Gaussian Processes and Limiting Linear Models

Robert B. Gramacy & Herbert K. H. Lee
Department of Applied Math & Statistics
University of California, Santa Cruz
Santa Cruz, CA 96054
{rbgramacy, herbie}@ams.ucsc.edu

Abstract

Gaussian processes (GPs) retain the linear model (LM) either as a special case, or in the limit. We show how this relationship can be exploited when the data are at least partially linear. However from the prospective of the Bayesian posterior, the GPs which encode the LM either have probability of nearly zero or are otherwise unattainable without the explicit construction of a prior with the limiting linear model (LLM) in mind. We develop such a prior, and show that its practical benefits extend well beyond the computational and conceptual simplicity of the LM. For example, linearity can be extracted on a per-dimension basis, or can be combined with treed partition models to yield a highly efficient nonstationary model. Our approach is demonstrated on synthetic and real datasets of varying linearity and dimensionality. Comparisons are made to other approaches in the literature.

1 Background

The Gaussian Process (GP) is a common model for fitting arbitrary functions or surfaces, because of its nonparametric flexibility [3]. This paper explores the connections between GPs and linear models. Combining this union with treed GPs [7] leads to a fully flexible yet computationally efficient model. Consider the following Bayesian hierarchical model for a GP for n inputs $\mathbf{X}$ of dimension m_X, and n responses $\mathbf{y}$:

$$\begin{aligned} \mathbf{y}|\boldsymbol{\beta}, \sigma^2, \mathbf{K} &\sim N(\mathbf{F}\boldsymbol{\beta}, \sigma^2\mathbf{K}) & \sigma^2 &\sim IG(\alpha_\sigma/2, q_\sigma/2) \\ \boldsymbol{\beta}|\sigma^2, \tau^2, \mathbf{W} &\sim N(\boldsymbol{\beta}_0, \sigma^2\tau^2\mathbf{W}) & \tau^2 &\sim IG(\alpha_\tau/2, q_\tau/2) \\ \boldsymbol{\beta}_0 &\sim N(\boldsymbol{\mu}, \mathbf{B}) & \mathbf{W}^{-1} &\sim W((\rho\mathbf{V})^{-1}, \rho) \end{aligned} \qquad (1)$$

with $\mathbf{F} = (\mathbf{1}, \mathbf{X})$, and $\mathbf{I}$ is a $(m_X + 1) \times (m_X + 1)$ matrix. N, IG and W are the Normal, Inverse-Gamma and Wishart distributions, respectively. Constants $\boldsymbol{\mu}, \mathbf{B}, \mathbf{V}, \rho, \alpha_\sigma, q_\sigma, \alpha_\tau, q_\tau$ are treated as known. The correlation matrix $\mathbf{K}$ is constructed from a correlation function $K(\cdot, \cdot)$ of the form $K(\mathbf{x}_j, \mathbf{x}_k) = K^*(\mathbf{x}_j, \mathbf{x}_k) + g\delta_{j,k}$ where $\delta_{\cdot,\cdot}$ is the Kronecker delta function, g is called the *nugget* parameter and is included in order to interject measurement error (or random noise) into the stochastic process, and K^* is a *true* correlation which we take to be from the separable power family (generalizations are straightforward):

$$K^*(\mathbf{x}_j, \mathbf{x}_k|\mathbf{d}) = \exp\left\{-\textstyle\sum_{i=1}^{m_X}(x_{ij} - x_{ik})^2/d_i\right\}. \qquad (2)$$

The specification of priors for K, K^*, and their parameters $\mathbf{d}$ and g will be deferred until later, as their construction will be a central part of this paper. With the separable power family some input variables can be modeled as more highly correlated than others. The (non-separable) isotropic exponential family is a special case (when $d = d_i$, for $i = 1, \ldots, m_X$).

