A Python library for simulating Gaussian random fields

Khaya Mpehle, Steven Maharaj

Here we will introduce a python library, `GaussRF`, concerned with simulating Gaussian Random fields in one-dimension and two-dimensional rectangular domains. The library will implement two sampling methods, one approximate and one exact. The Karhunen-Loève expansion, using the Nyström method to solve further numerical problems, is the approximate method, while the circulant embedding method is the exact sampling procedure. The document is structured as follows: a brief discussion on the theory random fields will start. The main part of the document will then introduce the key functions in `GaussRF` and give examples of their use. The document will finish by listing expansions to this library that should be prioritized.

1 Introductory theory

Let us denote by T some space, for example a two-dimensional rectangular domain $[a, b] \times [c, d]$, the surface of a sphere or the time interval $[t_{\min}, t_{\max}]$. A random field takes each member $t \in T$ of this space and assigns a random variable to it. A random field F_t therefore assigns a random variable to each point t in the index space T. If t is a one-dimensional index, then a random field reduces to a stochastic process(SP). The variations of topography would be modelled by a random field H_t on a two-dimensional domain $t \in \mathbb{R}^2$; Random fluctuations of the wind field across the Earth's surface would be modelled be a random field with its index set a subset of the surface of a sphere. In general then a random field is an extension of a stochastic process, allowing the index space T to be a manifold. A Gaussian random field X_t is one that has a Gaussian distribution for each $t \in T$.

It is our goal to produce a Python library for the numerical simulation of Gaussian random fields in one-dimensions and two-dimensional rectangular domains using two well established numerical techniques. The first is via truncation of the Karhunen-Loève(KL) expansion of the RF.

2 The Karhunen Loève approximation and the Circulant embedding method.

Let X_t be a zero mean Gaussian field, where $t \in T$, with covariance function

$$R(t, s) = E\left[X_t X_s\right]. \tag{1}$$

The Karhunen-Loève expansion theorem states that X_t can be expanded as

$$X_t = \sum_{k=1}^{\infty} \sqrt{\lambda_k} Z_k \phi_k(t). \tag{2}$$

In this expansion Z_k are iid standard normal variables $Z \sim N(0,1)$ while ϕ_k and λ_k are respectively the eigen-functions and -values of the RF's covariance function. Therefore ϕ_k and λ_k satisfy the eigenvalue problem

$$\int_{t\in T} R(t,s)\phi_k(s)ds = \lambda_k\phi_k(t). \tag{3}$$

The Karhunen-Loève expansion can roughly be thought of as "separating" the deterministic and random components of the Gaussian RF.

A means of approximating the RF X_t is to truncate its KL expansion after N terms. This is termed the Karhunen-Loève approximation of order N. With a finite number of terms in the expansion, the only remaining task is to find the eigenvalues and -functions of the convariance function. There are several ways to do this, here we shall focus on using the Nyström method[1,4]. The integral eigenvalue problem (3) is approximated by an M-point quadrature $\{w_i, x_i\}_{i=1}^M$, the resulting expression is considered on each of the quadrature points:

$$\sum_{i=1}^{M} w_i R(t_j, s_i)\phi_k(s_i) = \lambda_k\phi(t_j), \qquad j = 0, 1, 2, \ldots, M. \tag{4}$$

The discrete expression (4) reduces to the matrix eigenvalue problem

$$\boldsymbol{C}\mathbf{W}\boldsymbol{\phi}_\mathbf{k} = \lambda_\mathbf{k}\boldsymbol{\phi}_\mathbf{k} \tag{5}$$

where we have introduced the diagonal matrix of weights $\boldsymbol{W} = \mathrm{diag}\{w_1, \ldots, w_M\}$ and the covariance matrix $(\boldsymbol{C})_{ij} = R(t_i, t_j)$. Solving this eigenvalue problem will determine the eigenvectors and -values of the discretised system of (3) and allow the determination of the discrete KL approximation. We note as a practical consideration that (5) can be modified by left multiplying both sides by $\boldsymbol{W}^{1/2}$, such that the system

$$\boldsymbol{W}^{1/2}\boldsymbol{C}\boldsymbol{W}^{1/2}\boldsymbol{y}_k = \lambda_k\boldsymbol{y}_k, \tag{6}$$

where $\boldsymbol{y}_k = \boldsymbol{W}^{1/2}\boldsymbol{\phi}_k$, is obtained. This system is advantageous as $\boldsymbol{W}^{1/2}\boldsymbol{C}\boldsymbol{W}^{1/2}$ is a positive semi-definite symmetric matrix and therefore has orthogonal eigenvectors with positive,real eigen values.

