A Modified Random Forest Kernel for Highly Nonstationary Gaussian Process Regression with Application to Clinical Data

By

Jacob Paul VanHouten

Thesis
Submitted to the Faculty of the
Graduate School of Vanderbilt University
in partial fulfillment of the requirements
for the degree of

MASTER OF SCIENCE
in

Biostatistics
August, 2016
Nashville, Tennessee

Approved:

Christopher J. Fonnesbeck, PhD

Thomas A. Lasko, MD, PhD

ACKNOWLEDGEMENTS

This work was funded in part by the National Institutes of Health (Grant T32 GM07347) and the National Library of Medicine (Grant T15 LM007450).

I would like to thank my thesis committee for their guidance throughout this work and throughout my education at Vanderbilt. Chris Fonnesbeck, who pulled double duty as a member of PhD committee, has been a fantastic support through both degree programs, and has challenged me to improve as a researcher, a programmer, and a statistician. Tom Lasko, my primary advisor for my PhD, has also been completely supportive of my "double-life" of Biostatistics and Biomedical Informatics.

I owe a debt of gratitude to Michael Matheny, who was the first person to encourage me to pursue the additional MS degree in Biostatistics, and I am glad I listened to him. I would also like to thank others who have given me invaluable assistance and support. I would like to thank

I would also like to express my gratitude to the Departments of Biostatistics, Biomedical Informatics, and the Medical Scientist Training Program. I appreciate the support of the students, faculty, and staff, and am honored to call you all colleagues. I would not have made it through this training without you.

Finally, thank you to my lovely wife Courtney for support and love through this training program as well as through the others.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

An important category of electronic clinical data is repeated events or measurements collected over time as patients interact with the healthcare system. These can be of almost any form, including free text notes, laboratory values, billing codes, clinical communications, and many more. While a static snapshot of medical data can reveal the current health state of a patient, a longitudinal timeseries can incorporate additional information, such as frequency of interactions and trends. Using this more complete information, clinicians can more appropriately take into account the history of a patient, which can allow for a fuller description of the patient's current health state and improved predictions about future health states.

A challenge that arises when using clinical data for decision making is that timeseries are not typically neat representations of a clinical time course. Rather, data are sampled from patients at irregular intervals, often quite sparsely. Furthermore, clinical data are often noisy, containing significant sources of uncertainty and variability.

One solution to handling noisy, irregular, and sparse medical data has been to use Gaussian process regression, a Bayesian nonparametric method, to transform these observations into longitudinal probability distributions. This approach has been used for various machine learning tasks, and allows many standard techniques to be applied to data which would have been difficult to use otherwise [1].

A Gaussian process is a collection of random variables, any finite number of which have a joint Gaussian distribution. Alternatively, Gaussian processes can be thought of as a distribution over functions. They are fully specified by the mean and covariance functions:

$$m(x) = E[f(x)], \tag{1}$$

and

$$k(x, x') = E[(f(x) - m(x))(f(x') - m(x')]. \tag{2}$$

Combining these terms, a set of functions distributed as a Gaussian process can be written as:

$$f(x) \sim \mathcal{GP}\big(m(x), k(x, x')\big) \tag{3}$$

In the medical field, Gaussian process regression has been used for various machine learning tasks. Gaussian process regression has been used to provide predictions and uncertainty estimates for noisy heart rate data [2]. Ghassemi et al. [3] have used multi-task Gaussian processes to model correlations between clinical timeseries of intracranial pressure and arterial blood pressure to predict patient acuity in the intensive care unit. Lasko et al. [4] used time-warped Gaussian process regression to transform uric acid time series into a form suitable for unsupervised feature learning.

In practice, most Gaussian processes assume that the function from which the data are drawn is stationary – in other words, the statistical properties of the function to not vary over its input spaces. Gaussian process inference is straightforward under this assumption, and can be problematic without it.

Unfortunately, clinical data are particularly nonstationary. While homeostasis maintains a tight bound on human physiology during times of overall wellness, acute events can cause rapid, extreme, and sometimes short-lived changes in biological measurements and in rates of interaction with the healthcare systems. In the context of biomedical data retrieved from an electronic health record, standard stationary Gaussian processes are often insufficiently expressive.

Several approaches have been proposed for handling nonstationarity in Gaussian process regression. These have included warping two-dimensional space based on known covariances[5], mixture models [6], inferring an amplitude warping function [2], modeling time-varying length scales between observed points [7,8], inferring an input space warping function [9], and using local estimates of smoothness [10]. Some of these have been applied specifically to clinical data.

However, all of these methods for handling nonstationarity have drawbacks. All are too computationally inefficient to use on tens of sequences each for millions of patient records, and I have found none that are able to cope with the extreme nonstationarity of clinical data accurately enough to use in downstream research.

A promising approach to achieving efficient inference for highly nonstationary Gaussian process regression without these drawbacks is a covariance function using random forests to provide the estimates of covariances [11,12]. A covariance based on partitions from a random forest has been shown to be a valid positive semidefinite kernel [12,13]. The largest drawback of this approach is the piecewise-constant posterior it infers, which leads in turn to specific pathologic behavior in certain circumstances. As such, this work builds on these results and solves the problems caused by the piecewise-constant posterior.