Posterior inference and estimation is straightforward using the Metropolis-Hastings and Gibbs algorithms [7]. We shall not duplicate the estimation results here due to space constraints, but since some of the prediction equations will be useful later we remark that the predicted value of y at $\mathbf{x}$ is normally distributed with mean and variance

$$\hat{y}(\mathbf{x}) = \mathbf{f}^\top(\mathbf{x})\tilde{\boldsymbol{\beta}} + \mathbf{k}(\mathbf{x})^\top \mathbf{K}^{-1}(\mathbf{y} - \mathbf{F}\tilde{\boldsymbol{\beta}}), \quad \hat{\sigma}(\mathbf{x})^2 = \sigma^2[\kappa(\mathbf{x},\mathbf{x}) - \mathbf{q}^\top(\mathbf{x})\mathbf{C}^{-1}\mathbf{q}(\mathbf{x})], \tag{3}$$

where $\tilde{\boldsymbol{\beta}}$ is the posterior mean estimate of $\boldsymbol{\beta}$, $\mathbf{C}^{-1} = (\mathbf{K} + \tau^2\mathbf{F}\mathbf{F}^\top)^{-1}$, $\mathbf{q}(\mathbf{x}) = \mathbf{k}(\mathbf{x}) + \tau^2\mathbf{F}\mathbf{f}(\mathbf{x})$, and $\kappa(\mathbf{x},\mathbf{y}) = K(\mathbf{x},\mathbf{y}) + \tau^2\mathbf{f}^\top(\mathbf{x})\mathbf{f}(\mathbf{y})$, defining $\mathbf{f}^\top(\mathbf{x}) = (1, \mathbf{x}^\top)$, and $\mathbf{k}(\mathbf{x})$ is a $n-$vector with $\mathbf{k}_{\nu,j}(\mathbf{x}) = K(\mathbf{x}, \mathbf{x}_j)$, for all $\mathbf{x}_j \in \mathbf{X}$, the training data.

A treed GP [7] is a generalization of the CART (Classification and Regression Tree) model [1] that uses GPs at the leaves of the tree in place of the usual constant values. The Bayesian interpretation requires a prior be placed on the tree and GP parameterizations. Sampling commences with Reversible Jump (RJ) MCMC which allows for a simultaneous fit of the tree and the GPs at its leaves.

2 Linear Limiting Models

A special limiting case of the Gaussian process model is the standard linear model. Replacing the top (likelihood) line in the hierarchical model given in Equation (1)

$$\mathbf{y}|\boldsymbol{\beta}, \sigma^2, \mathbf{K} \sim N(\mathbf{F}\boldsymbol{\beta}, \sigma^2\mathbf{K}) \qquad \text{with} \qquad \mathbf{y}|\boldsymbol{\beta}, \sigma^2 \sim N(\mathbf{F}\boldsymbol{\beta}, \sigma^2\mathbf{I}),$$

where $\mathbf{I}$ is the $n \times n$ identity matrix, gives a parameterization of a linear model. From a phenomenological perspective, GP regression is more flexible than standard linear regression in that it can capture nonlinearities in the interaction between covariates ($\mathbf{x}$) and responses (y). From a modeling perspective, the GP can be more than just overkill for linear data. Parsimony and over-fitting considerations are just the tip of the iceberg. It is also unnecessarily computationally expensive, as well as numerically unstable. Specifically, it requires the inversion of a large covariance matrix— an operation whose computing cost grows with the cube of the sample size. Moreover, large finite $\mathbf{d}$ parameters can be problematic from a numerical perspective because, unless g is also large, the resulting covariance matrix can be numerically singular when the off-diagonal elements of $\mathbf{K}$ are nearly one.

It is common practice to scale the inputs ($\mathbf{x}$) either to lie in the unit cube, or to have a mean of zero and a range of one. Scaled data and mostly linear predictive surfaces can result in almost singular covariance matrices even when the range parameter is relatively small ($2 < d \ll \infty$). So for some parameterizations, the GP is operationally equivalent to the limiting linear model (LLM), but comes with none of its benefits (e.g. speed and stability). As this paper demonstrates, exploiting and/or manipulating such equivalence can be of great practical benefit. As Bayesians, this means constructing a prior distribution on $\mathbf{K}$ that makes it clear in which situations each model is preferred (i.e., when should $\mathbf{K} \rightarrow c\mathbf{I}$?). Our key idea is to specify a prior on a "jumping" criterion between the GP and its LLM, thus setting up a Bayesian model selection/averaging framework.

Theoretically, there are only two parameterizations to a GP correlation structure (K) which encode the LLM. Though they are indeed well-known, without intervention they are quite unhelpful from the perspective of *practical* estimation and inference. The first one is when the range parameter (d) is set to zero. In this case $\mathbf{K} = (1 + g)\mathbf{I}$, and the result is clearly a linear model. The other parameterization may be less obvious.

