ABSTRACT

Title of Thesis:	COMPARISON OF SPATIAL INTERPOLATION METHODS FOR ATMOSPHERIC CARBON MONITORING
	Philip Casey, Master of Arts, 2022
Thesis Directed By:	Dr. Benjamin Kedem Department of Mathematics, Statistics Program

Spatial interpolation is an important tool for prediction of unobserved points in space in the earth and environmental sciences. Three methods for spatial interpolation were atmospheric compared. The first two methods are ordinary kriging and Empirical Bayesian Kriging (EBK). The third method is the Bayesian Transformed Gaussian (BTG) model. The three methods are applied to remotely sensed satellite data of atmospheric carbon dioxide (XCO_2) provided by NASA's Orbiting Carbon Observatory (OCO-2) mission. Cross-validation on the data shows that the methods are close in terms of mean squared error (MSE) when applied to XCO_2 data.

COMPARISON OF SPATIAL INTERPOLATION METHODS FOR ATMOSPHERIC CARBON MONITORING

by

Philip Casey

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park, in partial fulfillment
of the requirements for the degree of
Master of Arts in Statistics
2022

Advisory Committee:
Professor Benjamin Kedem, Chair
Professor Vince Lyzinski
Professor Lawrence Washington

Dedication

To my family and Allie

Acknowledgements

I would like to thank Dr. Benjamin Kedem. It has been a wonderful opportunity to learn from his many years of experience. His teaching was my first exposure to spatial statistics and I have him to thank to putting me on an excellent track forward. He is a consummate teacher and innovator in the field.

I would also like to thank the NASA DEVELOP program for presenting such great opportunities for me bridge statistics and remote sensing.

Finally, I want to thank my parents and Allie Stavrolakis. I feel very privileged to have their love and support as I move forward with my education.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

1.1 Spatial Interpolation

Spatial interpolation is an important tool for prediction of unobserved points in space. Over the past 80 years, the popularity of statistical interpolation has continued to grow in a variety of fields such as the earth and environmental sciences. As more sophisticated techniques and methods are developed and advanced software makes statistical interpolation easier to implement, it is important to understand the underlying theory and applications of the techniques. In this investigation will look at three ways to do prediction: ordinary kriging, empirical Bayesian kriging (Krivoruchko and Gribov 2019), and the Bayesian Transformed Gaussian Model (BTG) (De Oliveira et al. 1997).

Chapter 2: Kriging

2.1 Stationary Random Fields

Stationary random fields are of important use in spatial prediction. Stationarity is not absolutely needed, but it helps with deriving processes.

Let $\{Z(\mathrm{s})\}, \mathrm{s} \in D$, be a spatial process or a random field. By D is a subset of $R^d, d \geq 1$, and $Z(\mathrm{s})$ is a random variable for each $s \in D$. A random field $\{Z(\mathrm{s})\}$ is Gaussian if for all $\mathrm{s}_1, \ldots, \mathrm{s}_n \in D$, the joint distribution of $\big(Z(\mathrm{s}_1), \ldots, Z(\mathrm{s}_n)\big)$ is multivariate normal. (Kedem and Fokianos p. 151)

$\{Z(\mathrm{s})\}$ is second-order stationary, when

$$\mathrm{E}(Z(\mathrm{s})) = \mu,\ \mathrm{s} \in D$$

and

$$\mathrm{Cov}\,(Z(\mathbf{s}+\mathbf{h}), Z(\mathbf{s})) \equiv C(\mathbf{h}),\ \mathbf{s}+\mathbf{h}, \mathbf{s} \in D$$

The function $C(\cdot)$ is called the covariogram or covariance function. (Kedem and Fokianos p. 151)

There are a couple of different useful functions depending on a parameter vector θ as follows.

Exponential correlation

$$K_\theta(l) = \exp\big(l^{\theta_2} \log \theta_1\big),$$

Where $\theta_1 \in (0,1)$ controls the correlation range and $\theta_2 \in (0,2]$ controls the smoothness of $\{Z(\mathrm{s})\}$.

Matern Correlation

$$K_{\boldsymbol{\theta}}(l) = \begin{cases} \frac{1}{2^{\theta_2-\Gamma(\theta_2)}}\left(\frac{l}{\theta_1}\right)^{\theta_2}\kappa_{\theta_2}\left(\frac{l}{\theta_1}\right) & \text{if } l \neq 0 \\ 1 & \text{if } l = 0 \end{cases}$$

where $\theta_1 > 0, \theta_2 > 0$, and κ_{θ_2} is a modified Bessel function of the third kind of order θ_2. The parameter θ_2 is considered the more critical parameter controlling the mean square differentiability of $\{Z(\text{ s})\}$.

Spherical Correlation:

$$K_{\theta}(l) = \begin{cases} 1 - \frac{3}{2}\left(\frac{l}{\theta}\right) + \frac{1}{2}\left(\frac{l}{\theta}\right)^3 & \text{if } l \leq \theta \\ 0 & \text{if } l > \theta \end{cases}$$

where $\theta > 0$ controls the correlation range.

Realization of the spatial fields is below using RStudio software:

```
1.  library(gstat)
2.  library(sp)
3.  library(grDevices)
4.
5.  xy <- expand.grid(1:100, 1:100)
6.  names(xy) <- c('x','y')
7.  g.dummy <- gstat(formula=z~1, locations=~x+y, dummy=T, beta=1,
    model=vgm(psill=0.025, range=5, model='Exp'), nmax=20)
8.  yy <- predict(g.dummy, newdata=xy, nsim=1)
9.  gridded(yy) = ~x+y
10. spplot(obj=yy[1])
11.
12. x <- seq(from = 1, to = 100, by = 1)
13. y <- seq(from = 1, to = 100, by = 1)
14. z <- matrix(0,100,100)
15. for(i in 1:100) {for(j in 1:100){ z[i,j] <- yy$sim1[100*(i-1)+j]}}
16.
17. persp(x, y, z,
18.       col="lightblue1", phi=20, theta=50,r=50, d=0.1,expand=0.5, ltheta=30,
19.       lphi=180, shade=0.15, ticktype="detailed", nticks=5)
20.
21. h.dummy <- gstat(formula=z~1, locations=~x+y, dummy=T, beta=1,
    model=vgm(psill=0.025, range=5, model='Sph'), nmax=20)
22. yy1 <- predict(h.dummy, newdata=xy, nsim=1)
23. gridded(yy1) = ~x+y
24. spplot(obj=yy1[1])
25.
26. x1 <- seq(from = 1, to = 100, by = 1)
27. y1 <- seq(from = 1, to = 100, by = 1)
```

```
z1 <- matrix(0,100,100)
for(i in 1:100) {for(j in 1:100){ z1[i,j] <- yy1$sim1[100*(j-1)+i]}}

persp(x1, y1, z1,
      col="lightblue1", phi=20, theta=50,r=50, d=0.1,expand=0.5, ltheta=30,
      lphi=180, shade=0.15, ticktype="detailed", nticks=5)

j.dummy <- gstat(formula=z~1, locations=~x+y, dummy=T, beta=1,
    model=vgm(psill=0.025, range=5, model='Mat'), nmax=20)
yy2 <- predict(j.dummy, newdata=xy, nsim=1)
gridded(yy2) = ~x+y
spplot(obj=yy2[1])

x2 <- seq(from = 1, to = 100, by = 1)
y2 <- seq(from = 1, to = 100, by = 1)
z2 <- matrix(0,100,100)
for(i in 1:100) {for(j in 1:100){ z2[i,j] <- yy2$sim1[100*(j-1)+i]}}

persp(x2, y2, z2,
      col="lightblue1", phi=20, theta=50,r=50, d=0.1,expand=0.5, ltheta=30,
      lphi=180, shade=0.15, ticktype="detailed", nticks=5)