If we further suppose X_t to be a stationary Gaussian random field, meaning its covariance function is of the form $R(s,t) = R(s-t)$, it may be preferable to use the *circulant embedding method*. The method is so-called because it exploits the fact that the covariance matrix of stationary RFs can be embedded into a larger circulant matrix. One can the use the Fast Fourier Transform(FFT) to compute the eigenvalues of the circulant matrix, and from there go on to simulate the RF. The details of the method are described in detail in [2,3]. What is desirable about the circulant embedding method is its generation of a sample that has the exact covariance structure of the underlying Gaussian RF, and its speed.

3 examples

We have written functions in Python, utilising the Numpy and Scipy libraries, to implement these numerical algorithms. The function `KL_1DNys` will return a one-dimensional array of simulated

values of a Gaussian RF X_t using the Karhunen-Loève in one-dimension. As an example, we will take Brownian motion whose covariance function is

$$R(s,t) = \min\{s,t\}. \tag{7}$$

Brownian motion is one of the few SPs whose Karhunen- Loeéve function can be calculated exactly. The eigenfunctions and -values are

$$\phi_k = \sqrt{2}\sin\left((k-\frac{1}{2})t\right), \qquad \lambda_k = \frac{1}{\pi^2(k-\frac{1}{2})^2}. \tag{8}$$

In figure 1, the eigenvalues of the Brownian motion process along with the first six eigenfunctions are compared to our EOLE Nystrm simulation. One can see good agreement between the exact eigenvalues and our numerical approximation.

The function `KL_2DNys` will return a two-dimensional array of simulated values of a Gaussian RF using the Karhunen-Loève expansion in two-dimensions. The example is the Gaussian RF with isotropic covariance

$$R(s,t) = \exp\left(-\frac{||s-t||}{\rho}\right) \tag{9}$$

where ρ is a scale radius. In figure 2, we see a plot of the eigenvalues of this covariance matrix along with the first 6 eigenfunctions simulated on the domain $[0,1]\times[0,1]$ with 50×50 points. Note that this is a small number of simulation points to be using, but this is all that the the relatively weak computers RAM allows to be used. This is a limitation of our current resources and a full convergence test with finer meshes, on a more powerful computer, is called for.

The function `circ_embed1D` will return a one-dimensional array containing the simulated Gaussian SP. Note that the sample size will need to be a power of two so that the speed of the FFT can be exploited. In figure 3 we plot a realisation of the exponential SP with the isotropic covariance function

$$R(s,t) = R(|s-t|) = \exp\left(\frac{|s-t|}{l}\right) \tag{10}$$

on the domain $[-5,5]$.

The function `circ_embed2D` will return a two-dimensional array containing the simulated Gaussian SP. An example given in Newsam and Dietrich[2] is the the Gaussian RF with anisotropic covariance function

$$R(s,t) = \exp\left((s-t)^T A(s-t)\right) \tag{11}$$

where A is the positive-definite symmetric matrix

$$\begin{pmatrix} 3 & 1 \\ 1 & 2 \end{pmatrix}.$$

Moreover, in applying the function (11), we rescale the x- and y-directions by writing $s = (s_1/l_1, s_2/l_2)$, $t = (t_1/l_1, t_2/l_2)$. Figure 3 shows a realisation of this process, simulated with `circ_embed2D`, with scale lengths $l_1 = 50$, $l_2 = 15$ and grid spacing $\Delta x = \Delta y = 1$.