The remainder of this chapter is dedicated to presenting the details of the random forest kernel as originally described, as well as demonstrating the pathology that results from using this kernel in a Gaussian process regression model. In Chapter II, I describe the modification that adds stochasticity to each tree in the random forest to overcome this pathology. I also describe the methods used to compare Gaussian process regressions using this new approach to Gaussian process regressions using the original random forest kernel and two other kernels. Finally, some conclusions and discussion are provided in Chapter III.

Random Forest Kernel

Consider a data set $\mathcal{D} = \{x_i, y_i\}_{i=1}^{N}$, where x_i are the input locations and y_i are the measured values at those locations. In these applications, the x_i are all on-dimensional measurement times, but these methods generalize to higher dimensions.

This kernel divides a data set into partitions via a forest of regression trees. Each tree is trained on a different subset of the data, sampled with replacement, and grown to a depth selected uniformly at random. If we use the notation $c_i(x)$ to denote the terminal leaf to which tree i assigns input instance x, then the proximity $r(x, x')$ between points x and x' is the fraction of time (over all M trees) that the points are assigned to the same leaf:

$$r(x, x') = \frac{1}{M} \sum I[c_i(x) = c_i(x')] \tag{4}$$

This proximity function can be converted to a covariance function by multiplying the signal variance scaling parameter σ_f^2, and defining the covariance function:

$$k(x, x') = \sigma_f^2 r(x, x') \tag{5}$$

This supervised kernel has only the signal variance hyperparameter σ_f^2 to learn. In some instances, the measurement noise parameter σ_n^2 in the likelihood function may also need to be learned. There is no covariance length scale hyperparameter, which highlights the fact that this kernel naturally handles changes in the length scale throughout the input space[12].

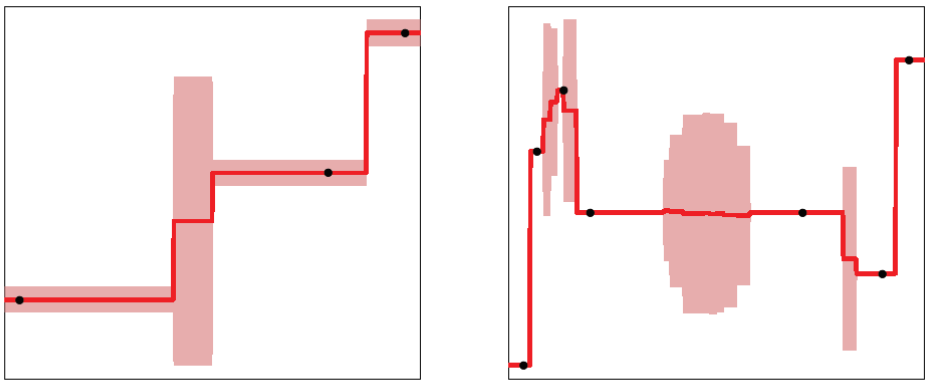
Pathologic Behavior

When used in a Gaussian process, the random forest kernel produces a posterior over piecewise-constant functions (Figure 1). For one-dimensional time data, the random forest partitioning over bootstrap samples causes each function to be potentially discontinuous at locations $d_{ij} = \frac{x_i + x_j}{2}$ for any $x_i \neq x_j \in D$, and constant everywhere else.

I call these points d_{ij} *split locations* because of their origin as points where the data set is split by a decision node in a tree. Despite the potential for $\frac{n(n-1)}{2}$ discontinuities in the function, most of them are not actually realized because they occur only between points that are neighbors in the bootstrap sample. The probability of realizing split location d_{ij} drops nonlinearly as the number of points between x_i and x_j increases.

I have identified a particular pathology of this kernel (Figure 1). Because of the restricted set of locations at which a discontinuity may occur in the latent function, the inferred distribution over those functions is quite unrealistic, with uncertainty being concentrated in restricted regions between observations as determined by the split locations d. In the following chapter, I propose a modification to the random forest kernel, and describe how it can be used to overcome this pathology.

Figure 1. The piecewise-constant nature of the random forest kernel leads to a reproducible pathology. In (a), there is a substantial increase in uncertainty between two points in the center of the graph, which are the only allowable split locations. In (b) there is a similar pathology, where the uncertainty is artificially small near the data points and only allowed to grow at points where the mean function can be discontinuous. Red line: posterior mean function. Red shading: 95% confidence interval. Black points: observed data points.



(a) (b)

CHAPTER II

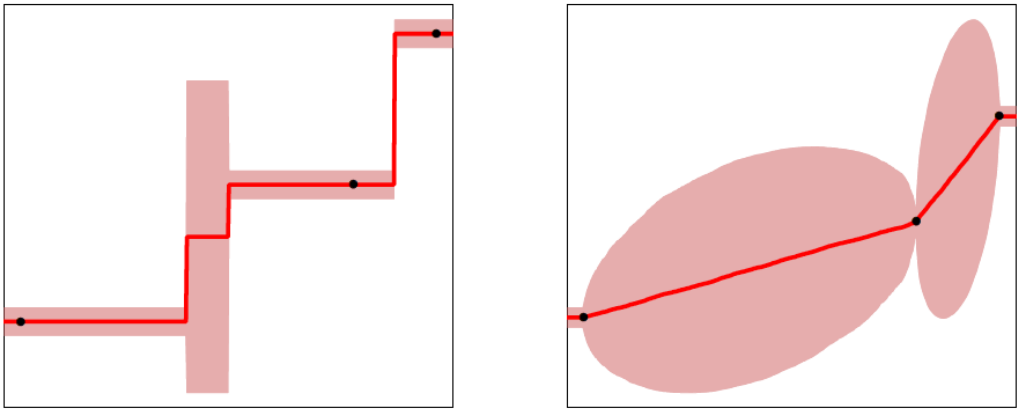
METHODS

Here, I present my proposed modification for the random forest kernel, and describe the methods by which I compared it to the original random forest kernel and two other kernels.