Cressie [3] (in Section 3.2.1) analyzes the "effect of variogram parameters on kriging" paying special attention to the nugget (g) and its interaction with the range parameter. He remarks that the larger the nugget the more the kriging interpolator smoothes and in the limit predicts with the linear mean. He later remarks on the interplay between the range and nugget parameter in determining the kriging neighborhood. Specifically, a large nugget coupled with a large range drives the interpolator towards the linear mean. This is refreshing since constructing a prior for the LLM by exploiting the former GP parameterization (range $d \to 0$) is difficult, and for the latter (nugget $g \to \infty$) near impossible. Cressie hints that an (essentially) linear model may be attainable with nonzero d and finite g.

3 Model selection prior

With the ideas outlined above, we set out to construct the prior for the "mixture" of the GP with its LLM. The key idea is an augmentation of the parameter space by m_X indicators $\mathbf{b} = \{b\}_{i=1}^{m_X} \in \{0,1\}^{m_X}$. The boolean b_i is intended to select either the GP ($b_i = 1$) or its LLM for the i^{th} dimension. The actual range parameter used by the correlation function is multiplied by $\mathbf{b}$: e.g. $K^*(\cdot,\cdot|\mathbf{b}^\top\mathbf{d})$. To encode our preference that GPs with larger range parameters be more likely to "jump" to the LLM, the prior on b_i is specified as a function of the range parameter d_i: $p(b_i, d_i) = p(b_i|d_i)p(d_i)$.

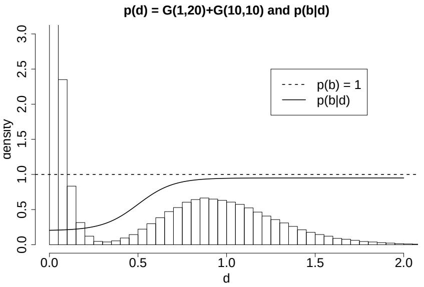

Figure 1: Prior distribution for the boolean (b) superimposed on $p(d)$.

Probability mass functions which increase as a function of d_i, e.g.,

$$p_{\gamma,\theta_1,\theta_2}(b_i = 0|d_i) = \theta_1 + (\theta_2 - \theta_1)/(1 + \exp\{-\gamma(d_i - 0.5)\}) \tag{4}$$

with $0 < \gamma$ and $0 \leq \theta_1 \leq \theta_2 < 1$, can encode such a preference by calling for the exclusion of dimensions i with with large d_i when constructing $\mathbf{K}$. Thus b_i determines whether the GP or the LLM is in charge of the marginal process in the i^{th} dimension. Accordingly, θ_1 and θ_2 represent minimum and maximum probabilities of jumping to the LLM, while γ governs the rate at which $p(b_i = 0|d_i)$ grows to θ_2 as d_i increases. Figure 1 plots $p(b_i = 0|d_i)$ for $(\gamma, \theta_1, \theta_2) = (10, 0.2, 0.95)$ superimposed on a convenient $p(d_i)$ which we take to be a mixture of Gamma distributions,

$$p(d) = [G(d|\alpha = 1, \beta = 20) + G(d|\alpha = 10, \beta = 10)]/2, \tag{5}$$

representing a population of GP parameterizations for wavy surfaces (small d) and a separate population of those which are quite smooth or approximately linear. We take θ_2 to be strictly less than one so as not to preclude a GP which models a genuinely nonlinear surface using an uncommonly large range setting.

The implied prior probability of the full m_X-dimensional LLM is

$$p(\text{linear model}) = \prod_{i=1}^{m_X} p(b_i = 0|d_i) = \prod_{i=1}^{m_X} \left[\theta_1 + \frac{\theta_2 - \theta_1}{1 + \exp\{-\gamma(d_i - 0.5)\}}\right]. \tag{6}$$

