# On log-transformation of a computer model data. Emulation of a positive continuous simulator

Ksenia N. Kyzyurova
kseniak.ucoz.net, ksenia.kyzyurova@gmail.com

December 25, 2019

**Abstract**

log-transformation of data is a common recommendation given by an academic statistician for analysis of positive-valued data. This article demonstrates that this recommendation may perform very badly on data coming from computer models. While providing the recommendation against the use of the log-transformation, the article concludes with the viable solution to the problem of emulation of a computer model represented by a positive continuous function.

## 1 Introduction

Computer models are numerical mathematical simulators, which are used for scientific purpose of analysis of various phenomenae. Examples and the context of such are introduced in, for instance, [1,4].

Emulation of a computer model which may be represented by a real-valued continuous function has been introduced in the monograph [3]: relying on a few observations from the model and using Gaussian process methodology, the statistical approximation, that is, an emulator of the model, is constructed. The monograph concentrates on many aspects of emulation of computer models, including emulation of multivariate output computer model, assessment of how good the statistical approximation is, emulation of a to-be coupled model with the developed methodology of a *linked emulator*, and the developed methodology of a *censored emulator* for the model whose output is non-negative with zero-output having a non-zero probability to occur.

In [1] the discussion of the modeling of the continuous positive function has been offered. One proposal is to employ truncation to the emulator's distribution. Alternative solution would be to employ transformation of the output data so that the positive values are transformed to the ones which belong to the whole real line (with subsequent construction of a Gaussian process emulator for the transformed output of a computer model). The focus is on the log-transformation which is arguably the most ubiquitos data transformation appearing in the literature. In particular, this was chosen for the models in geophysics the author was working with during her *philosophy doctorate* studies.

In this article the log-transformation is shown to be less promising than one would hope for: an example is provided showing that the transformation fails to capture the transformed output of small values and does not perform well for transformation of values which are substantially greater than zero.

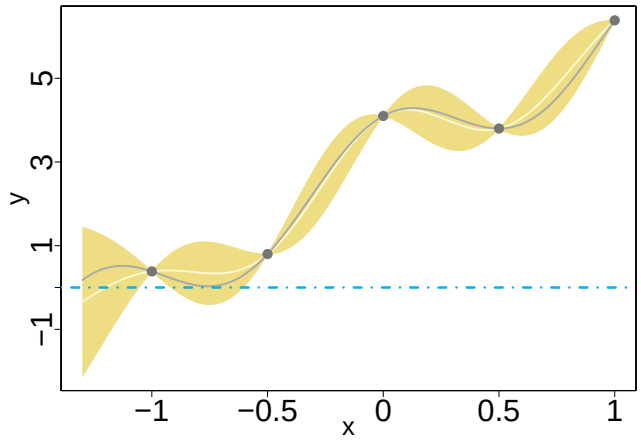


Figure 1: The Gaussian stochastic process emulator, whose credible 95% area is shown with the beige color, of the function $f(x) = 3x + \cos(5x) + 3.1$ in the domain $\mathcal{D} = (-1.3, 1)$ is constructed. The white curve indicates the mean of the emulator. The grey curve — the function itself. The threshold $c = 0$ on actual positivity of the output is shown with the blue dashed line.

**Illustration of the problem.** Function $f(x) = 3x + \cos(5x) + 3.1$ in the domain of $x \in \mathcal{D} = (-1.3, 1)$ is chosen as the simulator. Thus, the actual output takes only positive values although these values may be very small. If the information that a function takes only positive values is not available, then one proceeds by applying the Gaussian process methodology [1,5].

That is, evaluating the function (computer model) at a few inputs $\mathbf{x}$ and constructing a statistical approximation to outputs at all other inputs to the model. As for the illustrative example 5 points are chosen for model evaluation $\mathbf{x} = \{-1, -0.5, 0, 0.5, 1\}$ resulting in $f(\mathbf{x}) = \{f(x_1), f(x_2), \ldots, f(x_5)\}$. R package [2] has been used to construct and plot the result of the emulation which is shown in Figure 1. The emulator relies on the methodology of a (partially objective) Gaussian process emulator outlined in [1,5] which discuss the objective Bayesian procedure for estimation of parameters of the constructed statistical approximation as well.

