Simulating random functions

Felix Benning

January 29, 2026

Abstract

We want to simulate random functions and plot random functions from $\mathbb{R}^2$ to $\mathbb{R}$ as random landscapes. A random function from domain $\mathbb{X}$ into $\mathbb{R}$ is a collection of random variables $(\mathbf{f}(x))_{x\in\mathbb{X}}$ in $\mathbb{R}$. A Gaussian random function assumes every finite collection of these function values is multivariate Gaussian. Given a covariance function we will implement a simulation of centered Gaussian random functions and compare different approaches.

Prerequisites: "Probabilités" course

Nice to have: Multivariate Gaussian distribution, some Python (numpy and plotting) knowledge.

1 What is a Gaussian random function?

A random function $\mathbf{f}$ is a collection of random variables $(\mathbf{f}(x))_{x\in\mathbb{X}}$ indexed by the domain $\mathbb{X}$ of the function $\mathbf{f}$. Similarly, a random vector $X = (X_1, \dots, X_n)$ is indexed by the finite set $\{1, \dots, n\}$ and a random sequence $X = (X_k)_{k\in\mathbb{N}}$ is indexed by the natural numbers $\mathbb{N}$. However, while we are often interested in sequences of independent random variables, independence would make the function very discontinuous. For continuity it is necessary that function values at locations close to each other are highly correlated. A very important concept is therefore a 'covariance function'. $\mathcal{C}_\mathbf{f}$ is the covariance function of the random function $\mathbf{f}$ if

$$\mathcal{C}_\mathbf{f}(x, y) = \operatorname{Cov}(\mathbf{f}(x), \mathbf{f}(y)).$$

Not every function $k\colon \mathbb{X}\times\mathbb{X} \to \mathbb{R}$ can be a covariance function. For example it should be clear that $k(x, x) < 0$ cannot be allowed as this would imply a negative variance at x. A necessary and sufficient condition is that k is a **positive definite function**.

Definition 1.1 (Positive definite function)**.** A function $k\colon \mathbb{X}\times\mathbb{X} \to \mathbb{R}$ is *positive definite* if

- it is symmetric ($k(x, y) = k(y, x)$)
- for all $n \in \mathbb{N}$ and all selections of $a_i, a_j \in \mathbb{R}$ and $x_i, x_j \in \mathbb{X}$

$$\sum_{i,j=1}^{n} a_i a_j k(x_i, x_j) \geq 0 \tag{1}$$

Observe that for $\mathcal{C}_{\mathbf{f}}$ the requirement (1) is equivalent to

$$0 \leq \operatorname{Var}\Big(\sum_{i=1}^n a_i \mathbf{f}(x_i)\Big) = \sum_{i,j=1}^n a_i a_j \operatorname{Cov}(\mathbf{f}(x_i), \mathbf{f}(x_j))$$

by bilinearity of the covariance. This is of course a very reasonable requirement. Characterizations of such positive definite functions can be found in Chapter 2 of [1]. However for the purposes of simulating random functions you may want to focus on well known examples from Chapter 4 of [2].

A **Gaussian** random function is characterized by the fact that any finite sub-collection $(\mathbf{f}(x_1), \dots, \mathbf{f}(x_n))$ is a multivariate Gaussian random vector for any $x_1, \dots, x_n \in \mathbb{X}$. The covariance matrix of this vector is then given by

$$\Sigma = (\mathcal{C}_{\mathbf{f}}(x_i, x_j))_{i,j=1,\dots,n} \tag{2}$$

Exercise 1. Show that if $\mathcal{C}_{\mathbf{f}}$ is a positive definite function, Σ is always a positive definite matrix.

It is non-trivial to show that such a random function $\mathbf{f}$ exists for any *positive definite* covariance function and any mean function. This is a result of Kolmogorov's extension theorem [3, Sec. 14.3, esp. Thm. 14.36 or 4, Thm. 6.16]. For the purposes of this course you may simply believe this fact. It is similarly non-trivial to show smoothness properties about $\mathbf{f}$. Details can be found in [5, 6, 7, 8]. A rule of thumb (which is almost correct) is that a centered random function $\mathbf{f}$ is continuous if and only if $\mathcal{C}_{\mathbf{f}}$ is continuous and $\mathbf{f}$ is n-times differentiable if and only if $\mathcal{C}_{\mathbf{f}}(x, y)$ is n-times differentiable in x and simultaneously n-times differentiable in y, i.e. the following is well-defined

$$\partial_x^n \partial_y^n \mathcal{C}_{\mathbf{f}}(x, y).$$

The takeaway is that you can choose how rough or smooth the random function looks by the choice of covariance function. From now on we will further assume that the random function $\mathbf{f}$ is centered, i.e. has zero mean ($\mathbb{E}[\mathbf{f}(x)] = 0$ for all x).

Exercise 2. Why can we make this (centering) assumption without loss of generality?