Notice that the resulting process is still a GP if any of the booleans b_i are one. The primary computational advantage associated with the LLM is foregone unless all of the b_i's are zero. However, the intermediate result is a unique transitionary model lying somewhere between the GP and the LLM. It allows for the implementation of semiparametric stochastic processes like $Z(\mathbf{x}) = \boldsymbol{\beta} f(\mathbf{x}) + \varepsilon(\tilde{\mathbf{x}})$ representing a piecemeal spatial extension of a simple linear model. The first part ($\boldsymbol{\beta} f(\mathbf{x})$) of the process is linear in some known function of the the full set of covariates $\mathbf{x} = \{x_i\}_{i=1}^{m_X}$, and $\varepsilon(\cdot)$ is a spatial random process (e.g. a GP) which acts on a subset of the covariates $\tilde{\mathbf{x}}$. Such models are commonplace in the statistics community [4]. Traditionally, $\tilde{\mathbf{x}}$ is determined and fixed *a priori*. The separable boolean prior in (4) implements an adaptively semiparametric process where the subset $\tilde{\mathbf{x}} = \{x_i : b_i = 1, i = 1, \dots, m_X\}$ is given a prior distribution, instead of being fixed.

3.1 Prediction

Prediction under the limiting GP model is a simplification of Eq. (3) when it is known that $\mathbf{K} = (1+g)\mathbf{I}$. A characteristic of the standard linear model is that all input configurations $(\mathbf{x})$ are treated as independent conditional on knowing $\boldsymbol{\beta}$. Additionally, this implies that in (3) the terms $k(\mathbf{x})$ and $K(\mathbf{x},\mathbf{x})$ are zero for all $\mathbf{x}$. Thus, the predicted value of y at $\mathbf{x}$ is normally distributed with mean $\hat{y}(\mathbf{x}) = \mathbf{f}^\top(\mathbf{x})\tilde{\boldsymbol{\beta}}$ and variance

$$\sigma^2[1 + \tau^2\mathbf{f}^\top(\mathbf{x})\mathbf{f}(\mathbf{x}) - \tau^2\mathbf{f}^\top(\mathbf{x})\mathbf{F}^\top((1+g)\mathbf{I} + \tau^2\mathbf{F}\mathbf{F}^\top)^{-1}\mathbf{F}\mathbf{f}(\mathbf{x})\tau^2].$$

It is helpful to re-write the above expression for the variance as

$$\sigma^2\left[1 + \tau^2\mathbf{f}^\top(\mathbf{x})\mathbf{f}(\mathbf{x}) - \frac{\tau^2}{1+g}\mathbf{f}^\top(\mathbf{x})\mathbf{F}^\top\left(\mathbf{I} + \frac{\tau^2}{1+g}\mathbf{F}\mathbf{F}^\top\right)^{-1}\mathbf{F}\mathbf{f}(\mathbf{x})\tau^2\right].$$

Using a matrix inversion lemma called the Woodbury formula [see `Mathworld`: http://mathworld.wolfram.com/WoodburyFormula.html] one can show that

$$\hat{\sigma}(\mathbf{x})^2 = \sigma^2\left[1 - \mathbf{f}^\top(\mathbf{x})\left(\tau^{-2} + \mathbf{F}^\top\mathbf{F}/(1+g)\right)^{-1}\mathbf{f}(\mathbf{x})\right].$$

Not only is this a simplification of the predictive variance given in (3), but Gramacy et al. [7] give an expression for the posterior variance of the linear regression coefficients $\boldsymbol{\beta}$, namely $\mathbf{V}_{\tilde{\beta}}$, which should make it look more familiar. Writing $\mathbf{V}_{\tilde{\beta}}$ with $\mathbf{K}^{-1} = \mathbf{I}/(1+g)$ and setting $\mathbf{W} \equiv \mathbf{I}$ gives

$$\mathbf{V}_{\tilde{\beta}} = \left(\tau^{-2} + \mathbf{F}^\top\mathbf{F}(1+g)\right)^{-1} \qquad \text{and then:} \qquad \hat{\sigma}(\mathbf{x})^2 = \sigma^2\left[1 - \mathbf{f}^\top(\mathbf{x})\mathbf{V}_{\tilde{\beta}}\mathbf{f}(\mathbf{x})\right]. \quad (7)$$

This is just the usual posterior predictive density at $\mathbf{x}$ under the standard linear model: $y(\mathbf{x}) \sim N[\mathbf{f}^\top(\mathbf{x})\hat{\boldsymbol{\beta}}, \sigma^2(1 - \mathbf{f}^\top(\mathbf{x})\mathbf{V}_{\tilde{\beta}}\mathbf{f}(\mathbf{x}))]$. This means that we have a choice when it comes to obtaining samples from the posterior predictive distribution under the LLM. We prefer (7) over (3) because the latter involves inverting the $n \times n$ matrix $\mathbf{I} + \tau^2\mathbf{F}\mathbf{F}^\top/(1+g)$, whereas the former only requires the inversion of an $(m_X+1) \times (m_X+1)$ matrix.