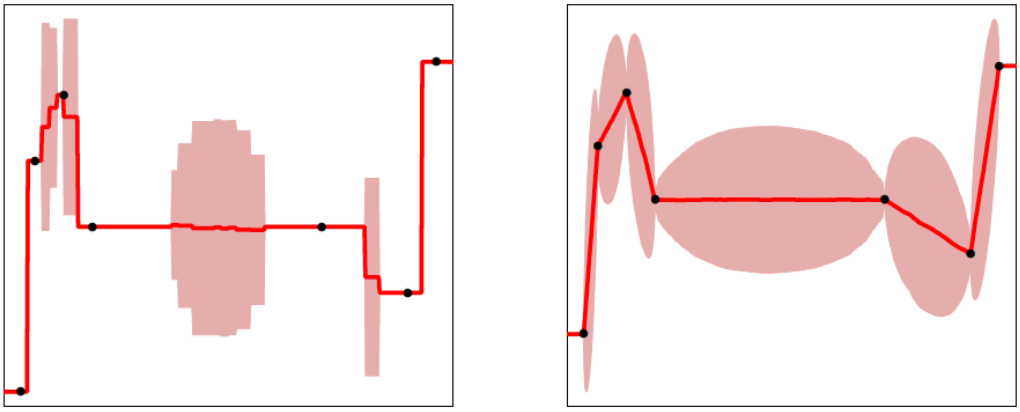
Modified Random Forest Kernel

The modified random forest kernel overcomes this pathology by randomizing the selection of the split locations d that determine the boundaries of $c(x)$. Instead of selecting $d_{ij} = \frac{x_i + x_j}{2}$, I instead selected $d_{ij} \sim Uniform(x_i, x_j)$, which gives more flexibility to the regression trees and removes the source of the pathology (Figure 2b, d). The form of the covariance function remains the same (1), but the means of partitioning $\mathcal{D}$ into leaves $c_i(x)$ as in (1) now uses an additional source of randomness. The distribution of possible split locations becomes a probability density instead of a discrete probability mass function.

Figure 2. Using the modified random forest kernel in a Gaussian process alleviates the pathology present in the original kernel. On the right hand side of the figure, we see the same two pathologic examples from Figure 1 (a, c). On the right, we see that the pathology is resolved in both cases by the modified random forest (b, d). Red line: posterior mean function. Red shading: 95% confidence interval. Black points: observed data points.



(a) (b)

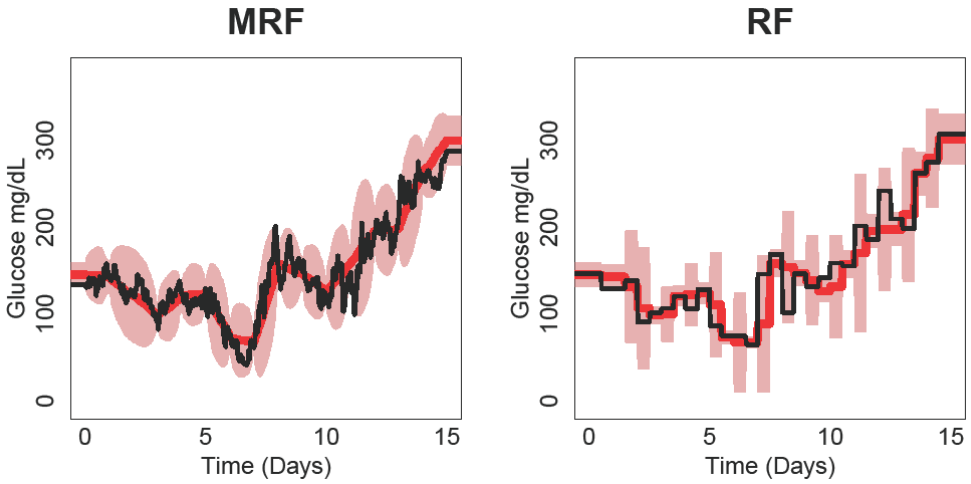


(c) (d)

It may be tempting to avoid the work of building the forest and simply select split locations uniformly across the entire range of x. However, that would provide a different and less useful result than using a random forest to define the splits. The random forest naturally uses the values y_i to take into account the degree of nonstationarity of the latent function, while a uniform distribution over all x_i does not make use of this information. It seems unlikely that a distribution of split locations that does not take into account the measured values at observed locations could ever capture nonstationarity.

The modified random forest kernel produces posterior mean functions that often appear piecewise linear (Figure 3), which is not a realistic estimate of how physiologic functions behave. Draws from the posterior, however, appear much more physiologically realistic, and demonstrate that in this case the posterior mean is not a typical draw. Similarities to both behaviors can be seen with the original random forest kernel, although they are constrained to approximate them using piecewise-constant functions.

Figure 3. The piecewise-linear posterior mean function using the modified random forest kernel is not a typical draw from that posterior (a). The posterior mean of the original random forest kernel (b) does appear to be a typical draw as a consequence of the piecewise-constant constraints it imposes. Red line: posterior mean function. Red shading: 95% confidence interval. Black line: draw from posterior distribution. Black points: observed data points.



