A Warped Gaussian Process Tutorial Using a Simple Example

Emma Foley[1,2]

[1]*Graduate Research Assistant, Bredesen Center, University of Tennessee, TN, United States*
[2]*Grid Communication and Security, Oak Ridge National Laboratory, TN, United States*

---***---

Abstract –*This paper describes an implementation of warped Gaussian process using a simple example. The results show the advantages of this method over a traditional Gaussian process. The flexibility of the warped Gaussian process maintains interpretability and mathematical foundation while accounting for non-Gaussian and/or noisy data. These features provide a basis for using this method in power grid applications, where this method is underrepresented in the literature. As exemplified by measurements taken in the Autonomous Intelligence Measurement Sensors and Systems (AIMS) project, data collected from sensors to support grid operations is non-Gaussian and noisy in nature. Warped Gaussian process is a flexible method suitable for analyzing multiple different types of power grid data.*

Key Words: Gaussian, warping function, tutorial, sensing, smart grid

1.INTRODUCTION

A Warped Gaussian Process (WGP) is a variation of the traditional Gaussian Process (GP). This method introduces a warping function that can account for non-Gaussian data, non-Gaussian noise, and capture more of the uncertainty in the data than the traditional Gaussian Process [1]. This tutorial will walk through the mathematical background and present a straightforward example using Python [2].

2. WARPED GAUSSIAN PROCESS

2.1 Define the Gaussian Process

In a traditional GP, we define a prior distribution over multiple functions so that we can infer a posterior distribution that better represents the actual underlying distribution. Once the posterior is determined, it can be used to calculate mean, variance, etc. GPs are particularly useful from a function space perspective [3], which is how we will proceed in this tutorial. First, define **X** to be a vector of random variables that follow a Gaussian distribution, then

$$\boldsymbol{X} = [X_1\, X_2\, \dots\, X_n\,]$$
$$\boldsymbol{X} \sim N(\mu, \Sigma)$$

where μ is the mean and Σ is the covariance matrix. Then there is a corresponding collection of function values that also follows a Gaussian distribution:

$$f(\mathbf{x}) = [f(x_1)\, f(x_2)\ \dots\ f(x_n)\,]^T$$

By definition, a GP is a collection of random variables which have a joint Gaussian distribution [4]. Thus, $f(\mathbf{x})$ is a Gaussian process and can be further defined as follows:

$$f(\mathbf{x}) \sim \mathrm{GP}(m(\mathbf{x}), k(\mathbf{x},\mathbf{x}'))$$
$$m(\mathbf{x}) = \mathrm{E}[f(\mathbf{x})]$$
$$k(\mathbf{x},\mathbf{x}') = \mathrm{E}[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))] + \delta_{ij}\sigma^2$$
$$\epsilon \sim \mathrm{N}(0, \sigma^2)$$

where $m(\mathbf{x})$ is the mean function, $k(\mathbf{x},\mathbf{x}')$ is the covariance function (also known as a kernel), δ is the Kronecker delta, and ϵ is the independent and identically distributed Gaussian noise.

2.2 Implementing the Warping Function

In order to warp the observation space to latent space, let **z** be a vector of latent target values modelled by a GP. The nonlinear monotonic function g maps all the entries from the actual target space to the latent space:

$$z_n = g(f(\mathbf{x}); \Psi)$$

where Ψ is a parameter. The function g can be any function but must be monotonic to maintain the validity of the probability distribution over $f(\mathbf{x})$. Once the target values have been warped into latent space, GP progresses as normal using **z.** The conditional distribution determines the predictive equations accounting for noise:

$$\mathbf{f}_*|\mathbf{Z},\mathbf{f} + \epsilon,\mathbf{Z}_* \sim \mathrm{N}(\mathbf{f}_*, cov(\mathbf{f}_*))$$
$$\bar{\mathbf{f}}_* \triangleq \mathrm{E}[\mathbf{Z},\mathbf{f} + \epsilon,\mathbf{Z}_*] = K(Z_*,Z)[K(Z,Z)+\sigma^2 I]^{-1}[\mathbf{f} + \epsilon]$$
$$cov(\mathbf{f}_*) = K(Z_*,Z_*) - K(Z_*,Z)[K(Z,Z)+\sigma^2 I]^{-1}K(Z,Z_*)$$

where the star subscript denotes data from the test subset. The marginal likelihood is defined below:

$$\log\big(p(\boldsymbol{f} + e|Z)\big) = -\frac{1}{2}[\mathbf{f} + \epsilon]^T[K(Z,Z) + \sigma^2]^{-1}[\mathbf{f} + \epsilon] - \frac{1}{2}\log[K(Z,Z) + \sigma^2 I] - \frac{n}{2}\log(2\pi)$$

