

RESEARCH ARTICLE

The pitfalls of using Gaussian Process Regression for normative modeling

Bohan Xu^{1,2*}, Rayus Kuplicki¹, Sandip Sen², Martin P. Paulus^{1,3,4}

1 Laureate Institute for Brain Research, Tulsa, OK, United States of America, **2** Department of Computer Science, Tandy School of Computer Science, University of Tulsa, Tulsa, OK, United States of America, **3** Department of Community Medicine, Oxley College of Health Sciences, University of Tulsa, Tulsa, OK, United States of America, **4** Department of Psychiatry, School of Medicine, University of California San Diego, San Diego, CA, United States of America

* bxu@laureateinstitute.org**OPEN ACCESS**

Citation: Xu B, Kuplicki R, Sen S, Paulus MP (2021) The pitfalls of using Gaussian Process Regression for normative modeling. PLoS ONE 16(9): e0252108. <https://doi.org/10.1371/journal.pone.0252108>

Editor: Chi-Hua Chen, Fuzhou University, CHINA

Received: May 6, 2021

Accepted: August 27, 2021

Published: September 15, 2021

Peer Review History: PLOS recognizes the benefits of transparency in the peer review process; therefore, we enable the publication of all of the content of peer review and author responses alongside final, published articles. The editorial history of this article is available here: <https://doi.org/10.1371/journal.pone.0252108>

Copyright: © 2021 Xu et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability Statement: The data is generated from a python script, and the python script is available on the Github (<https://github.com/nidaye1999/normative-model-GPR>).

Funding: This research was supported by the Laureate Institute for Brain Research and the

Abstract

Normative modeling, a group of methods used to quantify an individual's deviation from some expected trajectory relative to observed variability around that trajectory, has been used to characterize subject heterogeneity. Gaussian Processes Regression includes an estimate of variable uncertainty across the input domain, which at face value makes it an attractive method to normalize the cohort heterogeneity where the deviation between predicted value and true observation is divided by the derived uncertainty directly from Gaussian Processes Regression. However, we show that the uncertainty directly from Gaussian Processes Regression is irrelevant to the cohort heterogeneity in general.

Introduction

In case-control studies, participants are assigned labels and classified into one or more categories based on their similarities or common criteria, with little consideration for the heterogeneity within each cohort. Meanwhile, normative modeling is becoming increasingly popular. In a normative model, each observation is quantified as a normalized deviation with respect to the cohort heterogeneity. The growth chart [1, 2] is an example normative model as shown in Fig 1, where a series of percentile curves (normalized deviation) illustrate the distribution of selected body measurements in children. Another widely-used measure for normalized deviation is the z-score, which is calculated by dividing the difference between an observation and the reference model, i.e., residual, by a standard deviation that represents local heterogeneity and assumes residuals are Gaussian distributed locally.

The uncertainty sometimes can be classified into two categories: epistemic and aleatoric uncertainties. Epistemic uncertainty is known as systematic uncertainty and is due to things one could in principle know but do not in practice; aleatoric uncertainty is known as statistical uncertainty and is representative of unknowns that differ each time we run the same experiment [4]. Epistemic uncertainty is often introduced by the limited dataset size and can be reduced by adding more observations. On the other hand, aleatoric uncertainty represents a character of heterogeneity in the underlying distribution itself which is unrelated to

National Institute of General Medical Sciences
(P20GM121312, MP, RK).

Competing interests: The authors have declared that no competing interests exist.

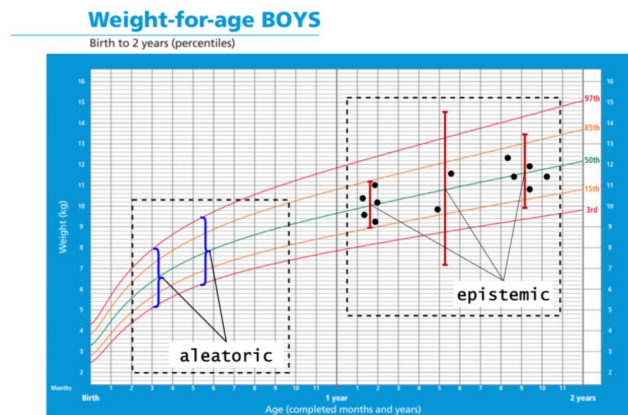


Fig 1. Weight-for-age boys: Birth to 2 years. Reprinted from [3] under a CC BY license, with permission from World Health Organization, original copyright (2021). The percentiles show the distribution of weights in boys from birth to 2 years. Black dots: observations; red error bars: epistemic uncertainty; blue curly brackets: aleatoric uncertainty.

<https://doi.org/10.1371/journal.pone.0252108.g001>

sample size, so it cannot be reduced by modifying the dataset, and this is the heterogeneity a normative model should measure. As shown in Fig 1, larger number and density of data points (black dots) reduce the epistemic uncertainty (red error bars), while the aleatoric uncertainty (blue curly brackets) is unrelated to the sample size or distribution. The confidence intervals obtained from most statistical tests and advanced machine learning models only capture epistemic uncertainty, while a normative model is designed to capture the aleatoric uncertainty.

Gaussian Process Regression (GPR) has been widely used in many domains. Schulz et al. [5] presented a tutorial on the GPR with the mathematics behind the model as well as several applications to real-life datasets/problems. Tonner et al. [6] developed a GPR based model and testing framework to capture the microbial population growth and shown their proposed approach outperformed primary growth models. Banerjee et al. [7] and Raissi et al. [8] introduced two novel approaches to improve the efficiency of GPR in “big data” problems.

However, some previous research implemented the GPR as a normative modeling approach and utilized the derived prediction variance to model the cohort heterogeneity. Ziegler et al. [9] attempted to build a normative model for diagnosing mild cognitive impairment and Alzheimer’s disease based on the normalized deviation of predicted brain volume from GPR. Marquand et al. [10] used delay discounting as covariates and reward-related brain activity derived from task Functional Magnetic Resonance Imaging (fMRI) as the target variable with GPR and extreme value statistics to identify the participants with Attention-Deficit/Hyperactivity Disorder (ADHD). Wolfers et al. [11] investigated the deviation of brain volume in an ADHD cohort from healthy control group (HC) with respect to age and gender, and they also explored the heterogeneous phenotype of brain volume for schizophrenia and bipolar disorder with GPR [12]. Zabihi et al. [13] studied Autism Spectrum Disorder (ASD) regarding the deviation of cortical thickness via a similar methodology.

In this paper, we introduce some background knowledge related to GPR. We then present a rigorous mathematical derivation and several examples to demonstrate that the variance from GPR cannot be used in a normative model alone. In the last section, we discuss the difficulties and disadvantages of modeling the cohort heterogeneity by modifying original GPR variance, and a misunderstanding existed in previous research.