```

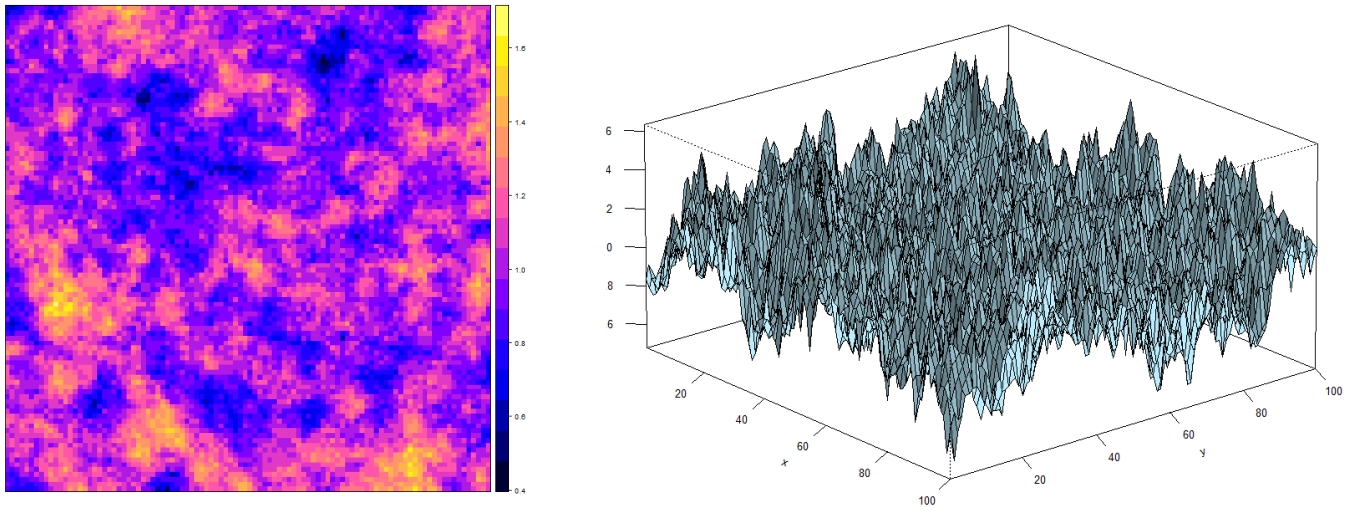

Figure 1. Gaussian(1,1) random field, exponential correlation

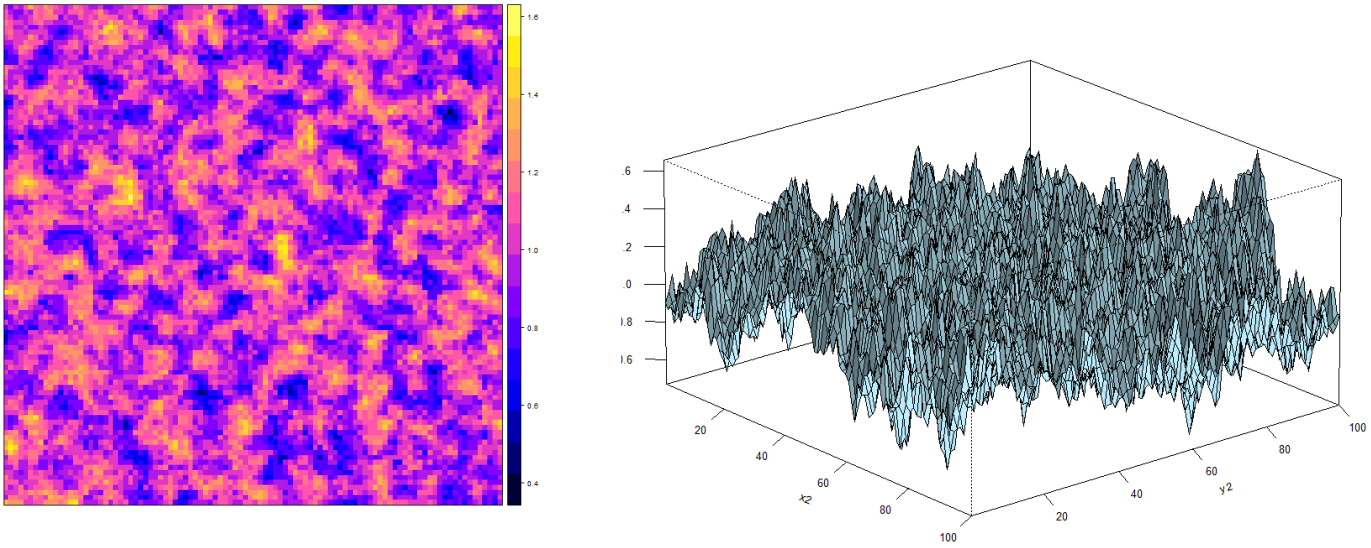

Figure 2. Gaussian(1,1) random field, spherical correlation

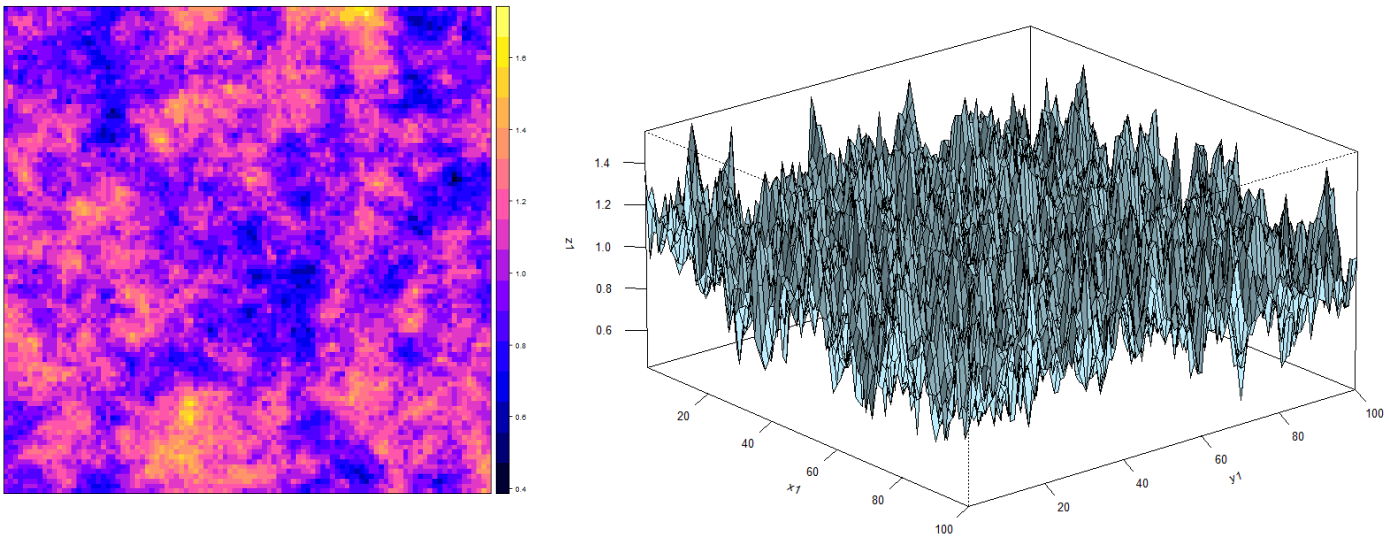

Figure 3. Gaussian(1,1) random field, Matern correlation

2.2 Ordinary Kriging

Given the data

$$\mathbf{Z} \equiv \big(Z(\mathbf{s}_1), \ldots, Z(\mathbf{s}_n)\big)'$$

observed at locations $\{\mathrm{s}_1, \ldots, \mathrm{s}_n\}$ in D, the problem is to estimate or predict $Z(\mathrm{s}_0)$ at location $\mathbf{s}_0$ using the best linear unbiased predictor (BLUP) obtained by minimizing

$$\mathrm{E}\left(Z(\mathrm{s}_0) - \sum_{i=1}^{n} \lambda_i Z(\mathrm{s}_i)\right)^2 \text{ subject to } \sum_{i=1}^{n} \lambda_i = 1$$

Thus is minimized

$$\mathrm{E}\left(Z(\mathbf{s}_0) - \sum_{i=1}^{n} \lambda_i Z(\mathbf{s}_i)\right)^2 - 2m\left(\sum_{i=1}^{n} \lambda_i - 1\right)$$

with respect to $\lambda_1, \ldots, \lambda_n$ and the Lagrange multiplier m. This is done by appealing to the second-order properties of the random field. (Kedem and Fokianos p. 156)

The condition that the λ_i sum to 1 guarantees an unbiased predictor,

$$\mathrm{E}\left(\sum_{i=1}^{n} \lambda_i Z(\mathrm{s}_i)\right) = \mathrm{E}\big(Z(\mathrm{s}_0)\big) \sum_{i=1}^{n} \lambda_i = \mu$$

