Ataskaita už 2022 doktorantūros metus

Doktorantė: Neringa Urbonaitė
Vadovas: Prof. habil. dr. Leonidas Sakalauskas

Vilniaus Universitetas
Duomenų mokslo ir skaitmeninių technologijų institutas

2022 kovo 24 d.

Disertacijos tema

Disertacijos tema:

Fraktalinio Brauno Lauko tyrimas ir taikymas daugiamačių duomenų modeliavime

Vadovas:

Prof. habil. dr. Leonidas Sakalauskas

Pradžios pabaigos metai:

2019-2023

Disertacijos rengimo planas

Studijų metai	Egzaminai		Dalyvavimas konferencijose		Publikacijos		
	Planas	Įvykdyta	Planas	Įvykdyta	Planas	Įvykdyta	Būklė [4]
I (2019/2020)	2	2	1	1	-	-	-
II (2020/2021)	2	2	1	1	1	1	Įteikta
III (2021/2022)	**-**	**-**	**1**	**2**	**0**	**2**	**Išsiųsta**
IV (2022/2023)	-	-	-	-	1	-	-
Iš viso:	4	4	2	4	2	1	-

Disertacijos rengimo planas

Einamieji studijų metai (III: 2021/2022).

Dalyvavimas konferencijose		
Planas	**Įvykdyta**	**Konferencijos tipas**
Joint ICTP-IUGG Workshop on Data Assimilation and Inverse Problems in Geophysical Sciences. Trieste, Italija. 18-29 Spalis, 2021.	Neringa Urbonaitė, Leonidas Sakalauskas *Investigation of Fractional Brownian Vector Fields.*	Tarptautinė
International Symposium on Applied Geoinformatics ISAG2021. Ryga, Latvija. 2-3 Gruodis, 2021.	Neringa Urbonaitė, Leonidas Sakalauskas *Multivariate Data Modeling: Methods and Challenges.*	Tarptautinė

Disertacijos rengimo planas

Publikacijos

Publikacijos			
Planas	**Įvykdyta**	**Būklė**	**Publikacijos tipas**
Applied Mathematics and Computation	Neringa Urbonaite, Leonidas Sakalauskas; A fractional kriging method for multicriteria data modeling	išsiusta	turi cituojamumo rodiklį
Baltic Journal of Modern Computing (BJMC)	Neringa Urbonaite, Leonidas Sakalauskas; Advances in Spatial Data Modeling: A Review	išsiusta	be cituojamumo rodiklio. CA

Mokslinių tyrimų ir disertacijos rengimo etapai

	Darbo pavadinimas	Atlikimo terminai	Pastabos
1.	Mokslinių tyrimų disertacijos tema apžvalga ir analizė (Lietuvoje ir užsienyje): 1. Atlikti daugiamačių Krigingo modelių apžvalgą. 2. Nustatyti (identifikuoti) mokslines problemas, kylančias uždaviniuose.	2019 m. spalio mėn. – 2020 m. rugsėjo mėn.	~18psl.
2.	Mokslinio tyrimo vykdymas: 2.1. Tyrimo metodikos sudarymas; 2.1.1Tyrimo metodikos iškeltam uždaviniui spręsti parinkimas; 2.1.2 Teorinio ir empirinio tyrimų suplanavimas pagal pasirinktą metodiką. 2.2. Teorinis tyrimas: 2.2.1 Fraktalinių Brauno laukų (FBL) savybių tyrimas; 2.2.2 FBL realizacijų kompiuterinio imitavimo metodo sukūrimas. 2.3. Empirinis tyrimas: 2.3.1 FBL pritaikymas daugiamačiam krigingui ir ekstrepoliavimui; 2.3.2 Atstikitinių laukų parametrų vertinimo metodų: semivariogramos ir didžiausio tikėtinumo algoritmų sukūrimas.	2020 m. spalio mėn. – 2021 m. balandžio mėn. 2020 m. gegužės mėn. – 2021 m. rugsėjo mėn. 2021 m. lapkričio mėn. – 2022 m. rugsėjo mėn.	~17psl. ~23-24psl.