Materials and methods

Gaussian Process Regression

The relation between the observation and the predictive model usually can be expressed as

$$y = f(\mathbf{x}) + \varepsilon, \tag{1}$$

where y is the observation (output), $f(\cdot)$ represents the predictive model, \mathbf{x} is a vector of independent variables (input) corresponding to the output y , and ε is the noise term which follows a normal distribution $\varepsilon \sim \mathcal{N}(0, \sigma_{\text{noise}}^2)$. Gaussian Process Regression (GPR) assumes a zero-mean normal distribution over the predictive model

$$f(\cdot) \sim \mathcal{N}(0, k(\cdot, \cdot)), \tag{2}$$

where $k(\cdot, \cdot)$ is some covariance (kernel) function. Given the training set input \mathbf{X} and testing set input \mathbf{X}_* , since both of them follow the same distribution, we have

$$f\left(\begin{bmatrix} \mathbf{X} \\ \mathbf{X}_* \end{bmatrix}\right) \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \mathbf{K}(\mathbf{X}, \mathbf{X}) & \mathbf{K}(\mathbf{X}, \mathbf{X}_*) \\ \mathbf{K}(\mathbf{X}_*, \mathbf{X}) & \mathbf{K}(\mathbf{X}_*, \mathbf{X}_*) \end{bmatrix}\right), \tag{3}$$

According to the Eq 1, the observation follows the summation of these two normal distributions

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{y}_* \end{bmatrix} = f\left(\begin{bmatrix} \mathbf{X} \\ \mathbf{X}_* \end{bmatrix}\right) + \begin{bmatrix} \varepsilon \\ \varepsilon_* \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \mathbf{K}(\mathbf{X}, \mathbf{X}) + \Sigma_{\text{train}}^2 & \mathbf{K}(\mathbf{X}, \mathbf{X}_*) \\ \mathbf{K}(\mathbf{X}_*, \mathbf{X}) & \mathbf{K}(\mathbf{X}_*, \mathbf{X}_*) + \Sigma_{\text{test}}^2 \end{bmatrix}\right), \tag{4}$$

where Σ_{train}^2 and Σ_{test}^2 are two square diagonal matrices that represent the variance of observation noise in training and testing sets, and all diagonal elements of Σ_{train}^2 and Σ_{test}^2 are identical and equal to σ_{noise}^2 . By the rules of conditional Gaussian distribution, the prediction of testing set \mathbf{y}_* follows a normal distribution $\mathbf{y}_* \sim \mathcal{N}(\boldsymbol{\mu}_*, \Sigma_*^2)$, where $\boldsymbol{\mu}_*$ and Σ_*^2 are defined as [14, 15]

$$\boldsymbol{\mu}_* = \mathbf{K}(\mathbf{X}_*, \mathbf{X})[\mathbf{K}(\mathbf{X}, \mathbf{X}) + \Sigma_{\text{train}}^2]^{-1} \mathbf{y}, \tag{5a}$$

$$\Sigma_*^2 = \mathbf{K}(\mathbf{X}_*, \mathbf{X}_*) + \Sigma_{\text{test}}^2 - \mathbf{K}(\mathbf{X}_*, \mathbf{X})[\mathbf{K}(\mathbf{X}, \mathbf{X}) + \Sigma_{\text{train}}^2]^{-1} \mathbf{K}(\mathbf{X}, \mathbf{X}_*). \tag{5b}$$

Kernel trick

Similar to Support Vector Machines (SVM), the kernel trick can also be implemented with GPR to project the input of data from the original space into a same or higher dimensional feature space via some mapping function $z(\cdot)$. Given a pair of inputs $(\mathbf{x}_1, \mathbf{x}_2)$, the kernel function calculates the inner product of the coordinates in the feature space, i.e., $k(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{z}(\mathbf{x}_1)\mathbf{z}(\mathbf{x}_2)^T$ [16, 17]. The kernel trick avoids the expensive computation of calculating the coordinate in the feature space for each input. We use the linear kernel and Radial Basis Function kernel (RBF) as examples to illustrate this advantage.

Linear kernel. The linear kernel is non-stationary and the simplest kernel, which is defined as

$$k(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{x}_1 \mathbf{x}_2^T, \tag{6a}$$

$$\mathbf{z}(\mathbf{x}) = \mathbf{x}, \tag{6b}$$

where the input is projected into a feature space according to Eq 6b, and the feature space is the original space.

Radial Basis Function kernel. The RBF kernel is a stationary kernel, which is also widely used and defined as [17]

$$k(\mathbf{x}_1, \mathbf{x}_2) = e^{-\frac{\|\mathbf{x}_1 - \mathbf{x}_2\|^2}{2l^2}}, \tag{7a}$$

$$z(\mathbf{x}) = \left[\frac{e^{-\frac{\|\mathbf{x}\|^2}{2l^2}}}{\sqrt{l^{2j} j!}} \sqrt{\frac{j!}{n_1! \dots n_k!}} x_1^{n_1} \dots x_k^{n_k} \right]_{j=0, \dots, \infty, \sum_{i=1}^k n_i = j}, \tag{7b}$$

where l is a free scaling parameter. The RBF kernel projects the input from the original space onto an infinite dimensional feature space where the mapping is defined by Eq 7b. It is impossible to exactly compute the coordinates in an infinite dimensional space, while Eq 7a still allows straightforward computation of the inner product for coordinate pairs in that feature space.

Matérn and Rational-Quadratic kernels. Matérn kernel is a generalization of the RBF kernel which is defined as [14]

$$k(\mathbf{x}_1, \mathbf{x}_2) = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}}{l} \|\mathbf{x}_1 - \mathbf{x}_2\| \right)^\nu K_\nu \left(\frac{\sqrt{2\nu}}{l} \|\mathbf{x}_1 - \mathbf{x}_2\| \right), \tag{8}$$

where the parameter ν controls the smoothness of the function, $\Gamma(\cdot)$ refers to the gamma function, and $K_\nu(\cdot)$ represents modified Bessel function. Rational-Quadratic kernel is another kernel based on the RBF kernel, which is given by

$$k(\mathbf{x}_1, \mathbf{x}_2) = \left(1 + \frac{\|\mathbf{x}_1 - \mathbf{x}_2\|^2}{2\alpha l^2} \right)^{-\alpha}, \tag{9}$$

where α is a scale mixture parameter. The Rational-Quadratic kernel can be considered as an infinite sum of RBF kernels with different length-scales l [18].

Estimated uncertainty for GPR

One benefit of using GPR to build a data-driven model is the predictions are associated with the derived variances as shown in Eq 5. However, we need to emphasize that this variance is only related to the kernel function $k(\cdot, \cdot)$ and distribution/coordinate of training set input \mathbf{X} , i.e., it cannot be utilized in a normative model approach alone to capture the variance introduced by the conditional distribution $Var(y|\mathbf{x})$.

We better illustrate and verify this statement through simplifying the Eq 5b. Since any kernel function $k(\cdot, \cdot)$ can be written as the inner product of a coordinate pair in the feature space by some mapping function $z(\cdot)$, we present our derivation in a general format. We define a variable \mathbf{x}_* which represents a testing input, then Eq 5b can be written as

$$\Sigma_*^2(\mathbf{x}_*) = k(\mathbf{x}_*, \mathbf{x}_*) + \sigma_{\text{test}}^2 - \mathbf{k}(\mathbf{x}_*, \mathbf{X}) [\mathbf{K}(\mathbf{X}, \mathbf{X}) + \Sigma_{\text{train}}^2]^{-1} \mathbf{k}(\mathbf{X}, \mathbf{x}_*) \tag{10a}$$

$$= \mathbf{z}(\mathbf{x}_*) \mathbf{z}(\mathbf{x}_*)^T + \sigma_{\text{test}}^2 - \mathbf{z}(\mathbf{x}_*) \mathbf{Z}(\mathbf{X})^T [\mathbf{Z}(\mathbf{X}) \mathbf{Z}(\mathbf{X})^T + \Sigma_{\text{train}}^2]^{-1} \mathbf{Z}(\mathbf{X}) \mathbf{z}(\mathbf{x}_*)^T. \tag{10a}$$

