Biometrics 000, 000–000 DOI: 000
000 0000

Supplemental Material for: Bayesian Additive Adaptive Basis Tensor Product Models for Modeling High Dimensional Surfaces: An application to high-throughput toxicity testing.

Matthew W. Wheeler

Risk Analysis Branch, National Institute for Occupational Safety and Health, Cincinnati, OH

email: mwheeler@cdc.gov

This paper has been submitted for consideration for publication in *Biometrics*

1. Full Sampling Algorithm

Define the vector of observations $Y = \{y_1(s_1, d_{11}), \ldots, y_1(s_1, d_{1C_1}), \ldots, y_n(s_n, d_{n1}), \ldots, y_n(s_n, d_{nC_n})\}'$ to be the $(N \times 1)$ vector of measurements across the n observed curves, where $N = \sum_{i=1}^{n} C_i$.

Likewise define

$$\mathbf{G} = \begin{bmatrix} 1 & g_1(s_1) & g_2(s_1) & \cdots & g_K(s_1) \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & g_1(s_1) & g_2(s_1) & \cdots & g_K(s_1) \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & g_1(s_n) & g_2(s_n) & \cdots & g_K(s_n) \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & g_1(s_n) & g_2(s_n) & \cdots & g_K(s_n) \end{bmatrix},$$

to be an $(N \times K + 1)$ matrix, where each row corresponds to the loadings for observation $y_i(s_i, d_{ic})$. Let

$$\mathbf{F} = \begin{bmatrix} f_0(d_{11}) & f_1(d_{11}) & \cdots & f_K(d_{11}) \\ \vdots & \vdots & & \vdots \\ f_0(d_{1C_1}) & f_1(d_{1C_1}) & \cdots & f_K(d_{1C_1}) \\ \vdots & \vdots & & \vdots \\ f_0(d_{n1}) & f_1(d_{n1}) & \cdots & f_K(d_{n1}) \\ \vdots & \vdots & & \vdots \\ f_0(d_{nC_n}) & f_1(d_{nC_n}) & \cdots & f_K(d_{nC_n}) \end{bmatrix},$$

be an $(N \times K + 1)$ matrix, where each row represents the basis functions evaluated at d_{ic}. Using these definitions, the model is

$$Y = \Big(\mathbf{G} \circ \mathbf{F}\Big) J' + \epsilon, \tag{1}$$

where $\circ$ is the Schur product, $J = (1, 1, \ldots, 1)$ is a $K + 1$ row vector, and $\epsilon = (\epsilon_{11}, \ldots, \epsilon_{iC_1}, \ldots, \epsilon_{n1}, \ldots, \epsilon_{nC_n})'$ is the $(N \times 1)$ vector of error terms.

Let $D = \{d_r^*\}_{r=1}^{R}$ be the set of R uniquely observed inputs across all observations in $\mathcal{D}$,

and define $\mathcal{I}^f$ to be an $(N \times R)$ matrix where the rows corresponds to each element in Y. For each row in $\mathcal{I}^f$, all entries are set to zero except at column r. This entry is set to one, and it corresponds to the observation $y_i(s_i, d_{ic})$ such that $d_r^* = d_{ic}$. Likewise, define the matrix $\mathcal{I}^g$ to be an $(N \times n)$ matrix. For each observation i, each entry of each column is set to zero except column i, which is set to one.

Sample from the posterior in a series of Gibbs and Metropolis within Gibbs steps as follows:

(1) For each k, $0 \leqslant k \leqslant K$, letting $Y^* = Y - (\mathbf{G}_{-k} \circ \mathbf{F}_{-k})J'_{-k}$, where $\mathbf{G}_{-k}$, $\mathbf{F}_{-k}$, and J_{-k} are $\mathbf{G}$, $\mathbf{F}$ and J without column k, sample $f_k \sim N(M, V)$ at $\{d_r^*\}_{r=1}^R$. Here

$$V = \Sigma_k(\tau \mathcal{G}'\mathcal{G}\Sigma_k + I)^{-1},$$

$$M = V(\tau \mathcal{G}'Y^*),$$

were Σ_k is the $(R \times R)$ covariance function constructed from $\sigma_k^f(\cdot, \cdot)$ and I is an $R \times R$ identity matrix. Further, $\mathcal{G}$ is an $(N \times R)$ matrix defined as $G_k \odot \mathcal{I}^f$, where $\odot$ takes the Schur product of the column vector G_k to each column in $\mathcal{I}^f$ resulting in a $(N \times R)$ matrix.