With the notation

$$
\begin{aligned}
\mathbf{1} &= (1,1,\ldots,1)', \ 1 \times n \text{ vector} \\
\mathbf{c} &= \left(C(\mathbf{s}_0 - \mathbf{s}_1), \ldots, C(\mathbf{s}_0 - \mathbf{s}_n)\right)' \\
\mathbf{C} &= \left(C(\mathbf{s}_i - \mathbf{s}_j)\right), \ i,j = 1,\ldots,n \\
\lambda &= (\lambda_1, \lambda_2, \ldots, \lambda_n)' \\
\mathbf{m} &= m1
\end{aligned}
$$

the minimization leads to

$$
\begin{cases}
\mathrm{C}\lambda = \mathrm{c} + \mathrm{m} \\
\sum_{i=1}^{n} \lambda_i = 1
\end{cases}
$$

So

$$
1 = \mathbf{1}'\boldsymbol{\lambda} = \mathbf{1}'\mathbf{C}^{-1}(\mathbf{c} + \mathbf{m}) = \mathbf{1}'\mathbf{C}^{-1}\mathbf{c} + \mathbf{1}'\mathbf{C}^{-1}m\mathbf{1}
$$

or

$$
m = \frac{1 - 1'\mathrm{C}^{-1}\mathrm{c}}{\mathbf{1}'\mathrm{C}^{-1}\mathbf{1}}
$$

and

$$
\hat{\lambda} = \mathbf{C}^{-1}\left(\mathbf{c} + \frac{1 - \mathbf{1}'\mathbf{C}^{-1}\mathbf{c}}{\mathbf{1}'\mathbf{C}^{-1}\mathbf{1}}\mathbf{1}\right)
$$

The ordinary kriging predictor is then

$$
\hat{Z}(\mathrm{s}_0) = \hat{\lambda}'\mathrm{Z}
$$

The minimized mean-square prediction error, denoted by $\sigma_k^2(\mathrm{s}_0)$, is called the kriging variance and is given by

$$\sigma_k^2(\mathrm{s}_0) = \mathrm{E}\left(Z(\mathrm{s}_0) - \hat{Z}(\mathrm{s}_0)\right)^2 = C(0) - \hat{\lambda}^{'}\mathrm{c} + m$$

It follows that when $Z(\mathrm{s})$ is Gaussian,

$$\hat{Z}(\mathrm{s}_0) \pm 1.96\sigma_k(\mathrm{s}_0)$$

Produce a 95% confidence interval for the prediction $\hat{Z}$. It is important to note that this may not be the case for non-Gaussian fields

2.3 Empirical Bayesian Kriging (EBK)

EBK is two geostatistical models. The intrinsic random function kriging (IRFK), and simple kriging with external trend (also known as linear mixed model or LMM). Both processes use the same algorithms so they can be combined into one computational model:

$$z_i = y(s_i) + \epsilon_i, i = \overline{1 \dots K},$$

where z_i is the actual observed value at the location $s_i, y(s)$ is the Gaussian process at the location s, ϵ_i is the error and K is the number of data measurements (Krivoruchko and Gribov 2019). With simple kriging with external trend, $y(s)$ is a constant or can be derived by the covariates. Since real world data is not always Gaussian, transformation to Gaussian distribution can be used for the simple kriging portion of the model.

The goal of the algorithm is to estimate the spatial process error $y(s)$.

Most spatial data measured from natural processes is by some extent nonstationary. Ignoring this produces bias in the covariance parameters (Krivoruchko and Gribov 2019).

2.4 Empirical Bayesian Kriging Algorithm

1. Parameters and semivariogram of the spatial process Θ, are estimated from the data.
2. Using the spatial process Θ, new values are simulated at each data location K_{sim} times.
3. New parameters $\Theta_i, i = \overline{1 \ldots . K_{\text{sim}}}$, for the spatial process are estimated from the newly simulated data and a histogram is generated and can be used as an estimate of prior distribution.
4. The simulated $\Theta_i, i = 1 \ldots K_{\text{sim}}$, is now known as the empirical prior distribution. The model parameters can only take Θ_i values, that is $f(\Theta' \mid \mathbf{z}) = 0$ for $\Theta' \neq \Theta_{\text{i}}$.
5. Each simulated model is use to create a weight using Bayes Rule, $w_i \propto f(\mathbf{z} \mid \Theta_i), w_i = \frac{f(\mathbf{z}|\Theta_i)}{\sum_{i=1}^{K_{\text{sim}}} f(\mathbf{z}|\Theta_i)}, \sum_{i=1}^{K_{\text{s=m}}} w_i = 1$, where $f(\mathbf{z} \mid \Theta_i)$ is the conditional probability of the data $\mathbf{z}$ given the model parameters Θ_i.
6. The predictions and standard errors of the predictions are generated at the data locations by the following.

$$E[\, y(s) \mid \mathbf{z}\,] = \hat{y}(s) = \sum_{i=1}^{K_{\text{sim}}} (w_i \cdot E[y(s) \mid \mathbf{z}, \Theta_i]),$$
$$\text{Var}[\, y(s) \mid \mathbf{z}\,] = \text{E}[(y(s) - \hat{y}(s))^2 \mid \mathbf{z}] =,$$

$$\sum_{i=1}^{K_{sim}} \left(w_i \cdot (\mathrm{Var}\left[y(s) \mid \mathbf{z}, \Theta_i\right] + (\mathrm{E}[y(s) \mid \mathbf{z}, \Theta_i] - \hat{y}(s))^2)\right)$$

where $\mathrm{E}[y(s) \mid \mathbf{z}, \Theta_i]$ and $\mathrm{Var}\left[y(s) \mid \mathbf{z}, \Theta_i\right]$ are given by kriging equations (Krivoruchko and Gribov 2019).

EBK was chosen for this investigation because of its recent incorporation into ArcGIS, a popular tool for many Geographical Informational Science (GIS) students and professionals.

Chapter 3: Bayesian Transform Gaussian

3.1 Model Description

Bayesian approach to kriging leads to a prediction at the original scale called the predictive density function

$$\begin{aligned} p(\mathbf{z}_o \mid \mathbf{z}) &= \int_{\Omega} p(\mathbf{z}_o, \boldsymbol{\eta} \mid \mathbf{z}) d\boldsymbol{\eta} \\ &= \int_{\Omega} p(\mathbf{z}_o \mid \boldsymbol{\eta}, \mathbf{z}) p(\boldsymbol{\eta} \mid \mathbf{z}) d\boldsymbol{\eta} \end{aligned}$$

where $\mathbf{z}_o = (z_{o1}, \dots, z_{ok})'$ is the value to be predicted, and η is the vector of model parameters taking values in Ω, $\{Z(\mathbf{s}), \mathbf{s} \in D\}, D \subset R^2$, is a random field observed at n locations in $D, \mathbf{Z} = \big(Z(\mathbf{s}_1), \dots, Z(\mathbf{s}_n)\big)'$, with the goal to predict the unobserved random vector $\mathbf{Z}_0 = \big(Z(\mathbf{s}_{01}), \dots, Z(\mathbf{s}_{0k})\big)'$, at known distinct locations $\mathbf{s}_{01}, \dots, \mathbf{s}_{0k}$ in D. The extra generality of predicting a vector rather than a scalar is obtained without any additional computational effort (De Oliveira et al. 1997).

Let $G = \{g_\lambda(\cdot) : \lambda \in \Lambda\}$ be a parametric family of nonlinear monotone transformations where each $g_\lambda(\cdot) \in G$ has a continuous derivative $g'_\lambda(x)$. Assume that for some unknown $\lambda \in \Lambda$ the process

$$\{Y(\mathbf{s}) = g_\lambda(Z(\mathbf{s})), \mathbf{s} \in D\}$$

is a Gaussian random field. To match format with log-Gaussian random field, the original field $\{Z(\mathbf{s}), \mathbf{s} \in D\}$ is referred to as g_λ-Gaussian random field.

$$\mathrm{E}(Y(\mathrm{s})) = \sum_{j=1}^{p} \beta_j f_j(\mathrm{s}) = \beta' \underline{f}(\mathrm{s}), \mathrm{s} \in D$$

where $\beta = (\beta_1, \dots, \beta_p)' \in R^p$ are unknown regression parameters, and $f(\mathbf{s}) = (f_1(\mathbf{s}), \dots, f_p(\mathbf{s}))'$ is a set of known location-dependent covariates.

Variance in terms of precision:

$$\mathrm{Var}\,(Y(\mathbf{s})) = \tau^{-1}$$

where τ is the precision of the random field.