Applying Singular Value Decomposition (SVD) on $\sum_{\text{train}}^{-1} \mathbf{Z}(\mathbf{X}) = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$, Eq 10 is reformulated as (the detailed derivation is presented in S1 Appendix)

$$\begin{aligned} & \Sigma_*^2(\mathbf{x}_*) \\ &= \mathbf{z}(\mathbf{x}_*)\mathbf{z}(\mathbf{x}_*)^T + \sigma_{\text{test}}^2 - \mathbf{z}(\mathbf{x}_*)\mathbf{Z}(\mathbf{X})^T[\mathbf{Z}(\mathbf{X})\mathbf{Z}(\mathbf{X})^T + \Sigma_{\text{train}}^2]^{-1}\mathbf{Z}(\mathbf{X})\mathbf{z}(\mathbf{x}_*)^T \\ &= \underbrace{\mathbf{z}(\mathbf{x}_*)\mathbf{V}[\mathbf{I} - \mathbf{\Sigma}^T(\mathbf{\Sigma}\mathbf{\Sigma}^T + \mathbf{I})^{-1}\mathbf{\Sigma}]\mathbf{V}^T\mathbf{z}(\mathbf{x}_*)^T}_{\text{quadratic term}} + \underbrace{\sigma_{\text{test}}^2}_{\text{constant}}. \end{aligned} \tag{11}$$

After simplification, the variance is reformulated as Eq 11, which is a summation of a quadratic term for $\mathbf{z}(\mathbf{x}_*)$ and a constant represents the noise, $\mathbf{\Sigma}$ and \mathbf{V} are constant matrices where the values are fully depended on training input \mathbf{X} , training noise Σ_{train} , and mapping function $\mathbf{z}(\cdot)$ or kernel function $k(\cdot, \cdot)$.

Modification of uncertainty from GPR

Regarding Eq 11, the variance calculated via Eq 5b is purely depended on kernel function and training data input, thus it is only able to capture the epistemic uncertainty which could be reduced by modifying or adding training data. The derived variance from GPR could be extended to model the heterogeneity $\text{Var}(y|\mathbf{x})$ for a normative model by adding an aleatoric variance term into Eq 5b

$$\begin{aligned} \text{Var}(y_*|\mathbf{X}_*) &= \underbrace{\mathbf{K}(\mathbf{X}_*, \mathbf{X}_*) + \Sigma_{\text{test}}^2 - \mathbf{K}(\mathbf{X}_*, \mathbf{X})[\mathbf{K}(\mathbf{X}, \mathbf{X}) + \Sigma_{\text{train}}^2]^{-1}\mathbf{K}(\mathbf{X}, \mathbf{X}_*)}_{\text{epistemic uncertainty}} \\ &+ \underbrace{\Sigma_{\text{aleatoric}}^2(\mathbf{X}_*)}_{\text{aleatoric uncertainty}}, \end{aligned} \tag{12}$$

where $\Sigma_{\text{aleatoric}}^2(\mathbf{X}_*)$ represents the data character of heterogeneity in output at given locations on the input space. This formula, however, is not implemented in any previous research as we know and we will discuss the difficulties and disadvantages in estimating the aleatoric uncertainty later.

Results

We apply the unmodified GPR (Eq 5) on several synthetic datasets where both input x and output y are one dimensional to facilitate visualization. Although the presented results are based on one dimensional input x , they are generalizable to any dimensional input. We illustrate the characteristics of the four kernels mentioned above, but we mainly focus on the linear and RBF kernels. We also present the results of two scenarios with known and unknown noise levels.

Dataset

Four synthetic datasets are generated and plotted in Fig 2, and each of them contains 1000 points with a noise level of $\sigma_{\text{noise}} = 0.05$. Four other undersampled datasets are plotted in Fig 3, each of which contains $1000 \times 5\% = 50$ points.

In Dataset 1, both input \mathbf{X} and output \mathbf{y} follow a Gaussian distribution $\mathcal{N}(0, 1^2)$ and are correlated with a Pearson coefficient of 0.75. Dataset 2 is transformed from Dataset 1, which moves the set of points where $x \geq 0$ in Dataset 1 along the line $y = x$ until the maximum input in that set equals 0, and moves the remaining points where $x < 0$ in Dataset 1 until the

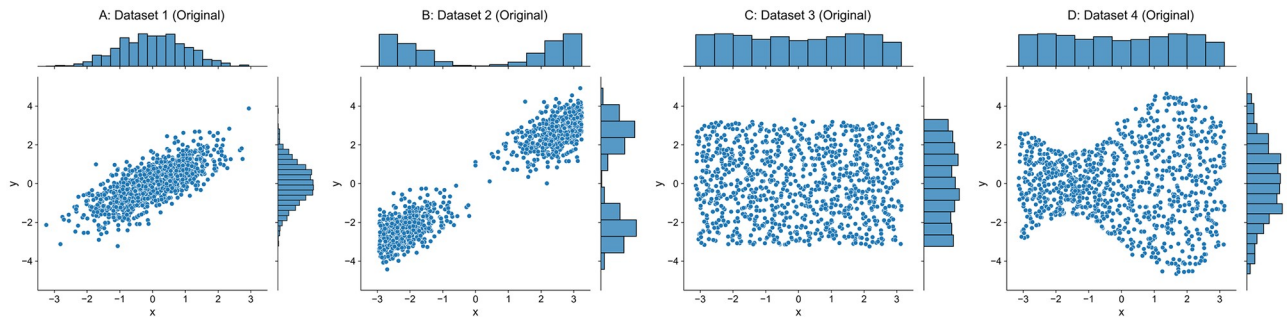


Fig 2. Original datasets.

<https://doi.org/10.1371/journal.pone.0252108.g002>

minimum input is 0. Dataset 3 has input X and output y uniformly distributed over a half-open interval $[-\pi, \pi)$. Output y of Dataset 4 is obtained by multiplying a factor function over output y from Dataset 3, which is defined as $f(x) = \sin(x)/2 + 1$ and x is the corresponding input. We should note that the inputs X of original Datasets 3–4 are exactly same as shown in Fig 2C and 2D, and the inputs X of corresponding undersampled Datasets 3–4 are also identical as shown in Fig 3C and 3D.

GPR with known noise level

Linear kernel. The regression surface of GPR with linear kernel is a hyperplane and the variance is a quadric hypersurface defined by Eq 11 in feature/original spaces, where the hyperplane always passes the origin, the variance is a function only with respect to the coordinate of testing input x and a unique minimum is located at $x^* = 0$. Figs 4 and 5 present results for GPR with linear kernel on the one dimensional synthetic datasets, where top sub-figures plot the reference models/predictions (red lines) overlapped on the data (blue dots), middle sub-figures show the derived variances (blue curves) across the original input space, and the bottom sub-figures shows the corresponding “z-score” for training set which is computed via Eq 13 if the residual $(y - y_{\text{reference}})$ is mistakenly normalized by standard deviation Σ directly from GPR (Eq 5b).

$$z\text{-score} = \frac{y - y_{\text{reference}}}{\Sigma} \tag{13}$$

The mapping function of linear kernel projects an input to itself (Eq 6b). For one dimensional input, Σ in Eq 11 is an $m \times 1$ matrix, where m is the size of training set. The only non-

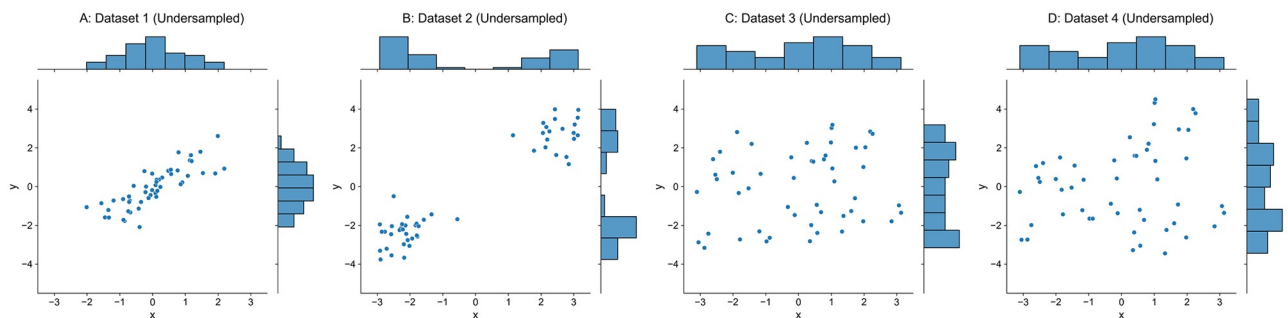


Fig 3. Undersampled datasets.