4 Implementation, results, and comparisons

Here, the GP with jumps to the LLM (hereafter GP LLM) is illustrated on synthetic and real data. This work grew out of research focused on extending the reach of the treed GP model presented by Gramacy et al. [7], whereby the data are recursively partitioned and a separate GP is fit in each partition. Thus most of our experiments are in this context, though in Section 4.3 we demonstrate an example without treed partitioning. Partition models are an ideal setting for evaluating the utility of the GP LLM as linearity can be extracted in large areas of the input space. The result is a uniquely tractable nonstationary spatial model.

Sampling from the posterior can be accomplished by Gibbs steps for all but $\mathbf{d}$ and g [7]. Proposals for the booleans $\mathbf{b}$ are drawn from the prior, conditional on $\mathbf{d}$, and accepted and rejected on the basis of the constructed covariance matrix $\mathbf{K}$. The same prior parameterizations are used for all experiments with a couple reasonable exceptions, the idea being to develop a method that works "right out of the box" as much as possible.

4.1 Synthetic exponential data

Consider the 2-d input space $[-2,6] \times [-2,6]$ in which the true response is given by $Y(\mathbf{x}) = x_1\exp(-x_1^2 - x_2^2) + \epsilon$, where $\epsilon \sim N(0, \sigma = 0.001)$. Figure 2 summarizes the consequences

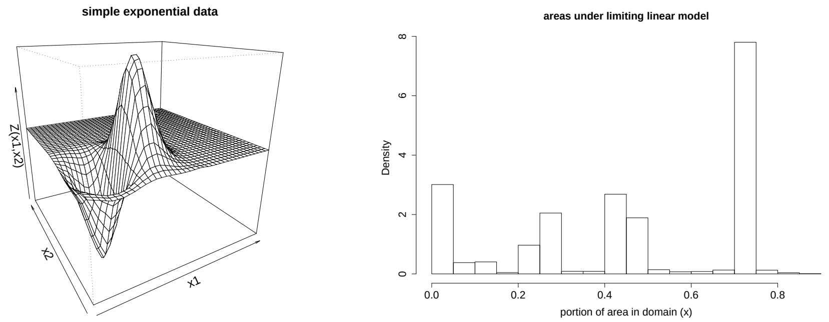

Figure 2: *Left:* exponential data GP LLM fit. *Right:* histogram of the areas under the LLM.

of estimation and prediction with the treed GP LLM for a $n = 200$ sub-sample of this data from a regular grid of size 441. The partitioning structure of the treed GP LLM first splits the region into two halves, one of which can be fit linearly. It then recursively partitions the half with the action into a piece which requires a GP and another piece which is also linear. The *left* pane shows a mean predictive surface wherein the LLM was used over 66% of the domain (on average) which was obtained in less than ten seconds on a 1.8 GHz Athalon. The *right* pane shows a histogram of the areas of the domain under the LLM over 20-fold repeated experiments. The four modes of the histogram clump around 0%, 25%, 50%, and 75% showing that most often the obvious three-quarters of the space are under the LLM, although sometimes one of the two partitions will use a very smooth GP. The treed GP LLM was 40% faster than the treed GP alone when combining estimation and sampling from the posterior predictive distributions at the remaining $n' = 241$ points from the grid.

4.2 Motorcycle Data

The Motorcycle Accident Dataset [10] is a classic for illustrating nonstationary models. It samples the acceleration force on the head of a motorcycle rider as a function of time in the first moments after an impact. Figure 3 shows the data, and a fit using the treed GP LLM. The *top* pane shows the mean predictive surface, with 90% quantile error-bars. From the *bottom* pane, which shows the difference in 95% and 5% quantiles, it is clear that the tree structure typically partitions the space into three parts. On average, 29% of the domain was under the LLM, split between the left low-noise region (before impact) and the noisier right region.

