Journal of Data Science 20 (4), 439–460
October 2022
DOI: 10.6339/22-JDS1076
Statistical Data Science

# The Second Competition on Spatial Statistics for Large Datasets

Sameh Abdulah[1,*], Faten Alamri[2], Pratik Nag[3], Ying Sun[1,3], Hatem Ltaief[1], David E. Keyes[1], and Marc G. Genton[1,3]

[1]*Extreme Computing Research Center, King Abdullah University of Science and Technology, Thuwal 23955-6900, Saudi Arabia*
[2]*Mathematical Science Department, Princess Nourah bint Abdulrahman University, Riyadh 84428, Saudi Arabia*
[3]*Statistics Program, King Abdullah University of Science and Technology, Thuwal 23955-6900, Saudi Arabia*

**Abstract**

In the last few decades, the size of spatial and spatio-temporal datasets in many research areas has rapidly increased with the development of data collection technologies. As a result, classical statistical methods in spatial statistics are facing computational challenges. For example, the kriging predictor in geostatistics becomes prohibitive on traditional hardware architectures for large datasets as it requires high computing power and memory footprint when dealing with large dense matrix operations. Over the years, various approximation methods have been proposed to address such computational issues, however, the community lacks a holistic process to assess their approximation efficiency. To provide a fair assessment, in 2021, we organized the first competition on spatial statistics for large datasets, generated by our *ExaGeoStat* software, and asked participants to report the results of estimation and prediction. Thanks to its widely acknowledged success and at the request of many participants, we organized the second competition in 2022 focusing on predictions for more complex spatial and spatio-temporal processes, including univariate nonstationary spatial processes, univariate stationary space-time processes, and bivariate stationary spatial processes. In this paper, we describe in detail the data generation procedure and make the valuable datasets publicly available for a wider adoption. Then, we review the submitted methods from fourteen teams worldwide, analyze the competition outcomes, and assess the performance of each team.

**Keywords** *Gaussian process; multivariate; nonstationary; prediction; space-time; spatial*

## 1 Introduction

With the explosion of spatial and spatio-temporal data coming from different sources such as sensors and satellites, it has become crucial to handle these data in a robust manner. In the last few decades, studies focusing on spatial statistics have followed one main direction to deal with large geospatial data: adopting an approximation approach to reduce the modeling and prediction process complexity. However, these methods have been mostly benchmarked using small and medium-sized datasets because of the prohibitive computation of the exact solution for comparison purposes.

*Corresponding author. Email: sameh.abdulah@kaust.edu.sa.


Received August 14, 2022; Accepted October 29, 2022

Abdulah et al. (2018a) proposed the *ExaGeoStat* software to perform large-scale statistical modeling and prediction for geospatial data on leading-edge parallel hardware architectures. *ExaGeoStat* is able to deal with millions of spatial locations and complete the statistical estimation and prediction in exact and approximate formats (Abdulah et al., 2018b, 2021). It is also supported by a data generation tool that is able to generate synthetic spatial datasets from different realizations with millions of locations. As mentioned by Vu et al. (2021), *ExaGeoStat* can be described as the gold standard for spatial statistical data modeling tools since it can generate and analyze large geospatial data with exact computations for millions of spatial locations. Thus, in 2021, we organized the first KAUST spatial statistics competition for large synthetic datasets for the spatial statistics community to assess existing geospatial modeling methods using *ExaGeoStat*. The competition involved a set of synthetic datasets generated from univariate spatial processes with up to 1M locations. Out of twenty-nine research teams worldwide who registered to participate in the competition, twenty-one teams successfully submitted their results; see Huang et al. (2021) for the analysis of the submissions. Over the past year, *ExaGeoStat* has been further developed to support richer classes of models for spatial and spatio-temporal processes. Therefore, we organized a second competition in 2022 by providing geospatial datasets with new features.

This work follows several studies that have aimed to assess the efficiency of existing tools and methods to perform kriging for geospatial datasets. For instance, in Englund (1990), a collection of spatial datasets were sent to twelve investigators to analyze them and perform spatial inference for missing locations. The "Walker Lake" dataset (Srivastava, 1987) was the core source of the study datasets. The author illustrated the clear variability in the submitted results, motivating and encouraging the statistics community to develop a set of performance-based guidelines to assess the quality of the statistical analysis with different models and methods. The work in Weber and Englund (1992) is an extension of Englund's work, where the relative accuracy of fifteen inference methods was assessed for analyzing the fifty-four datasets of the "Walker Lake" data. For example, the spatial predictions based on the inverse distance were compared to those obtained by the kriging method. The study explained in detail the pros and cons of each method for these datasets. More recently, a study by Heaton et al. (2019) assessed existing kriging tools with simulated and real datasets by introducing one competition to a pre-selected set of research groups. Twelve methods were compared in the competition, with the code and details of each method made available to the statistics community. Wikle et al. (2017) proposed the "secret sauce" to build a benchmarking framework for objective comparison of spatial prediction methods, which has been called a Common Task Framework (CTF). The article argues that any fair comparison between spatial prediction methods should follow this framework by including a set of publicly available spatial datasets, predefined prediction assessing rules, and an automatic scoring referee. The datasets should be complete and well described, and the scoring referee should be unique for all the submissions. In the literature, other studies for the assessment of geospatial methods exist but with different perspectives and performed on other datasets, such as the work in Li and Heap (2011, 2014), Shahbeik et al. (2014), and Bradley et al. (2016).

Following the success of the 2021 competition, this year we organized another competition by providing synthetic datasets generated by *ExaGeoStat*. These datasets include three types of data: spatial nonstationary datasets, space-time and bivariate spatial stationary datasets. The evaluation criteria focuses on point prediction performance only. Specifically, we organized six sub-competitions, two sizes for each type of data: one medium-sized and the other large-sized. The competition was launched on March 1st, 2022, with twenty teams registered to participate in one or more sub-competitions. This year, the competition was hosted on the Kaggle platform,

which allows fast and accurate assessment of the teams' submissions. The competition ended on May 1st, 2022, and fourteen teams successfully submitted their results to the platform.

The rest of the paper is organized as follows. In Section 2, we provide an overview of the *ExaGeoStat* software. In Section 3, we describe in detail the models used for the generation of the competition datasets. In Section 4, we highlight the competition results and list the ranks of different teams based on their performance in different sub-competitions. In Section 5, we summarize the methods used by different teams in this competition. Finally, in Section 6, we conclude with observations and discussions.

## 2 The ExaGeoStat Software

Gaussian processes (GPs) are widely used to model geospatial data, for which the covariance function plays an important role. The modeling process relies on likelihood-based methods that require constructing a covariance matrix whose entries represent the covariance between any two observations via a predefined covariance function. Suppose the covariance function has a parametric form $C(\boldsymbol{s}_i, \boldsymbol{s}_j; \boldsymbol{\theta})$, where $\boldsymbol{s}_i, \boldsymbol{s}_j \in \mathbb{R}^2$ $(i, j = 1, \ldots, n)$ are the spatial locations and $\boldsymbol{\theta}$ is a parameter vector of interest. The Gaussian log-likelihood function is:

$$l(\boldsymbol{\theta}) = -\frac{n}{2}\log(2\pi) - \frac{1}{2}\log|\boldsymbol{\Sigma}(\boldsymbol{\theta})| - \frac{1}{2}\mathbf{Z}^\top \boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}\mathbf{Z}, \tag{1}$$

and needs to be maximized with respect to $\boldsymbol{\theta}$ to obtain its maximum likelihood estimator (MLE). Here $\boldsymbol{\Sigma}(\boldsymbol{\theta}) = \{C(\boldsymbol{s}_i, \boldsymbol{s}_j; \boldsymbol{\theta})\}_{i,j=1}^{n}$ is the $n \times n$ covariance matrix (symmetric and positive definite) of the $n$-dimensional data vector $\mathbf{Z}$, $n$ represents the number of spatial locations, and $|\boldsymbol{\Sigma}(\boldsymbol{\theta})|$ is the determinant of $\boldsymbol{\Sigma}(\boldsymbol{\theta})$.

The computation of the log-likelihood (1) is prohibitive for large sample size $n$ as the complexity of computing the inverse of the covariance matrix, $\boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}$, is $O(n^3)$, and requires $O(n^2)$ memory. This computation becomes even more expensive when considering the multivariate and spatio-temporal case because the sample size becomes $pnm$, where $p$ is the number of spatial variables and $m$ is the number of time points. Thus, dealing with large geospatial datasets requires more advanced computational techniques, and integrating the capabilities of HPC with existing spatial statistics methods becomes mandatory with the vast increase in the sizes of geospatial datasets. Abdulah et al. (2018a) introduced the *ExaGeoStat* parallel software to help in scaling the geospatial operations using leading-edge parallel hardware architectures. *ExaGeoStat* relies on the state-of-the-art parallel linear algebra libraries to allow fast and efficient computation of the log-likelihood function. *ExaGeoStat* also depends on modern runtime systems such as *StarPU* and *PaRSEC* to improve the portability of the code on the different parallel hardware architectures, including GPUs. *ExaGeoStat* has driven breakthroughs in the spatial statistics field by supporting rich classes of covariance models for large-scale spatial and spatio-temporal data; see, e.g., Salvaña et al. (2021), Mondal et al. (2022), and Salvaña et al. (2022).