<https://doi.org/10.1371/journal.pone.0252108.g003>

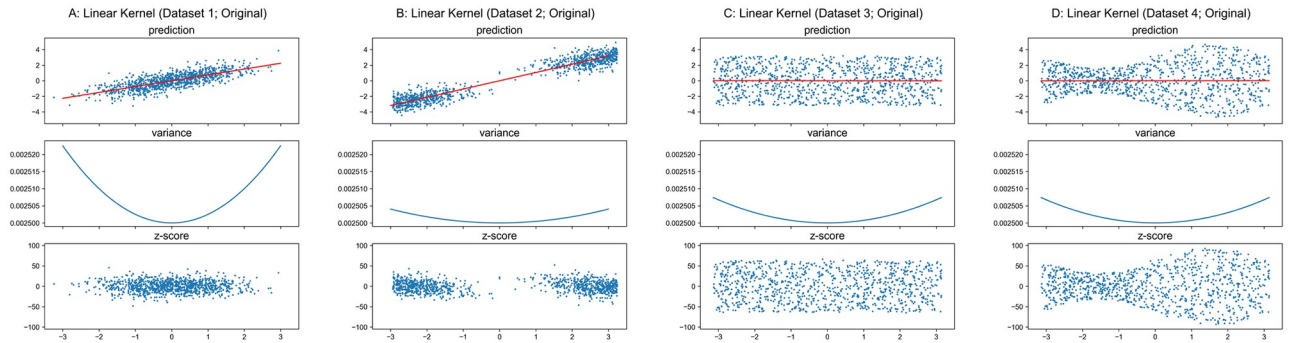


Fig 4. GPR with linear kernel on original datasets.

<https://doi.org/10.1371/journal.pone.0252108.g004>

zero element $\Sigma_{1,1}$ equals the only non-zero singular value σ of $\Sigma_{\text{train}}^{-1} \mathbf{X}$, and \mathbf{V} is a 1×1 identity matrix. Therefore, Eq 11 can be further reduced to

$$\begin{aligned} \Sigma_*^2(x_*) &= [1 - \Sigma^T(\Sigma\Sigma^T + \mathbf{I})^{-1}\Sigma]x_*^2 + \sigma_{\text{test}}^2 \\ &= \left(1 - \frac{\sigma^2}{\sigma^2 + 1}\right)x_*^2 + \sigma_{\text{test}}^2 \\ &= \frac{x_*^2}{\sigma^2 + 1} + \sigma_{\text{test}}^2. \end{aligned} \tag{14}$$

As shown in Figs 4 and 5, the variance is a univariate function of coordinate of the testing input x^* where the shape is a quadratic curve, and the global minimum is always located at $x^* = 0$ with a value of $\sigma_{\text{test}}^2 = 0.05^2$ as Eq 14 formulated. The result of GPR with linear kernel presents a good example which illustrates the derived variance from GPR does not model the conditional variance $\text{Var}(y|x)$, thus corresponding z-score cannot be utilized as a normalized deviation in a normative model.

As previously mentioned, the predicted variance for testing set from GPR only depends on the training set input and the kernel function. As the original as well as the undersampled Datasets 3–4 have identical inputs \mathbf{X} , the variance curves in Figs 4C, 4D, 5C and 5D are respectively identical.

RBF kernel. Unlike the linear kernel, RBF kernel mapping function (Eq 7b) defines a feature space which is different from the original space. Regarding the original space, the

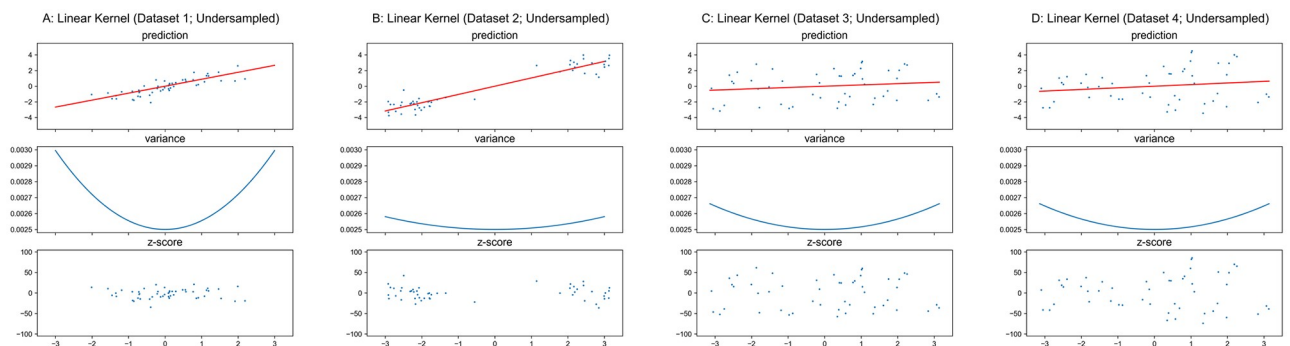


Fig 5. GPR with linear kernel on undersampled datasets.

<https://doi.org/10.1371/journal.pone.0252108.g005>

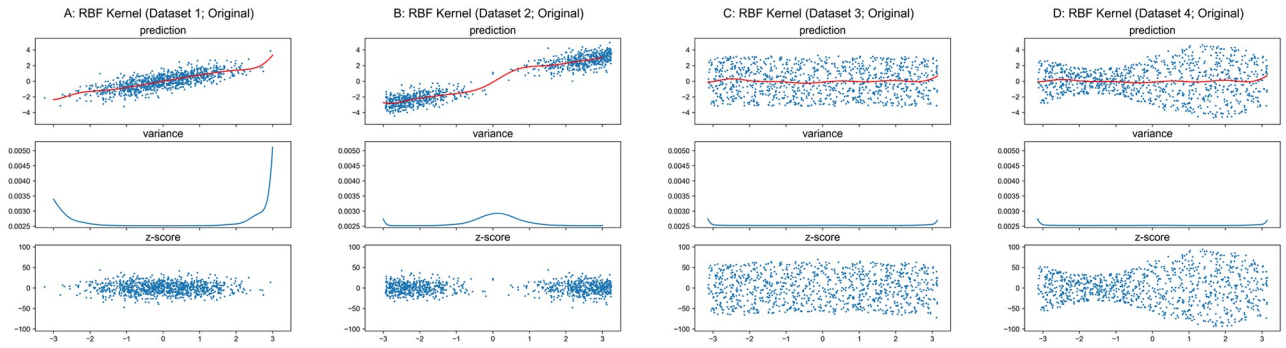


Fig 6. GPR with RBF kernel on original datasets.