Rasmussen & Ghahramani [9] analyzed this data by using a Dirichlet process mixture of Gaussian process (DPGP) experts which reportedly took one hour on a 1 GHz Pentium. Such times are typical of nonstationary modeling because of the computational effort required to construct and invert large covariance matrices. In contrast, the treed GP LLM fits this dataset with comparable accuracy but in less than one minute on a 1.8 GHz Athalon.

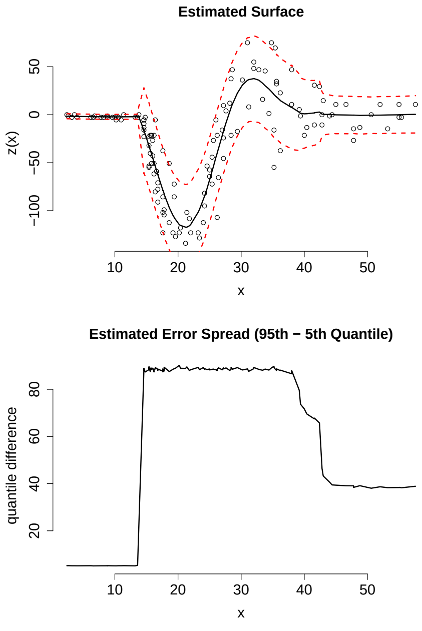

Figure 3: *Top:* Motorcycle Data fit by treed GP LLM. *Bottom:* and quantile differences.

We identify three things which make the treed GP LLM so fast relative to most nonstationary spatial models. (1) Partitioning fits models to less data, yielding smaller matrices to invert. (2) Jumps to the LLM mean fewer inversions all together. (3) MCMC mixes better because under the LLM the parameters $\mathbf{d}$ and g are out of the picture and all sampling can be performed via Gibbs steps.

4.3 Friedman data

This Friedman data set is the first one of a suite that was used to illustrate MARS (Multivariate Adaptive Regression Splines) [6]. There are 10 covariates in the data ($\mathbf{x} = \{x_1, x_2, \dots, x_{10}\}$), but the function that describes the responses (Y), observed with standard Normal noise,

$$E(Y|\mathbf{x}) = \mu = 10\sin(\pi x_1 x_2) + 20(x_3 - 0.5)^2 + 10x_4 + 5x_5 \tag{8}$$

depends only on $\{x_1, \dots, x_5\}$, thus combining nonlinear, linear, and irrelevant effects. We make comparisons on this data to results provided for several other models in recent literature. Chipman et al. [1] used this data to compare their linear CART algorithm to four other methods of varying parameterization: linear regression, greedy tree, MARS, and neural networks. The statistic they use for comparison is root mean-square error (RMSE)

$$\text{MSE} = \textstyle\sum_{i=1}^{n} (\mu_i - \hat{Y}_i)^2/n \qquad\qquad \text{RMSE} = \sqrt{\text{MSE}}$$

where $\hat{Y}_i$ is the model-predicted response for input $\mathbf{x}_i$. The $\mathbf{x}$'s are randomly distributed on the unit interval. RMSE's are gathered for fifty repeated simulations of size $n = 100$ from (8). Chipman et al. provide a nice collection of boxplots showing the results. However, they do not provide any numerical results, so we have extracted some key numbers from their plots and refer the reader to that paper for their full results.

We duplicated the experiment using our GP LLM. For this dataset, we use a single model, not a treed model, as the function is essentially stationary in the spatial statistical sense (so if we were to try to fit a treed GP, it would keep all of the data in a single partition). Linearizing boolean prior parameters $(\gamma, \theta_1, \theta_2) = (10, 0.2, 0.9)$ were used, which gave the LLM a relatively low prior probability of 0.35, for large range parameters d_i. The RMSEs that we obtained for the GP LLM are summarized in the table below.

	Min	1st Qu.	Median	Mean	3rd Qu.	Max
GP LLM	0.4341	0.5743	0.6233	0.6258	0.6707	0.7891
Linear	1.710	2.165	2.291	2.325	2.500	2.794

Results on the linear model are reported for calibration purposes, and can be seen to be essentially the same as those reported by Chipman et al. RMSEs for the GP LLM are on average significantly better than *all* of those reported for the above methods, with lower variance. For example, the best mean RMSE shown in the boxplot is ≈ 0.9. That is 1.4 times higher than the worst one we obtained for GP LLM. Further comparison to the boxplots provided by Chipman et al. shows that the GP LLM is the clear winner.