Several studies have demonstrated the capabilities of *ExaGeoStat* in utilizing today's high-performance hardware for spatial statistics. Figure 1 explains in detail the statistical tasks supported by *ExaGeoStat* and a visual literature review of the software portability on different hardware architectures. The figure shows at a high-level the five main components currently included in the software, i.e., data generation, MLE modeling, geospatial prediction, mean loss of efficiency (MLOE) and mean misspecification of the mean square error (MMOM) tools, and the Fisher information matrix (FIM) computing. The MLOE/MMOM tools are used to assess the loss of the prediction efficiency by using the approximated or misspecified covariance

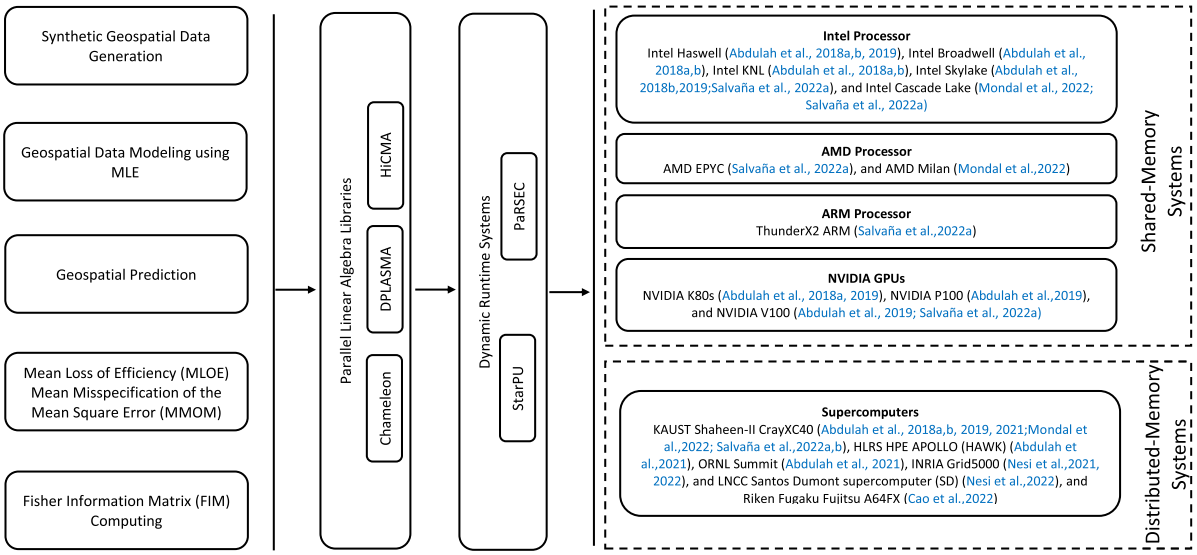



Figure 1: Statistical tasks supported by *ExaGeoStat* with a visual literature review of the software portability on different hardware architectures.

models (Hong et al., 2021). The Fisher information matrix provides information about the importance of a single observation in estimating unknown statistical parameters, i.e., uncertainty quantification. Figure 1 also highlights the parallel linear algebra libraries on which *ExaGeoStat* relies to perform the required covariance matrix operations, i.e., Chameleon and DPLASMA for dense matrix computations and HiCMA for Tile Low-Rank (TLR) matrix approximations. In addition, as seen Figure 1, *ExaGeoStat* relies on two different runtime systems, i.e., *StarPU* and *PaRSEC*.

# 3 Detailed Description of the Competition Datasets

In this competition, we generated three types of data: univariate nonstationary spatial data, univariate stationary space-time data, and bivariate stationary spatial data. We consider different settings for each type. The following subsections give more details about the data generating models and the associated settings. A summary of the six sub-competitions can be found in Table 3 at the end of this section.

## 3.1 Univariate Nonstationary Spatial Model (Sub-competitions 1a and 1b)

We consider a univariate Gaussian random field (GRF), $Z(\boldsymbol{s})$, $\boldsymbol{s} \in [0,1]^2$, modeled as:

$$Z(\boldsymbol{s}) = m(\boldsymbol{s}) + Y(\boldsymbol{s}) + \epsilon(\boldsymbol{s}), \tag{2}$$

where $m(\cdot)$ is the mean function, $Y(\cdot)$ is a spatially dependent and zero-mean GRF with covariance function $C(\cdot,\cdot)$, $\epsilon(\cdot) \sim \mathcal{N}(0,\tau^2)$ is an independent noise process with $\tau^2$ denoting the nugget effect, and $\epsilon(\cdot)$ is independent of $Y(\cdot)$. In Sub-competitions 1a and 1b, four nonstationary datasets were generated under the model in equation (2).

For datasets 1a-1 and 1b-1, the nonstationarity is in the deterministic mean function as described in Xiong et al. (2007) and Ba and Joseph (2012). Data were generated by $Z(\boldsymbol{s}) = m(\boldsymbol{s}) + \epsilon(\boldsymbol{s})$, $\boldsymbol{s} = (s_x, s_y)$ on a regular grid. For the dataset 1a-1 of size 100K, $\tau = 0.1$ and $m(\boldsymbol{s})$ is:

$$\begin{aligned} m(\boldsymbol{s}) = 5\sin\left\{30\left(\frac{s_x+s_y}{2}-0.9\right)^3\right\}\cos\left\{20\left(\frac{s_x+s_y}{2}-0.9\right)^4\right\} \\ + \frac{1}{2}\exp\{\sin(30s_x)+\sin(13s_y)\} + \frac{1}{2}\left(\frac{s_x+s_y}{2}-0.2\right). \end{aligned} \tag{3}$$

For the dataset 1b-1 of size 1M, $\tau = 0.3$ and $m(\boldsymbol{s})$ takes the form:

$$\begin{aligned} m(\boldsymbol{s}) = 3\sin\left\{20\left(\frac{s_x+s_y}{2}+1.9\right)\right\}\cos\left\{20\left(\frac{s_x+s_y}{2}-1.2\right)^6\right\} \\ + \frac{3}{5}\exp\{\sin(25s_x)+\sin(13s_y)\} + \frac{1}{2}\left(\frac{s_x+s_y}{2}-0.2\right). \end{aligned} \tag{4}$$

Datasets 1a-2 (100K) and 1b-2 (1M) were generated at irregular locations from a zero-mean Gaussian process $Z(\boldsymbol{s})$ with a nonstationary Matérn covariance function (Li and Sun, 2019, and references therein):

$$\begin{aligned} C^{NS}(\boldsymbol{s}_i, \boldsymbol{s}_j; \boldsymbol{\theta}) = \tau^2 \mathbb{1}_{[i=j]}(\boldsymbol{s}_i, \boldsymbol{s}_j) + \frac{\sigma(\boldsymbol{s}_i)\sigma(\boldsymbol{s}_j)}{\Gamma(\nu)2^{\nu-1}}|\boldsymbol{\Sigma}(\boldsymbol{s}_i)|^{1/4}|\boldsymbol{\Sigma}(\boldsymbol{s}_j)|^{1/4} \\ \times \left|\frac{\boldsymbol{\Sigma}(\boldsymbol{s}_i)+\boldsymbol{\Sigma}(\boldsymbol{s}_j)}{2}\right|^{-1/2}\left(2\sqrt{\nu Q_{ij}}\right)^{\nu}\mathcal{K}_{\nu}\left(2\sqrt{\nu Q_{ij}}\right), \end{aligned} \tag{5}$$

where $\sigma(\boldsymbol{s}_i)$ is the spatially varying standard deviation, $\boldsymbol{\Sigma}(\boldsymbol{s}_i)$ is the kernel matrix at $\boldsymbol{s}_i$, $\mathcal{K}_{\nu}$ is the modified Bessel function of the second kind of order $\nu > 0$, $\nu$ is the smoothness parameter, and $Q_{ij}$ is the square Mahalanobis distance between $\boldsymbol{s}_i$ and $\boldsymbol{s}_j$.

The nonstationarity of the GRF is controlled by the spatially varying parameters $\theta(\boldsymbol{s}_i) \in \{\boldsymbol{\Sigma}(\boldsymbol{s}_i), \sigma(\boldsymbol{s}_i)\}$. The kernel matrices are obtained through a spectral decomposition:

$$\boldsymbol{\Sigma}(\boldsymbol{s}_i) = \begin{bmatrix}\cos(\phi) & -\sin(\phi)\\ \sin(\phi) & \cos(\phi)\end{bmatrix}\begin{bmatrix}\lambda_1(\boldsymbol{s}_i) & 0\\ 0 & \lambda_2(\boldsymbol{s}_i)\end{bmatrix}\begin{bmatrix}\cos(\phi) & \sin(\phi)\\ -\sin(\phi) & \cos(\phi)\end{bmatrix},$$

where $\lambda_1(\boldsymbol{s}_i) > 0$ and $\lambda_2(\boldsymbol{s}_i) > 0$ are eigenvalues that represent spatial ranges and $\phi \in [0, \pi/2]$ represents the angle of rotation.

The generation process depends on dividing the spatial region into a grid of subregions centered at reference locations $(\tilde{\boldsymbol{s}}_k)_{k=1}^M$. A spatially varying parameter $\theta$ at location $\boldsymbol{s}_i$ is defined as follows:

$$\theta(\boldsymbol{s}_i) = \sum_{k=1}^{M} w(\boldsymbol{s}_i, \tilde{\boldsymbol{s}}_k)\theta_k, \qquad w(\boldsymbol{s}_i, \tilde{\boldsymbol{s}}_k) = \frac{K(\boldsymbol{s}_i, \tilde{\boldsymbol{s}}_k)}{\sum_{k=1}^{M} K(\boldsymbol{s}_i, \tilde{\boldsymbol{s}}_k)},$$

where $M$ represents the number of subregions, $\theta_k$ is the parameter value at the reference location $\tilde{\boldsymbol{s}}_k$ associated with the $k$-th subregion, $w(\boldsymbol{s}_i, \tilde{\boldsymbol{s}}_k)$ is a weight function, and $K(\cdot)$ denotes a bivariate kernel function. Herein, we chose a Gaussian kernel defined as $K(\boldsymbol{s}_i, \tilde{\boldsymbol{s}}_k) = \exp\{-\|\boldsymbol{s}_i - \tilde{\boldsymbol{s}}_k\|^2/(2h)\}$, where $h > 0$ is the bandwidth parameter fixed to 0.09.

We chose $M = 4$ subregions with four reference locations with the coordinates $(0.25, 0.25)$, $(0.25, 0.75)$, $(0.75, 0.25)$, $(0.75, 0.75)$ in the unit square. Parameter settings at the 4 reference

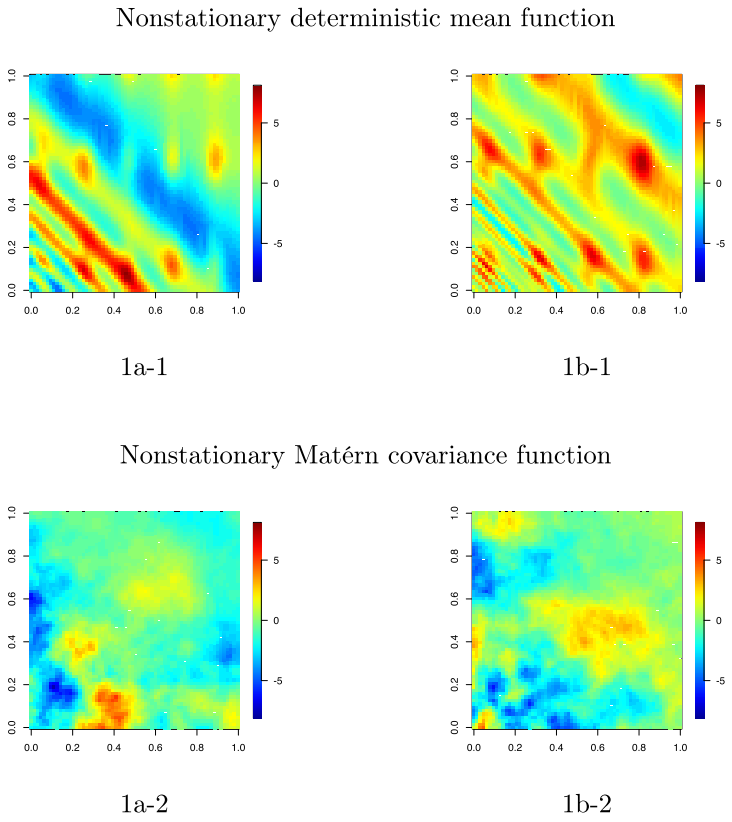



Figure 2: Synthetic univariate nonstationary spatial datasets in Sub-competitions 1a and 1b: (1a-1) A 100K nonstationary dataset generated using the deterministic mean function in (3); (1b-1) A 1M nonstationary dataset generated using the deterministic mean function in (4); (1a-2) A 100K nonstationary dataset generated using the covariance function in (5); and (1b-2) A 1M nonstationary dataset generated using the covariance function in (5).

locations are $\sigma = (3.5, 1.9, 1.8, 0.7)$, $\lambda_1 = \lambda_2 = (0.03, 0.07, 0.1, 0.3)$, with constant $\nu = 0.7$, $\tau^2 = 0.3$, and $\phi = \pi/2$.