3.2 Parametric Correlation:

$$\mathrm{Cov}\,(Y(\mathbf{s}), Y(\mathbf{u})) = \frac{1}{\tau} K_{\boldsymbol{\theta}}(\| \mathbf{s} - \mathbf{u} \|),\ \mathbf{s}, \mathbf{u} \in D$$

The parameter $\theta = (\theta_1, \dots, \theta_q)' \in \Theta \subset R^q$ is a structural parameter controlling the range of correlation and/or the smoothness of the random field, and for every $\theta \in \Theta$, $K_\theta(\cdot)$ is an isotropic correlation function. Isotropy is not essential, unlike ordinary kriging and any $K_\theta(\cdot)$ will suffice.

For any vector $\mathbf{a} = (a_1, \dots, a_n)$ define $g_\lambda(\mathbf{a}) \equiv (g_\lambda(a_1), \dots, g_\lambda(a_n))$. Then the Gaussian assumption about $Y(\mathbf{s})$ implies,

$$\left(\underline{g_\lambda}(Z_0), \underline{g_\lambda}(Z) \mid \beta, \tau, \theta, \lambda\right)' \sim \mathcal{N}_{k+n}\left(\begin{pmatrix} \mathbf{X}_0\beta \\ \mathbf{X}\beta \end{pmatrix}, \frac{1}{\tau}\begin{pmatrix} \mathbf{E}_{\boldsymbol{\theta}} & \mathbf{B}_\theta \\ \mathbf{B}_\theta & \Sigma_\theta \end{pmatrix}\right)$$

for some $\lambda \in \Lambda$ and $(\beta, \tau, \theta)' \in R^p \times (0, \infty) \times \Theta$. The matrices $\mathbf{X}$ and $\mathbf{X}_0$ are known $n \times p$ and $k \times p$ design matrices, respectively, defined by $X_{ij} = f_j(\mathbf{s}_i), X_{0,ij} =$

$f_j(\mathrm{s}_{0i})$, and $\mathrm{E}_\theta, \mathrm{B}_\theta$ and Σ_θ are respectively $k \times k, k \times n$, and $n \times n$, correlation matrices defined as

$$\mathbf{E}_{\boldsymbol{\theta},ij} = K_\theta\left(\|\mathbf{s}_{0i} - \mathbf{s}_{0j}\|\right)$$
$$\mathbf{B}_{\theta,ij} = K_\theta\left(\|\mathbf{s}_{0i} - \mathbf{s}_j\|\right)$$
$$\boldsymbol{\Sigma}_{\theta,ij} = K_\theta\left(\|\mathbf{s}_i - \mathbf{s}_j\|\right)$$

It is assumed that X has full rank and that the matrix $\Sigma_\theta, \theta \in \Theta$, is nonsingular.

The likelihood $L(\eta; \mathbf{z}) \equiv p(\mathbf{z} \mid \eta)$ of the model parameters $\eta = (\beta, \tau, \theta, \lambda)'$ based on the original data $\mathbf{z} = (z_1, \ldots, z_n)'$ is given by

$$L(\boldsymbol{\eta}; \mathbf{z}) = \left(\frac{\tau}{2\pi}\right)^{n/2} |\boldsymbol{\Sigma}_{\boldsymbol{\theta}}|^{-1/2}$$

$$\times \exp\left\{-\frac{\tau}{2}\left(\underline{g}_\lambda(\mathbf{z}) - \mathbf{X}\boldsymbol{\beta}\right)' \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1}\left(\underline{g}_\lambda(\mathbf{z}) - \mathbf{X}\boldsymbol{\beta}\right)\right\} J_\lambda$$

for $z_i \in g_\lambda^{-1}(R)$, where

$$J_\lambda = \prod_{i=1}^n |g_\lambda'(z_i)|$$

is the Jacobian of the transformation (De Oliveira et al. 1997).

The Bayesian paradigm calls for the update of the prior distribution by means of Bayes theorem taking into account the likelihood of the observed data.

$$p(\boldsymbol{\beta}, \tau, \boldsymbol{\theta}, \lambda) \propto \frac{p(\boldsymbol{\theta})p(\lambda)}{\tau J_\lambda^{p/n}}$$

where $\boldsymbol{p}(\boldsymbol{\theta})$ and $\boldsymbol{p}(\boldsymbol{\lambda})$ are the prior marginals of $\boldsymbol{\theta}$ and $\boldsymbol{\lambda}$, respectively.

The joint posterior distribution of the model parameters given the data is obtained from its factors,

$$p(\boldsymbol{\eta} \mid \mathbf{z}) = p(\boldsymbol{\beta}, \tau, \boldsymbol{\theta}, \lambda \mid \mathbf{z}) = p(\boldsymbol{\beta}, \tau \mid \boldsymbol{\theta}, \lambda, \mathbf{z}) p(\boldsymbol{\theta}, \lambda \mid \mathbf{z}).$$

A compact $\Theta \times \Lambda$ is a sufficient condition for this posterior to be proper.

From the fact that conditional on θ and λ, $\underline{g}_{\lambda}(\mathbf{z})$ is a linear model in terms of the transformed data

$$(\boldsymbol{\beta} \mid \tau, \boldsymbol{\theta}, \lambda, \mathbf{z}) \sim \mathcal{N}_p \left(\hat{\boldsymbol{\beta}}_{\boldsymbol{\theta},\lambda}, \frac{1}{\tau} (\mathbf{X}' \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} \mathbf{X})^{-1} \right)$$

$$(\tau \mid \boldsymbol{\theta}, \lambda, \mathbf{z}) \sim Ga \left(\frac{n-p}{2}, \frac{2}{\tilde{q}_{\boldsymbol{\theta},\lambda}} \right)$$

where

$$\hat{\boldsymbol{\beta}}_{\boldsymbol{\theta},\lambda} = (\mathbf{X}' \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} \mathbf{X})^{-1} \mathbf{X}' \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} \underline{g}_{\lambda}(\mathbf{z})$$

is the weighted least squares estimate of parameter β based on the transformed data (De Oliveira et al. 1997).

Prediction of an Unknown Point Z_0

With the joint posterior distribution $p(\eta \mid z)$ from the last section, the predictive density $p(\mathbf{z}_o \mid \mathbf{z})$ requires $p(\mathbf{z}_o \mid \boldsymbol{\eta}, \mathbf{z})$. The latter can be derived from the parametric correlation.

$$\left(\underline{g}_{\lambda}(\mathbf{Z}_0) \mid \boldsymbol{\beta}, \tau, \boldsymbol{\theta}, \lambda, \mathbf{z} \right) \sim \mathcal{N}_k \left(\mathbf{M}_{\boldsymbol{\beta},\boldsymbol{\theta},\lambda}, \frac{1}{\tau} \mathbf{D}_{\boldsymbol{\theta}} \right)$$

where

$$\mathbf{M}_{\boldsymbol{\beta},\boldsymbol{\theta},\lambda} = \mathbf{B}_{\boldsymbol{\theta}} \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} \underline{g}_{\lambda}(\mathrm{z}) + \mathbf{H}_{\boldsymbol{\theta}} \boldsymbol{\beta}$$

$$\mathbf{H}_{\boldsymbol{\theta}} = \mathbf{X}_0 - \mathbf{B}_{\boldsymbol{\theta}} \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} \mathbf{X}$$

$$\mathrm{D}_{\boldsymbol{\theta}} = \mathrm{E}_{\boldsymbol{\theta}} - \mathrm{B}_{\boldsymbol{\theta}} \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} \mathbf{B}_{\boldsymbol{\theta}}'.$$

So,

$$p(\mathbf{z}_o \mid \boldsymbol{\eta}, \mathbf{z}) = \left(\frac{\tau}{2\pi}\right)^{k/2} |\mathbf{D}_{\boldsymbol{\theta}}|^{-1/2} \prod_{j=1}^{k} \left|g'_{\lambda}(z_{oj})\right|$$
$$\times \exp\left\{-\frac{\tau}{2}\left(\underline{g}_{\lambda}(\mathbf{z}_o) - \mathbf{M}_{\boldsymbol{\beta},\boldsymbol{\theta},\lambda}\right)' \mathbf{D}_{\boldsymbol{\theta}}^{-1}\left(\underline{g}_{\lambda}(\mathbf{z}) - \mathbf{M}_{\boldsymbol{\beta},\boldsymbol{\theta},\lambda}\right)\right\}$$