Once you have the Gaussian based on the latent space, that Gaussian is passed back through the warping function to get the distribution in the observation space, the shape of which will depend on the warping function. Extracting the mean and median is described below:

$$f(x_n+1)^{med} = g^{-1}(z\hat{}_n+1)$$

$$\begin{aligned} E(f(x_{n+1})) &= E(g^{-1}) \\ &= \int dz\, g^{-1}(z) N_z(\hat{z}_{n+1},\ \sigma^2_{n+1}) \end{aligned}$$

In cases where the inverse of the warping function is undefined, it will need to be approximated.

3. A SIMPLE EXAMPLE

In this section we will use a simple example to demonstrate the benefits of WGP. The function $f(x) = x * sin(x)$ defines the dataset over $-\pi$ to π with additional Gaussian noise inserted. Chart 1 shows the underlying function and noisy observation data. The data is split into training and test data for analysis. The chosen warping function is $z = (f(\boldsymbol{x}))^{\frac{1}{3}}$ (see Chart 2). Once the data has been warped into latent space, GP progresses as described in Section 1, treating **z** as the target values. Chart 3 shows the heatmaps of the covariance matrices for the warped targets that will be used to calculate predictions. The predictions from the warped target values are then passed back through the warping function to retrieve the predictions in the true observation space where $f(\mathbf{x})$ are the target values. Chart 4 shows the results of the WGP method as well as a traditional GP method for comparison. The predictions are charted with dashed lines and the 95% confidence intervals are shaded in light gray (GP) and dark gray (WGP).