For Sub-competitions 1a and 1b, 90% of the data were randomly selected as the training datasets, and the remaining 10% were used as testing datasets. Training datasets were given to the participants to perform prediction on the locations of the testing data. Figure 2 shows the visual images of the four training datasets in Sub-competitions 1a and 1b that illustrate different features of nonstationarity.

### 3.2 Univariate Stationary Space-Time Model (Sub-competitions 2a and 2b)

The second part of the competition includes eighteen univariate space-time datasets generated from a zero-mean Gaussian process with a non-separable stationary space-time covariance function (Gneiting, 2002) at space-time locations $(\boldsymbol{s}, t) \in [0, 1]^2 \times \mathbb{R}$:

$$C(\boldsymbol{h}, u; \boldsymbol{\theta}) = \frac{\sigma^2}{a_t |u|^{2\alpha} + 1} \mathcal{M}_\nu \left\{ \frac{\|\boldsymbol{h}\|/a_s}{(a_t |u|^{2\alpha} + 1)^{\beta/2}} \right\}, \tag{6}$$

Table 1: The settings to generate datasets in Sub-competitions 2a and 2b using a non-separable stationary space-time Matérn covariance model.

| Dataset | # spatial locations | # time slots | $\sigma^2$ | $\alpha$ | $\beta$ | $\nu$ | $a_s$ | $a_t$ | Time EffRange | Space EffRange | Prediction setting |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ST1 (2a-1) | 1K | 100 | 0.9 | 0.6 | 0.9 | 1 | 0.02 | 1 | 11.63 | 0.08 | RS |
| ST2 (2a-2) | 1K | 100 | 0.9 | 0.6 | 0.9 | 1 | 0.08 | 0.24 | 38.20 | 0.32 | RS |
| ST3 (2a-3) | 1K | 100 | 0.9 | 0.08 | 0.9 | 1 | 0.4 | 1 | $\infty$ | 1.6 | RS |
| ST4 (2a-4) | 1K | 100 | 0.9 | 0.6 | 0.9 | 1 | 0.02 | 1 | 11.63 | 0.08 | RST |
| ST5 (2a-5) | 1K | 100 | 0.9 | 0.6 | 0.9 | 1 | 0.08 | 0.24 | 38.20 | 0.32 | RST |
| ST6 (2a-6) | 1K | 100 | 0.9 | 0.08 | 0.9 | 1 | 0.4 | 1 | $\infty$ | 1.6 | RST |
| ST7 (2a-7) | 1K | 100 | 0.9 | 0.6 | 0.9 | 1 | 0.02 | 1 | 11.63 | 0.08 | T10 |
| ST8 (2a-8) | 1K | 100 | 0.9 | 0.6 | 0.9 | 1 | 0.08 | 0.24 | 38.20 | 0.32 | T10 |
| ST9 (2a-9) | 1K | 100 | 0.9 | 0.08 | 0.9 | 1 | 0.4 | 1 | $\infty$ | 1.6 | T10 |
| ST10 (2b-1) | 10K | 100 | 0.9 | 0.6 | 0.9 | 1 | 0.02 | 1 | 11.63 | 0.08 | RS |
| ST11 (2b-2) | 10K | 100 | 0.9 | 0.6 | 0.9 | 1 | 0.08 | 0.24 | 38.20 | 0.32 | RS |
| ST12 (2b-3) | 10K | 100 | 0.9 | 0.08 | 0.9 | 1 | 0.4 | 1 | $\infty$ | 1.6 | RS |
| ST13 (2b-4) | 10K | 100 | 0.9 | 0.6 | 0.9 | 1 | 0.02 | 1 | 11.63 | 0.08 | RST |
| ST14 (2b-5) | 10K | 100 | 0.9 | 0.6 | 0.9 | 1 | 0.08 | 0.24 | 38.20 | 0.32 | RST |
| ST15 (2b-6) | 10K | 100 | 0.9 | 0.08 | 0.9 | 1 | 0.4 | 1 | $\infty$ | 1.6 | RST |
| ST16 (2b-7) | 10K | 100 | 0.9 | 0.6 | 0.9 | 1 | 0.02 | 1 | 11.63 | 0.08 | T10 |
| ST17 (2b-8) | 10K | 100 | 0.9 | 0.6 | 0.9 | 1 | 0.08 | 0.24 | 38.20 | 0.32 | T10 |
| ST18 (2b-9) | 10K | 100 | 0.9 | 0.08 | 0.9 | 1 | 0.4 | 1 | $\infty$ | 1.6 | T10 |

where $\sigma^2 > 0$ is the variance, $\nu > 0$ and $\alpha \in (0, 1]$ are the smoothing parameters, $a_s$ and $a_t > 0$ are the range parameters in space and time, respectively, $\beta \in (0, 1]$ is the space-time interaction parameter, and $\mathcal{M}_\nu$ is the univariate Matérn correlation function:

$$\mathcal{M}_\nu(r) = \frac{1}{2^{\nu-1}\Gamma(\nu)} r^\nu \mathcal{K}_\nu(r).$$

The sizes of the generated datasets were 1K and 10K locations, with 100 time-slots. We considered different spatial and temporal dependencies: strong, moderate and weak for a non-separable model with $\beta = 0.9$. The parameter settings for each case were as follows: weak-strong $(a_s, a_t) = (0.02, 1)$, moderate-moderate $(a_s, a_t) = (0.08, 0.24)$, strong-strong $(a_s, a_t) = (0.4, 1)$, $\alpha = (0.08, 0.6)$, and $(\sigma^2, \beta, \nu) = (0.9, 0.9, 1)$. Table 1 summarizes the parameter settings used for each dataset, as well as the time/space effective range (EffRange).

We considered three different scenarios of leaving out space-time points for prediction:

1. Random spatial locations with all times left out (RS);
2. Random locations in space/time left out (RST);
3. All spatial locations are missing on the last 10 time points (T10).

There were eighteen space-time datasets generated by considering different parameter settings and leaving out schemes. Figure 3 visualizes datasets at $t = 0$, $t = 1$, and $t = 2$ generated under the first three settings of each sub-competition in Table 1 with RS left out. Other datasets (under RST and T10) were generated using the same parameter settings.

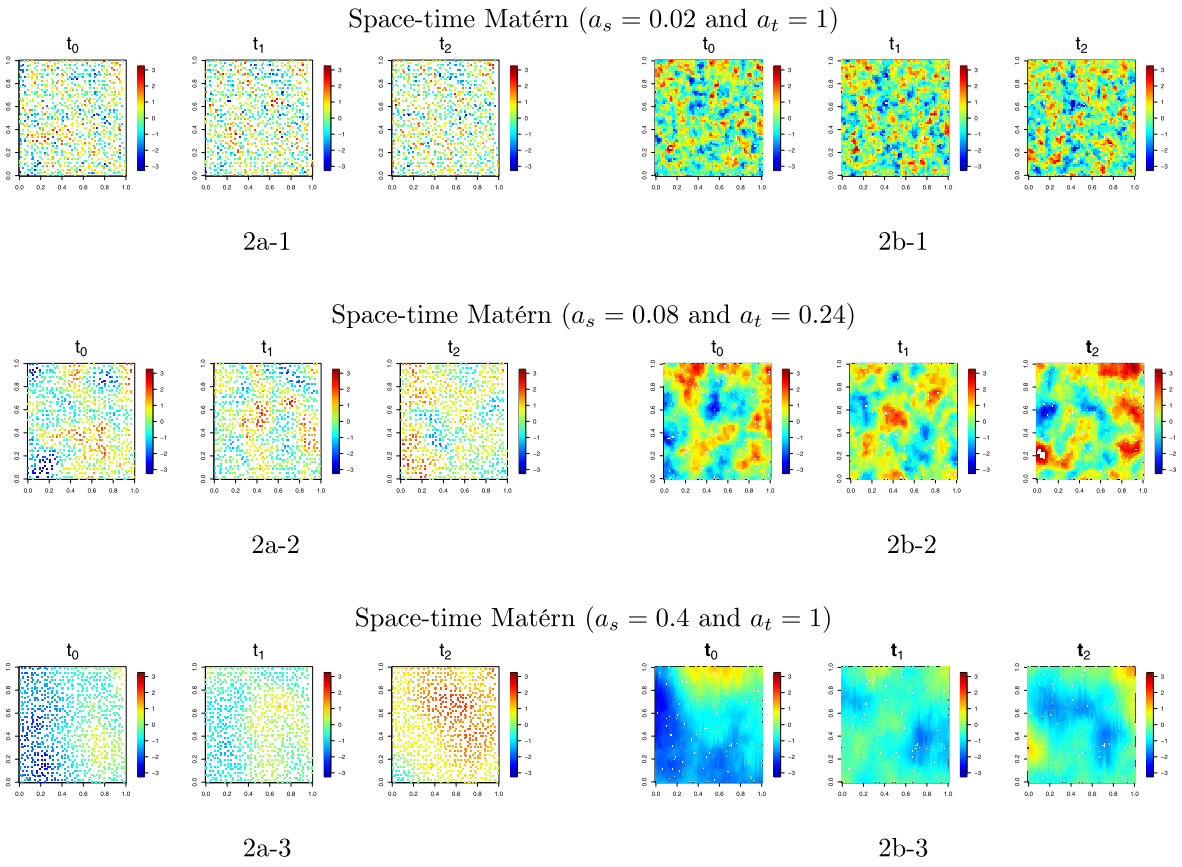



Figure 3: (2a-1, 2a-2, 2a-3) Synthetic space-time datasets with 1K locations and 100 time-slots generated from a non-separable space-time covariance function; (2b-1, 2b-2, 2b-3) Synthetic space-time datasets with 10K locations and 100 time-slots generated from a non-separable space-time covariance function. The figure visualizes the datasets at $t = 0$, $t = 1$, $t = 2$ generated under the first three settings of each sub-competition in Table 1 with RS left out.

