreases
overlap → 1.60	Want large if the goal is to have the Wendland basis functions overlap by a large number of lattice units	Weak linear relationship- as you increase or decrease overlap the computation time increases/decreases
nu → .001	Want large if you want the tails of your basis function to be small	Weak linear relationship- as you increase or decrease nu the computation time increases/decreases

Results - Tuning Parameter Differences

The table below describes the average trend observed with an increase in each tuning parameter

Tuning Parameter	**MSE**	**Coverage**
NC	Decreased*	Increased*
nlevel	Decreased*	Increased*
a.wght	About the same	Increased*
overlap	Concave down	Concave down
nu	About the same	Decreased*

*Note: This is just the general trend. Most of these parameters plateau at different values eventually if they are inputted as extreme values, ie. NC = 100 does not give a much higher coverage than NC = 80

Results - Computation Time

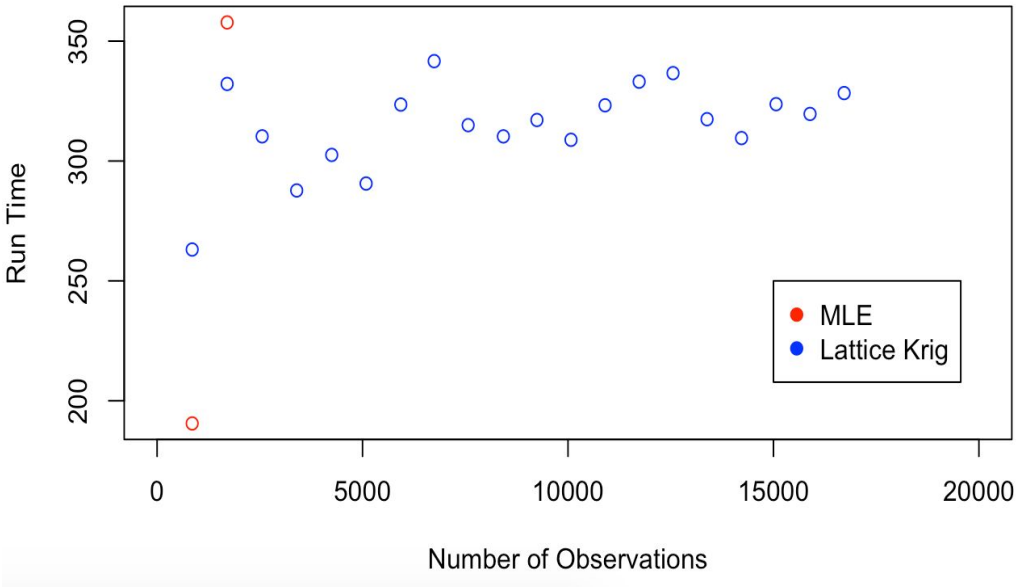
Lattice Krig

Training Set	Compute Time (seconds)
1	281.101
2	291.843
3	283.167
...	...
19	338.803
20	342.309
Total Time	1 hr. 45 min.

MLE

Training Set	Compute Time (seconds)
1	190.548
2	357.814
3	46616.575*
Total Time	13 hrs. +

Computation Time between Lattice Krig and MLE



*Note: Since the computation time for 3rd training set of MLE was extreme outlier, we did not plot it for the purposes of presenting an interpretable plot

Results - Prediction Performance

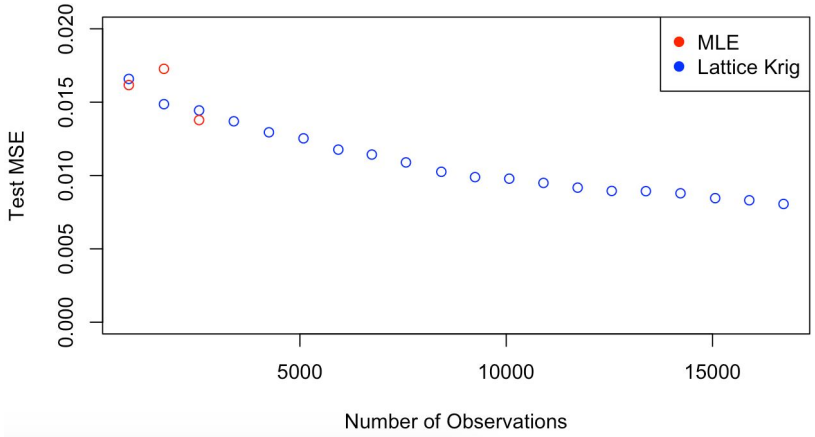
Test MSE

- Test MSE improved as the number of observations increased
- On average, Test MSE was better for Lattice Krig
- Average LK MSE: **0.01096188**

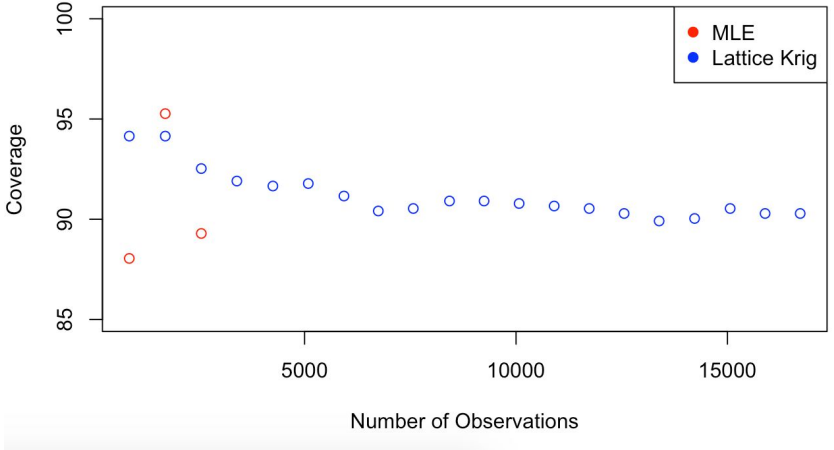
Coverage

- Coverage decreased as the number of observations increased.
- On average, the coverage was better for Lattice Krig
- Average LK Coverage: **91.17061**

Test MSE between Lattice Krig and MLE



Coverage between Lattice Krig and MLE



Conclusions

- Lattice Krig performs far better than MLE in terms of **Computational Time**
  - Promotes Operational Excellence in the workplace
- The **results** from LK and MLE are comparable
  - Results did not suffer from computational time being significantly cut
- Recommend using LK for **Big Data**