(2) For each k, $1 \leqslant k \leqslant K$ and letting $Y^* = Y - (\mathbf{G}_{-k} \circ \mathbf{F}_{-k})J'_{-k}$, sample $g_k \sim N(M, V)$ at $\{s_i\}_{i=1}^n$. Here

$$V = \Sigma_k(\tau \mathcal{F}'\mathcal{F}\Sigma_k + I)^{-1},$$

$$M = V(\tau \mathcal{F}'Y^*),$$

where Σ_k is the $(n \times n)$ covariance matrix formed from $\sigma_k^g(\cdot, \cdot)$, $\mathcal{F}$ is an $(N \times n)$ matrix defined to be $(FJ') \odot \mathcal{I}^g$, and I is the $n \times n$ identity matrix.

(3) Sample $\phi \sim \text{Ga}(c, d)$, where

$$c = K\frac{n}{2} + a_1,$$

$$d = \left[\sum_{m=1}^{K}\left(\prod_{l=1}^{m}\delta_j\right)G_m'\Sigma_m G_m\right] + 1,$$

G_m is a column vector from column m of G, and Σ_i is the matrix formed from $\exp(-\theta_m||s-s'||^2)$

(4) For each k sample $\delta_k \sim \text{Ga}(c,d)$, where

$$c = (K-k+1)\frac{n}{2} + a_1,$$
$$d = \left[\sum_{m=k}^{K}, \phi\Big(\prod_{l=1,l\neq k}^{i}\delta_l\Big)G_m'\Sigma_m G_m\right] + 1$$

G_m is a column vector from column m of G, Σ_i is the matrix formed from $\exp(-\theta_i||s-s'||^2)$, and $\left(\prod_{l=1,l\neq 1}^{1}\delta_j\right) = 1$.

(5) For each k, $0 \leqslant k \leqslant K$, sample ω_k using a Metropolis-Hastings step. Sample $\delta_k \sim N(0,a)$; let $\omega_k^* = \exp[\log(\omega_k) + \delta_k]$, and construct the covariance matrices Σ_k^* and Σ_k from ω_k^* and ω_k respectively. Letting

$$\ell^* = -\frac{1}{2}\left\{f_k'\Sigma_k^* f_k + \log(|\Sigma_k^*|)\right\} + r(\omega_k^*) + q(\omega_k^* \to \omega_k),$$

and

$$\ell = -\frac{1}{2}\left\{f_k'\Sigma_k f_k + \log(|\Sigma_k|)\right\} + r(\omega_k) + q(\omega_k \to \omega_k^*), \tag{2}$$

where $r(\cdot)$ and $q(\cdot)$ are the log probability density functions for the prior and proposal distributions respectively. Accept the new proposal ω_k^* with probability $\min[1, \exp(\ell^* - \ell)]$. Note in the case of the sampling algorithm used in the manuscript, $q(s \to t)$ is the log probability density function of a log-Gaussian distribution with $\mu = \log(\text{s})$ and $\sigma^2 = 0.15$.

(6) For each $k, 1 \leqslant k \leqslant K$, sample θ_k using a Metropolis step. Sample $\theta_k \sim DiscreteUniform(A)$ a discrete uniform distribution over the set A, and construct the covariance matrices Σ_k^* and Σ_k from θ_k^* and θ_k respectively. Letting

$$\ell^* = -\frac{1}{2}\left\{g_k'\Sigma_k^* g_k + \log(|\Sigma_k^*|)\right\},$$

and

$$\ell = -\frac{1}{2}\left\{g_k'\Sigma_k g_k + \log(|\Sigma_k|)\right\}, \tag{3}$$

Accept the new proposal θ_k^* with probability $\min[1, \exp(\ell^* - \ell)]$.

(7) For τ, let $\Lambda = \mathbf{G} \circ \mathbf{F}$ and set

$$A = \frac{n}{2} + \frac{\nu}{2}, \tag{4}$$

$$B = \frac{(Y - \Lambda)'(Y - \Lambda)}{2} + \frac{\nu}{2}, \tag{5}$$

sampling τ from $Ga(A, B)$, where $\nu = 2$ in the manuscript.

The algorithm is written in the `R` programming language (R Core Team, 2015) using the `Rcpp` `C++` extensions (Eddelbuettel, 2013) and the `RcppArmadillo` (Eddelbuettel and Sanderson, 2014) linear algebra extensions. All code is available as a supplement.