Disertacijos uždaviniai

Tyrimo objektas: Fraktalinis vektorinis Brauno laukas

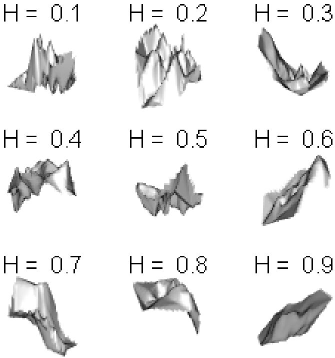
Tikslas: Skirtų daugiamačiams duomenims ir jų sąsajoms vertinti, generuoti, prognozuoti, taikyti, algoritmų kūrimas, remiantis Fraktalinio vektorinio Brauno lauko modeliu, kai yra stebėjimų trūkumas.

Uždaviniai:

1. Sukurti FVBL'o modelį;
2. Sudaryti FVBL'o realizacijų generavimo algoritmą;
3. Sudaryti FVBL'o vertinimo algoritmą taikant didžiausio tikėtinumo metodą (DT) ir palyginti jį su variogramos (V) metodu;
4. Sudaryti metodą daugiamačių duomenų ekstrepoliavimui taikant FVBL'o modelį;
5. Pritaikyti sukurtą metodą praktiniams uždaviniams.

Choleckio metodas

Examples of fractional Brownian field surfaces were generated by applying the FBvf simulation method and choosing different values of the Hurst parameter, e.g. $H = 0.1$, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, or 0.9, as depicted in Pav 1.



pav. 1: Surfaces of FBvf and their dependence on the Hurst parameter.

Fraktalinis vektorinis Brauno laukas

Definition

An FBvf with a Hurst parameter, $H \in [0,1]$, is a spatial process based on the Gaussian distribution and distances between the points of observation,
.

$Z = (z^1, z^2, ..., z^K)$, $z^i = z(x^i)$, $1 \leqslant i, j \leqslant K$, $A = [A_{i,j}]_{i,j=1}^K$ and calculated between X as a set of vector pairs:

$$A = ((x^i - x^j) \cdot (x^i - x^j))^H. \tag{1}$$

Now, consider the column vector of ones, $\mathbf{1}_i = 1$,
$a = ((x^i - x^0) \cdot (x^i - x^0))^H$:

$$F = \mathbf{1}a^\top + a\mathbf{1}^\top - A \tag{2}$$

FvBI generavimas

In this work, we aim to indicate the importance and benefits of distance matrix A. Hence, we present expressions of recursive algorithm. To obtain values Z^1 and Z^2 the algorithm starts with indicated x^1 and x^2 .

$$Z^1 = \beta_{chol} \cdot \zeta^1 \cdot 2^{0.5-H} \cdot ((x^1 - x^2)^T \cdot (x^1 - x^2))^{\frac{H}{2}},$$

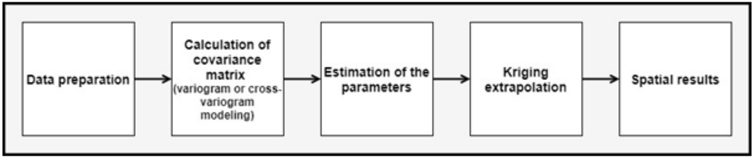
$$Z^2 = Z^1 \cdot (1 - 2^{2H-1}) + \beta_{chol} \cdot \zeta^2 \cdot \sqrt{1 - (2^{0.5-H} - 2^{H-0.5})^2}$$

$$\cdot 2^{1-2H} \cdot ((x^1 - x^2)^T \cdot (x^1 - x^2))^{\frac{H}{2}}.$$

Generalized expression of the recursive algorithm is the following and properties of kernel matrix:

$$Z^i = \hat{z}_A(x^i) + \beta_{chol} \cdot \zeta^i \cdot \beta_A(x^i). \tag{3}$$