### 3.3 Bivariate Stationary Spatial Model (Sub-competitions 3a and 3b)

The third part of the competition involved generating datasets from bivariate zero-mean Gaussian random fields $(Z_1(\boldsymbol{s}), Z_2(\boldsymbol{s}))^\top$ using two different cross-covariance functions (without nugget): the parsimonious (Gneiting et al., 2010) and the flexible (Apanasovich et al., 2012) Matérn.

The parsimonious Matérn cross-covariance function has the following form:

$$C_{ij}(\boldsymbol{h};\boldsymbol{\theta}) = \frac{\rho_{ij}\sigma_{ii}\sigma_{jj}}{2^{\nu_{ij}-1}\Gamma(\nu_{ij})}\left(\frac{\|\boldsymbol{h}\|}{a}\right)^{\nu_{ij}}\mathcal{K}_{\nu_{ij}}\left(\frac{\|\boldsymbol{h}\|}{a}\right), \tag{7}$$

for $i, j = 1, 2$. Here $\boldsymbol{\theta}$ consists of the following parameters: marginal smoothnesses $\nu_{ii} > 0$, a common range $a > 0$, marginal variances $\sigma_{ii}^2 > 0$, and the collocated correlation $\rho_{ij}$, $i \neq j$. The cross-smoothness $\nu_{ij} = \frac{1}{2}(\nu_{ii} + \nu_{jj})$ and $\rho_{ij}$ are related:

$$\rho_{ij} = \beta_{ij}\frac{\Gamma(\nu_{ii}+\frac{d}{2})^{\frac{1}{2}}}{\Gamma(\nu_{ii})^{\frac{1}{2}}}\frac{\Gamma(\nu_{jj}+\frac{d}{2})^{\frac{1}{2}}}{\Gamma(\nu_{jj})^{\frac{1}{2}}}\frac{\Gamma\{\frac{1}{2}(\nu_{ii}+\nu_{jj})\}}{\Gamma\{\frac{1}{2}(\nu_{ii}+\nu_{jj})+\frac{d}{2}\}},$$

Table 2: The settings to generate datasets in Sub-competitions 3a and 3b using a parsimonious/flexible Matérn covariance model.

| Dataset | Size | $\sigma_{11}^2$ | $\sigma_{22}^2$ | $\beta_{12}$ | $\nu_{11}$ | $\nu_{22}$ | $a_{11}$ | $a_{22}$ | EffRange of $C_{11}$ | EffRange of $C_{22}$ | $\rho_{12}$ | Matérn Model |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 3a-1 | 50K | 0.9 | 0.9 | 0.9 | 0.6 | 1.4 | 0.03 | 0.03 | 0.097 | 0.138 | 0.824 | Parsimonious |
| 3b-1 | 500K | 0.9 | 0.9 | 0.9 | 0.6 | 1.4 | 0.03 | 0.03 | 0.097 | 0.138 | 0.824 | Parsimonious |
| 3a-2 | 50K | 0.9 | 0.9 | 0.9 | 0.9 | 0.9 | 0.02 | 0.3 | 0.077 | 1.15 | 0.9 | Flexible |
| 3a-3 | 50K | 0.9 | 0.9 | 0.9 | 0.6 | 1.4 | 0.03 | 0.1 | 0.097 | 0.46 | 0.824 | Flexible |
| 3b-2 | 500K | 0.9 | 0.9 | 0.9 | 0.9 | 0.9 | 0.02 | 0.3 | 0.077 | 1.15 | 0.9 | Flexible |
| 3b-3 | 500K | 0.9 | 0.9 | 0.9 | 0.6 | 1.4 | 0.03 | 0.1 | 0.097 | 0.46 | 0.824 | Flexible |

where $d = 2$ and $(\beta_{ij})_{i,j=1}^{p}$ is a symmetric and positive definite correlation matrix. More descriptions can be found in Salvaña et al. (2021).

The flexible Matérn cross-covariance function has the following form:

$$C_{ij}(\boldsymbol{h};\boldsymbol{\theta}) = \frac{\rho_{ij}\sigma_{ii}\sigma_{jj}}{2^{\nu_{ij}-1}\Gamma(\nu_{ij})}\left(\frac{\|\boldsymbol{h}\|}{a_{ij}}\right)^{\nu_{ij}}\mathcal{K}_{\nu_{ij}}\left(\frac{\|\boldsymbol{h}\|}{a_{ij}}\right), \tag{8}$$

where $a_{ij}^2 = (a_{ii}^2 + a_{jj}^2)/2 + \bar{\tau}(a_{ii} - a_{jj})^2$, $0 \leqslant \bar{\tau} < \infty$. In this case, the marginal ranges $a_{ii}$ are no longer constant across variables.

Table 2 shows two different settings to generate a 50K (3a-1) and a 500K (3b-1) datasets using the parsimonious bivariate model, with $\beta_{12} = 0.9$, $a = a_{11} = a_{22} = 0.03$, $\nu_{11} = 0.6$ and $\nu_{22} = 1.4$. Table 2 also shows two different settings of the flexible bivariate Matérn model with $\beta_{12} = 0.9$ and $\bar{\tau} = 0$:

- Same smoothness parameters $\nu_{11} = \nu_{22} = 0.9$ and different range parameters with $a_{11} = 0.02$ indicating weak dependence and $a_{22} = 0.3$ indicating strong dependence (3a-2 and 3b-2);
- Mixed parameters setting: smoothness parameters $\nu_{11} = 0.6$ and $\nu_{22} = 1.4$, and range parameters $a_{11} = 0.03$ and $a_{22} = 0.1$ (3a-3 and 3b-3).

Figure 4 shows visual images of datasets generated from the parsimonious and the flexible Matérn models, respectively. For prediction, 90% of each dataset were provided as training data, while 10% were kept as testing data with both variables left out simultaneously.

### 3.4 Assessment Metric

We hosted the competition this year on the Kaggle machine learning and data science platform. Six different Kaggle competitions were created to represent the 6 sub-competitions as follows:

- Sub-competition 1a: two 100K univariate nonstationary spatial datasets: https://www.kaggle.com/competitions/2022-kaust-ss-competition-1a
- Sub-competition 1b: two 1M univariate nonstationary spatial datasets: https://www.kaggle.com/competitions/2022-kaust-ss-competition-1b
- Sub-competition 2a: nine 1K spatial locations at 100 time points (space-time) datasets: https://www.kaggle.com/competitions/2022-kaust-ss-competition-2a
- Sub-competition 2b: nine 10K spatial locations at 100 time points (space-time) datasets: https://www.kaggle.com/competitions/2022-kaust-ss-competition-2b

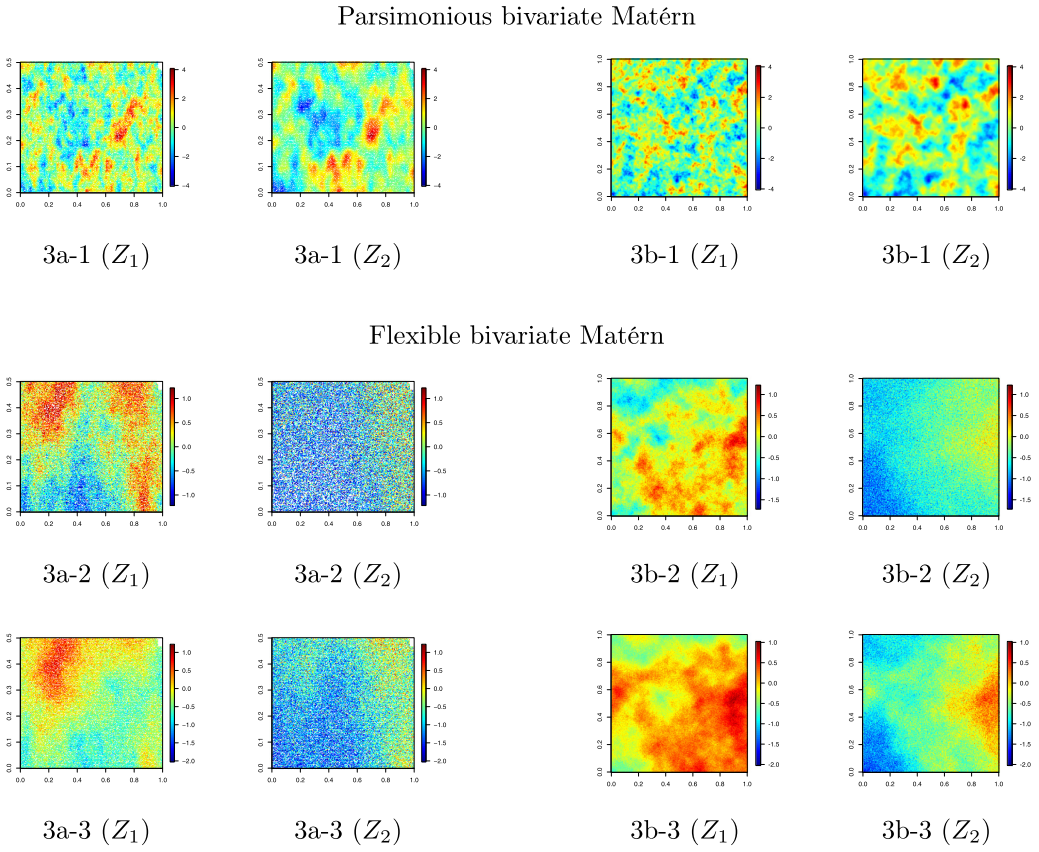



Figure 4: Bivariate datasets $(Z_1(\boldsymbol{s}), Z_2(\boldsymbol{s}))^\top$ using the parsimonious and flexible Matérn models with two sizes: 50K (3a-1, 3a-2, and 3a-3) and 500K (3b-1, 3b-2, and 3b-3).

- Sub-competition 3a: three 50K bivariate spatial datasets: https://www.kaggle.com/competitions/2022-kaust-ss-competition-3a
- Sub-competition 3b: three 500K bivariate spatial datasets: https://www.kaggle.com/competitions/2022-kaust-ss-competition-3b

The participating teams were ranked independently for each sub-competition. We used the Root Mean Square Error (RMSE) criterion to evaluate the prediction accuracy for each sub-competition:

$$\text{RMSE} = \sqrt{\frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} \left(\hat{Z}_i - Z_i\right)^2},$$

where $\hat{Z}_i$ and $Z_i$ are respectively the predicted and true realization values in the testing dataset, and $N_{\text{test}}$ is the total number of data points in the testing dataset. Depending on the sub-competitions, $\hat{Z}_i$ and $Z_i$ are either spatial only or spatio-temporal variables. The final score of each team for a given sub-competition is calculated in Kaggle using the Mean Columnwise Root Mean Squared Error (MCRMSE), i.e., the averaged RMSE over datasets for each sub-competition.

Table 3: Summary of the six sub-competitions.