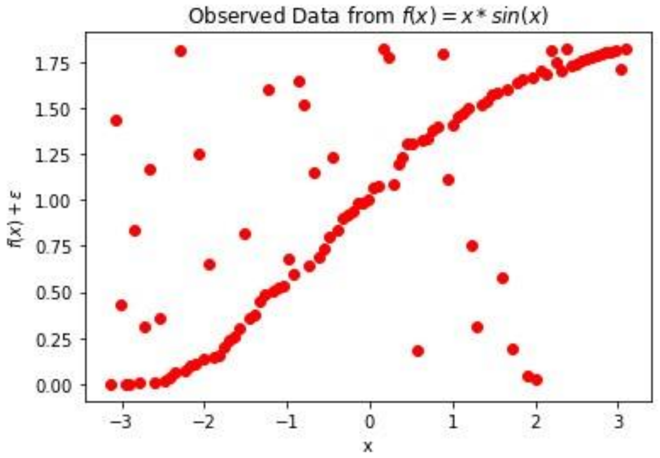


Chart -1: Observed data $f(x) = x * sin(x) + \epsilon$

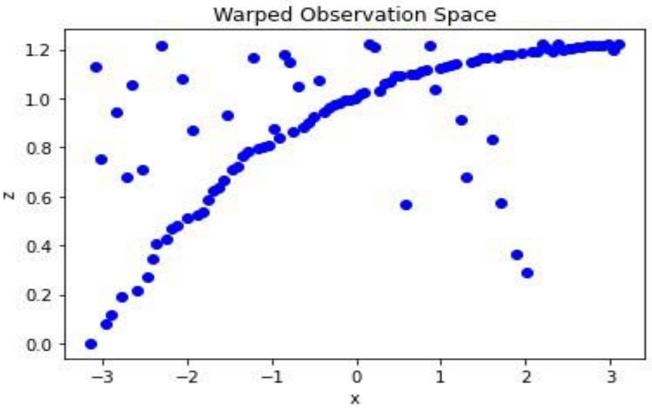
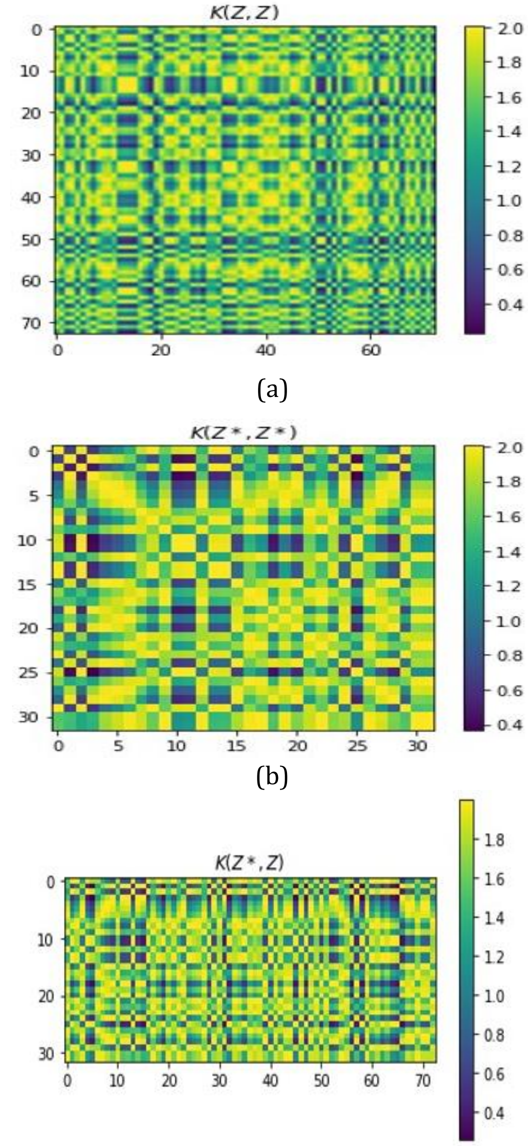


Chart -2: Observed data after being transformed by the warping function $z = (f(\boldsymbol{x}))^{\frac{1}{3}}$



(a)

(b)

(c)

Chart-3: Covariance calculations on the warped targets, **z**: (a) *K(Z, Z)*, (b) *K(Z*, Z*)*, (c) *K(Z*, Z)*.

The results of this example show the advantage WGP has in modeling noise. As shown in Chart 4, the GP predictions do closely follow the underlying function $f(x) = x * sin(x)$, but at the expense of capturing the noisiness of the data. In a real-world scenario, noise is often a factor and must be modelled accurately. In this example, the added noise was Gaussian, but that need not be the case. The nature of WGP allows it to model non-Gaussian noise. The WGP better approximates all the data over the entire distribution, which leads to lower error values (shown in Table 1) in both mean absolute error and mean square error.

Table -1: Results of an example using $f(x) = x * sin(x)$: WGP clearly outperforms GP in terms of mean absolute error and mean square error.

Method	MAE	MSE
GP	0.725	0.797
WGP	0.588	0.421

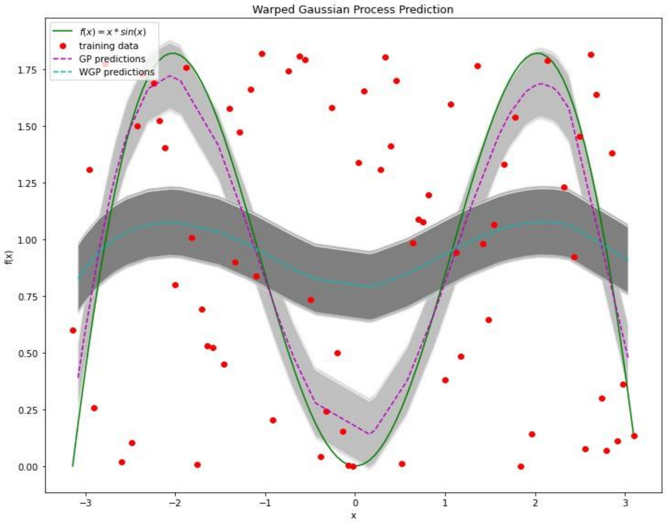


Chart-4: Results of an example using $f(x) = x*sin(x)$: The WGP method better accounts for noise, leading to improved MAE and MSE.