To start the MCMC sampler, the model parameters are initialized as follows: The $\{f_k\}_{k=0}^K$ are initialized to zero across $\mathcal{D}$; ϕ as well as $\{\delta_k\}_{k=1}^K$ are sampled from their priors. The covariance kernels width parameters initialized at a fixed value, and the $\{g_k\}_{k=1}^K$ are drawn from a $N(0, 1)$ distribution. All of the $\{f_k\}_{k=0}^K$ are initialized to zero as initializing them from the prior may give models with very low posterior probability causing the algorithm to fall into numerical issues before the MCMC sampler can reach the regions of high posterior probability. To test the speed of convergence of the algorithm, the model was started at different locations than those above. In all cases, the model converged to the same posterior within 500 iterations. Additionally, for estimates of $h(s, d)$ the effective sample size varied across d from 20 in 100 samples to 95 in 100 samples, with the median effective sample size being near 60 per 100 samples.

When the number of unique observations is small, it is possible to develop a block Gibbs sampler for all of the $\{f_k\}_{k=0}^K$ and $\{g_k\}_{k=1}^K$. In large problems, this requires the inversion of a large matrix offsetting the benefits of the increased computational efficiency.

1.1 *Predictive Inference*

The posterior predictive distribution of n^* unobserved cross sections of $h(s,d)$ at $\grave{S} = \{\grave{s}_1, \ldots, \grave{s}_{n^*}\}$ is estimated through MCMC. Let the vector $g_k = (g_k(s_1), \ldots, g_k(s_n))'$ be observed at $S = (s_1, \ldots, s_n),'$ with $\grave{g}_k = (g_k(\grave{s}_1), \ldots, g_k(\grave{s}_{n^*}))$being of interest; g_k and $\grave{g}_k$ are jointly distributed as

$$\begin{bmatrix} g_k \\ \grave{g}_k \end{bmatrix} \sim N\left(0, \begin{bmatrix} \Sigma_k^g(S,S) & \Sigma_k^g(S,\grave{S}) \\ \Sigma_k^g(\grave{S},S) & \Sigma_k^g(\grave{S},\grave{S}) \end{bmatrix}\right).$$

Here $\Sigma_k^g(S,S)$, $\Sigma_k^g(\grave{S},\grave{S})$, $\Sigma_k^g(\grave{S},S)$ and $\Sigma_k^g(S,\grave{S})$ represent covariance matrices given S, $\grave{S}$ and $\sigma_k^g(\cdot,\cdot)$. Using properties of the multivariate normal distribution, conditionally on g_k

$$\begin{aligned}\grave{g}_k \mid g_k \sim & N\Big[\Sigma_k^g(\grave{S},S)\Sigma_k^g(S,S)^{-1}g_k, \Sigma_k^g(\grave{S},\grave{S}) \\ & - \Sigma_k^g(\grave{S},S)\Sigma_k^g(S,S)^{-1}\Sigma_k^g(S,\grave{S})\Big].\end{aligned}$$

For each iteration, this expression is used to draw $\{\grave{g}_1, \ldots, \grave{g}_K\}$. Given this draw, as well as $\{f_0, \ldots, f_K\}$ which represents $f_0(d), f_1(d)$ etc., evaluated at $D = \{d_r^*\}_{r=1}^R$, one can estimate the posterior predicted distribution of $h(\grave{S}, D)$. If new $\grave{D}$ are of interest, the same technique can be applied to estimating $\{\grave{f}_0, \ldots, \grave{f}_K\}$. These values can be used with $\{\grave{g}_k\}_{k=1}^K$ to provide estimates for $h(\grave{S}, \grave{D})$.

2. Additional Data Example Tests

The method was also compared to standard QSAR approaches, which model a single data point. Similar to the standard QSAR methodology, a model was fit to each data-set i and the response associated with a given dose was computed to be observation y_i. This value was then used in the model

$$y_i = x(s_i) + \epsilon_i,$$

where $x(s_i)$ is a Gaussian process with squared exponential covariance kernel. The model was fit on the same 669 observations in the training set and predictions were made of the response at that dose. Both this QSAR approach as well as the adaptive tensor product approach were compared to the hold out samples using the correlation coefficient, a standard practice in the QSAR literature. For the dose of $20\mu M$ the standard QSAR approach had a correlation of 0.42 as compared to the co-mixture approaches' correlation of 0.48. For the dose of $100\mu M$ a similar 0.06 increase was seen, and, when observations that have a chemical within the training set defined as 'close' (i.e., a relative distance between two chemicals less than 2.2), this improvement in the correlation coefficient is almost 0.1 (i.e, 0.5 compared to 0.6). This indicates that one sees 10% to 20% improvements in the correlation coefficient using the proposed approach.

References

Eddelbuettel, D. (2013). *Seamless R and C++ integration with Rcpp*. Springer.

Eddelbuettel, D. and Sanderson, C. (2014). RcppArmadillo: Accelerating R with high-performance C++ linear algebra. *Computational Statistics and Data Analysis* **71,** 1054–1063.

R Core Team (2015). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.