# KAUST Competition on Spatial Statistics for Large Datasets

Huang Huang, Sameh Abdulah, Marc G. Genton,
Ying Sun, Hatem Ltaief, and David E. Keyes

November, 2020

## 1 Introduction

With the development of observing techniques and computing devices, it has become easier and more common to obtain large datasets. Statistical inference in spatial statistics becomes computationally challenging. For decades, various approximation methods have been proposed to model and analyze large-scale spatial data when the exact computation is infeasible. However, in the literature, the performance of the statistical inference using those proposed approximation methods was usually assessed with small and medium datasets only, for which the exact solution can be obtained. Then, for real-world large datasets, the exact computation was no longer feasible. The inference with approximation methods was often validated empirically or via prediction accuracy with the fitted model.

In this competition, the goal is to reassess existing approximation methods on large spatial datasets in a uniform way that guarantees a fair comparison. The results will be compared to the exact solution provided by the *ExaGeoStat* [1] software (`https://github.com/ecrc/exageostat`). We generated a collection of synthetic datasets on a large scale from a set of selected true models. We aim at validating the statistical performance of the state-of-the-art approximation methods in terms of modeling, inference, and prediction. The selected true models cover disparate spatial properties to ensure a fair comparison among all the competitors' methods.

# 2 Datasets

This competition focuses on two parts. The first part is to infer the parameters of the spatial covariance of Gaussian random fields and then to make predictions at new locations. The second part solely focuses on making predictions at new locations without restrictions of Gaussian process models.

## 2.1 Model inference and prediction for Gaussian random fields

### 2.1.1 Parameter estimation (Sub-competition 1a)

We have generated 16 datasets from different zero-mean stationary isotropic Gaussian random fields with a Matérn covariance using *ExaGeoStat*. The spatial domain is the unit square $[0,1] \times [0,1]$ in Euclidean space. The training dataset consists of 90,000 location coordinates and associated values. **The participant team is asked to provide the estimated values of the four parameters in the Matérn covariance function shown in Equation (1) (the partial sill $\sigma^2$, the range $\beta > 0$, the smoothness $\nu > 0$, and the nugget $\tau^2$) for each dataset.**

$$\operatorname{cov}\{Z(\boldsymbol{s}_i), Z(\boldsymbol{s}_j)\} = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \Big( \frac{\|\boldsymbol{s}_i - \boldsymbol{s}_j\|}{\beta} \Big)^{\nu} K_{\nu} \Big( \frac{\|\boldsymbol{s}_i - \boldsymbol{s}_j\|}{\beta} \Big) + \tau^2 \mathbb{1}_{\{i=j\}}, \tag{1}$$

where $\operatorname{cov}\{Z(\boldsymbol{s}_i), Z(\boldsymbol{s}_j)\}$ is the Matérn covariance between realizations of $Z(\cdot)$ at locations $\boldsymbol{s}_i$ and $\boldsymbol{s}_j$, $K_{\nu}(\cdot)$ is the modified Bessel function of the second kind of order $\nu$, $\Gamma(\cdot)$ is the Gamma function, and $\mathbb{1}$ is the indicator function.

### 2.1.2 Prediction (Sub-competition 1b)

For each of the 16 training datasets, we also provide the corresponding testing dataset. Each of the 16 testing datasets consists of 10,000 new location coordinates. **The participant team is asked to make predictions at the 10,000 locations.**

1. **If the participant team also participates in Sub-competition 1a, the prediction is then asked to be made based on the inferred model in Sub-competition 1a.**

2. **If the participant team does not participate in Sub-competition 1a and the adopted method cannot provide parameter estimation in Sub-competition 1a, then the participant team is free to choose any models or algorithms for making the prediction.**

### 2.2 Prediction for random fields (Sub-competitions 2a and 2b)

We provide two datasets generated from different random fields in each of Sub-competitions 2a and 2b. The two datasets in Sub-competitions 2b are bigger in size. The spatial domain is still the unit square $[0,1] \times [0,1]$ in Euclidean space.