<https://doi.org/10.1371/journal.pone.0252108.g006>

regression surface is no longer a hyperplane and the variance is no more a quadric hypersurface for the RBF kernel, although regression surface is always a hyperplane and variance is always a quadric hypersurface for any kernels in the feature space. Because the mapping function of RBF kernel is very complicated, we only briefly describe the characteristics of the regression surface and variance in the original space. For a test input \mathbf{x}^* , the prediction is a summation of discounted outputs of all training points where each corresponding discount factor is determined by the Euclidean distance between \mathbf{x}^* and that training input, and the predicted value converges to 0 if \mathbf{x}^* is far away from all training inputs. On the other hand, the variance depends only on the density of training inputs at \mathbf{x}^* , and higher density results in lower variance. Therefore, the variance of GPR with RBF kernel is related of the relative location to the training inputs rather than the absolute location specified by coordinate.

The results for GPR with RBF kernel applied to these synthetic datasets are shown in Figs 6 and 7. We should note that the value of hyper-parameter l in the RBF kernel function (Eq 7a) does not affect the main idea of this paper, thus we used a fixed value of 1.0 instead of utilizing hyper-parameter optimization in this section. As shown in Figs 6 and 7, the variance is unrelated to the conditional variance $Var(y|x)$. Therefore, z-scores based on this model do not represent normalized deviation. However, unlike the quadratic curves whose unique minimum is always located at $x = 0$ in Figs 4 and 5 for linear kernel, the variance function of GPR with RBF kernel regarding the original input space is related to the distribution of training input \mathbf{X} . The denser inputs at the middle of Dataset 1 and two ends of Dataset 2 lead to lower variances at those locations in Fig 6A and 6B, while the uniformly distributed inputs of Datasets 3–4 result in relatively flat curves in and Fig 6C and 6D. According to Eq 7a and given an arbitrary input \mathbf{x}^* , the RBF kernel function returns a larger value for a point in \mathbf{X} that is closer to \mathbf{x}^* , and $\mathbf{k}(\mathbf{x}^*, \mathbf{X})$ and $\mathbf{k}(\mathbf{X}, \mathbf{x}^*)$ have more large elements if \mathbf{x}^* is close to more points in \mathbf{X} . Due to $\mathbf{K}(\mathbf{X}, \mathbf{X}) + \sum_{\text{train}}^2$ is a symmetric positive definite matrix, both result in the decrease of the value for Eq 10a, i.e., to smaller variance.

Similar to the result for the linear kernel, the theoretical minimum of variance is $\sigma_{\text{test}}^2 = 0.05^2$, and the variance curves are exactly identical in Figs 6C, 6D, 7C and 7D respectively.

Matérn and Rational-Quadratic kernels. The properties of Matérn and Rational-Quadratic kernels are similar to the RBF kernel. Therefore, we only present the results in Figs 8–11 for these two kernels without the detailed analysis. Similar to RBF kernel, the hyper-parameters of Matérn and Rational-Quadratic kernels are also fixed in this section, where $\nu = 1.5$ for Matérn kernel, $\alpha = 1.0$ for Rational-Quadratic kernel, and $l = 1.0$ for both kernels. The

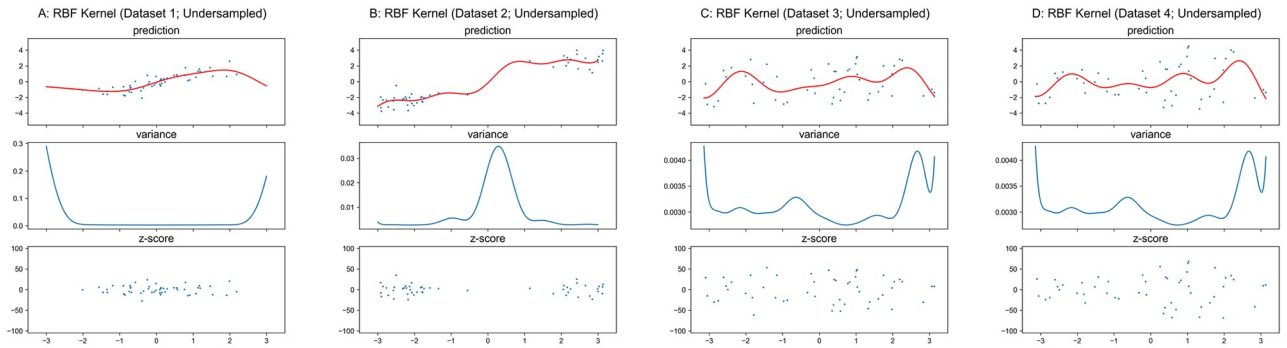


Fig 7. GPR with RBF kernel on undersampled datasets. Scales of Y-axis for variance plots are different.

<https://doi.org/10.1371/journal.pone.0252108.g007>

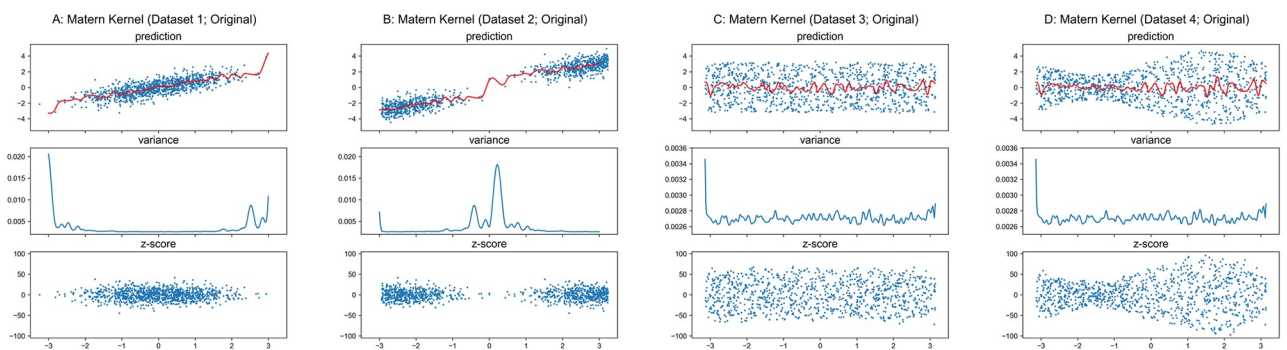


Fig 8. GPR with matern kernel on original datasets. Scales of Y-axis for variance plots are different.

<https://doi.org/10.1371/journal.pone.0252108.g008>

variance curves shown in Figs 8C, 8D, 9C, 9D, 10C, 10D, 11C and 11D are exactly identical, respectively.

GPR with unknown noise level

The noise level can be included as a hyper-parameter when it is unknown. However, the derived variance from GPR still does not model the heterogeneity $Var(y|\mathbf{x})$, although it could be a good approximation in some special cases.

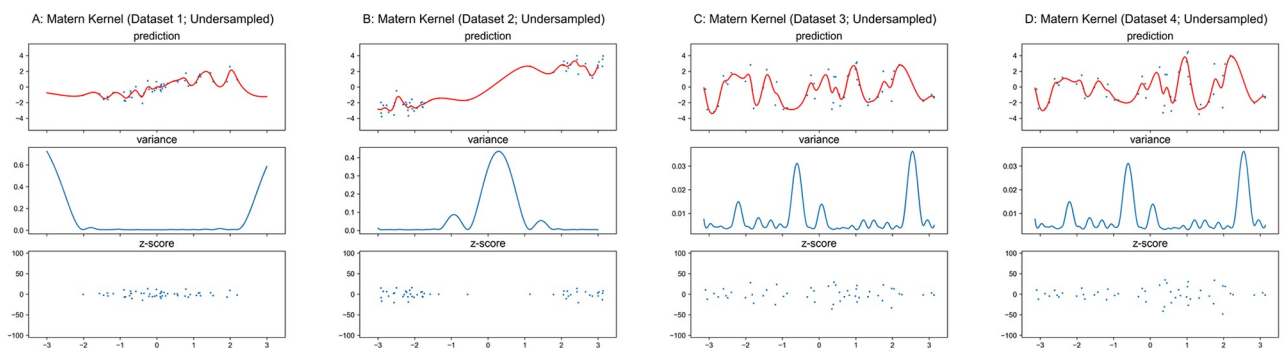


Fig 9. GPR with matern kernel on undersampled datasets. Scales of Y-axis for variance plots are different.