3.1 Comments

For simplicity, I used a one-dimensional problem with a well-known function. In a real-world scenario, both the GP and WGP have parameters that will need to be tuned and investigated to determine what effect changing them has on the analysis. GP performance can also be influenced by choice of basis and kernel functions. Additionally, the warping function itself is something that has not been deeply investigated in the literature. The choice of a power warping function versus and radial basis function may not have much effect on MAE or MSE but would change the shape of the final distribution. Another avenue would be to learn the warping function from the data prior to analysis to determine if an "ideal" function can be determined. Using WGP methodology should also be evaluated for overfitting. Accounting for noise can improve accuracy, but if the model is too reliant on training data it may not be usable on unseen data. Finally, one major drawback of WGP is the computational complexity, which was not evaluated here since the example was simple (the dataset contained 105 data points).

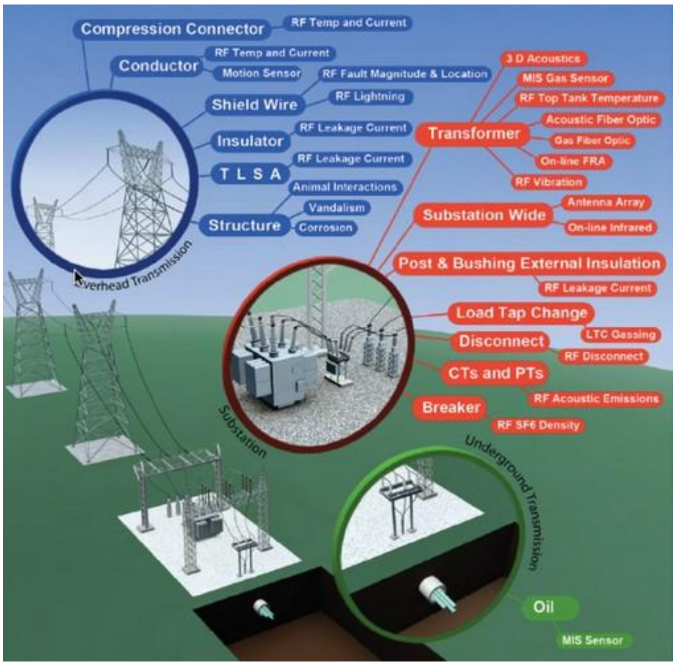


Fig -1: Graphical representation of the elements in a power grid and their associated data [5]

4. POWER GRID APPLICABILITY

Data management and analysis is a key issue for the grid since the power grid has a large magnitude of diverse data associated with its operation. Figure 1 highlights the areas where data collection can occur using different sensors. The quantity and diversity of this data suggests that there will be non-Gaussian representations, non-Gaussian noise, asymmetrical and non-stationary data, all of which can be served by the flexibility and interpretability of WGP. This method is underutilized in power grid research.

5. CONCLUSION

WGP is a method well suited to power grid analysis due to its ability to model non-stationary, non-symmetric, non-Gaussian data. Given how crucial grid operations are to managing infrastructure, the interpretable nature of WGP is a strength. WGP has all the benefits of traditional GPs with the advantage of higher accuracy since this method is more adept at capturing noise. Incorporating this method into analysis for the Autonomous Intelligence Measurement Sensors and Systems (AIMS) project will provide research opportunities for applications on a diverse set of real-world data.

REFERENCES

[1] E. Snelson, Z. Ghahramani, and C. Rasmussen, “Warped gaussian processes,” in *Advances in Neural Information Processing Systems* (S. Thrun, L. Saul, and B. Schoellkopf, eds.), vol. 16, MIT Press, 2003.

[2] J. Maucher, “Gaussian process: Implementation in python.” GitHub, 2022.

[3] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, *Dive into Deep Learning*. Cambridge, United Kingdom: Cambridge University Press, 2023.

[4] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA: Massachusetts Institute of Technology, 2006.

[5] E. P. R. Institute, “Sensor technologies for a smart transmission system.” An EPRI White Paper, 2009.