In Sub-competitions 2a, for each dataset, the training data consist of 90,000 location coordinates and associated values, and the testing data are for 10,000 new location coordinates. **The participant team is asked to make predictions at the 10,000 locations based on their choice of models or algorithms.**

In Sub-competitions 2b, for each dataset, the training data consist of 900,000 location coordinates and associated values, and the testing data are for 100,000 new location coordinates. **The participant team is asked to make predictions at the 100,000 locations based on their choice of models or algorithms.**

## 3 Assessment

### 3.1 Assessment of methods for Sub-competition 1a

Mean Loss of Efficiency (MLOE) and Mean Misspecification of the Mean Square Error (MMOM) [2] are used to assess the quality of the estimated model. MLOE characterizes the average loss of prediction efficiency when the approximated model is used to predict instead of the true model. MMOM characterizes the average misspecification of the mean square error when calculated under the approximated model. Details of these two metrics calculation can be found in the Appendix.

Assuming that we have $K^{(1a)}$ participant teams for Sub-competition 1a in the end, let

$P_{ki1}, P_{ki2}$, $k = 1, \ldots, K^{(1a)}, i = 1, ..., 16$, denote the MLOE and MMOM from team $k$ for dataset $i$, respectively. Then, for each dataset $i$ and metric $j = 1, 2$, we sort $P_{kij}$, $k = 1, \ldots, K^{(1a)}$ by the absolute values in ascending order and assign rank $R_{kij}^{(1a)}$ to each team (an averaged rank is used for ties).

The final score for team $k$ in Sub-competition 1a is calculated as $S_k^{(1a)} = \sum_{i=1}^{16} \left( R_{ki1}^{(1a)} + R_{ki2}^{(1a)} \right)$, and the final rank is assigned by sorting $S_k^{(1a)}$ in ascending order.

## 3.2 Assessment of methods for Sub-competition 1b

The Root Mean Square Error (RMSE) is used to evaluate the prediction accuracy,

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

where $\hat{Z}(\boldsymbol{s}_i)$ and $Z(\boldsymbol{s}_i)$ are respectively the predicted and true realization values at the prediction location $\boldsymbol{s}_i$ in the testing dataset, and $N_{\text{test}}$ is the total number of locations in the testing dataset.

Assuming that we have $K^{(1b)}$ participant teams for Sub-competition 1b in the end, let $\text{RMSE}_{ki}^{(1b)}$, $k = 1, \ldots, K^{(1b)}, i = 1, ..., 16$, denote the RMSE from team $k$ for dataset $i$. For each dataset $i$, we sort $\text{RMSE}_{ki}^{(1b)}$, $k = 1, \ldots, K^{(1b)}$ in ascending order and assign rank $R_{ki}^{(1b)}$ to each team (an averaged rank is used for ties).

The final score for team $k$ in Sub-competition 1b is calculated as $S_k^{(1b)} = \sum_{i=1}^{16} R_{ki}^{(1b)}$, and the final rank is assigned by sorting $S_k^{(1b)}$ in ascending order.

## 3.3 Assessment of methods for Sub-competition 2a

The Root Mean Square Error (RMSE) is used to evaluate the prediction accuracy. Assuming that we have $K^{(2a)}$ participant teams for Sub-competition 2a in the end, let $\text{RMSE}_{ki}^{(2a)}$, $k = 1, \ldots, K^{(2a)}, i = 1, 2$, denote the RMSE from team $k$ for all the 10,000 testing data points in dataset $i$. For each dataset $i$, we sort $\text{RMSE}_{ki}^{(2a)}$, $k = 1, \ldots, K^{(2a)}$ in ascending order and assign rank $R_{ki}^{(2a)}$ to each team (an averaged rank is used for ties).