Krigingas



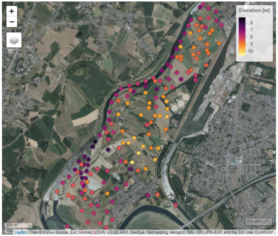
Suppose that the set of FBvf values $Z = (z^1, z^2, ..., z^K)$, $z^i \in \mathfrak{R}^m$, $m \geqslant 1$ is measured at the mutually distinctive points $X = (x^1, x^2, ..., x^K)$, $x^i \in \mathfrak{R}^d$, $d \geqslant 1$, $1 \leqslant i \leqslant K$, $b = ((x^i - x) \cdot (x^i - x))^H$. Then, predictions via kriging are obtained:

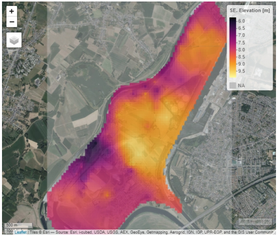
$$\hat{z}_F(x) = Z^\top \cdot F^{-1} \cdot (f + \mathbf{1} \cdot \frac{1 - \mathbf{1}^\top \cdot F^{-1} \cdot f}{\mathbf{1}^\top \cdot F^{-1} \cdot \mathbf{1}}), \tag{4}$$

$$\hat{z}_A(x) = Z^\top \cdot A^{-1} \cdot (b + \mathbf{1} \cdot \frac{1 - \mathbf{1}^\top \cdot A^{-1} \cdot b}{\mathbf{1}^\top \cdot A^{-1} \cdot \mathbf{1}}), \tag{5}$$

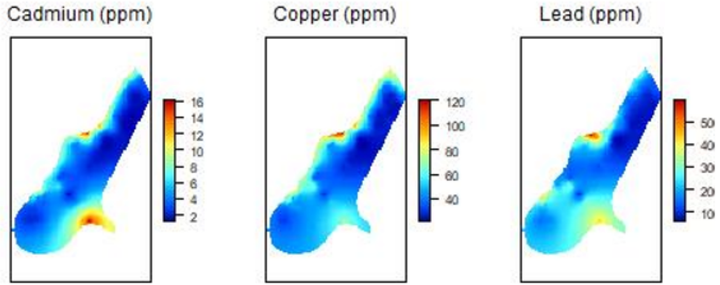
Interpoliacija

- Process of predicting values at unknown locations using values at known location;
- Transforms measurements of a continuous phenomenon into continuous surface;
- Interpolation predicts within region; Extrapolation predicts outside region;