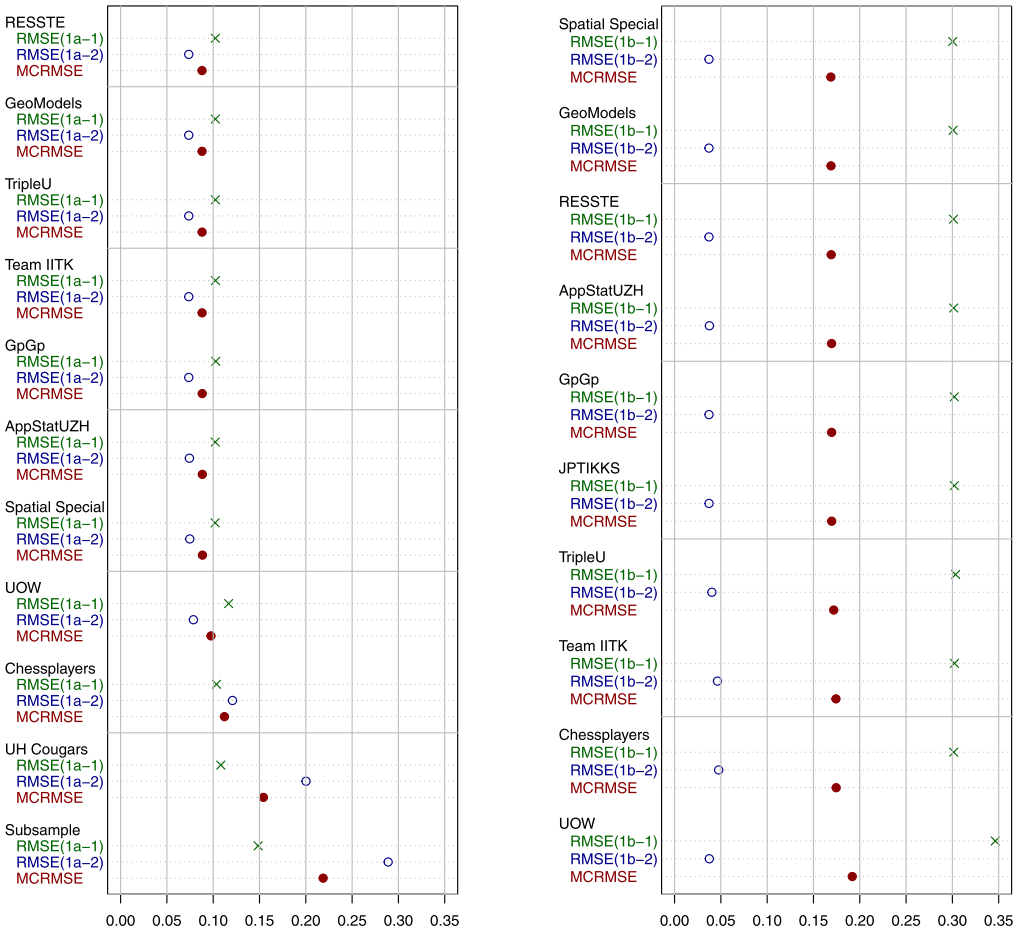
| Sub-competition | Setting | True Data Model | # of Datasets | Training Data Size | Testing Data Size |
|---|---|---|---|---|---|
| 1a | Univariate Nonstationary Spatial | GP with Nonstationary Mean or Cov | 2 | 90K | 10K |
| 1b | Univariate Nonstationary Spatial | GP with Nonstationary Mean or Cov | 2 | 900K | 100K |
| 2a | Univariate Stationary Space-Time | GP with Non-Separable Cov | 9 | 90K | 10K |
| 2b | Univariate Stationary Space-Time | GP with Non-Separable Cov | 9 | 900K | 100K |
| 3a | Bivariate Stationary Spatial | GP with Parsimonious or Flexible Matérn Cross-Cov | 3 | 45K | 5K |
| 3b | Bivariate Stationary Spatial | GP with Parsimonious or Flexible Matérn Cross-Cov | 3 | 450K | 50K |

## 4 Analysis of the Submitted Results

In this section, we summarize the results submitted from different teams (list provided in Table S1 of the Supplementary Material) and compare their performances. Specially, we have designed visual graphs to highlight the teams' performance related to each dataset and its impact on the teams' final rank in each sub-competition. The RMSE values for different teams on all datasets are provided in the Supplementary Material (Tables S2 to S11) as well as those obtained with *ExaGeoStat* for reference purpose.

### 4.1 Results of Sub-competitions 1a and 1b (Univariate Nonstationary Spatial)

In Sub-competition 1a, there were two 100K datasets, 1a-1 and 1a-2, where 1a-1 has been generated using the deterministic mean function in (3), while 1a-2 has been generated using the nonstationary covariance function in (5) with specific settings as mentioned in Section 3.1. Figure 5a shows the competition results of the participating teams in Sub-competition 1a. All the teams except the `SubSample` team got close RMSE values for all the datasets. The `SubSample` team utilized a subsampling technique to fit the given data. However, their approach seems inappropriate for the given nonstationary data. In dataset 1a-2, the top eight teams, i.e., `RESSTE`,




(a) 1a (b) 1b

Figure 5: Sub-competitions 1a and 1b leaderboard.

`GeoModels`, `TripleU`, `Team IITK`, `GpGp`, `AppStatUZH`, `Spatial Special`, and `UOW` obtained very close RMSEs with a variation of around $7 \times 10^{-6}$. The next three teams, i.e., `Chessplayers`, `UH Cougars`, and `SubSample` have larger RMSEs than the top eight teams. More details are needed to understand how to improve the performance of these three teams.

In Sub-competition 1b, we had two 1M datasets, 1b-1 and 1b-2, where dataset 1b-1 has been generated using the deterministic mean function in (4), while dataset 1b-2 has been generated using the nonstationary covariance function in (5) with the settings mentioned in Section 3.1. Figure 5b summarizes the results from different teams. As shown by the figure, all the teams except the `UOW` team were able to obtain very close RMSEs in datasets 1b-1 and 1b-2 with variation around $10^{-5}$. In dataset 1b-2, the `UOW` team performed better than `TripleU`, `Team IITK`, and `Chessplayers` teams with RMSE equal to 0.0375 compared to 0.0403, 0.0463, and 0.0476 for the three teams, respectively. However, the overall MCRMSEs are better for the three teams than the `UOW` team, as shown in the figure, because of the performance of the `UOW` team in dataset 1b-1.

### 4.2 Results of Sub-competitions 2a and 2b (Univariate Stationary Space-time)

In Sub-competition 2a, nine space-time datasets were generated using the non-separable stationary space-time covariance function in (6) with 1K locations and 100 time-slots. The generated datasets were divided into a training dataset (90%) and a testing dataset (10%). The testing datasets have been chosen with three settings, i.e., RS, RST, and T10, as described in Section 3.2. We have seven participants in this sub-competition and the results are presented in Figure 6. The `Envstat.ai` team was able to obtain the best MCRMSE, i.e., 0.2573, for all the nine datasets in Sub-competition 2a. This result is 1.8× better than the second-ranked team, i.e., `GpGp`. With a closer examination of the performance of the `Envstat.ai` team in different datasets, we observe that the team was able to obtain the best RMSEs in datasets 2a-7, 2a-8, and 2a-9, where all the spatial locations at the last 10 time-slots were missing (i.e., forecasting case). The improvements compared to the `GpGp` team are 1.36×, 2.08×, and 3.47× in datasets 2a-7, 2a-8, and 2a-9, respectively, which also shows that the `Envstat.ai` team performed better with strong space correlation and strong time correlation. The RMSEs were close to each other for different datasets for all other teams.

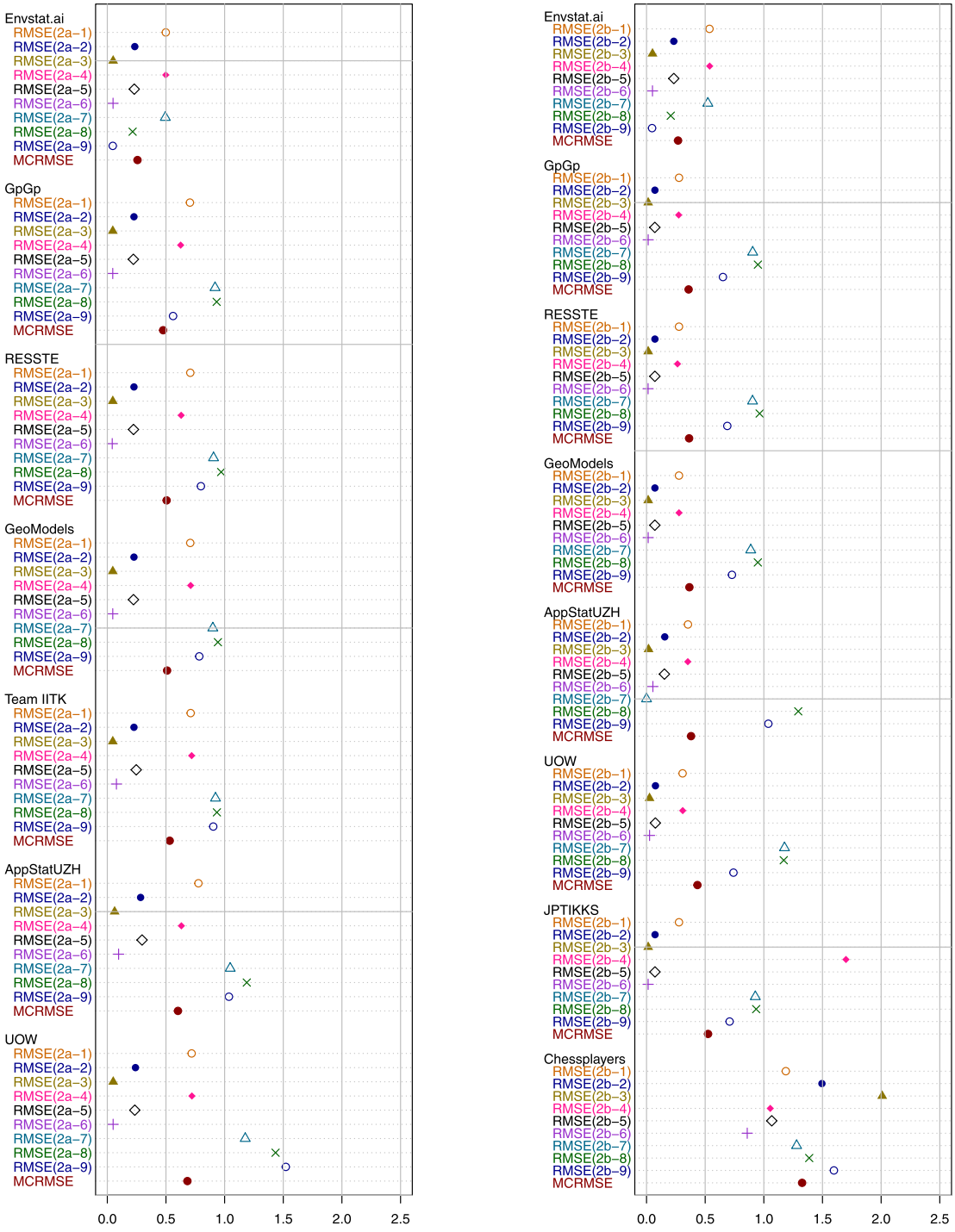
In Sub-competition 2b, nine space-time datasets were generated using the non-separable space-time covariance function in (6) with 10K locations and 100 time-slots. Eight teams have successfully submitted their results to this sub-competition. The ranks of the first four teams were the same as in Sub-competition 2a. The `Envstat.ai` team obtained the first rank with a total MCRMSE value equals to 0.2687, which is 1.33× better than the second-ranked team, i.e., `GpGp`. The `Envstat.ai` team outperformed the `GpGp` team in only three datasets, 2b-7, 2b-8, and 2b-9, by 1.32×, 2.14×, and 3.71×. Despite the fact that the `GpGp` team was able to outperform the `Envstat.ai` team in six datasets 2b-1, 2b-2, 2b-3, 2b-4, 2b-5, and 2b-6 by 1.39×, 1.80×, 1.90×, 1.40×, 1.81×, 1.89×, the differences in RMSE values were not enough to have an MCRMSE value lower than the `Envstat.ai` team.