Experiments

I performed several experiments using both synthetic and real data to evaluate the performance of the modified random forest kernel.

Methods

I compared the modified random forest kernel to the original random forest kernel, as well as a standard squared exponential Gaussian kernel and a treed Gaussian process regression that fits segments of the input data in a piecewise fashion. I chose treed Gaussian process regression as my comparator nonstationary approach is based on the fact that both Gaussian process regression with modified random forest kernel and treed Gaussian process regression separate the input space as part of their modeling. Thus, it is a reasonable standard for comparison.

Gaussian process regression with a squared exponential kernel was performed with the R package gptk [14], and the treed Gaussian process regression was performed using the R package tgp [15]. I used the randomForest package to produce the proximity matrices for the original and modified random forest kernels [16]. All analysis was carried out using R version 3.0.1 [17]. Experiments were run on a single machine with an Intel Core i7-26000 3.4 Ghz and 16 GB DDR4.

Evaluation Measures

I evaluated the four approaches on synthetic data where the true function $f(x)$ was known for all x, using two measures of fit and one measure of runtime. I evaluated the approaches in terms of negative log probability:

$$\text{NLP} = \sum_i -\log p(y_i^*|D, x_i^*) = \sum_i \frac{1}{2}\log(2\pi\sigma_*^2) + \frac{\left(y_i^* - \hat{f}(x_i^*)\right)^2}{2\sigma_*^2}, \quad (6)$$

where σ_*^2 is the posterior variance, (x_i^*, y_i^*) is a test point, and $\hat{f}(x_i^*)$ is the posterior mean at the test point x_i^*. I also compared the approaches using standardized mean squared error:

$$SMSE = \sum_i \left(\hat{f}(x_i) - f(x_i)\right)^2 / \sum_i \left(f(x_i) - \bar{f}\right)^2, \quad (7)$$

where $\hat{f}(x_i)$ is the posterior mean at the training point x_i, and $\bar{f} = \frac{1}{n}\sum_i f(x_i)$. I also measured the runtime reported in seconds.

Synthetic Data

I tested these approaches on sets of synthetic data, created by adding random Gaussian noise $N(0, \sigma^2)$ for some choice of σ^2 to known functions to create new data sets. The first three of these

functions have previously been used to compare different methods of nonstationary Gaussian process regression [8,10,18]. I used them here because they have become a *de facto* test suite.

The fourth function displays a much higher degree of nonstationarity, manifest as rapid, bursty changes in the underlying function. This function was designed to mimic a specific kind of extreme nonstationarity seen in clinical data, highlighting the issues I seek to address with my kernel.

Function 1. The true function is a spline with three internal knots at (0.2, 0.6, 0.7) and coefficients $\beta =$ (20, 4, 6, 11, 6), with σ = 0.9, and 101 data points sampled for each new data set.

Function 2. The true function is

$$f(x) = \sin(x) + 2\exp(-30x^2), x \in [-2, 2],$$

with $\sigma =$ 0.3, with 101 points sampled for each new data set.

Function 3. The true function is a spline with five knots located at (0.4, 0.4, 0.4, 0.4, 0.7) and coefficients (2, -5, 5, 2, -3, -1, 2) and σ = 0.55, with 201 points sampled for each new data set.

Function 4. The true function is a sine wave with discontinuous jumps,

$$f(x) = \sin(0.05x) + I[100 < x < 135] * -0.1x + I[250 < x < 275] * 0.1x$$
$$+I[300 < x < 310 * (-1 * (x - 305)^2 - 25),$$

with σ = 0.5 and 100 points sampled for each new data set.

I compared Gaussian process regression using the modified random forest kernel to the three other approaches described above. Applying these methods to data drawn from these known functions, I evaluated the performance of the regression methods using negative log probability (3),