Now the integral of $p(\mathbf{z}_o \mid \boldsymbol{\eta}, \mathbf{z})p(\boldsymbol{\eta} \mid \mathbf{z})$ needed for $p(\mathbf{z}_o \mid \mathbf{z})$. By integrating out β and τ in the simplified form is obtained:

$$\begin{aligned} p(\mathbf{z}_o \mid \mathbf{z}) &= \int_{\Lambda}\int_{\Theta} p(\mathbf{z}_o \mid \boldsymbol{\theta}, \lambda, \mathbf{z}) p(\boldsymbol{\theta}, \lambda \mid \mathbf{z}) d\boldsymbol{\theta} d\lambda \\ &= \frac{\int_{\Lambda}\int_{\Theta} p(\mathbf{z}_o \mid \boldsymbol{\theta}, \lambda, \mathbf{z}) p(\mathbf{z} \mid \boldsymbol{\theta}, \lambda) p(\boldsymbol{\theta}) p(\lambda) d\boldsymbol{\theta} d\lambda}{\int_{\Lambda}\int_{\Theta} p(\mathbf{z} \mid \boldsymbol{\theta}, \lambda) p(\boldsymbol{\theta}) p(\lambda) d\boldsymbol{\theta} d\lambda} \end{aligned}$$

Now,

$$\left(\underline{g}_{\lambda}(\mathbf{Z}_0) \mid \boldsymbol{\theta}, \lambda, \mathbf{z}\right) \sim T_k\left(n - p, \mathbf{m}_{\boldsymbol{\theta},\lambda}, \left(\tilde{q}\boldsymbol{\theta}_{,\lambda}\mathbf{C}_{\boldsymbol{\theta}}\right)^{-1}\right)$$

is a k-variate Student t-distribution with $n - p$ degrees of freedom (De Oliveira et al. 1997).

Applying BTG

The btg program is a software implementation of the BTG model. In its current state, the program allows the user to:

1. Predict $Z(\mathbf{s}_0)$ and its uncertainty.
2. Compute the 95% prediction intervals based on the standard errors of the above uncertainty.
3. Cross-validate the predictions of $Z(\mathbf{s}_0)$.

The btg program has some limitations. The range of Z_0 must be positive since the Box-Cox transformation functions are positive real numbers. Due to issues in compiling with the original code produce by De Oliveria the program no longer produces a 3D

grid map of the data but does allow for prediction and cross-validation. The code is also computationally expensive so around maximum of 50 data points can be used for the cross-validation analysis (De Oliveira et al. 1997).

Chapter 4: Comparison

4.1 Atmospheric CO_2

Atmospheric Carbon Dioxide (XCO_2) is one of the primary greenhouse gases on Earth. Since industrialization, XCO_2 (measured in parts per million) has increased by over a third from the highest previous concentration of 300ppm over the past 800,000 years (Lüthi, et al., 2008). It has been well documented that the sudden increase of XCO_2 over the past 150 years has decreased the amount of thermal energy that radiates back into space causing a consistent warming trend that has caused rapid changes in the Earth's climate. The Earth naturally maintains a balance of sources (emitters of XCO_2) and sinks (areas that remove XCO_2) known as the carbon cycle (NOAA 2021). Because of the exponential increase of XCO_2 over the past 150 years, the natural carbon sinks are not enough to keep the XCO_2 in balance. Figure 4 gives contexts to this increased at a fixed spatial point at Mauna Lao Observatory.

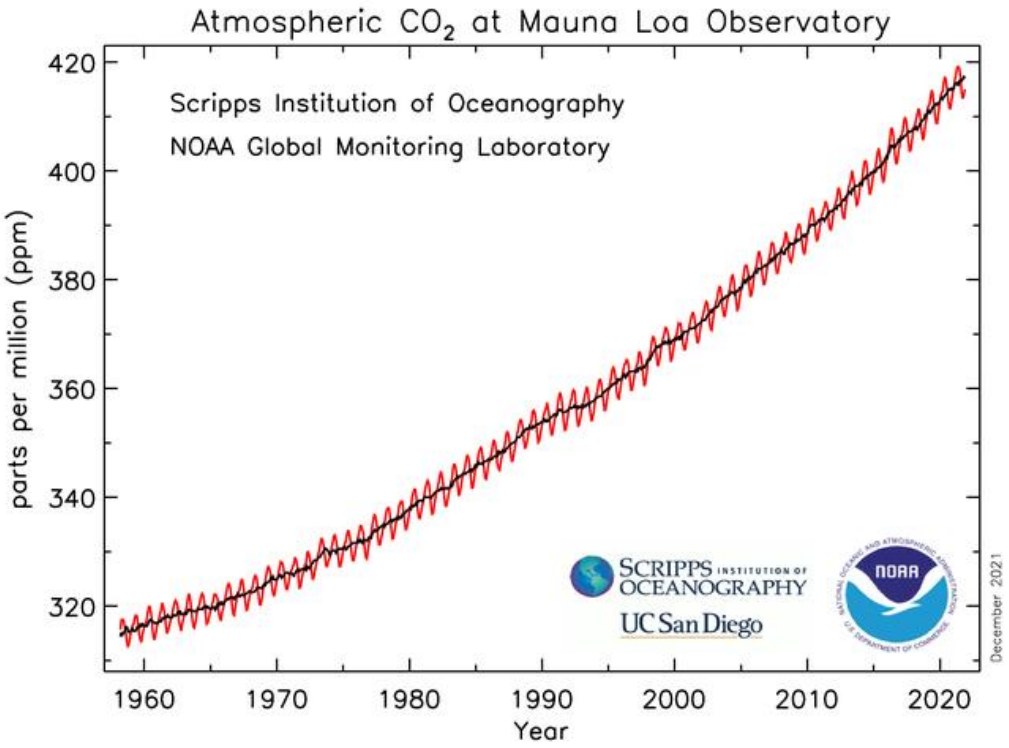

Figure 4. Atmospheric CO_2 Measured At Mauna Loa Observatory (Source: NOAA)

4.2 OCO-2 Application

In order to mitigate the changing climate, there must be 1.) a reduction of the burning of fossil fuels that is contributing to the continued rise of XCO_2 and 2.) an expansion of carbon sinks to remove the XCO_2 already in the atmosphere.

Carbon sinks are areas that absorb more carbon than they release. This might include land-based sinks (such as forests, grasslands, soil) or ocean-based sinks (such as mangrove forests, seagrass, salt marshes, and ocean absorbed CO_2). Via photosynthesis and geologic processes, these sinks slowly remove the atmospheric CO_2. As countries and corporations tout "carbon neutral" activities that often come up short in their reporting consistent and remote measurements of XCO_2 is important to

understand the location of both sources and sinks. In 2014, NASA launched the Orbiting Carbon Observatory-2 (OCO-2) to fulfill this task. OCO-2 carries 3 spectrometers that collects high-resolution spectra from reflected sunlight that passes through oxygen molecules in O_2 and CO_2. Measurements from the three different spectrometers are combined into a sounding and then run through a physical retrieval algorithm to estimate the XCO_2. The measurements of XCO_2 are as a column averaged to produce the measurement in column-averaged dry mole fraction in ppm. OCO-2 is a polar orbiting satellites it hits the same geolocation every 16 days. It should be noted that since the retrievals are based on infrared radiance, clouds obscure measurements of XCO_2 limiting the total coverage data (NASA 2020).

OCO-3 builds on the OCO-2 mission while adding measurements of Solar Induced Florescence (SIF). The OCO-3 instrument was added to the International Space Station (ISS) in 2019. The orbit of the ISS allows for viewing between latitudes less than 52 degrees. The instrument is essentially the same spectrometer of the OCO-2 but the configuration has a pointing mechanism that allows rapid transition of the viewing angle to allow for more focused, targeted sampling locations. This also allows for significant overlapping data with OCO-2 (NASA 2020).

Data retrievals are from both instruments are provided by NASA's Jet Propulsion Laboratory and can be accessed through NASA Goddard DAAC user interface (NASA 2020). Additionally, JPL has provided users with the CO_2 Virtual Science Data Environment that provides quick access to data from any of the NASA products focusing on satellite observations of carbon dioxide.