The final score for team $k$ in Sub-competition 2a is calculated as $S_k^{(2a)} = R_{k1}^{(2a)} + R_{k2}^{(2a)}$, and the final rank is assigned by sorting $S_k^{(2a)}$ in ascending order.

### 3.4 Assessment of methods for Sub-competition 2b

The Root Mean Square Error (RMSE) is used to evaluate the prediction accuracy. Assuming that we have $K^{(2b)}$ participant teams for Sub-competition 2b in the end, let $\text{RMSE}_{ki}^{(2b)}$, $k = 1, \ldots, K^{(2b)}, i = 1, 2$, denote the RMSE from team $k$ for all the 100,000 testing data points in dataset $i$. For each dataset $i$, we sort $\text{RMSE}_{ki}^{(2b)}$, $k = 1, \ldots, K^{(2b)}$ in ascending order and assign rank $R_{ki}^{(2b)}$ to each team (an averaged rank is used for ties).

The final score for team $k$ in Sub-competition 2b is calculated as $S_k^{(2b)} = R_{k1}^{(2b)} + R_{k2}^{(2b)}$, and the final rank is assigned by sorting $S_k^{(2b)}$ in ascending order.

## 4 Rules

- Teams from any background are welcome to participate. Participant teams can choose to participate in one or more sub-competitions among Sub-competitions 1a, 1b, 2a, and 2b. Separate rankings will be used for these four sub-competitions.
- In each sub-competition, all the required results need to be submitted before the deadline to secure a rank.
- Each team is allowed and encouraged to have more than one submission if different methods are used to solve the given problem.
- The execution time will not be used to rank the teams in this competition. It only provides some insights into the method's computational efficiency.

## 5 Results

The ranks of participant teams will be announced on our KAUST web page. One representative member from the top-ranked team in each sub-competition will be invited to KAUST

to present their work in a workshop dedicated to this competition when the COVID-19 situation is better and travel is possible.

# 6 Getting Started

Please accept or reject the invitation for this competition by filling in the Registration Google Form (`https://bit.ly/3pGdyBS`) by December 15, 2020. The datasets links will be sent to each registered team via e-mail after the registration (but no earlier than November 23, 2020).

# 7 Timeline

All submissions should be received by 11:59pm (UTC±00:00) January 15, 2021.

# 8 Deliverables

Submission examples can be found at `https://bit.ly/32JYObi`. All required files should be uploaded to Google drive via the Results Google Form (`https://bit.ly/2KminAb`). The required files are:

- A text description file "TeamName-Description.txt", showing the participating sub-competitions and briefly describing the adopted methods. If the adopted methods include priors, hyper-parameters, tuning parameters, etc., an explicit description of their choices should be provided. If data preprocessing is used, it should also be explained.
- Other files for each sub-competition described in the following sections.

## 8.1 Sub-competition 1a

- One "TeamName-1a-hardware.txt" text file is needed explaining the hardware platform (CPU, memory, etc.) that is used for Sub-competition 1a.

- One "TeamName-1a.csv" file (delimited by commas) is needed with 16 rows and 5 columns. One "TeamName-1a.csv" file (delimited by commas) is needed with 16 rows and 5 columns, and an additional header row. The header row is fixed as "sigma squared, beta, nu, tau squared, time in seconds". Then, row $i$ reports the estimated $\hat{\sigma}^2, \hat{\beta}, \hat{\nu}, \hat{\tau}^2$, and the execution time (in seconds), for dataset $i$. Each value should be reported with six digits after the decimal point.

### Sub-competition 1b

- One "TeamName-1b-hardware.txt" text file is needed indicating the hardware platform (CPU, memory, etc.) that is used for Sub-competition 1b.

- One "TeamName-1b-time.csv" file (delimited by commas) with 16 rows and 1 column, and an additional header row is needed. The header row is fixed as "time in seconds". Then, row $i$ reports the total execution time (in seconds, with six digits after the decimal point) of prediction at the 10,000 new locations for dataset $i$.