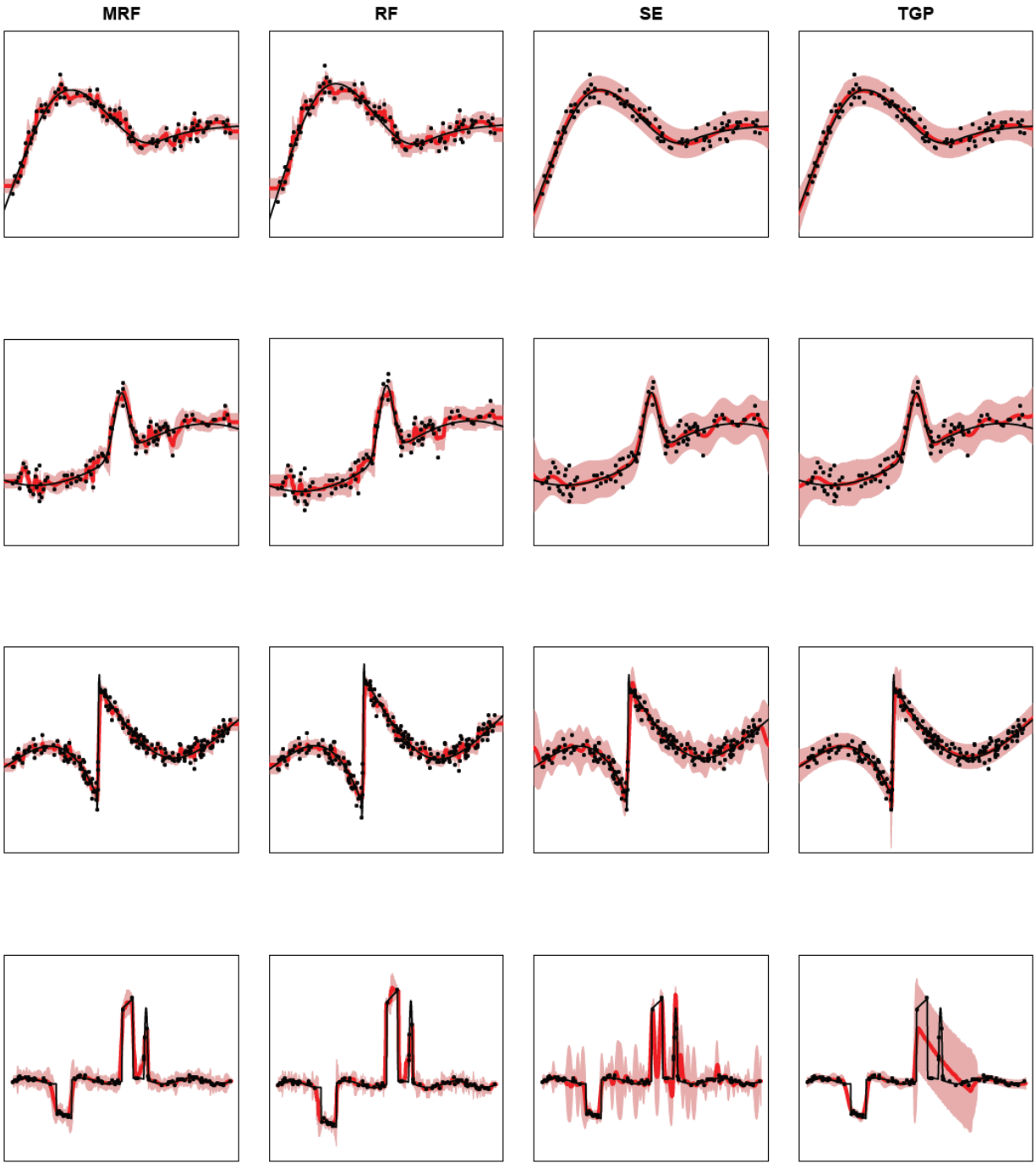
standardized mean squared error (4), and runtime (Table 1). All functions were evaluated at a regularly spaced grid of 500 points, and both random forest kernels used forests with 500 trees.

For the first three functions, I optimized the fit of both the signal variance σ_f^2 and noise parameter σ_n^2 for all methods. For the fourth function, I fixed the measurement noise parameter to its known value. This is consistent with the approach I use later for medical data, where the value of the measurement noise is generally known. Typical regression fits to the data for each method are shown in Figure 3.

Table 1. Modified random forest kernel regression performs favorably, even on smooth, stationary data, and excels on extremely nonstationary data. MRF: modified random forest kernel. RF: random forest kernel. SE: squared exponential kernel. TGP: treed Gaussian process regression. NLP: negative log probability. SMSE: standardized mean squared error.

	MRF	RF	SE	TGP
Function 1				
NLP	1.92 [1.66, 2.44]	2.25 [1.96, 3.97]	1.46 [1.41, 1.49]	1.38 [1.33, 1.43]
SMSE	0.14 [0.12, 0.17]	0.15 [0.13, 0.18]	0.03 [0.01, 0.05]	0.02 [0.01, 0.03]
Runtime(s)	12.7 [12.4, 12.9]	12.5 [12.3, 12.7]	0.9 [0.8, 1.0]	44.0 [39.8, 46.8]
Function 2				
NLP	0.53 [0.39, 0.73]	0.57 [0.48, 0.84]	0.39 (0.35, 0.43)	0.57 [0.56, 0.61]
SMSE	0.05 [0.04, 0.06]	0.06 [0.05, 0.07]	0.04 [0.03, 0.05]	0.03 [0.02, 0.04]
Runtime(s)	12.7 [12.6, 12.9]	12.3 [12.1, 12.4]	0.4 [0.4, 0.5]	28.0 [21.7, 36.2]
Function 3				
NLP	1.11 [1.05, 1.33]	1.28 [1.07, 1.52]	1.22 [1.17, 1.31]	1.04 [0.99, 1.09]
SMSE	0.10 [0.08, 0.13]	0.11 [0.09, 0.14]	0.31 [0.30, 0.32]	0.11 [0.07, 0.12]
Runtime(s)	19.8 [19.3, 20.2]	19.5 [18.9, 19.8]	1.9 [1.8, 2.3]	92.6 [86.0, 100.0]
Function 4				
NLP	1.83 [1.63, 2.08]	4.30 [2.55, 6.04]	2.43 [2.21, 2.77]	2.39 [2.06, 2.96]
SMSE	0.12 [0.09, 0.14]	0.15 [0.10, 0.19]	0.18 [0.14, 0.25]	0.23 [0.13, 0.32]
Runtime(s)	10.9 [10.7, 11.0]	10.6 [10.5, 10.8]	0.5 [0.5, 0.6]	21.4 [19.7, 25.7]

Figure 4. The modified random forest kernel produces posterior distributions which are not smooth, but which naturally handle nonstationarity. Each row represents one data set drawn from each of the four synthetic functions. For these data sets the horizontal and vertical scales are arbitrary, and therefore omitted for clarity. Red line: posterior mean function. Red shading: 95% confidence interval. Black line: known function. Black points: observed data points.