The emulator performs well capturing $f$ on the whole domain, producing visually small discrepancy between the predictive mean (shown with a white colored line) and the function (grey coloured line) as well as providing reasonable estimates of uncertainty around the prediction — emulator's mean — 95% central credible area.

If prior information that $f(x) > 0$ for all $x \in \mathcal{D}$ is available, then this figure reveals that the Gaussian stochastic process emulator is giving a wrong statistical approximation providing a substantial non-zero probabilistic estimate that an output of a function may be negative. This is expected since the Gaussian process prior does not have any constraints on the range of a statistical approximation whose values within its framework belong to $\mathcal{R} = (-\infty, \infty)$ anyway. This formulates the problem for the emulation purpose of small positive output values under the prior information on positivity (or non-negativity) of the function continuous in all points in its domain of interest.

In this article, of particular interest is to consider a monotone transformation $g$ applied to the output of $f(x) \in (a, b)$, so that values of $g(f(x)) \in (-\infty, \infty)$. For instance, in the last simulation example $(a, b) \in (0, \infty)$. One commonly used transformation to apply for the positive valued function $f$ is $g(f) = \log(f)$.[1] Then Gaussian emulator is constructed for a transformed output of the computer model.

[1]Other proposals in the literature is to first shift upward the function $f$ before applying the log-transformation, that is $g = \log(f + \Delta)$, where $\Delta > 0$. However, one usually does not know which value of $\Delta$ to set it at in order to obtain an accurate emulator of the transformed output $g$.

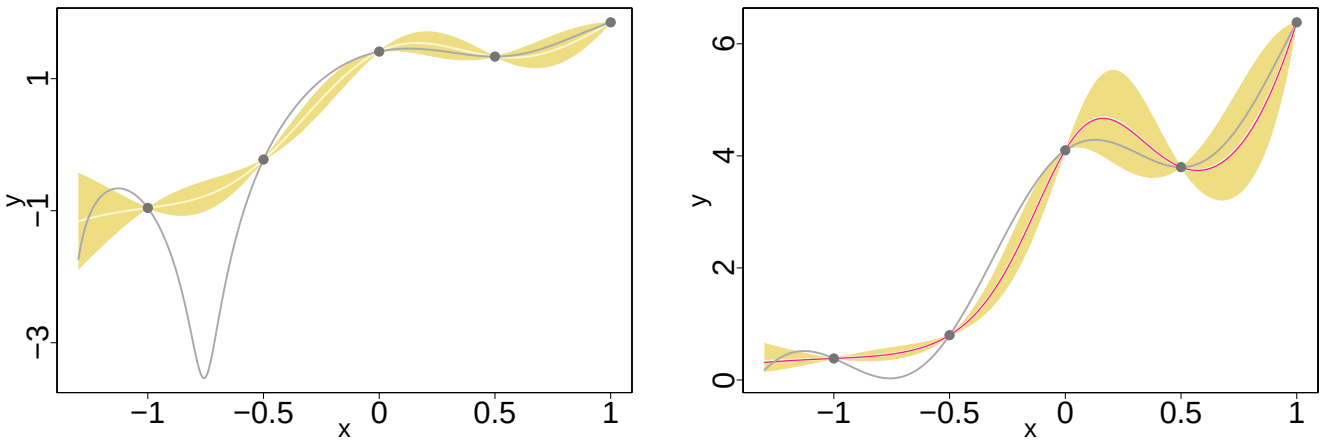


Figure 2: Left: The Gaussian stochastic process emulator constructed for a transformed function $g$, whose credible 95% area is shown with the beige color. The white curve indicates the mean of the emulator. The grey curve — the function itself. Right: The transformed emulator $f^M = \exp g^M$ along with the simulator $f$ is shown in the input domain $\mathcal{D}$. The emulator's colors repeat those of the emulator in the left panel. The functon $f$ is shown with the red color.