Because the many variables such as cloud and sunlight reflection, the current data products for both OCO-2 and OCO-3 are presented in point data over the measured areas (Figure 5). This presents a unique opportunity to use spatial interpolation to fill in the gaps between the point data. The comparison data was created from a satellite pass over Los Angeles in September 2019. The map of the area is below:

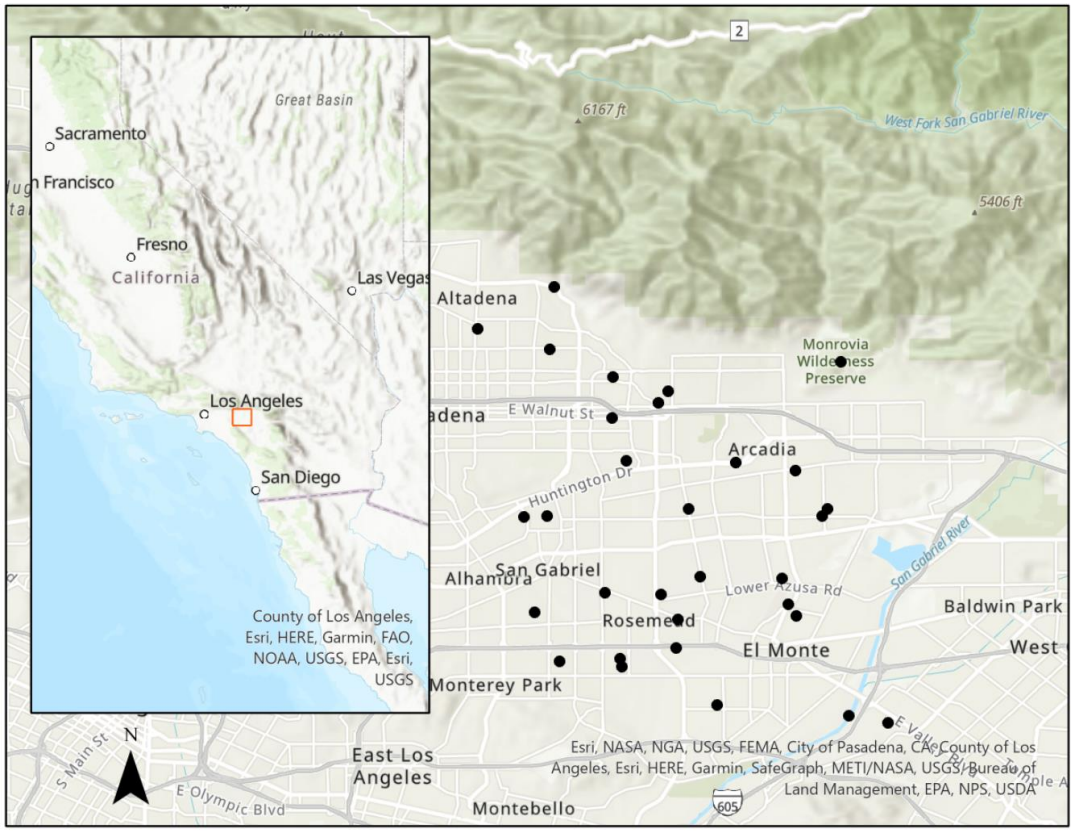

Figure 5. Atmospheric CO_2 Measured At Mauna Loa Observatory (Source: NOAA)

With the resulting histogram and interpolation of the data points.