- Sixteen files are needed with names "TeamName-1b-1.csv",..., "TeamName-1b-16.csv" (delimited by commas), where file "TeamName-1b-$i$.csv" contains the prediction results for the dataset $i$. Each file has 10,000 rows and 3 columns, and an additional header row. The header row is fixed as "x, y, predicted values". Then, each row reports the $x$ and $y$ coordinates of the prediction locations and the predicted values. Each value should be reported with six digits after the decimal point.

## 8.2 Sub-competition 2a

- One "TeamName-2a-hardware.txt" text file is needed indicating the hardware platform (CPU, memory, etc.) that is used for Sub-competition 2a.

- One "TeamName-2a-time.csv" (delimited by commas) file with 2 rows and 1 column, and an additional header row is needed. The header row is fixed as "time in seconds". Then, row $i$ reports the total execution time (in seconds, with six digits after the decimal point) of prediction at the 10,000 new locations for dataset $i$.

- Two files are needed with names "TeamName-2a-1.csv" and "TeamName-2a-2.csv" (delimited by commas), where file "TeamName-2a-$i$.csv" contains the prediction results for the dataset $i$. Each file has 10,000 rows and 3 columns, and an additional header row. The header row is fixed as "x, y, predicted values". Then, each row reports the $x$ and $y$ coordinates of the prediction locations and the predicted values. Each value should be reported with six digits after the decimal point.

### Sub-competition 2b

- One "TeamName-2b-hardware.txt" text file is needed indicating the hardware platform (CPU, memory, etc.) that is used for Sub-competition 2b.
- One "TeamName-2b-time.csv" (delimited by commas) file with 2 rows and 1 column, and an additional header row is needed. The header row is fixed as "time in seconds". Then, row $i$ reports the total execution time (in seconds, with six digits after the decimal point) of prediction at the 100,000 new locations for dataset $i$.
- Two files are needed with names "TeamName-2b-1.csv" and "TeamName-2b-2.csv" (delimited by commas), where file "TeamName-2b-$i$.csv" contains the prediction results for the dataset $i$. Each file has 100,000 rows and 3 columns, and an additional header row. The header row is fixed as "x, y, predicted values". Then, each row reports the $x$ and $y$ coordinates of the prediction locations and the predicted values. Each value should be reported with six digits after the decimal point.

# 9 Contact

If you have any questions about this competition, you can contact us at kaustcompspat@gmail.com.

# References

[1] Sameh Abdulah, Hatem Ltaief, Ying Sun, Marc G Genton, and David E Keyes. Exageostat: A high performance unified software for geostatistics on manycore systems.

*IEEE Transactions on Parallel and Distributed Systems*, 29(12):2771–2784, 2018.

[2] Yiping Hong, Sameh Abdulah, Marc G Genton, and Ying Sun. Efficiency assessment of approximated spatial predictions for large datasets. *arXiv preprint arXiv:1911.04109*, 2019.

# Appendix

Let $\tilde{\boldsymbol{s}}_1, \ldots, \tilde{\boldsymbol{s}}_{100}$ denote 100 predetermined, randomly-chosen locations in the unit square. The MLOE and MMOM are calculated as follows,

$$\text{MLOE} = \frac{1}{100}\sum_{j=1}^{100} \text{LOE}(\tilde{\boldsymbol{s}}_j), \ \text{MMOM} = \frac{1}{100}\sum_{j=1}^{100} \text{MOM}(\tilde{\boldsymbol{s}}_j),$$