In fitting the model, the Markov Chain quickly keyed in on the fact that only the first three covariates contribute nonlinearly. After burn-in, the booleans $\mathbf{b}$ almost never deviated from $(1, 1, 1, 0, 0, 0, 0, 0, 0, 0)$. From the following table summarizing the posterior for the linear regression coefficients ($\boldsymbol{\beta}$) we can see that the coefficients for x_4 and x_5 (between double-bars) were estimated accurately, and that the model correctly determined that $\{x_6, \dots x_{10}\}$ were irrelevant (i.e. not included in the GP, and had β's close to zero).

		x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
$\boldsymbol{\beta}$	5% Qu.	8.40	2.60	-1.23	-0.89	-1.82	-0.60	- 0.91
	Mean	9.75	4.59	-0.190	0.049	-0.612	0.326	0.066
	95% Qu.	10.99	9.98	0.92	1.00	0.68	1.21	1.02

For a final comparison we consider an SVM method [5] illustrated on this data and compared to Bagging. We note that the SVM method required cross-validation (CV) to set some of its parameters. In the comparison, 100 randomized training sets of size $n = 200$ were used, and RMSEs were collected for a (single) test set of size $n' = 1000$. An average MSE of 0.67 is reported, showing the SVM to be uniformly better the Bagging method with an MSE of 2.26. We repeated the experiment for the GP LLM (which requires no CV!), and obtained an average MSE of 0.293, which is 2.28 times better than the SVM, and 7.71 times better than Bagging.

4.4 Boston housing data

A commonly used data set for validating multivariate models is the Boston Housing Data [8], which contains 506 responses over 13 covariates. Chipman et. al [1] showed that their (Bayesian) linear CART model gave lower RMSEs, on average, compared to a number of popular techniques (the same ones listed in the previous section). Here we employed a treed GP LLM, which is a generalization of their linear CART model, retaining the original linear CART as an accessible special case. Though computationally more intensive than linear CART, the treed GP LLM gives impressive results. To mitigate some of the computational demands, the LLM can be used to initialize the Markov Chain by breaking the larger data set into smaller partitions. Before treed GP burn-in begins, the model is fit using only the faster (limiting) linear CART model. Once the treed partitioning has stabilized, this fit is taken as the starting value for a full MCMC exploration of the posterior for the treed GP LLM. This initialization process allows us to fit GPs on smaller segments of the data, reducing the size of matrices that need to be inverted and greatly reducing computation time. For the Boston Housing data we use $(\gamma, \theta_1, \theta_2) = (10, 0.2, 0.95)$, which gives the LLM a prior probability of $0.95^{13} \approx 0.51$, when the d_i's are large.

Experiments in the Bayesian linear CART paper [1] consist of calculating RMSEs via 10-fold CV. The data are randomly partitioned into 10 groups, iteratively trained on 9/10 of the data, and tested on the remaining 1/10. This is repeated for 20 random partitions, and boxplots are shown. Note that the logarithm of the response is used and that CV is only used to assess predictive error, not to tune parameters. Samples are gathered from the posterior predictive distribution of the linear CART model for six parameterizations using 20 restarts of 4000 iterations. This seems excessive, but we followed suit for the treed GP LLM in order to obtain a fair comparison. Our “boxplot” for training and testing RMSEs are summarized in the table below. As before, linear regression (on the log responses) is used for calibration.

		Min	1st Qu.	Median	Mean	3rd Qu.	Max
train	GP LLM	0.0701	0.0716	0.0724	0.0728	0.0730	0.0818
	Linear	0.1868	0.1869	0.1869	0.1869	0.1869	0.1870
test	GP LLM	0.1321	0.1327	0.1346	0.1346	0.1356	0.1389
	Linear	0.1926	0.1945	0.1950	0.1950	0.1953	0.1982

Notice that the RMSEs for the linear model have extremely low variability. This is similar to the results provided by Chipman et al. and was a key factor in determining that our experiment was well-calibrated. Upon comparison of the above numbers with the boxplots in Chipman et al., it can readily be seen that the treed GP LLM is leaps and bounds better than linear CART, and *all* of the other methods in the study. Our worst training RMSE is almost two times lower than the best ones from the boxplot. All of our testing RMSEs are lower than the lowest ones from the boxplot, and our median RMSE (0.1346) is 1.26 times lower than the lowest median RMSE (≈ 0.17) from the boxplot.