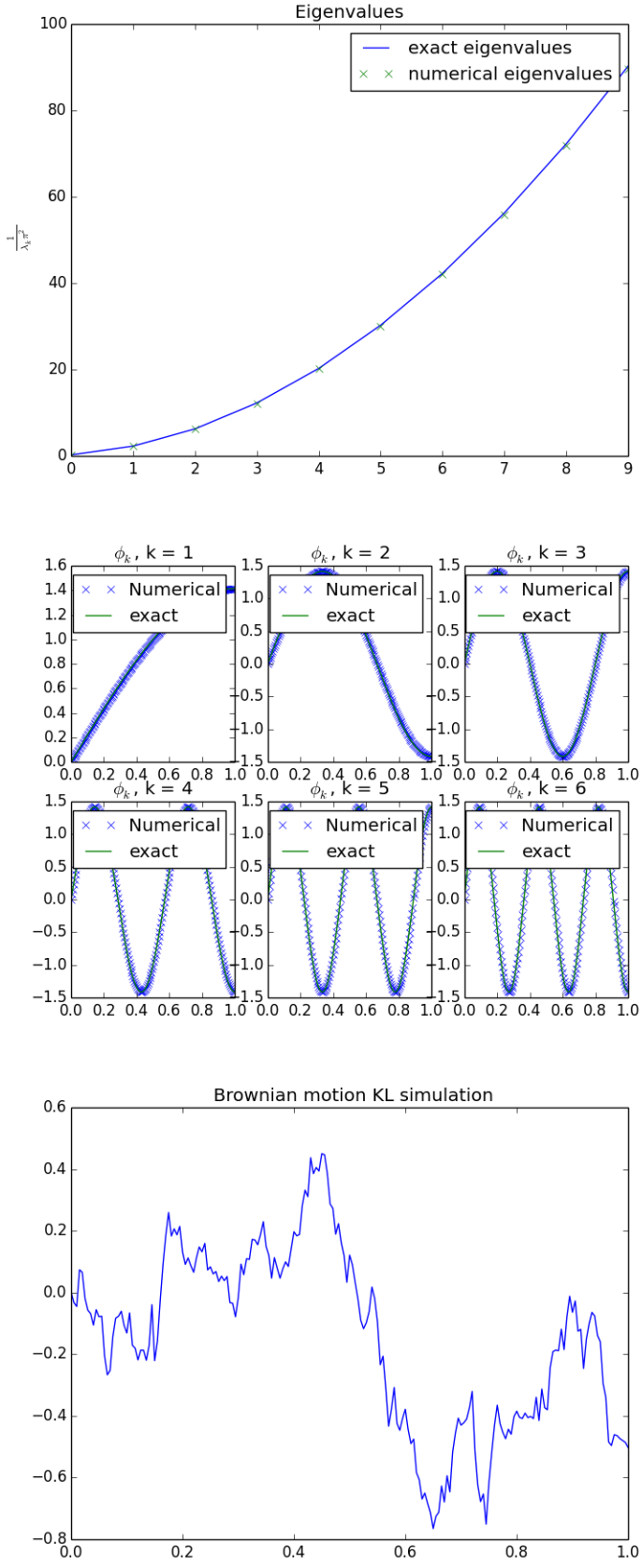

Figure 1: Top: Comparison of the numerical eigenvalues to the exact eigenvalues in the Brownian motion Karhunen-Loève function expansion.Middle: Comparison of the exact and numerical 5th eigenfunction. Bottom: The approximate Brownian motion in an $N = 200$ term expansion. Here N=200 quadrature points were used.