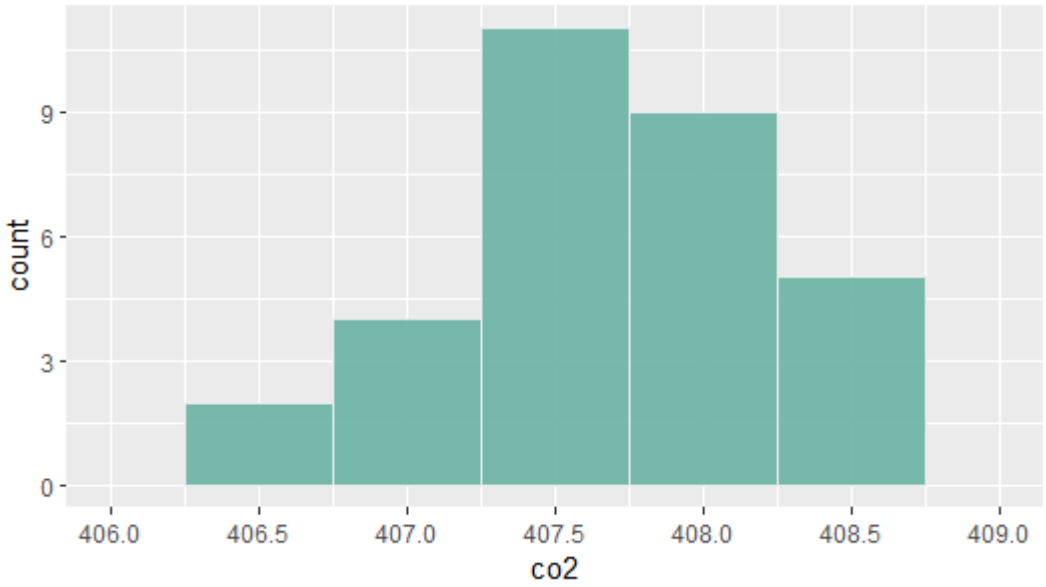

Figure 6. Histogram of sampled data points

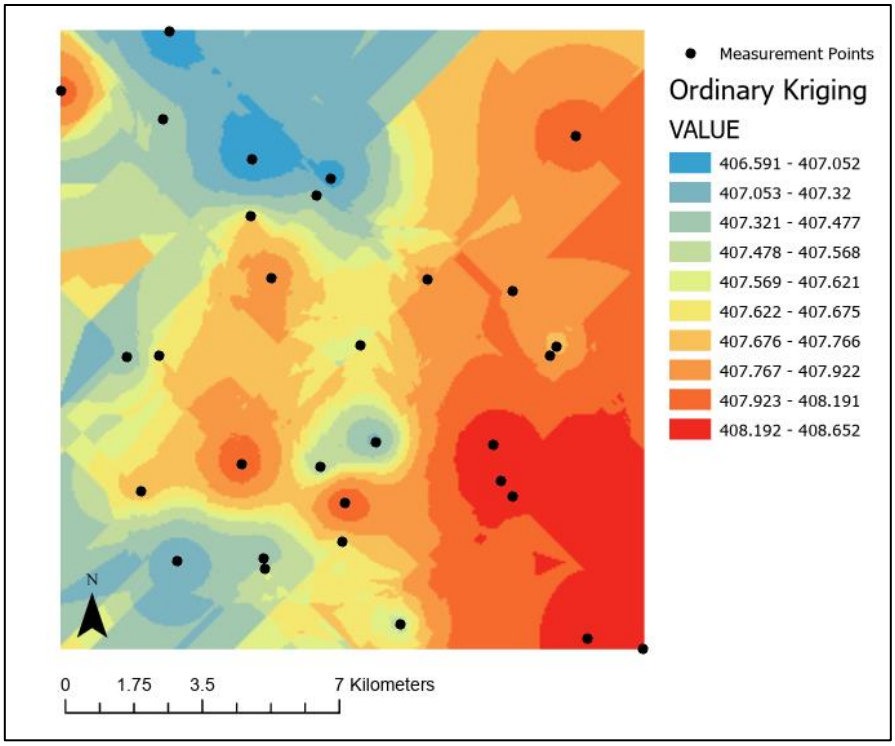

Figure 7. Contour of ordinary kriging interpolation

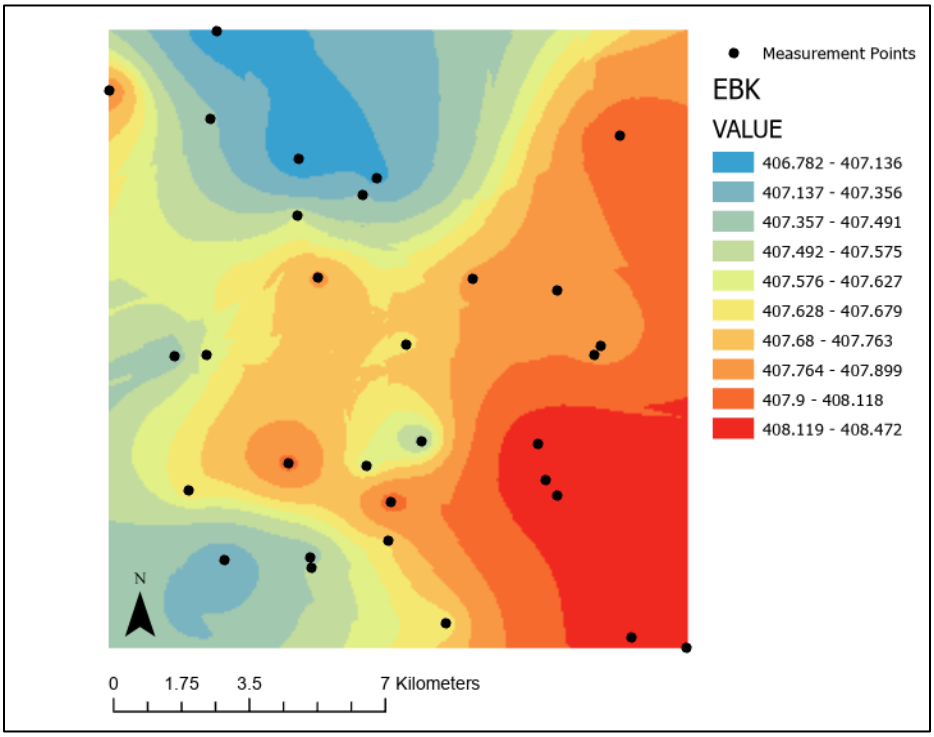

Figure 8. Contour of empirical Bayesian kriging interpolation

Measured	Predicted OK	StdError OK	Predicted EBK	StdError EBK	Predicted BTG	StdError BTG
406.65802	407.4307421	0.549530785	407.5294681	0.568853736	407.805	1.311208206
406.75742	407.5037859	0.428242383	407.4448046	0.45421862	407.806	1.09382314
406.96472	407.7345647	0.543879834	407.6367422	0.516746755	407.855	0.800523878
407.15866	407.9502608	0.505167499	407.9326588	0.481243213	407.789	0.39614436
407.21277	407.6256519	0.352134967	407.6109734	0.419481043	407.788	0.333900221
407.32413	407.5849573	0.474026431	407.6494244	0.483870772	407.784	0.215557776
407.4303	407.845229	0.471369439	407.8169504	0.460542996	407.777	0.142283612
407.45914	407.3112458	0.533317208	407.26909	0.510219794	407.781	0.122912647
407.46332	407.0420493	0.414767007	407.1819925	0.453263271	407.779	0.101986977
407.48535	407.9830875	0.56227345	407.9210368	0.545520886	407.779	0.102048303
407.59534	407.8936771	0.476950926	407.8063752	0.468959846	407.779	0.083246099
407.5971	407.5826827	0.526422672	407.7607858	0.503889231	407.775	0.030537912
407.61017	407.8971315	0.364237317	407.9413369	0.431665907	407.775	0.030635851
407.61526	407.3056452	0.355724387	407.3758586	0.435417841	407.775	0.02737635
407.68265	407.4825397	0.46857465	407.5001931	0.471000762	407.792	0.012595124
407.71698	407.365805	0.490298955	407.3522444	0.476813365	407.779	0.003433866
407.76596	407.448425	0.539608057	407.4760367	0.523352576	407.769	9.99934E-07
407.84314	407.7551363	0.364737216	407.8369556	0.433032219	407.744	0.009164769
407.85538	407.6754396	0.529433787	407.6990739	0.500531395	407.766	0.008870626
407.91187	407.8279609	0.527383142	407.8238445	0.498111896	407.764	0.021278057
407.95682	407.5490166	0.513280196	407.5482036	0.498096788	407.753	0.042645967
408.19055	407.2399721	0.572713261	407.1600921	0.573375036	407.754	0.189803121
408.20923	407.4758584	0.516911601	407.5377465	0.491936135	407.752	0.209557951
408.24356	407.4995083	0.585171439	407.5048039	0.647401496	407.756	0.234341192
408.2459	408.4308398	0.418060191	408.3521213	0.468838039	407.752	0.247556003
408.25754	407.5660832	0.448339296	407.6002789	0.455380614	407.659	0.361785408
408.44186	408.1601164	0.529178499	408.1821352	0.521878223	407.746	0.481190555
408.4735	408.2287481	0.541359672	408.2297138	0.552051148	407.745	0.525188641
408.63864	408.2901479	0.390742074	408.2078818	0.444214546	407.737	0.815022562
408.7057	408.1700996	0.483401278	408.0607319	0.477314908	407.737	0.946487711

Table 1. Table of cross-validation results

Data from each of the three interpolation methods ordinary kriging (OK), Empircal Bayesian kriging (EBK), and Bayesian Transform Guassian (BTG) in ppm over the LA region study area.

Method	Mean Squared Error (MSE)
Ordinary Kriging	0.2557928198
Empirical Bayesian Kriging	0.2498231109
BTG	0.319744823

Table 2. MSE Comparisons

Example of validation exercise and raster maps creation using EBK default model. The following python code is used to process larger amounts of data and compare between the ordinary kriging and Empirical Bayesian Kriging. This dataset is over the full month of August 2019 in Northern California.

```
1.  import arcpy
2.  import os
3.  import time
4.
5.  workspace = os.getcwd()
6.  arcpy.env.workspace = workspace
7.
8.  start = time.time()
9.
10. # Process: Empirical Bayesian Kriging
11. arcpy.EmpiricalBayesianKriging_ga("California CO2.shp", z_field="C)2",
    out_ga_layer="GALayer", out_raster="", cell_size="6431.3811596",
    transformation_type="NONE", max_local_points="100", overlap_factor="1",
    number_semivariograms="100", search_neighborhood="NBRTYPE=StandardCircular
    RADIUS=602402.270130742 ANGLE=0 NBR_MAX=15 NBR_MIN=10 SECTOR_TYPE=ONE_SECTOR",
    output_type="PREDICTION", quantile_value="0.5", threshold_type="EXCEED",
    probability_threshold="", semivariogram_model_type="POWER")
12.
13. # Process: GA Layer To Rasters
14. tempEnvironment0 = arcpy.env.mask
15. arcpy.env.mask = " California CO2.shp shp"
16. arcpy.GALayerToRasters_ga(in_geostat_layer="GALayer",
    out_raster="ResPrediction", output_type="PREDICTION",
    quantile_probability_value="", cell_size="5000", points_per_block_horz="1",
    points_per_block_vert="1", additional_rasters="'ResStandErr'
    PREDICTION_STANDARD_ERROR #", out_elevation="")
17. arcpy.env.mask = tempEnvironment0
18.
19. # Process: Copy Raster
```

```
20. arcpy.CopyRaster_management(in_raster="ResPrediction",
    out_rasterdataset="Prediction.tif", config_keyword="", background_value="",
    nodata_value="-3.402823e+38", onebit_to_eightbit="NONE", colormap_to_RGB="NONE",
    pixel_type="", scale_pixel_value="NONE", RGB_to_Colormap="NONE", format="TIFF",
    transform="NONE", process_as_multidimensional="CURRENT_SLICE",
    build_multidimensional_transpose="NO_TRANSPOSE")
21.
22. # Process: Copy Raster
23. arcpy.CopyRaster_management(in_raster="ResStandErr",
    out_rasterdataset="StandardError.tif", config_keyword="", background_value="",
    nodata_value="-3.402823e+38", onebit_to_eightbit="NONE", colormap_to_RGB="NONE",
    pixel_type="", scale_pixel_value="NONE", RGB_to_Colormap="NONE", format="TIFF",
    transform="NONE", process_as_multidimensional="CURRENT_SLICE",
    build_multidimensional_transpose="NO_TRANSPOSE")
24.
25. end = time.time()
26.
27. print(datetime.timedelta(seconds = round(end - start)))
28.
```