Figure 7 shows the individual leaderboards in Sub-competition 2a and 2b for all the participating teams. It clearly shows that the `Envstat.ai` team was able to obtain the lowest RMSE values in the T10 datasets in both sub-competitions, i.e., forecasting case, with a noticeable improvement compared to the other teams. However, it did not perform well in 3 out of the 4 remaining datasets. The figure also shows that the `GpGp` team performed quite well in all the datasets compared to the `Envstat.ai` team. The performance of the `Envstat.ai` team in 2a-7, 2a-8, 2a-9, 2b-7, 2b-8, and 2b-9 was good enough to win both sub-competitions.

### 4.3 Results of Sub-competitions 3a and 3b (Bivariate Stationary Spatial)

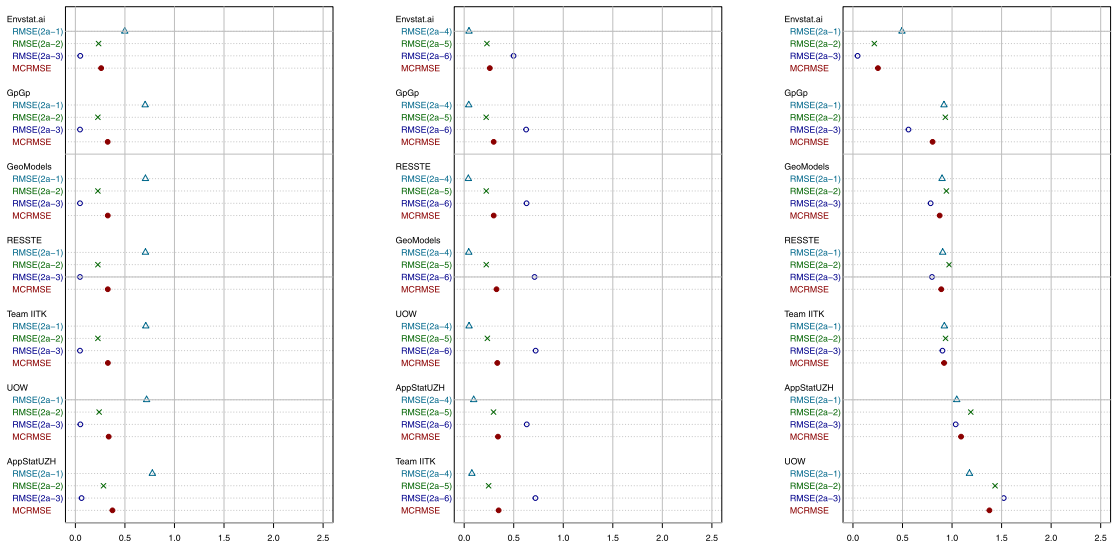
In Sub-competition 3a, we generated one 50K bivariate datasets using the parsimonious Matérn covariance function in (7), i.e., 3a-1, and two 50K bivariate datasets using the flexible Matérn covariance function in (8), i.e., 3a-2 and 3a-3. Six teams submitted their results to this sub-competition. Figure 8 shows the two leaderboard lists of Sub-competitions 3a and 3b. As shown, the `GpGp` team outperformed all the other teams, slightly improving the final MCRMSE. The difference between teams was minimal. The only observation is that the `TripleU` team has a high RMSE in 3a-2, representing the flexible Matérn model under $\nu_{11} = 0.6$ and $\nu_{11} = 1.4$, compared to the other teams.

In Sub-competition 3b, we generated one 500K bivariate dataset using the parsimonious Matérn covariance function in (7), i.e., 3b-1, and two 500K bivariate datasets using the flexible Matérn covariance function in (8), i.e., 3b-2 and 3b-3. Six teams submitted their results

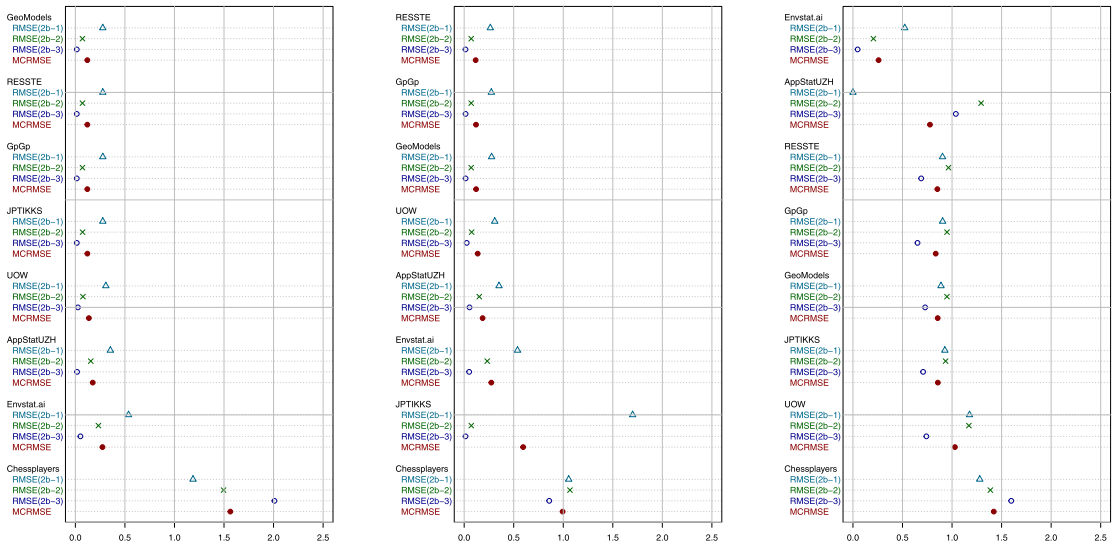



(a) 2a (b) 2b

Figure 6: Sub-competitions 2a and 2b leaderboard.



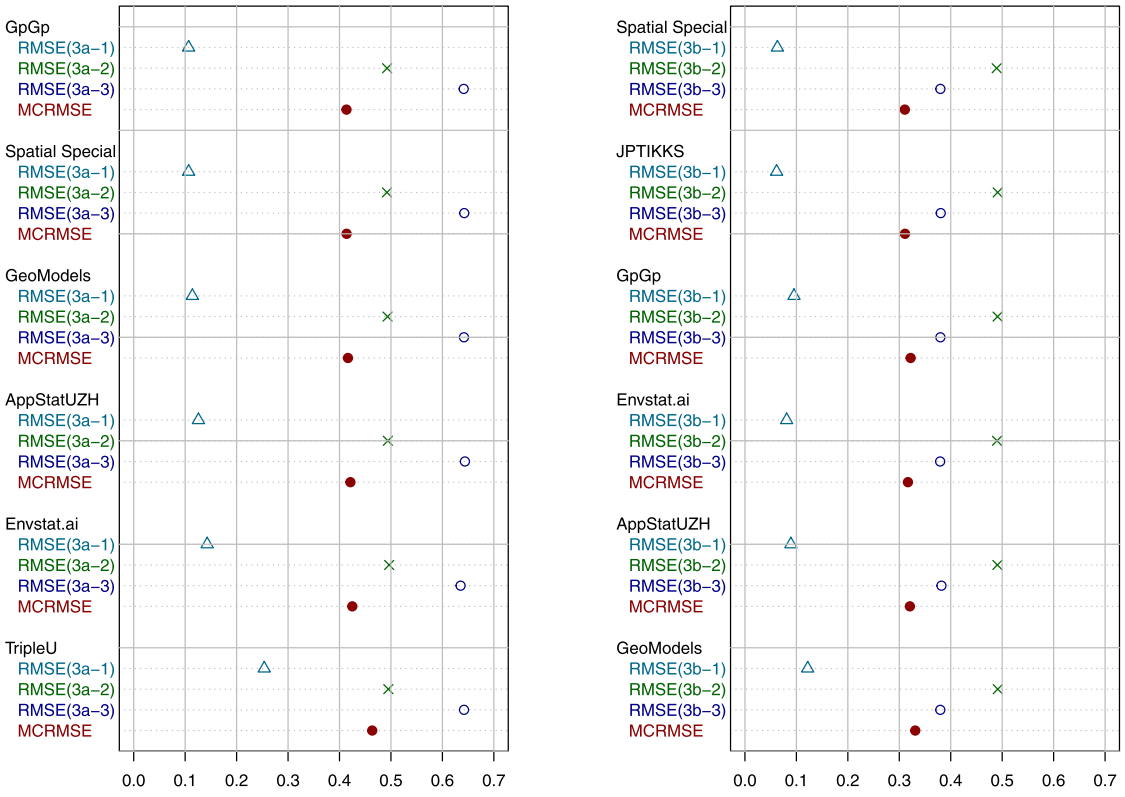

(a) 2a-1, 2a-2, and 2a-3 (RS) (b) 2a-4, 2a-5, and 2a-6 (RST) (c) 2a-7, 2a-8, and 2a-9 (T10)




(d) 2b-1, 2b-2, and 2b-3 (RS) (e) 2b-4, 2b-5, and 2b-6 (RST) (f) 2b-7, 2b-8, and 2b-9 (T10)

Figure 7: Sub-competitions 2a and 2b individual leaderboards where we combine datasets with the same scenario of leaving out missing points for prediction (i.e., RS, RST, and T10).

to this sub-competition, i.e., `Spatial Special`, `JPTIKKS`, `GpGp`, `Envstat.ai`, `AppStatZH`, and `GeoModels`. Although the main change in 3b compared to 3a is the sizes of the datasets, the ranks of the teams changed in 3b vs 3a. The `Spatial Special` team ranked first and the `GpGp` team ranked second in 3b, compared to third and first in 3a, respectively. The difference in the MCRMSE was very small, and there is no special observation from the final results for individual datasets.