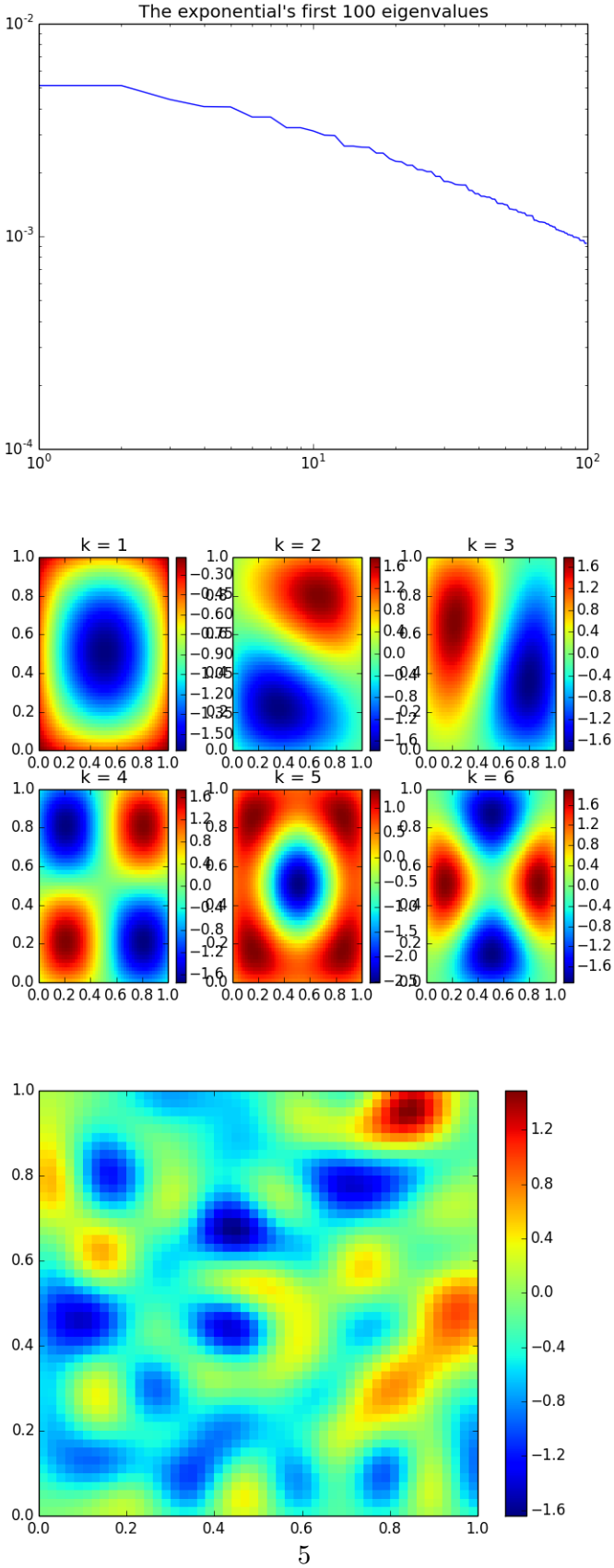

Figure 2: Top: Eigenvalues of the 2D exponential covariance function Gaussian RF. Middle: The first 6 eigenfunctions. Bottom: The random field realisation, using interpolated eigenfunctions. There are 50×50 points and the order of the expansion is $N = 100$.