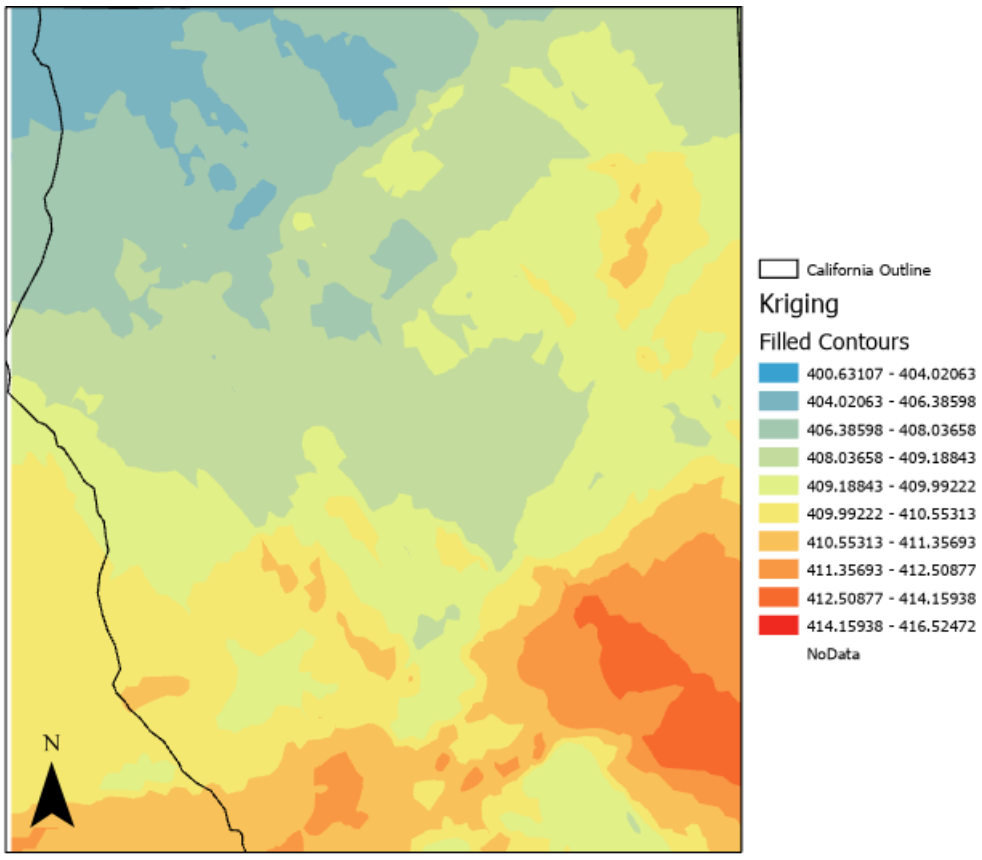

Figure 9. Contour of ordinary kriging interpolation of Northern California

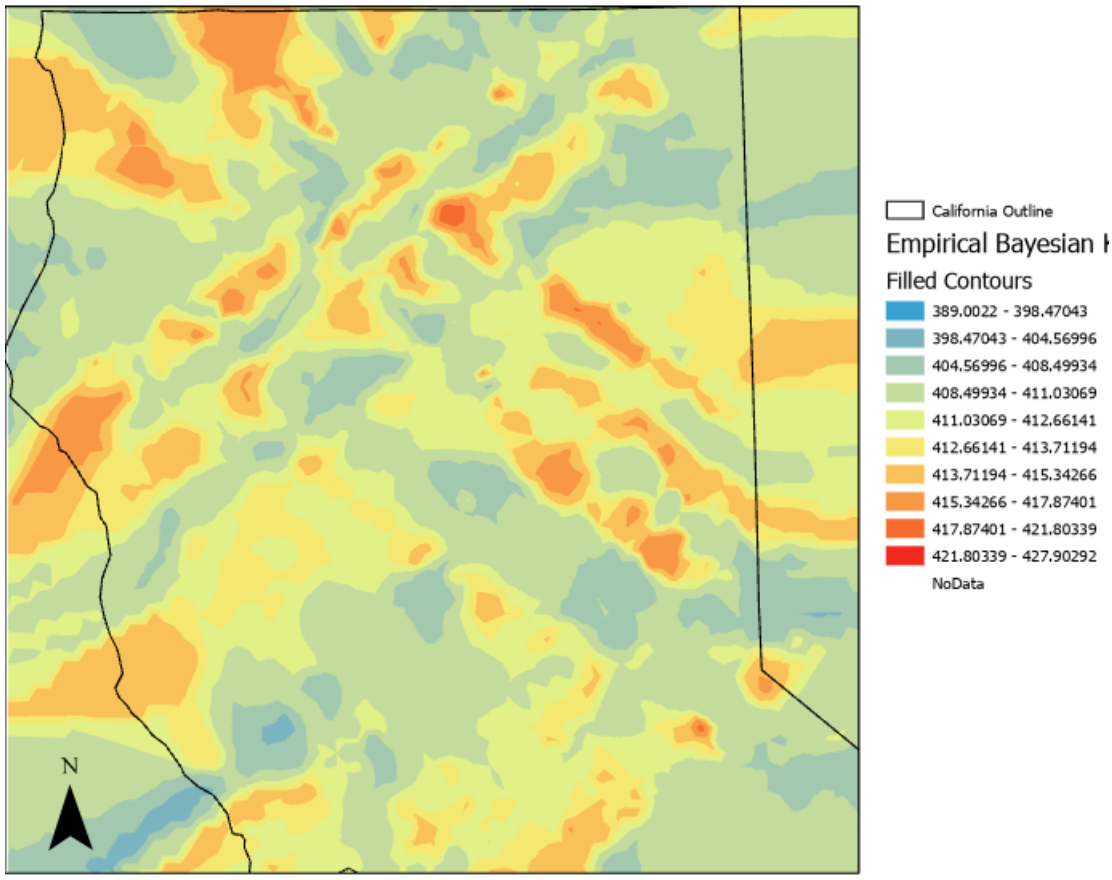

Figure 9. Contour of EBK interpolation of Northern California

Chapter 5: Conclusion

The Gaussian assumption for the spatial data involves covariance parameters and transformation parameter if the data is not Gaussian.On the other hand, the BTG and EBK method do not require the specification of parameters, since their estimation is integrated into the method. EBK uses simulated data to estimate the parameters. BTG only falls slightly behind the other methods as far as MSE

BTG is useful in situations when there is a large number of unknown parameters and a small number of data points. The BTG algorithm, which uses Monte-Carlo method for integration, is its computational complex. An update of the BTG code would be an interesting addition for future investigation.

EBK is useful when there is a large amount of datapoints over a larger spatial field as the subsetting of simulated data allows for non-stationality to be incorporated into the predicted data. For further studies, it would be interesting to compare BTG and EBK on environmental data that is not Gaussian.

Bibliography

Bindel, D., De Oliveira, V., Kedem, B. (1997), An Implementation of the Bayesian Transformed Gaussian Spatial Prediction Model

Daya Sagar, B. S., Cheng, Q., & Agterberg, F. (Eds.). (2018). *Handbook of Mathematical Geosciences: Fifty Years of IAMG*. Springer International Publishing. https://doi.org/10.1007/978-3-319-78999-6

De Oliveira, V., Kedem, B., and Short, D. (1997), Bayesian Prediction of Transformed Gaussian Random Fields," *Journal of the American Statistical Association*, 92, 440, 1422-1433.

Kostopoulou, E. (2021). Applicability of ordinary Kriging modeling techniques for filling satellite data gaps in support of coastal management. *Modeling Earth Systems and Environment*, *7*(2), 1145–1158. https://doi.org/10.1007/s40808-020-00940-5

Krivoruchko, K., & Gribov, A. (2019). Evaluation of empirical Bayesian kriging. *Spatial Statistics*, *32*, 100368. https://doi.org/10.1016/j.spasta.2019.100368

Liu, Y., Wang, X., Guo, M., & Tani, H. (2012). Mapping the FTS SWIR L2 product of XCO_2 and XCH_4 data from the GOSAT by the Kriging method – a case study in East Asia. *International Journal of Remote Sensing*, *33*(10), 3004–3025. https://doi.org/10.1080/01431161.2011.624132

National Aeronautics and Space Administration (2020). Orbiting Carbon Observatory -2 & 3. *Data Product User's Guide, Operational Level 2*.

Siabi, Z., Falahatkar, S., & Alavi, S. J. (2019). Spatial distribution of XCO2 using OCO-2 data in growing seasons. *Journal of Environmental Management*, *244*, 110–118. https://doi.org/10.1016/j.jenvman.2019.05.049

US Department of Commerce, N. (n.d.). *Global Monitoring Laboratory—Carbon Cycle Greenhouse Gases*. Retrieved November 19, 2021, from https://gml.noaa.gov/ccgg/trends/

Wu, Y., & Chan, C. W. (2011). Analysis of data for the carbon dioxide capture domain. *Engineering Applications of Artificial Intelligence*, *24*(1), 154–163. https://doi.org/10.1016/j.engappai.2010.09.002