## 2 log-transformation of the computer model

Suppose that the Gaussian process emulator has been constructed for a simulator, using its transformed output $g$. Let denote predictive Gaussian distribution of the transformed output of a computer model at a new point $x^*$ as

$$g^M(x^*) \sim \mathcal{N}(\mu^*, \sigma^{*2}), \tag{1}$$

where $\mu^*$ is the expectation of this distribution, and $\sigma^{*2}$ is its variance. For the specific illustrative example, using the same inputs $\mathbf{x}$ as before, the constructed Gaussian process approximation to $g$ is shown in the left panel of Figure 2.

*In the area of interest where the output is small (the positive values of $f$ are less than 1) the $g = \log f$ produces negative output, which quickly grows large in absolute value.* This is why the constructed emulator of $g$ completely loses this region: smooth and slowly changing function becomes a highly volatile one in $(-1, -0.5)$, causing difficulty for the standard emulation. Besides, the emulator of $g = \log f$ misses the interval nearby as well. The overall behaviour of the emulator is bad: the nominal 95% frequency coverage corresponds to about 60% empirically of capturing the emulator's mean; predictive intervals in region $x > 0$ are large and are advisable to be shortened.

Going back to the original scale of the output of the simulator $f$, one must employ the inverse transformation of $g$ (assuming this exists), that is, $f = g^{-1}(g(f))$.

For the log function, the inverse is the exponential one. Therefore, the emulator $f^M = \exp g^M$ is the transformed emulator of the (1). In case of a Gaussian emulator $g^M$ (namely, normal marginal prredictive distribution $g^M(x^*)$ at any new point $x^*$), the exponential transformation gives that marginal posterior predictive distribution, that of the emulator $f^M$, follows a log-normal distribution with parameters $\mu^*$ and $\sigma^{*2}$.

$$f^M(x^*) \sim \mathcal{LN}(\mu^*, \sigma^{*2}). \tag{2}$$

Useful is to note the following properties of the log-normal distribution:

1. Transformed median, that of $f^M(x^*)$, equals to $\exp(\mu^*)$.
2. Transformed mean, that is, $\mathrm{E}f^M(x^*) = \exp(\mu^* + \sigma^{*2}/2)$.
3. Transformed variance, $\mathrm{V}f^M(x^*) = (\exp(\sigma^{*2} - 1))\exp(2\mu^* + \sigma^{*2})$.
4. Quantiles are transform-invariant under exponential transformation.

Figure 2, the right panel, demonstrates the transformed emulator $f^M$ in the domain $\mathcal{D}$ (2).

Unattractive property of the predictive distribution $f^M(x^*)$ is that $\sigma^{*2}$ enters the expression for its mean. If $\sigma^{*2}$ is comparable to $\mu^*$ or simply large, then the transformed mean may explode, as well as the transformed variance: too large of uncertainty around the mean (Figure 2, right panel) compared to those estimates provided by the Gaussian stochastic process emulator of $f$ (Figure 1) in the domain $x > 0$. One must seek very low-uncertainty emulator $g^M$ of the log-transformed output of a computer model in order to obtain an acceptable emulator $f^M$.

**$\mathcal{T}$-process emulator and its transformation.** If $g^M$ is a $\mathcal{T}$-process emulator (introduced in [1,3]), then backwards transformation results in a process $f^M = \exp g^M$ which is a $\log -\mathcal{T}$ process. The unattractive property of the $\log -\mathcal{T}$ distribution is that it has no positive moments. Thus, applying exponential backwards transformation for a $\mathcal{T}$-process would require some *ad hoc* truncation of means and variances at very large values which makes the methodology no longer coherent, hence, unattractive.

# 3 Conclusion

The main message of this article is the general recommendation against the log-transformation of a computer model output for emulation of data coming from numerical simulators. The overall performance of the log-transformation is not satisfactory because the problem of emulation of *a priori* known positive continuous function is hardly solved.

As for the general application of the log-transformation for other types of data: meaningful is to log-transform the exponentially distributed data, because such a transformation leads to normally distributed data which are subject to standard, "safe" inferential procedures (see examples in [4]). Of mathematical interest is to understand and quantify the errors one obtains if working with the log-transformation of data coming from distributions other than exponential.