<https://doi.org/10.1371/journal.pone.0252108.g009>

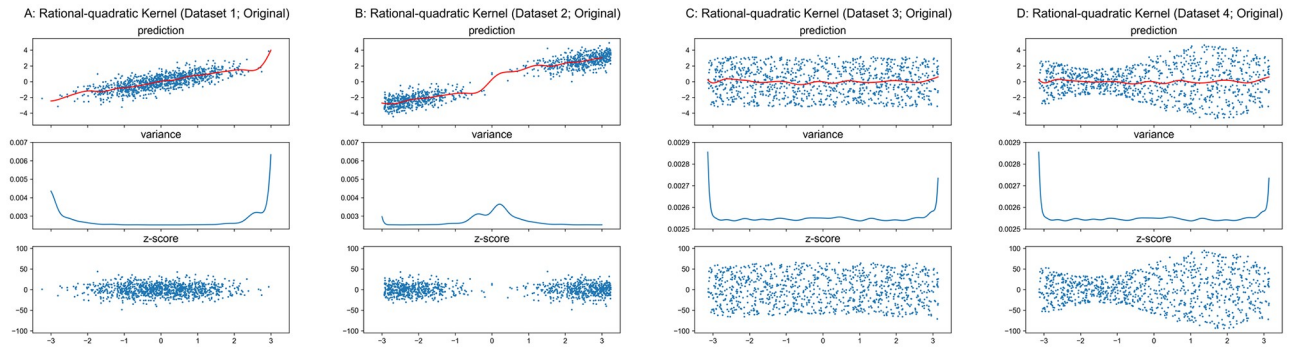


Fig 10. GPR with Rational-Quadratic kernel on original datasets. Scales of Y-axis for variance plots are different.

<https://doi.org/10.1371/journal.pone.0252108.g010>

As the basic properties of linear and RBF kernels have been introduced, a hybrid kernel is utilized in the following analysis which is defined as

$$k_{\text{hybrid}}(\cdot, \cdot) = w_{\text{linear}}k_{\text{linear}}(\cdot, \cdot) + w_{\text{RBF}}k_{\text{RBF}}(\cdot, \cdot) + k_{\text{white}}(\cdot, \cdot), \tag{15}$$

where w_{linear} and w_{RBF} represent adjustable weights on linear and RBF kernels, and $k_{\text{white}}(\cdot, \cdot)$ refers to a white-noise kernel that represents the independently and identically normally-distributed observation noise, i.e., $\mathbf{K}_{\text{white}}(\mathbf{X}, \mathbf{X}) = \sum_{\text{train}}^2$, $\mathbf{K}_{\text{white}}(\mathbf{X}_*, \mathbf{X}_*) = \sum_{\text{test}}^2$, $\mathbf{K}_{\text{white}}(\mathbf{X}, \mathbf{X}_*) = \mathbf{0}$, and $\mathbf{K}_{\text{white}}(\mathbf{X}_*, \mathbf{X}) = \mathbf{0}$. Because the Matérn and Rational-Quadratic kernels are both based on the RBF kernel, so we only include the RBF kernel in the hybrid kernel. The Eq 5b can be reformulated as

$$\Sigma_*^2 = \mathbf{K}_{\text{hybrid}}(\mathbf{X}_*, \mathbf{X}_*) - \mathbf{K}_{\text{hybrid}}(\mathbf{X}_*, \mathbf{X})\mathbf{K}_{\text{hybrid}}(\mathbf{X}, \mathbf{X})^{-1}\mathbf{K}_{\text{hybrid}}(\mathbf{X}, \mathbf{X}_*). \tag{16}$$

Original Datasets 3–4 in Fig 2C and 2D are perfect for testing whether a model captures the heterogeneity $\text{Var}(y|\mathbf{x})$, as the large number of instances and uniformly distributed data over the input space lead to negligible epistemic uncertainty in certain input range, and the true reference model $y = 0$ is very simple as well. Besides, the results of two more complex datasets with quadratic reference models are presented in the S1 Appendix. In this section, the hyper-parameters are tuned by maximizing the likelihood $P(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta})$, where $\boldsymbol{\theta}$ represents all hyper-parameters in the model. The results are plotted in Fig 12, and the optimized hyper-parameters are listed in Table 1 as well as the overall variances of residual $\text{Var}(y - y_{\text{reference}})$.

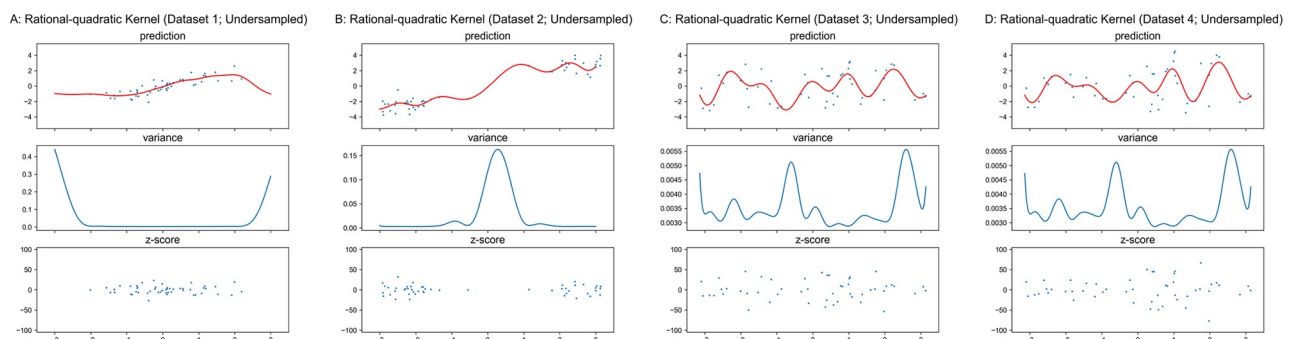


Fig 11. GPR with Rational-Quadratic kernel on undersampled datasets. Scales of Y-axis for variance plots are different.