2 Plotting a random function

To plot a function you typically select a finite grid $(x_1, \dots, x_n)$ of points to evaluate the function on. You then plot the function values $(\mathbf{f}(x_1), \dots, \mathbf{f}(x_n))$ on this grid. This fortunately makes our lives easier: We do not need to simulate the entire function $\mathbf{f}$, it is sufficient to simulate the random vector $(\mathbf{f}(x_1), \dots, \mathbf{f}(x_n))$. Recall that this is simply a Gaussian random vector with covariance matrix Σ as defined in (2). By our assumption that $\mathbf{f}$ is centered the mean is also zero.

Exercise 3. Consider the vector X defined by

$$X := LY$$

for the Gaussian random vector $Y = (Y_1, \dots, Y_n)^T$ with Y_k independent standard normal and L an $n \times n$ matrix. Show that the covariance matrix of X is given by LL^T.

With the exercise above it is easy to simulate $(\mathbf{f}(x_1), \dots, \mathbf{f}(x_n)) \sim \mathcal{N}(0, \Sigma)$, if we can simulate independent standard normal random variables and are able to obtain a decomposition of the covariance matrix $\Sigma = LL^T$.

Exercise 4. Calculate the computational complexity of the Cholesky decomposition and verify that it is applicable to our situation.

Exercise 5. Why does a spectral decomposition $\Sigma = UDU^T$ with orthonormal matrix U and diagonal matrix D exist? Verify that $L = U\sqrt{D}$, where the root is applied to entry wise to every element on the diagonal, results in another valid decomposition. What is the computational complexity of this algorithm?

Exercise 6. Compare the spectral decomposition with the Cholesky decomposition.

Exercise 7. Plot a random function $\mathbf{f}$ with domain $\mathbb{X} = [0, 1]^2 \subseteq \mathbb{R}^2$. For this, choose a covariance function from Chapter 4 in [2] (or some other if you want).

2.1 Improving the computational complexity

In the following I list some references for directions to improve the computational complexity of the sampler. I have not tried these approximations myself yet and they likely come with caveats that we can explore.

Iterative methods Pick a method from the review [9] to simulate a multivariate Gaussian vector $\mathcal{N}(0, \Sigma)$. Compare this to our baseline established above.

Since we only want to plot a few samples of the function, the following iterative samplers are probably the most likely to yield an improvement:

1. Chebyshev
2. Lanczos
3. Conjugate Gradient
4. Conjugate Gradient with Gradient scan

Turning band method [10, chapter 4] and [11]

3 Beyond simulating a finite number of points

Conditional sampling TODO

Karhunen-Loève Decomposition TODO (see [7])

References

[1] Zoltán Sasvári. *Multivariate Characteristic and Correlation Functions*. Number 50 in De Gruyter Studies in Mathematics. Walter de Gruyter, Berlin/Boston, March 2013. ISBN 978-3-11-022399-6.

[2] Carl Edward Rasmussen and Christopher K.I. Williams. *Gaussian Processes for Machine Learning*. Number 3 in Adaptive Computation and Machine Learning. MIT Press, Cambridge, Massachusetts, 2 edition, 2006. ISBN 0-262-18253-X. URL http://gaussianprocess.org/gpml/chapters/RW.pdf.

[3] Achim Klenke. *Probability Theory: A Comprehensive Course*. Universitext. Springer, London, 2014. ISBN 978-1-4471-5360-3 978-1-4471-5361-0. doi: 10.1007/978-1-4471-5361-0.

[4] Olav Kallenberg. *Foundations of Modern Probability*. Probability and Its Applications. Springer, New York, NY, 2002. ISBN 978-1-4419-2949-5 978-1-4757-4015-8. doi: 10.1007/978-1-4757-4015-8.

[5] Nathaël Da Costa, Marvin Pförtner, Lancelot Da Costa, and Philipp Hennig. Sample Path Regularity of Gaussian Processes from the Covariance Kernel, February 2024.

[6] Michael Scheurer. *A Comparison of Models and Methods for Spatial Interpolation in Statistics and Numerical Analysis*. PhD thesis, Göttingen, 2009.

[7] Robert J. Adler and Jonathan E. Taylor. *Random Fields and Geometry*. Springer Monographs in Mathematics. Springer New York, New York, NY, 2007. ISBN 978-0-387-48112-8. doi: 10.1007/978-0-387-48116-6.

[8] Michel Talagrand. Regularity of gaussian processes. *Acta Mathematica*, 159:99–149, January 1987. ISSN 0001-5962, 1871-2509. doi: 10.1007/BF02392556.

[9] Maxime Vono, Nicolas Dobigeon, and Pierre Chainais. High-Dimensional Gaussian Sampling: A Review and a Unifying Approach Based on a Stochastic Proximal Point Algorithm. *SIAM Review*, 64(1):3–56, February 2022. ISSN 0036-1445. doi: 10.1137/20M1371026.

[10] G. Matheron. The intrinsic random functions and their applications. *Advances in Applied Probability*, 5(3):439–468, December 1973. ISSN 0001-8678, 1475-6064. doi: 10.2307/1425829.

[11] Aristotelis Mantoglou and John L. Wilson. Simulation of Random Fields with the Turning Bands Method. July 1981. URL https://dspace.mit.edu/handle/1721.1/143010.