## 3.1 Solution to the problem of emulation of a positive continuous function

Relying on the construction of the GASP emulator of the output first, one truncates the emulator's marginal distributions at any new input from below. Great is to obtain results such that the lowest quantile of interest is numerically indistinguishable (or very close to) that of the Gaussian emulator of $f$, because in this case the *ad hoc* truncations act as very small adjustments to the GASP mean, variance and quantiles. If the GASP lowest quantile is negative, then one must consider to improve on the design of the computer experiment by, e.g., adding design points, to achieve the numerically and practically sufficient approximation to the model.

The example of the successful application of the improved design methodology is presented in Figure 3. The same function $f$ as before is chosen as the simulator. The number of design points has been increased by one point added in the middle of the region where the emulator's (the one in the first illustrative example) lowest quantile was negative, namely, between $x_1 = -1$ and $x_2 = -0.5$. Partially objective Gaussian process emulator is constructed as a statistical approximation of $f$ in the domain $\mathcal{D}$.

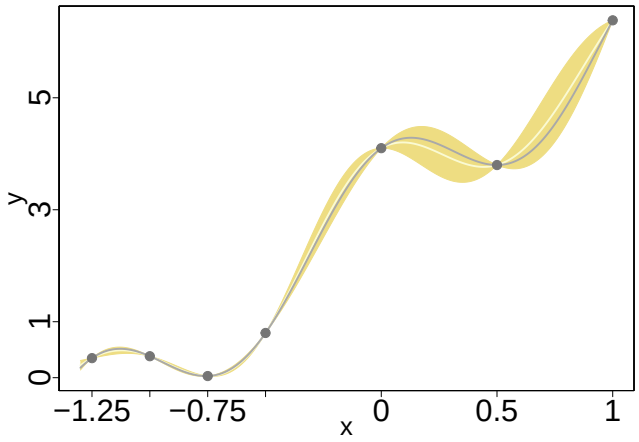


Figure 3: GASP emulator of the function $f(x) = 3x + \cos(5x) + 3.1$ in the domain $\mathcal{D} = (-1.3, 1)$. The number of design points $\{\mathbf{x}, f(\mathbf{x})\}$ has been increased in the area where the 2.5% quantile was negative (compared to that of the GASP emulator given the previous design, shown in Figure 1), resulting in excellent approximation of the *a priori* known positive-valued function $f$.

Generalization of the emulation of a continuous computer model truncated at other thresholds $c \neq 0$ from either below, above or both sides is straightforward. The closed-form expressions for the mean, variance and any quantile of a truncated normal distribution are given in [1].

Although specification of a stochastic process for an infinite-dimensional positive vector is theoretically possible, the solution is unlikely to be found in analytic or closed-form mathematical expressions resulting in that the numerical implementation of such a process is a prohibitive task. Therefore, the problem of emulation of a continuous positive computer model is unlikely to be solved in a principled way. For example, the stochastic process such that all finite-dimensional multivariate distributions are joint truncated multivariate normal does not exist. The author is not aware of other multivariate distributions for positive vectors whose marginal and conditional distributions are subject to analytic or closed-form mathematical expressions.

# Bibliography

[1] Kyzyurova K.N. (2017), On Uncertainty Quantification for Systems of Computer Models, PhD thesis, Duke University.

[2] Kyzyurova K.N. (2018), *LinkedGASP: Linked Emulator of a Coupled System of Simulators: R package version 1.0*, Comprehensive R Archive network (CRAN) repository.

[3] Kyzyurova K.N. (2019a), *Analysis of scientific computer models. Methodology in numerical simulator data analysis*, ■.

[4] Kyzyurova K.N. (2019b), *Calibration of mathematical computer models*, ■.

[5] Kyzyurova K.N., Berger J.O. & Wolpert R.L. (2018), 'Coupling Computer Models through Linking Their Statistical Emulators', *SIAM/ASA Journal on Uncertainty Quantification* **6**(3), 1151-1171.