Real Data

After obtaining IRB approval, I extracted the full sequence of measurements for a small number of laboratory tests from Vanderbilt's Synthetic Derivative, the deidentified mirror of the electronic health record [19], which contains up to 30 years of clinical data on over 2 million patients.

I extracted measurements of blood urea nitrogen and glucose, which are commonly measured in many circumstances and display the kind of bursty nonstationarity that makes medical data challenging to fit with stationary regression techniques.

Rather than inferring a parameter for the measurement noise σ_n^2, I fixed the parameter at the value defined by the national Clinical Laboratory Improvement Amendments (CLIA) requirements for laboratory tests[20]. While measurement error for a given test or even a given machine can vary significantly over time, CLIA limits have remained largely unchanged for over a decade, providing a stable bound on the measurement error associated with various laboratory tests. For blood urea nitrogen, the CLIA-defined uncertainty is the maximum of 9% of the laboratory measurement value and 2mg/dL. For glucose, it is the maximum of 10% of the laboratory measurement value and 6mg/dL. As a result of this variable uncertainty, I set the noise parameter to a different value at each observed data point; this is allowed by most Gaussian process regression packages. All functions were evaluated at a regularly spaced grid of 1000 points. The original random forest kernel and modified random forest kernel used forests with 1000 trees. Typical regression fits to selected sequences that display varying degrees of bursty nonstationarity are shown in Figures 5 and 6. As I fit the regression models using negative log probability minimization and I did not know the true underlying function, I could not compare these approaches on the same evaluation measures I used on the synthetic data. Instead, I compared them qualitatively to describe differences in fit (Figures 5 and 6).

Figure 5. The modified random forest kernel handles many different nonstationary functions on real clinical data. While the squared exponential kernel and treed Gaussian process regression often underfit nonstationary functions, the random forest kernel and modified random forest kernel do not demonstrate this limitation. Red line: posterior mean function. Red shading: 95% confidence interval. Black points: observed data points.