Interpoliacija

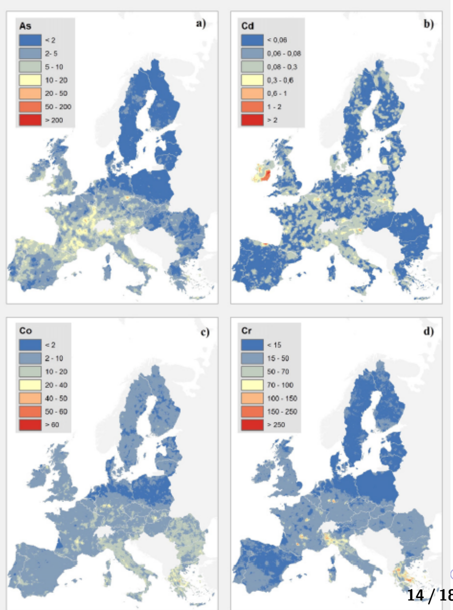


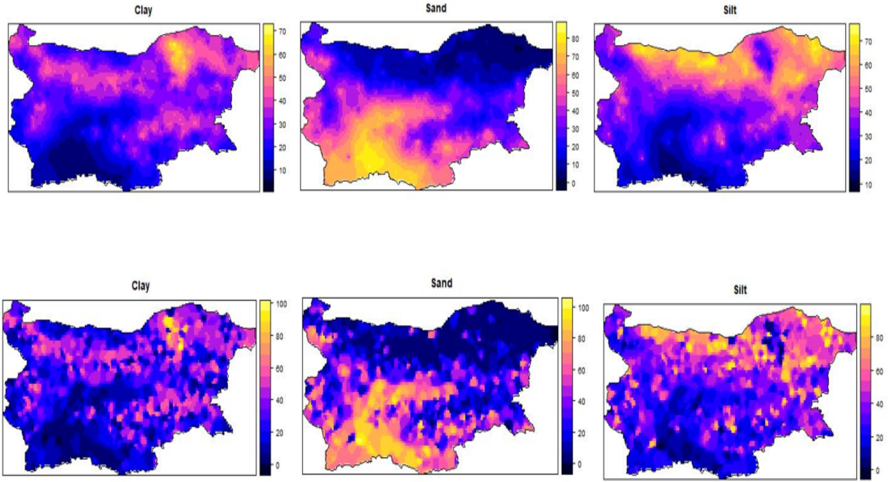
Lentelė 1: Summary statistics of the results of extrapolation.

	Cd	Cu	Pb	Zn	Elevation
$RMSE_{FK}$	4.14	20.23	134.13	399.260	1.47
$RMSE_{SE}$	4.21	22.88	142.10	416.56	1.63

LUCAS

- The LUCAS Topsoil Survey of the European Union;
- 1 site$/200km^2$ sampling density;
- Kriging was used to map;
- SAGA GIS.




Clay
70
60
50
40
30
20
10
Sand
80
70
60
50
40
30
20
10
0
Silt
70
60
50
40
30
20
10
Clay
100
80
60
40
20
0
Sand
100
80
60
40
20
0
Silt
80
60
40
20
0

Comprehensive R Archive Network (CRAN)

- In principle, packages must pass R CMD check without warnings or significant notes to be admitted to the main CRAN package area.

FracKrigingR: Spatial Multivariate Data Modeling

Aim is to provide fractional Brownian vector field generation algorithm, Hurst parai

Version:	1.0.0
Imports:	psych, clusterGeneration, graphics, stats
Suggests:	knitr, gstat, sp, rmarkdown, raster
Published:	2021-11-08
Author:	Neringa Urbonaite [aut, cre], Leonidas Sakalauskas [aut]
Maintainer:	Neringa Urbonaite <neringa.urbonaite at mif.vu.lt>
License:	GPL-2
Copyright:	Vilnius University Institute of Data Science and Digital Technol
URL:	https://github.com/NidaGreen/FracKriging
NeedsCompilation:	no
CRAN checks:	FracKrigingR results

Documentation:

Reference manual: FracKrigingR.pdf

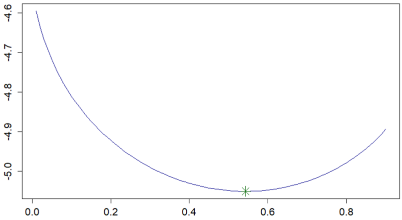
Downloads:

Package source:	FracKrigingR_1.0.0.tar.gz
Windows binaries:	r-devel: FracKrigingR_1.0.0.zip, r-release: FracKrigingR_1.0.0.:
macOS binaries:	r-release (arm64): FracKrigingR_1.0.0.tgz, r-release (x86_64): F

Linking:

Please use the canonical form https://CRAN.R-project.org/package=FracKrigingR t

Comprehensive R Archive Network (CRAN)




```
library(sp)
library(gstat)
library(FracKrigingR)

data(meuse)
xy<-cbind(meuse$x,meuse$y)
X<-xy[1:50,]

Z<-as.matrix(zz1[1:50,])

K<-50
#Hurst parameter estimation
H<-MaxLikelihood(X,Z)
Xnew<-xy[51:100,]
results<- FracKrig(X,Z,Xnew,H)
```

```
results[,2], top =
max(data[,2]), bottom =
min(data[,2])
)
z3 = denormalize(
results[,3], top =
max(data[,3]), bottom =
min(data[,3])
)
RMSE<-function(z,prognosis){
rmse<-
sqrt(((1/(length(z))))*sum((z-
prognosis)^2))
rmse
}
Cd<-RMSE(data[,1],z1)
Cu<-RMSE(data[,2],z2)
Pb<-RMSE(data[,3],z3)
```

Ačiū!