(a) 3a (b) 3b

Figure 8: Sub-competitions 3a and 3b leaderboard.

# 5 Methods Used in the Competition: A Closer Look

The participants have relied on different methods to tackle the various sub-competitions. In this section, we comment on the performance of the top-ranked teams and the strategies they adopted based on the description they submitted to us.

## 5.1 Sub-competition 1a (Univariate Nonstationary Spatial, 100K)

Sub-competition 1a included two 100K nonstationary datasets: one was generated using a predefined deterministic mean function and the other using a nonstationary covariance function under certain settings. The `RESSTE` team ranked first in this sub-competition. The team applied the Vecchia composite likelihood approximation method (Vecchia, 1988; Katzfuss et al., 2020; Katzfuss and Guinness, 2021) to model the datasets in 1a. The Vecchia approximation method generally depends on a sparse covariance matrix instead of the usual dense matrix to allow faster computation of the matrix inverse operation. The `RESSTE` team used the *GpGp* R package (Guinness et al., 2018) to implement their model for datasets 1a-1 and 1a-2. For 1a-1, the team realized the nonstationarity of the given data and decided to split the region into two parts, i.e., upper and lower, by the line $s_x + s_y = 0.85$. The intercept 0.85 was chosen by experimenting

with several options and selecting the one with the lowest mean square prediction error values on one part of the training dataset. They used an anisotropic Matérn covariance function in the upper area, while an isotropic Matérn function was chosen for the lower area. To apply the Vecchia approximation to 1a-1, the team set the number of neighbors to 10 for estimation and to 500 for prediction. For 1a-2, an isotropic Matérn covariance function was used for the whole region without splitting. Herein, the team set the number of neighbors to 50 for estimation and to 70 for prediction.

The `GeoModels` team ranked second in Sub-competition 1a. The team applied the nearest neighbor weighted composite likelihood method to model both datasets in 1a. The analysis and the results were obtained using the *GeoModels* package (Bevilacqua et al., 2018). In 1a, the team considered Gaussian random fields with a geometrically anisotropic Matérn correlation model and Tukey non-Gaussian random fields.

### 5.2 Sub-competition 1b (Univariate Nonstationary Spatial, 1M)

Sub-competition 1b had two 1M datasets: 1b-1 was generated using a deterministic mean function, while 1b-2 was generated using a nonstationary covariance function in (4).

The `Spatial Special` team ranked first in this sub-competition. The team was able to find that dataset 1b-1 is nonstationary, and they adopted a deformation technique to map the locations from the original space to a latent space as follows: $r = \sqrt{s_x + s_y}$ and $\theta = \tan^{-1}(\sqrt{s_y/s_x})$. This deformation allowed the team to assume the data are stationary in the latent space.

The prediction was performed using a deep feed-forward neural network model with two inputs (i.e., $r$ and $\theta$), four hidden layers (i.e., 50, 50, 20, and 20 neurons, respectively), and one output (i.e., MSPE). The objective function was constructed as follows:

$$MSPE = \frac{1}{N}\sum_{i=1}^{N}\{\hat{Z}(r_i, \theta_i) - Z(r_i, \theta_i)\}^2,$$

where $\hat{Z}(r_i, \theta_i)$ is the predicted value, $Z(r_i, \theta_i)$ is the true value at the location $(r_i, \theta_i)$, the activation function was the hyperbolic tangent function, and the optimizer was the Adam optimizer (Kingma and Ba, 2015). All the experiments were performed using TensorFlow 1.15.

In 1b-2, the `Spatial Special` team followed a different strategy since it assumed that the data are stationary. The team fit the data with a Gaussian process, taking a constant mean $\mu$ and a Matérn covariance function with nugget $\tau^2$. Parameter estimation was done using the *GpGp* package with 10-neighbors approximation, then the 30-neighbors approximation was maximized using the 10-neighbors estimates as starting values. For the prediction, 200-neighbors have been used to obtain predicted values. The parameter estimates are $\hat{\mu} = -0.8148$, $\hat{\sigma}^2 = 3.1046$, $\hat{\beta} = 0.0031$, $\hat{\nu} = 0.7250$, and $\hat{\tau} = 8.01 \times 10^{-5}$.

The `GeoModels` team was ranked second in Sub-competition 2b. They also applied the nearest neighbor weighted composite likelihood-based methods to model both datasets in 1b using the *GeoModels* R package. The strategy of modeling and prediction was the same as in 1a.

The `RESSTE` team was ranked third in this sub-competition. They also applied the Vecchia approximation methods as in Sub-competition 1a. For dataset 1b-1, the data region was divided using the line $s_x + s_y = 0.85$ as in 1a-1. However, an anisotropic Matérn model was chosen for the lower region, and an isotropic Matérn model was chosen for the upper region. For Vecchia approximation, the number of neighbors was set to 10 and the estimation step to 500. For 1b-2, an anisotropic Matérn covariance function was used for the whole region without splitting. The team set the number of neighbors to 50 and the estimation step to 70.

### 5.3 Sub-competitions 2a and 2b (Univariate Stationary Space-time)

We combine two sub-competitions in this subsection since they have the same top-ranked teams adopting the same methods to deal with the datasets. Sub-competition 2a has nine 1K datasets in 100 time-lots, using three different settings. Sub-competition 2b has a larger size, 10K datasets in 100 time-lots. The description of the datasets in both sub-competitions is shown in Section 3.2.

The `Envstat.ai` was ranked first in Sub-competitions 2a and 2b. The `Envstat.ai` relied on a Deep Neural Network (DNN) for spatio-temporal predictions, an extension of the spatial version in Chen et al. (2022). They used basis functions to capture the spatio-temporal dependence. For interpolation, they relied on regression, whereas they used a 2-stage modeling approach for forecasting at new locations. First, they interpolated at the new locations for the existing time points. Then they trained Long-Short Term Memory Network (LSTM) on these points to get the forecast for future time points. The results obtained by the `Envstat.ai` team show a considerable improvement compared to the following ranked teams in this sub-competition.

The `GpGp` team was ranked second in Sub-competitions 2a and 2b. The team initially fit the Matérn space-time covariance function in the *GpGp* package, i.e., *matern_spacetime(args)* to estimate the model parameters. Then, they used the two estimated range parameters to rescale the coordinates to select the ordering and neighbors for Vecchia's approximation. They relied on 30-neighbors to fit the model for prediction purpose. The team used the prediction function in the *GpGp* package, i.e., *predictions(args)*, where the number of selected neighbors was $m = 60$.

The `RESSTE` team was ranked third in both sub-competitions. The team performed an exploratory data analysis step on the given data using a space-time covariance function and found that the data were generated from a positively non-separable space-time kernel. Based on this finding, the team chose to rely on a covariance function from the Gneiting class, with a Matérn spatial covariance and a Cauchy temporal covariance. Due to the size of the given datasets, the team used a block-composite likelihood to estimate the spatial and temporal parameters separately. Then, the team used the estimated parameters to fit the space-time model and estimate the full set of parameters. To predict the missing values, the `RESSTE` team applied ordinary kriging conditioned on the 1/9 nearest data points in space and 3 preceding, current, and 3 following time-slots in time for 2a, and 1/100 nearest data points in space and 2 preceding, and 3 following time-slots in time for 2b. The team applied slight changes to datasets 7, 8, and 9 in 2a and 2b by using 20 time-slots and 10 time-slots for 2a and 2b, respectively.

### 5.4 Sub-competition 3a (Bivariate Stationary Spatial, 50K)

Sub-competition 3a included three bivariate spatial datasets, i.e., 3a-1, 3a-2, and 3a-3. Dataset 3a-1 was generated using the parsimonious Matérn cross-covariance function in (7). Datasets 3a-2 and 3a-3 were generated using the flexible Matérn cross-covariance function in (8). The size of each dataset was 50K.

The `GpGp` team was ranked first in Sub-competition 3a. The team relied on the flexible bivariate Matérn model in Apanasovich et al. (2012) to fit the data in 3a-1, 3a-2, and 3a-3. It applied the Fisher scoring algorithm (Guinness, 2021) to perform the optimization of the likelihood function through Vecchia's approximation method. The team used the *GpGp* package to perform the modeling and the prediction tasks using 30-neighbors and 100-neighbors, respectively.

The `Spatial Special` team was ranked second in this sub-competition. The team considered $Z_1$ and $Z_2$ as individual variables and modeled them separately and not jointly. The team used the Matérn covariance function and its variogram to fit all the datasets in 3a (except

for the $Z_2$ variable in 3a-1) by weighted least squares for parameter estimation. It defined the weights $n_h/h^2$, where $n_h$ is the number of point pairs and $h$ is the distance. The team used the *gstat* package (Pebesma, 2004) to implement their method. For 3a-1 ($Z_2$), the team relied on the same method it used in 1b-2, as described in Section 5.3. The only change is the number of considered neighbors when performing the prediction, for which the team used 100-neighbors instead of 200.

The `GeoModels` team was ranked third in this sub-competition. They applied the methods of the weighted composite likelihood based on pairs, where the weight function was based on the spatial nearest neighbors. The covariance function was assumed to be the bivariate isotropic Matérn covariance function. The prediction was performed using local kriging.

### 5.5 Sub-competition 3b (Bivariate Stationary Spatial, 500K)

Sub-competition 3b included three bivariate spatial datasets, i.e., 3b-1, 3b-2, and 3b-3. Dataset 3b-1 was generated using the parsimonious Matérn cross-covariance function in (7). Datasets 3b-2 and 3b-3 were generated using the flexible Matérn cross-covariance function in (8). The size of each dataset was 500K.

The `Spatial Special` team was ranked first. For 3b-1 ($Z_1$) and 3b-1 ($Z_2$), the team used the same strategy as in 1b, with the number of nearest neighbors being 100 for prediction. For 3b-2 ($Z_1$), 3b-2 ($Z_2$), 3b-3 ($Z_1$), and 3b-3 ($Z_2$), the team divided the prediction region into 100 smaller prediction subregions. The team also used the ordinary kriging method to predict the missing data in each subregion independently. The missing values at the intersection between subregions were obtained by averaging the predicted values. The team fit the data using the Matérn covariance function and its variogram. The weighted least squares method was used for parameter estimation where the weights were again $n_h/h^2$. The modeling and the prediction were implemented using the *gstat* R package.