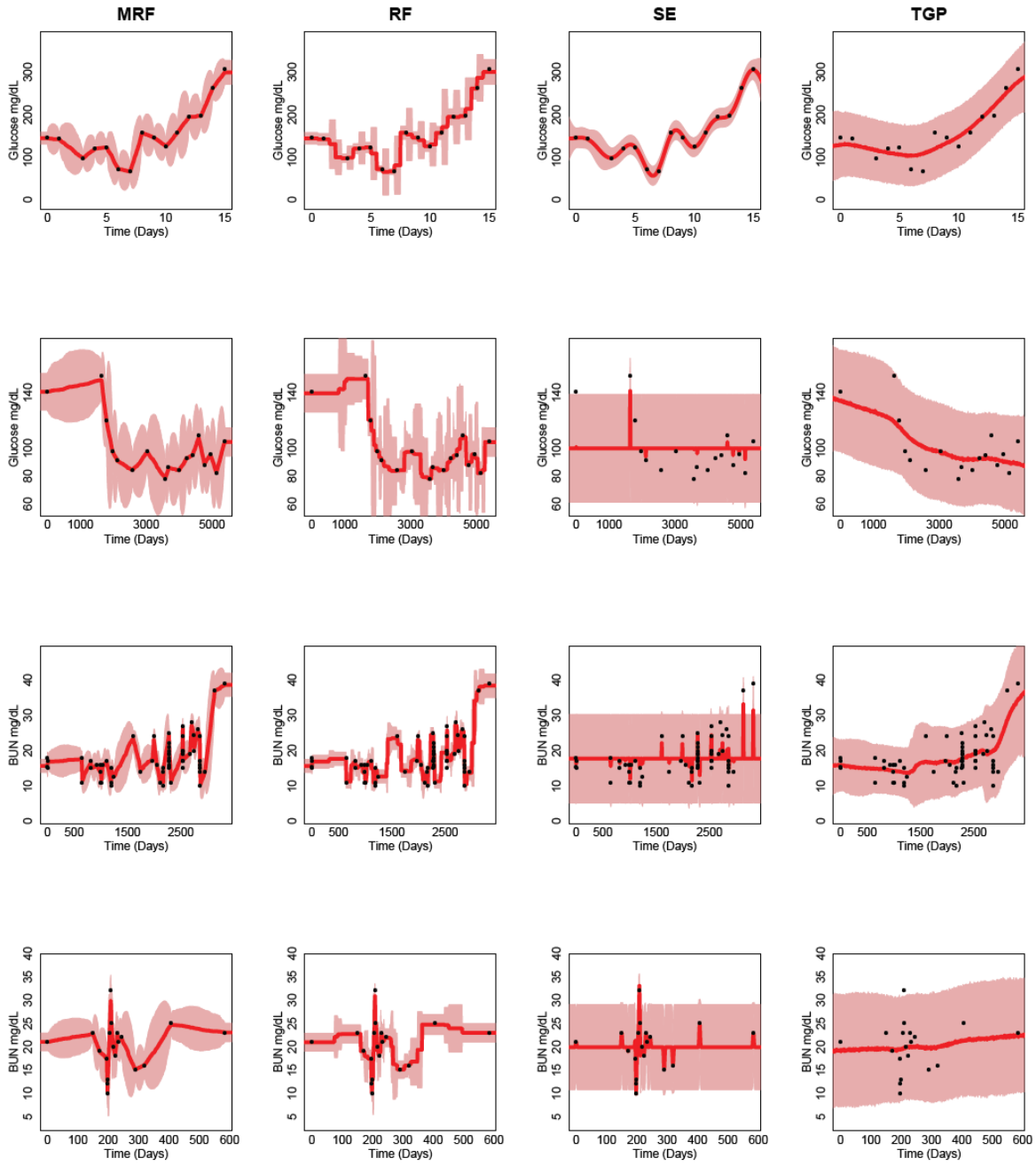
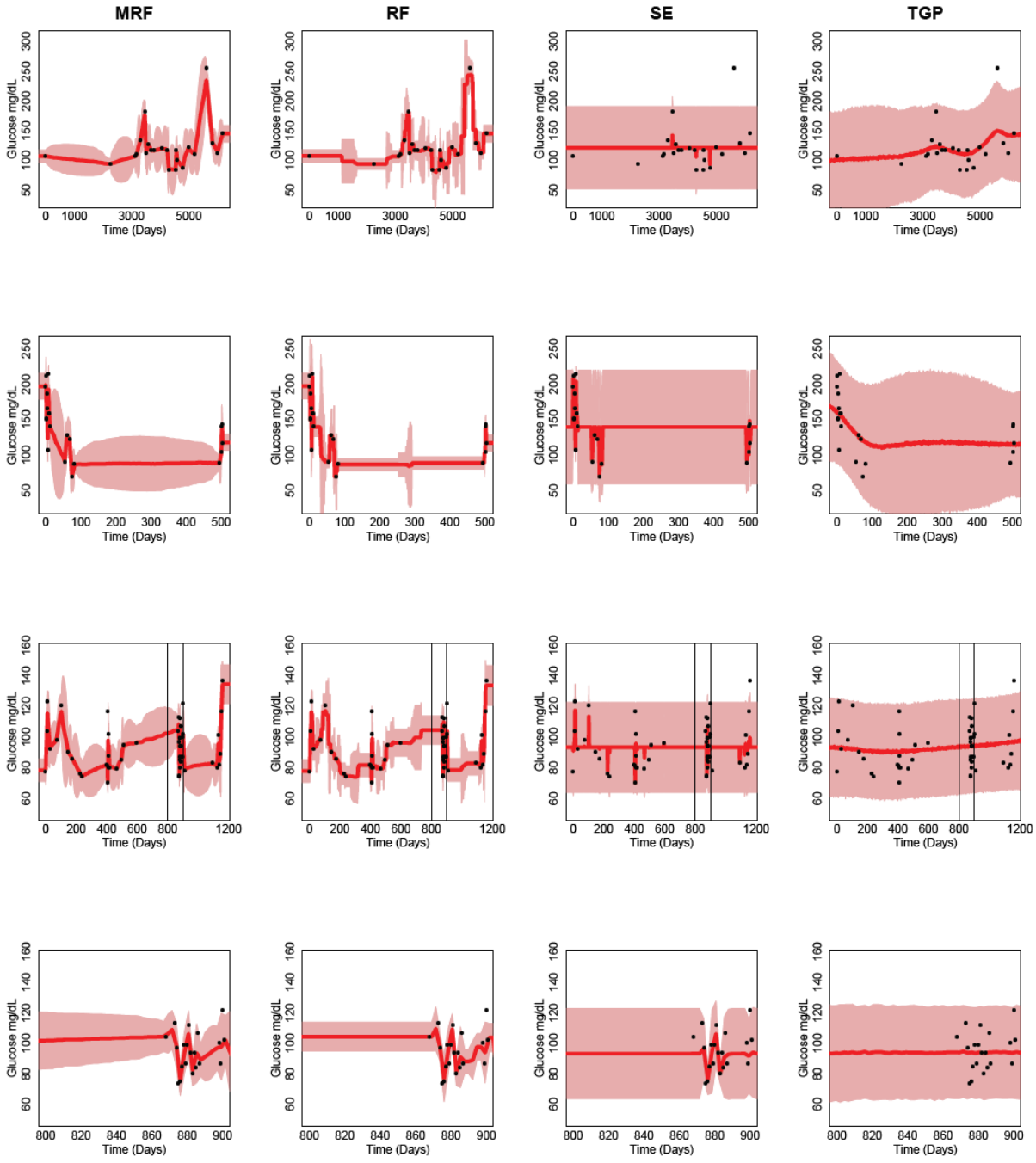


Figure 6. Additional fits of real clinical data. In the second row, we see an example of the modified random forest kernel overcoming the pathology of random forest kernels described earlier in this thesis. The fourth row is a magnification of the third row, of the boxed area between days 800 and 900. Red line: posterior mean function. Red shading: 95% confidence interval. Black points: observed data points.



CHAPTER III

DISCUSSION

In this thesis, I present a modified random forest kernel. When used in Gaussian process regression, this new kernel remains fast and nonstationary, and also removes a pathology that results from the piecewise-constant posterior created by the original random forest kernel. For data sets with very few instances or for which there are long stretches of time between measurements, the pathology of the original random forests is easily seen and the modified kernel demonstrates a clear benefit. However, the pathology exists to some degree in every data set.