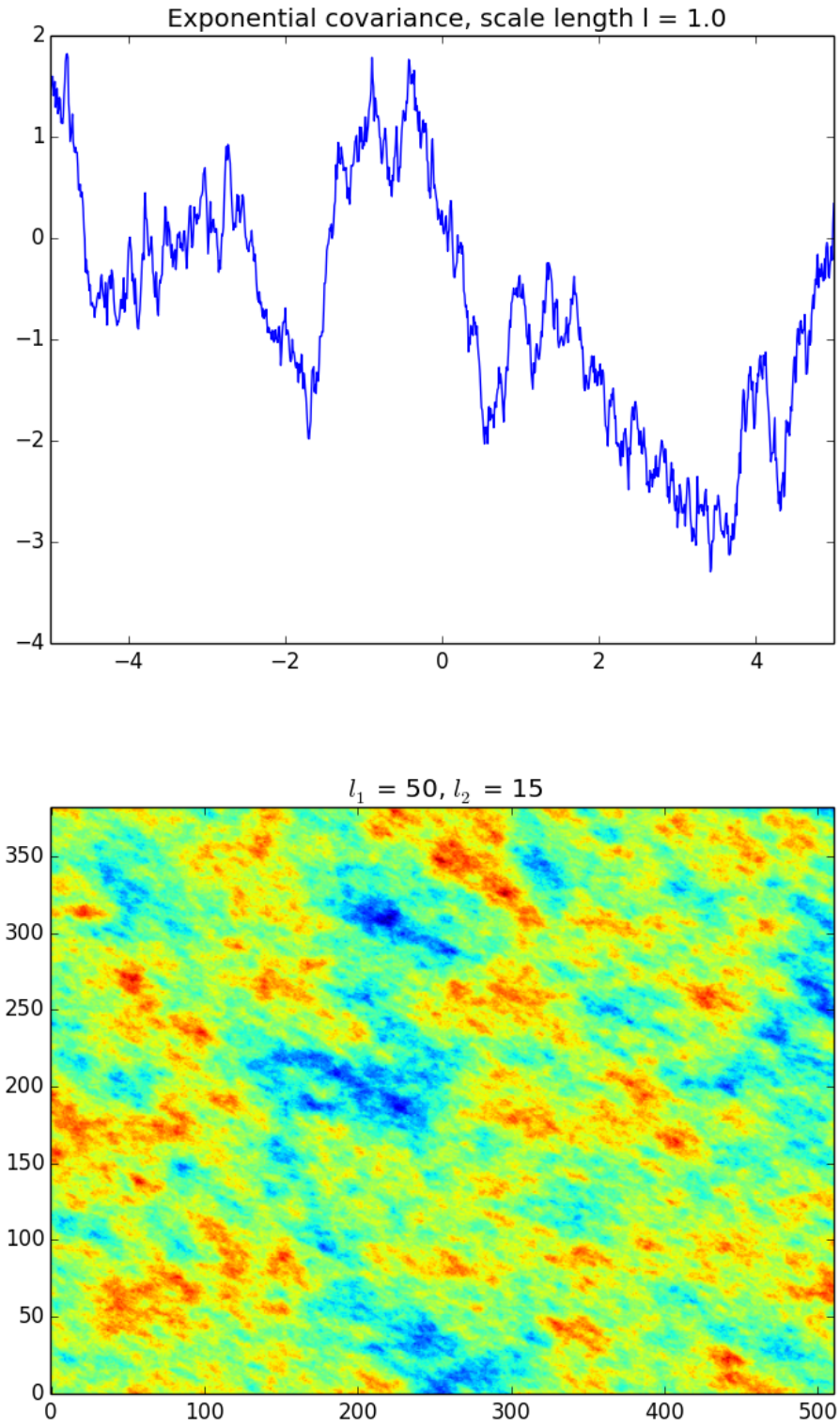

Figure 3: Top: A Realisation of the 1-D exponential random process.Bottom: A realisation of the 2D homogeneous Gaussian RF with anisotropic covariance (9).

4 Extensions

We list here extensions to the library of functions that should be prioritised. In no particular order we should add functions to:

- Implement the Cholesky decomposition simulation procedure in one- and two-dimensions
- Implement Galerkin projection methods in one- and two-dimensions[1].
  - In particular, we should implement Haar-wavelet basis functions[5] due to the potential of a speed boost compared to other finite-element basis functions.
- Add features for spatial statistics.

References

[1] Betz, W., Papaioannou, I., & Straub, D.(2014). Numerical methods for the discretization of random fields by means of the Karhunen=Loève expansion. *Computer Methods in Applied Mechanics and Engineering, 271*, 109-129.

[2] Dietrich, C.R., & Newsam, G. N. (1993). A fast and exact method for multidimensional Gaussian stochastic simulations. *Water Resources Research, 29(8)*, 2861-2869.

[3] Chan, G., & Wood, A.T. (1999). Simulation of stationary Gaussian vector fields. *Statistics and computing, 9(4)*, 265-268.

[4] Atkinson, K.E. (1967). The numerical solution of Fredholm integral equations of the second kind. *Siam Journal on Numerical Analysis, 4(3)*, 337-348.

[5] Phoon, K. K., Huang, S. P., & Quek, S. T. (2002). Implementation of Karhunen-Loeve expansion for simulation using a wavelet-Galerkin scheme. *Probabilistic Engineering Mechanics, 17(3)*, 293-303.