More recently, Chu et al. [2] performed a similar experiment (see Table V), but instead of 10-fold CV, they randomly partitioned the data 100 times into training/test sets of size 481/25 and reported average MSEs on the un-transformed responses. They compare their

Bayesian SVM regression algorithm (BSVR) to other high-powered techniques like Ridge Regression, Relevance Vector Machine, GPs, etc., with and without ARD (automatic relevance determination). Repeating their experiment for the treed GP LLM gave an average MSE of 6.96 compared to that of 6.99 for the BSVR with ARD, making the two algorithms by far the best in the comparison. However, without ARD the MSE of BSVR was 12.34, 1.77 times higher than the treed GP LLM, and the worst in the comparison. The reported results for a GP with (8.32) and without (9.13) ARD showed the same effect, but to a lesser degree. Thus our GP LLM might similarly benefit from an ARD-like approach. Perhaps not surprisingly, the average MSEs do not tell the whole story. The 1st, median, and 3rd quantile MSEs we obtained for the treed GP LLM were 3.72, 5.32 and 8.48 respectively, showing that its distribution had a heavy right-hand tail. We take this as an indication that several responses in the data are either misleading, noisy, or otherwise very hard to predict.

5 Conclusions

Gaussian processes are a flexible modeling tool which can be overkill for many applications. We have shown how its limiting linear model can be both useful and accessible in terms of Bayesian posterior estimation, and prediction. The benefits include speed, parsimony, and a relatively straightforward implementation of a semiparametric model. Combined with treed partitioning, the GP LLM extends linear CART, resulting in a uniquely nonstationary, tractable, and highly accurate regression tool.

We believe that a large contribution of the GP LLM will be in the domain of sequential design of computer experiments [7] which was the inspiration for much of the work presented here. Empirical evidence suggests that many computer experiments are nearly linear. That is, either the response is linear in most of its input dimensions, or the process is entirely linear in a subset of the input domain. Supremely relevant, but largely ignored in this paper, is that the Bayesian treed GP LLM provides a *full* posterior predictive distribution (particularly a nonstationary and thus region-specific estimate of predictive variance) which can be used towards active learning in the input domain. Exploitation of these characteristics should lead to a efficient framework for the adaptive exploration of computer experiment parameter spaces.

References

[1] Hugh A. Chipman, Edward I. George, and Robert E. McCulloch. Bayesian treed models. *Machine Learning*, 48:303–324, 2002.

[2] W. Chu, S. S. Keerthi, and C. J. Ong. Bayesian support vector regression using a unified loss function. *IEEE Transactions on Neural Networks*, 15(1):29–44, 2004.

[3] Noel A. Cressie. *Statistics for Spatial Data*. John Wiley and Sons, Inc., 1991.

[4] Dipak Dey, Peter Müller, and Debajyoti Sinha. *Practical nonparametric and semiparametric Bayesian statistics*. Springer-Verlag New York, Inc., New York, NY, USA, 1998.

[5] Harris Drucker, Christopher J. C. Burges, Linda Kaufman, Alex J. Smola, and Vladimir Vapnik. Support vector regression machines. In *NIPS*, pages 155–161. MIT Press, 1996.

[6] J. H. Freidman. Multivariate adaptive regression splines. *Annals of Statistics*, 19, No. 1:1–67, March 1991.

[7] R. B. Gramacy, Herbert K. H. Lee, and William Macready. Parameter space exploration with Gaussian process trees. In *ICML*, pages 353–360. Omnipress & ACM Digital Library, 2004.

[8] D. Harrison and D. L. Rubinfeld. Hedonic housing prices and the demand for clean air. *Journal of Environmental Economics and Management*, 5:81–102, 1978.

[9] Carl Edward Rasmussen and Zoubin Ghahramani. Infinite mixtures of Gaussian process experts. In *NIPS*, volume 14, pages 881–888. MIT Press, 2002.

[10] B. W. Silverman. Some aspects of the spline smoothing approach to non-parametric curve fitting. *Journal of the Royal Statistical Society Series B*, 47:1–52, 1985.