and

$$\begin{aligned}
\text{LOE}(\tilde{\boldsymbol{s}}_j) &= \frac{\mathbb{E}_t[\{\hat{Z}_a(\tilde{\boldsymbol{s}}_j) - Z(\tilde{\boldsymbol{s}}_j)\}^2]}{\mathbb{E}_t[\{\hat{Z}_t(\tilde{\boldsymbol{s}}_j) - Z(\tilde{\boldsymbol{s}}_j)\}^2]} - 1, \\
\text{MOM}(\tilde{\boldsymbol{s}}_i) &= \frac{\mathbb{E}_a[\{\hat{Z}_a(\tilde{\boldsymbol{s}}_j) - Z(\tilde{\boldsymbol{s}}_j)\}^2]}{\mathbb{E}_t[\{\hat{Z}_a(\tilde{\boldsymbol{s}}_j) - Z(\tilde{\boldsymbol{s}}_j)\}^2]} - 1,
\end{aligned}$$

where $\hat{Z}_t(\tilde{\boldsymbol{s}}_j)$ and $\hat{Z}_a(\tilde{\boldsymbol{s}}_j)$ are respectively kriging prediction at $\tilde{\boldsymbol{s}}_j$ using the true and approximated model (plugging in the true parameters and estimated parameters in the covariance function), and $\mathbb{E}_t$ and $\mathbb{E}_a$ are respectively the expectation using the true and approximated model.

More specifically, let $\boldsymbol{z}$ denote the $90,000$-dimensional vector for the values in the training dataset; $\boldsymbol{k}_t(\tilde{\boldsymbol{s}}_j) = \text{cov}_t\big\{\boldsymbol{z}, Z(\tilde{\boldsymbol{s}}_j)\big\}$, $k_t(\tilde{\boldsymbol{s}}_j) = \text{cov}_t\big\{Z(\tilde{\boldsymbol{s}}_j), Z(\tilde{\boldsymbol{s}}_j)\big\}$, and $K_t = \text{cov}_t(\boldsymbol{z}, \boldsymbol{z})$ using the true model; $\boldsymbol{k}_a(\tilde{\boldsymbol{s}}_j) = \text{cov}_a\big\{\boldsymbol{z}, Z(\tilde{\boldsymbol{s}}_j)\big\}$, $k_a(\tilde{\boldsymbol{s}}_j) = \text{cov}_a\big\{Z(\tilde{\boldsymbol{s}}_j), Z(\tilde{\boldsymbol{s}}_j)\big\}$, and $K_a = \text{cov}_a(\boldsymbol{z}, \boldsymbol{z})$ using the approximated model. Then,

$$\begin{aligned}
\mathbb{E}_t[\{\hat{Z}_t(\tilde{\boldsymbol{s}}_j) - Z(\tilde{\boldsymbol{s}}_j)\}^2] &= k_t(\tilde{\boldsymbol{s}}_j) - \boldsymbol{k}_t(\tilde{\boldsymbol{s}}_j)^\top K_t^{-1} \boldsymbol{k}_t(\tilde{\boldsymbol{s}}_j), \\
\mathbb{E}_a[\{\hat{Z}_a(\tilde{\boldsymbol{s}}_j) - Z(\tilde{\boldsymbol{s}}_j)\}^2] &= k_a(\tilde{\boldsymbol{s}}_j) - \boldsymbol{k}_a(\tilde{\boldsymbol{s}}_j)^\top K_a^{-1} \boldsymbol{k}_a(\tilde{\boldsymbol{s}}_j), \\
\mathbb{E}_t[\{\hat{Z}_a(\tilde{\boldsymbol{s}}_j) - Z(\tilde{\boldsymbol{s}}_j)\}^2] &= k_t(\tilde{\boldsymbol{s}}_j) - 2\boldsymbol{k}_t(\tilde{\boldsymbol{s}}_j)^\top K_a^{-1} \boldsymbol{k}_a(\tilde{\boldsymbol{s}}_j) + \boldsymbol{k}_a(\tilde{\boldsymbol{s}}_j)^\top K_a^{-1} K_t K_a^{-1} \boldsymbol{k}_a(\tilde{\boldsymbol{s}}_j).
\end{aligned}$$