This modified random forest kernel performs favorably against other methods. The modified random forest kernel consistently outperforms the original random forest kernel in terms of negative log probability and standardized mean squared error, possibly due to the more realistic posterior uncertainty estimates and the flexibility of the posterior mean function. On the first and second functions, which are fairly close to stationary, the squared exponential kernel and treed Gaussian process regression outperform both random forest kernels by a wide margin in terms of both negative log probability and standardized mean squared error. On the third function in particular, treed Gaussian process regression provides a superior fit to the data. This makes some sense, given this methods' approach of fitting Gaussian processes over individual segments of the input data. On the fourth function, which was constructed to mimic the bursty nonstationarity of medical data, the regression using the modified random forest kernel performed much better than the others.

I did not optimize the implementation of the modified random forest kernel for speed, and thus this method was slightly slower than the original random forest kernel due to the additional

processing needed to draw the split points from a uniform distribution. Both of the random forest kernel regresisons were slower than squared exponential Gaussian process regression, but both were also faster than the treed Gaussian process regression. Future work on the modified random forest kernel, such as byte compiling and growing the forests in parallel, will likely improve the speed of this method.

The modified random forest kernel resolves the pathology seen with the original random forest kernel; an example of this can be seen in the second row of Figure 5. When faced with the task of fitting highly nonstationary data, the squared exponential kernel and treed Gaussian process methods each converged to one of the two possible modes: short length scale with high signal variance σ_f^2, or moderate length scale with high measurement noise σ_n^2. Both are suboptimal fits, but are the only two reasonable alternatives given their stationarity constraints [1]. Regression using either of the random forest kernels found subjectively better fits.

Using the modified random forest kernel and Gaussian process regression on large sets of continuous medical data could allow for automatic, fast construction of compact longitudinal data representations from noisy, sparse, and irregular observations. Such representations of medical data could be used as input to standard machine learning algorithms, allowing improved mining of clinical relationships.

REFERENCES

[1] Rasmussen CE, Williams CKI. Gaussian Processes for Machine Learning. 2006.

[2] Adams RP, Stegle O. Gaussian process product models for nonparametric nonstationarity. Proc. 25th Int. Conf. Mach. Learn. - ICML '08, New York, New York, USA: ACM Press; 2008, p. 1–8. doi:10.1145/1390156.1390157.

[3] Ghassemi M, Pimentel M. A Multivariate Timeseries Modeling Approach to Severity of Illness Assessment and Forecasting in ICU with Sparse, Heterogeneous Clinical Data. Proc. Twenty-Ninth AAAI Conf. Artif. Intell., 2015.

[4] Lasko TA, Denny JC, Levy MA. Computational phenotype discovery using unsupervised feature learning over noisy, sparse, and irregular clinical data. PLoS One 2013;8:e66341. doi:10.1371/journal.pone.0066341.

[5] Sampson PD, Guttorp P. Nonparametric Estimation of Nonstationary Spatial Covariance Structure. J Am Stat Assoc 1992;87:108. doi:10.2307/2290458.

[6] Pfingsten T, Kuss M, Rasmussen CE. Nonstationary Gaussian Process Regression using a Latent Extension of the Input Space ∗ n.d.

[7] Gibbs MN. Bayesian Gaussian Processes for Regression and Classification. Dissertation 1997.

[8] Paciorek CJ, Schervish MJ. Nonstationary covariance functions for Gaussian process regression. Adv Neural Inf Process Syst 16 Proc 2003 Conf 2004:273–80.

[9] Lasko TA. Nonstationary Gaussian Process Regression for Evaluating Clinical Laboratory Test Sampling Strategies. Proc. Twenty-Ninth AAAI Conf. Artif. Intell., 2015, p. 1777–83.

[10] Plagemann C, Kersting K, Burgard W. Nonstationary Gaussian Process Regression Using Point Estimates of Local Smoothness. Mach. Learn. Knowl. Discov. Databases, Berlin, Heidelberg: Springer Berlin Heidelberg; 2008, p. 204–19. doi:10.1007/978-3-540-87481-2_14.

[11] Torkkola K, Tuv E. Ensemble Learning with Supervised Kernels. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 3720 LNAI, 2005, p. 400–11. doi:10.1007/11564096_39.

[12] Davies A, Ghahramani Z. The Random Forest Kernel and other kernels for big data from random partitions 2014.

[13] Scornet E. Random Forests and Kernel Methods. IEEE Trans Inf Theory 2016;62:1485–500. doi:10.1109/TIT.2016.2514489.

[14] Alfredo Kalaitzis A, Alfredo Kalaitzis M. gptk: Gaussian Processes Tool-Kit 2015.

[15] Gramacy RB, Lee HKH. Bayesian Treed Gaussian Process Models With an Application to Computer Modeling. J Am Stat Assoc 2008;103:1119–30. doi:10.1198/016214508000000689.

[16] Liaw A, Wiener M. Classification and Regression by randomForest. R News 2002;2:18–22.

[17] R Core Team. R: A Language and Environment for Statistical Computing 2013.

[18] Dimatteo I, Genovese CR, Kass RE. Bayesian curve-fitting with free-knot splines. Biometrika 2001;88:1055–71. doi:10.1093/biomet/88.4.1055.

[19] Roden D, Pulley J, Basford M, Bernard G, Clayton E, Balser J, et al. Development of a Large-Scale De-Identified DNA Biobank to Enable Personalized Medicine. Clin Pharmacol Ther 2008;84:362–9. doi:10.1038/clpt.2008.89.

[20] CLIA program; simplifying CLIA regulations relating to accreditation, exemption of laboratories under a state licensure program, proficiency testing, and inspection--HCFA. Final rule. Fed Regist 1998;63:26722–38.