<https://doi.org/10.1371/journal.pone.0252108.g011>

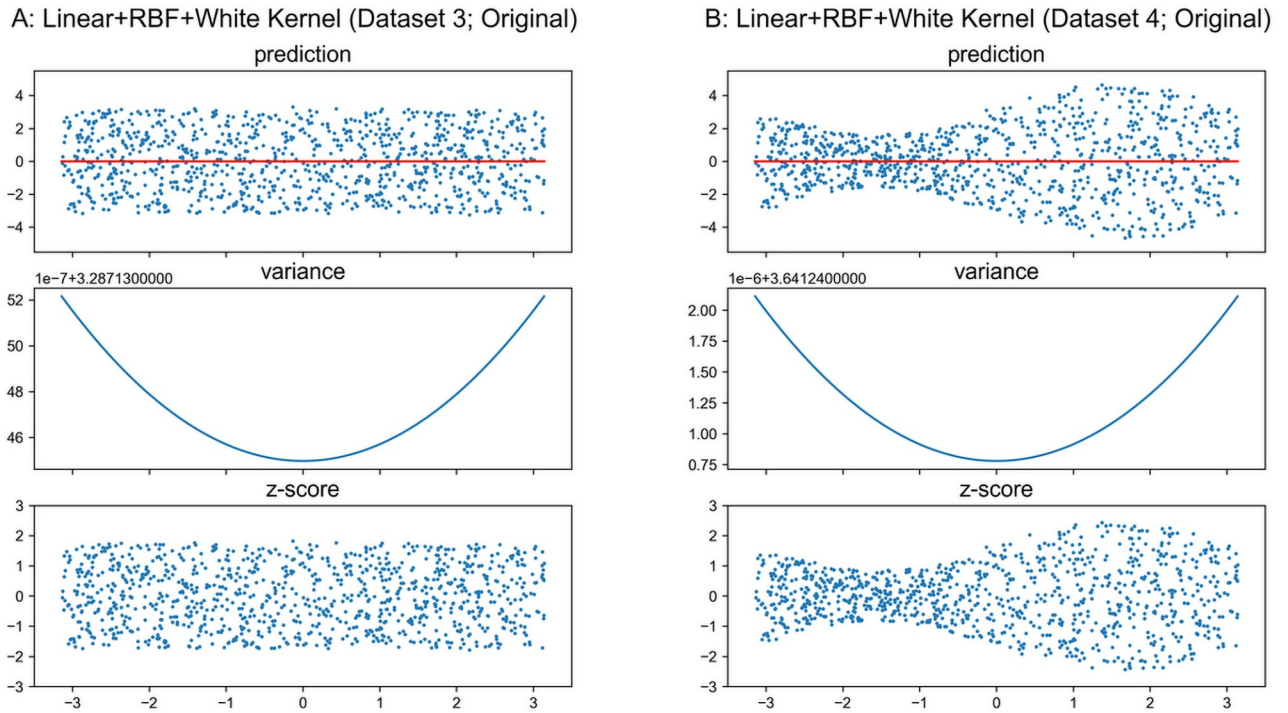


Fig 12. GPR with hybrid kernel on original datasets 3–4. Scales of Y-axis for variance plots are different.

<https://doi.org/10.1371/journal.pone.0252108.g012>

As shown in Fig 12, the GPR accurately estimates the reference models, i.e., $y_{\text{reference}} \approx y_{\text{reference,true}}$. The variance curves are nearly quadratic, since the w_{linear} is relatively larger than w_{RBF} while w_{RBF} is not exact zero as listed in Table 1. However, the domination of $k_{\text{white}}(\cdot, \cdot)$ over $k_{\text{linear}}(\cdot, \cdot)$ and $k_{\text{RBF}}(\cdot, \cdot)$ due to small optimized weights flattens the curves, i.e., the value of the curve is almost constant over the plotted input range in this example. Particularly, the σ_{noise}^2 is very close to the overall residual variance $\text{Var}(y - y_{\text{reference}})$, and the explanation will be presented later. Therefore, $\mathbf{K}_{\text{hybrid}}(\mathbf{X}_*, \mathbf{X}_*) \approx \sum_{\text{test}}^2 \mathbf{K}_{\text{hybrid}}(\mathbf{X}, \mathbf{X}) \approx \sum_{\text{train}}^2 \mathbf{K}_{\text{hybrid}}(\mathbf{X}^*, \mathbf{X}) \approx \mathbf{0}$ and $\mathbf{K}_{\text{hybrid}}(\mathbf{X}, \mathbf{X}^*) \approx \mathbf{0}$, which result in $\sum_*^2 \approx \sum_{\text{test}}^2 = \sum_{\text{noise}}^2$ for Eq 16.

Regarding Eq 1, since the noise is included as a tunable hyper-parameter without any constraints, the optimizer will adjust reference model $f(\cdot)$ as well as bias σ_{noise}^2 to $\text{Var}(y - y_{\text{reference}})$ to maximize the likelihood $P(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta})$. Even the σ_{noise} refers to the observation noise level in GPR while the optimizer handles it as a variable without considering its meaning in a model.

In Dataset 3, σ_{noise}^2 is biased to the overall residual variance $\text{Var}(y - y_{\text{reference}})$, and $\text{Var}(y - y_{\text{reference}})$ is well matched with the homoskedastic heterogeneity $\text{Var}(y|\mathbf{x})$. So the z-scores plotted in Fig 12A show the GPR works as a normative model approach in this special case. However, in Dataset 4, σ_{noise}^2 is also biased to the overall residual variance $\text{Var}(y - y_{\text{reference}})$, while $\text{Var}(y - y_{\text{reference}})$ does not approximate the heteroskedastic

Table 1. Optimized hyper-parameters for hybrid kernel on original datasets 3–4.

	w_{linear}	w_{RBF}	l	σ_{noise}^2	$\text{Var}(y - y_{\text{reference}})$
Dataset 3	7.29e-8	4.88e-10	2.94e2	3.29	3.29
Dataset 4	1.35e-7	1.98e-17	1.54e-5	3.64	3.64

<https://doi.org/10.1371/journal.pone.0252108.t001>

heterogeneity $Var(y|\mathbf{x})$. So the z -scores plotted in Fig 12B do not represent a measure of normalized deviation in general.

Discussion

Although GPR could be extended and to model the heterogeneity as presented in this work, it is either: (1) hard to estimate the aleatoric uncertainty accurately when the data are sparse, e.g., at the middle of Dataset 2; or (2) unnecessary to model the conditional variance by Eq 12 when the data are dense, e.g., Datasets 3–4. One approach to estimate $\sigma_{\text{aleatoric}}^2(\mathbf{x}_*)$ is using the sliding window technique, but it is hard to choose the window size for each dimension of input. For Scenario 1, even if the optimal window sizes can be obtained, it is hard to accurately estimate $\sigma_{\text{aleatoric}}^2(\mathbf{x}_*)$ when the window centered at \mathbf{x}^* only covers a small number of training data points, e.g., \mathbf{x}^* is far away from all points in \mathbf{X} . If the window centered at \mathbf{x}^* covers a large number of training data points, e.g., Scenario 2, $Var(y|\mathbf{x}^*)$ should almost equal $\sigma_{\text{aleatoric}}^2(\mathbf{x}_*)$ and epistemic uncertainty is insignificant. Then $Var(y|\mathbf{x}^*)$ can be simply approximated as a local variance over a space defined by the window. There are more sophisticated algorithms than the naive sliding window technique, e.g., LOcal regrESSion (LOESS) [19, 20] and Generalized Additive Models of Location Shape and Scale (GAMLSS) [21, 22]. However, these methods still need densely distributed data over the input space based on our experience.

Another misunderstanding we found in the literature is interpreting the noise term σ_{noise}^2 as aleatoric uncertainty. When the observation noise is considered as a hyper-parameter, it will likely bias the overall residual variance $Var(y - y_{\text{reference}})$. The overall residual variance is a good approximation of homoskedastic aleatoric uncertainty $Var(y|\mathbf{x})$. It is, however, not valid for cases with heteroskedastic residuals, which is the main motivation for using normative modeling. Although the value of the noise term is biased to estimate overall residual variance during the optimization, the mathematical/physical meanings are pre-defined by the model. Moreover, in homoskedastic aleatoric uncertainty cases, further investigation is needed to verify whether $\mathbf{K}(\mathbf{X}_*, \mathbf{X}_*) - \mathbf{K}(\mathbf{X}_*, \mathbf{X}) [\mathbf{K}(\mathbf{X}, \mathbf{X}) + \sum_{\text{noise}}^2]^{-1} \mathbf{K}(\mathbf{X}, \mathbf{X}_*)$ will still be a good approximation of epistemic uncertainty with such a biased estimation of observation noise level.