The `JPTIKKS` team was ranked second in this sub-competition. The team used covariance tapering with a bivariate Matérn covariance function to estimate the parameters for the first dataset, 3b-1. For datasets 3b-2 and 3b-3, they adopted the sparse version of the scalable Geographically Weighted Regression (GWR) method (Murakami et al., 2020) with the help of *scgwr* R package to model the data. The sparsity helped in dealing with large datasets and reduced the computation complexity. Local regression was used to estimate spatially varying intercepts. The kernel was considered as a linear combination of known exponential sub-kernels. Each sub-kernel used 100-neighbors to estimate the parameter vector. The team used the leave-one-out cross-validation method for estimation.

The `GpGp` team was ranked third in this sub-competition. The team fit the flexible bivariate Matérn model to the data and used Vecchia's approximation through the *GpGp* package. They chose 30-neighbors when fitting the model and 100-neighbors to perform predictions.

## 6 Discussion

This work proposed a framework for designing and assessing a competition for predicting missing values in large spatial and spatio-temporal datasets. Thanks to the *ExaGeoStat* software, very large datasets with various settings were generated from popular spatial and spatio-temporal models with exact computations. We have made all the datasets publicly available online (http://dx.doi.org/10.25781/KAUST-4ADYZ) for future assessments of other existing or new methods as needed. This work also serves as a reference for comparing new results with the

results obtained in this study. The competition covered three types of data: nonstationary spatial data, stationary space-time data, and bivariate stationary spatial data. We have reviewed the methods used by each participating team and ranked their performances based on prediction accuracy. The ranks provide a fair comparison among these teams, which may shed light on developing even better prediction methods.

In the process of evaluating the performance of different teams, we noticed the two top performers in Sub-competitions 2a and 2b, outperforming other teams by a large margin. After contacting them, we learned that both teams had combined several datasets from the same sub-competition to perform predictions. Since the same model actually generated different datasets in each sub-competition, the prediction accuracy was significantly improved by combing datasets. Therefore, we decided to exclude both teams from the leaderboard of Sub-competition 2 for fair comparisons. We believe this was an unintended flaw in the competition and such data-generating problems should be avoided in future competitions.

For the second year, we had another successful competition with a wonderful response from the community. We believe that conducting such competitions provides opportunities for researchers to assess their methods on the same large datasets that are generated by exact computations. These datasets are valuable to the community, and we have made them publicly available. We also believe it is worthwhile to build benchmarking tools to assess the performance of existing approximation methods, which will significantly help to better understand the advantages and disadvantages of these methods. Although we focused on point forecasts, there is value in probabilistic forecasting for uncertainty quantification purpose and we plan to explore this topic in a future competition.

## Supplementary Material

In the Supplementary Material, we list the members of all the teams participating in this competition in Table S1. Moreover, Tables S2 to S11 summarize the RMSE values obtained by different teams in each dataset of different sub-competitions, as well as those obtained with *ExaGeoStat* for reference purpose.

## Acknowledgement

We want to thank the Supercomputing Laboratory (KSL) for providing computational resources to this project on the Shaheen-II Cray XC40 Supercomputer. Finally, the authors would like to thank the KSL team for their valuable help in running the experiments in this publication.

## Funding

The research in this manuscript was funded by the King Abdullah University of Science and Technology (KAUST) in Thuwal, Saudi Arabia. We want to thank the Supercomputing Laboratory (KSL) at KAUST (https://www.hpc.kaust.edu.sa/) for supporting this research by providing the hardware resources, including the Shaheen-II Cray XC40 supercomputer used to generate the datasets in this competition.

## References


Abdulah S, Cao Q, Pei Y, Bosilca G, Dongarra J, Genton MG, et al. (2021). Accelerating geostatistical modeling and prediction with mixed-precision computations: A high-productivity approach with parsec. *IEEE Transactions on Parallel and Distributed Systems*, 33(4): 964–976.

Abdulah S, Ltaief H, Sun Y, Genton MG, Keyes DE (2018a). ExaGeoStat: A high performance unified software for geostatistics on manycore systems. *IEEE Transactions on Parallel and Distributed Systems*, 29(12): 2771–2784.

Abdulah S, Ltaief H, Sun Y, Genton MG, Keyes DE (2018b). Parallel approximation of the maximum likelihood estimation for the prediction of large-scale geostatistics simulations. In: *2018 IEEE International Conference on Cluster Computing (CLUSTER)*, 98–108. IEEE.

Abdulah S, Ltaief H, Sun Y, Genton MG, Keyes DE (2019). Geostatistical modeling and prediction using mixed precision tile Cholesky factorization. In: *2019 IEEE 26th International Conference on High Performance Computing, Data, and Analytics (HiPC)*, 152–162. IEEE.

Apanasovich TV, Genton MG, Sun Y (2012). A valid Matérn class of cross-covariance functions for multivariate random fields with any number of components. *Journal of the American Statistical Association*, 107(497): 180–193.

Ba S, Joseph VR (2012). Composite Gaussian process models for emulating expensive functions. *The Annals of Applied Statistics*, 6: 1838–1860.

Bevilacqua M, Morales-Oñate V, Caamaño-Carrillo C (2018). *GeoModels: Procedures for Gaussian and Non Gaussian Geostatistical (Large) Data Analysis*. R package version 1.0.0.

Bradley JR, Cressie N, Shi T (2016). A comparison of spatial predictors when datasets could be very large. *Statistics Surveys*, 10: 100–131.

Cao Q, Abdulah S, Alomairy R, Nag P, Pei Y, Bosilca G, et al. (2022). Reshaping geostatistical modeling and prediction for extreme-scale environmental applications. In: *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis*.

Chen W, Li Y, Reich BJ, Sun Y (2022). DeepKriging: Spatially dependent deep neural networks for spatial prediction. *Statistica Sinica*, to appear.

Englund EJ (1990). A variance of geostatisticians. *Mathematical Geology*, 22(4): 417–455.

Gneiting T (2002). Nonseparable, stationary covariance functions for space-time data. *Journal of the American Statistical Association*, 97(458): 590–600.

Gneiting T, Kleiber W, Schlather M (2010). Matérn cross-covariance functions for multivariate random fields. *Journal of the American Statistical Association*, 105(491): 1167–1177.

Guinness J (2021). Gaussian process learning via Fisher scoring of Vecchia's approximation. *Statistics and Computing*, 31(3): 1–8.

Guinness J, Katzfuss M, Fahmy Y (2018). GpGp: fast Gaussian process computation using Vecchia's approximation. *R package version 0.1.0*.

Heaton MJ, Datta A, Finley A, Furrer R, Guhaniyogi R, Gerber F, et al. (2019). A case study competition among methods for analyzing large spatial data. *Journal of Agricultural, Biological and Environmental Statistics*, 24: 398–425.

Hong Y, Abdulah S, Genton MG, Sun Y (2021). Efficiency assessment of approximated spatial predictions for large datasets. *Spatial Statistics*, 43: 100517.

Huang H, Abdulah S, Sun Y, Ltaief H, Keyes DE, Genton MG (2021). Competition on spatial statistics for large datasets (with discussion). *Journal of Agricultural, Biological and Environmental Statistics*, 26(4): 580–595.

Katzfuss M, Guinness J (2021). A general framework for Vecchia approximations of Gaussian processes. *Statistical Science*, 36(1): 124–141.

Katzfuss M, Guinness J, Gong W, Zilber D (2020). Vecchia approximations of Gaussian-process predictions. *Journal of Agricultural, Biological and Environmental Statistics*, 25(3): 383–414.

Kingma DP, Ba J (2015). Adam: A method for stochastic optimization. In: *International Conference on Learning Representations*, San Diego.

Li J, Heap AD (2011). A review of comparative studies of spatial interpolation methods in environmental sciences: Performance and impact factors. *Ecological Informatics*, 6(3–4): 228–241.

Li J, Heap AD (2014). Spatial interpolation methods applied in the environmental sciences: A review. *Environmental Modelling & Software*, 53: 173–189.

Li Y, Sun Y (2019). Efficient estimation of nonstationary spatial covariance functions with application to high-resolution climate model emulation. *Statistica Sinica*, 29(3): 1209–1231.

Mondal S, Abdulah S, Ltaief H, Sun Y, Genton MG, Keyes DE (2022). Parallel approximations of the Tukey g-and-h likelihoods and predictions for non-Gaussian geostatistics. In: *International Parallel and Distributed Processing Symposium*, 379–389.

Murakami D, Tsutsumida N, Yoshida T, Nakaya T, Lu B (2020). Scalable gwr: A linear-time algorithm for large-scale geographically weighted regression with polynomial kernels. *Annals of the American Association of Geographers*, 111(2): 459–480.

Nesi L, Legrand A, Mello Schnorr L (2021). Exploiting system level heterogeneity to improve the performance of a geostatistics multi-phase task-based application. In: *50th International Conference on Parallel Processing*, 1–10.

Nesi L, Schnorr LM, Legrand A (2022). Multi-phase task-based HPC applications: Quickly learning how to run fast. In: *IPDPS 2022 – 36th IEEE International Parallel & Distributed Processing Symposium*.

Pebesma EJ (2004). Multivariable geostatistics in S: The gstat package. *Computers & Geosciences*, 30: 683–691.

Salvaña MLO, Abdulah S, Huang H, Ltaief H, Sun Y, Genton MG, et al. (2021). High performance multivariate geospatial statistics on manycore systems. *IEEE Transactions on Parallel and Distributed Systems*, 32(11): 2719–2733.

Salvaña MLO, Abdulah S, Ltaief H, Sun Y, Genton MG, Keyes DE (2022). Parallel space-time likelihood optimization for air pollution prediction on large-scale systems. In: *Platform for Advanced Scientific Computing Conference (PASC'22)*, 1–11. Basel, Switzerland, Article No. 17.

Shahbeik S, Afzal P, Moarefvand P, Qumarsy M (2014). Comparison between ordinary kriging (OK) and inverse distance weighted (IDW) based on estimation error. Case study: Dardevey iron ore deposit, NE Iran. *Arabian Journal of Geosciences*, 7(9): 3693–3704.

Srivastava RM (1987). A non-ergodic framework for variograms and covariance functions, Master's thesis, Stanford University, Stanford, California.

Vecchia AV (1988). Estimation and model identification for continuous spatial processes. *Journal of the Royal Statistical Society: Series B (Methodological)*, 50(2): 297–312.

Vu Q, Cao Y, Jacobson J, Pearse AR, Zammit-Mangion A (2021). Discussion on "Competition on spatial statistics for large datasets". *Journal of Agricultural, Biological and Environmental Statistics*, 26(4): 614–618.

Weber D, Englund E (1992). Evaluation and comparison of spatial interpolators. *Mathematical Geology*, 24(4): 381–391.

Wikle CK, Cressie N, Zammit-Mangion A, Shumack C (2017). A common task framework (CTF) for objective comparison of spatial prediction methodologies. *Statistics Views*.

Xiong Y, Chen W, Apley D, Ding X (2007). A non-stationary covariance-based kriging method for metamodelling in engineering design. *International Journal for Numerical Methods in Engineering*, 71(6): 733–756.