Conclusions

In this paper, we present the mathematical derivation with a general formula to demonstrate that the derived prediction variance from GPR does not model the heterogeneity $Var(y|\mathbf{x})$, which in general is necessary for a normative model. GPR with a linear kernel and an RBF kernel are used as examples to illustrate this statement on one dimensional input datasets. Overall, the derived variance from GPR cannot be utilized in a normative model alone.

Supporting information

S1 Appendix. This file contains data/code availability, Eq S1, S1 Fig, and S1 Table. All datasets are generated from the same Python script, which contains the code for analysis as well; Eq S1 is a detailed derivation for Eq 11; S1 Fig and S1 Table are results for modified Datasets 3–4.
(PDF)

Acknowledgments

I would sincerely appreciate my families and friends from Laureate Institute for Brain Research, the University of Tulsa, and Brain Technologies Inc. for helping me through my

brain tumor surgery and the subsequent recovery during the revision of this paper. Special appreciation to Kaiping Burrows and Leandra Figueroa-Hall.

We would also like to thank the journal editor and anonymous reviewers for insightful discussions and feedback that have improved our study and manuscript.

Author Contributions

Formal analysis: Bohan Xu.

Investigation: Bohan Xu.

Methodology: Bohan Xu.

Writing – original draft: Bohan Xu.

Writing – review & editing: Rayus Kuplicki, Sandip Sen, Martin P. Paulus.

References

1. Kuczmarski RJ, Ogden CL, Grummer-Strawn LM, Flegal KM, Guo SS, Wei R, et al. CDC growth charts: United States. *Advance data from vital and health statistics*; 314. Hyattsville, Maryland: National Center for Health Statistics; 2000.
2. Kuczmarski RJ, Ogden CL, Guo SS, Grummer-Strawn LM, Flegal KM, Mei Z, et al. 2000 CDC growth charts for the United States: methods and development. *Vital Health Stat* 11(246). Hyattsville, Maryland: National Center for Health Statistics; 2002.
3. World Health Organization. Reproduced from “Weight-for-Age Boys: Birth to 2 years (percentiles)” accessed 6-May-2021; Copyright (2021). Available from: [https://cdn.who.int/media/docs/default-source/child-growth/child-growth-standards/indicators/weight-for-age/boys-charts—weight-for-age-birth-to-2-years-\(percentiles\).pdf](https://cdn.who.int/media/docs/default-source/child-growth/child-growth-standards/indicators/weight-for-age/boys-charts—weight-for-age-birth-to-2-years-(percentiles).pdf).
4. Wikipedia contributors. Uncertainty quantification—Wikipedia, the free encyclopedia; 2021. Available from: https://en.wikipedia.org/w/index.php?title=Uncertainty_quantification&oldid=1015674163.
5. Schulz E, Speekenbrink M, Krause A. A tutorial on Gaussian process regression: modelling, exploring, and exploiting functions. *Journal of Mathematical Psychology*. 2018; 85:1–16. <https://doi.org/10.1016/j.jmp.2018.03.001>
6. Tonner PD, Darnell CL, Engelhardt BE, Schmid AK. Detecting differential growth of microbial populations with Gaussian process regression. *Genome Research*. 2017; 27(2):320–333. <https://doi.org/10.1101/gr.210286.116>
7. Banerjee A, Dunson DB, Tokdar ST. Efficient Gaussian process regression for large datasets. *Biometrika*. 2013; 100(1):75–89. <https://doi.org/10.1093/biomet/ass068>
8. Raissi M, Babae H, Karniadakis GE. Parametric Gaussian process regression for big data. *Computational Mechanics*. 2019; 64(2):409–416. <https://doi.org/10.1007/s00466-019-01711-5>
9. Ziegler G, Ridgway GR, Dahnke R, Gaser C, Alzheimer’s Disease Neuroimaging Initiative. Individualized Gaussian process-based prediction and detection of local and global gray matter abnormalities in elderly subjects. *NeuroImage*. 2014; 97:333–348. <https://doi.org/10.1016/j.neuroimage.2014.04.018>
10. Marquand AF, Rezek I, Buitelaar J, Beckmann CF. Understanding heterogeneity in clinical cohorts using normative models: beyond case-control studies. *Biological Psychiatry*. 2016; 80(7):552–561. <https://doi.org/10.1016/j.biopsych.2015.12.023>
11. Wolfers T, Beckmann CF, Hoogman M, Buitelaar JK, Franke B, Marquand AF. Individual differences v. the average patient: mapping the heterogeneity in ADHD using normative models. *Psychological Medicine*. 2020; 50(2):314–323. <https://doi.org/10.1017/S0033291719000084>
12. Wolfers T, Doan NT, Kaufmann T, Alnæs D, Moberget T, Agartz I, et al. Mapping the heterogeneous phenotype of schizophrenia and bipolar disorder using normative models. *JAMA Psychiatry*. 2018; 75(11):1146–1155. <https://doi.org/10.1001/jamapsychiatry.2018.2467> PMID: 30304337
13. Zabihi M, Oldehinkel M, Wolfers T, Frouin V, Goyard D, Loth E, et al. Dissecting the heterogeneous cortical anatomy of autism spectrum disorder using normative models. *Biological Psychiatry: Cognitive Neuroscience and Neuroimaging*. 2019; 4(6):567–578. <https://doi.org/10.1016/j.bpsc.2018.11.013> PMID: 30799285
14. Rasmussen CE, Williams CKI. *Gaussian processes for machine learning*. MIT Press; 2006.

15. Do CB. Gaussian processes; 2008. Available from: http://cs229.stanford.edu/section/cs229-gaussian_processes.pdf.
16. Mohri M, Rostamizadeh A, Talwalkar A. Foundations of machine learning. MIT press; 2018.
17. Shashua A. Introduction to machine learning: class notes 67577; 2009.
18. Duvenaud DK. Automatic model construction with Gaussian processes. Ph.D. Dissertation, University of Cambridge; 2014. Available from: <https://www.cs.toronto.edu/~duvenaud/thesis.pdf>.
19. Lefebvre A, Delorme R, Delanoë C, Amsellem F, Beggiano A, Germanaud D, et al. Alpha waves as a neuromarker of autism spectrum disorder: the challenge of reproducibility and heterogeneity. *Frontiers in neuroscience*. 2018; 12:662. <https://doi.org/10.3389/fnins.2018.00662> PMID: 30327586
20. Maruani A, Dumas G, Beggiano A, Traut N, Peyre H, Cohen-Freoua A, et al. Morning plasma melatonin differences in autism: beyond the impact of pineal gland volume. *Frontiers in psychiatry*. 2019; 10:11. <https://doi.org/10.3389/fpsy.2019.00011> PMID: 30787884
21. Rigby RA, Stasinopoulos DM. Generalized additive models for location, scale and shape. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*. 2005; 54(3):507–554. <https://doi.org/10.1111/j.1467-9876.2005.00510.x>
22. Borghi E, de Onis M, Garza C, Van den Broeck J, Frongillo EA, Grummer-Strawn L, et al. Construction of the World Health Organization child growth standards: selection of methods for attained growth curves. *Statistics in medicine*. 2006; 25(2):247–265. <https://doi.org/10.1002/sim.2227> PMID: 16143968