Bayesian regression methods

Motivation

We are interested in using Bayesian regression techniques to characterize the nature of complex interactions. In addition to being an interesting statistical challenge, complex interactions are also relevant from both a mechanistic and public health point of view.

- mechanistic
- public health

Drawing from the mechanistic and public health contexts, there are three classes of interactions that are of interest.

- different types of interactions

However, non-additive interactions are difficult to detect.

First, there are different forms that an interaction can take on. For instance, in the linear regression setting, the form of the interaction must be explicitly specified; most commonly, this is done by multiplying two variables together. However,

Moreover, the number of possible interactions to consider quickly becomes intractable in high-dimensional settings. For instance, consider modelling 10 exposures in a linear regression setting. In order to be assessed, each interaction must be explicitly specified as a new term in the model. Including all the possible two-way interactions would involve adding $\binom{10}{2} = 45$ additional terms to the model, and all possible three-way interactions would add $\binom{10}{3} = 120$ additional terms.

- interactions are difficult to pick up
- consider the number of interactions that would have to be encoded explicitly in MLR

It is important also to acknowledge, here, that there is a limit to how many variables can be included in an interaction before it becomes incomprehensible to most humans.

Bayesian kernel machine regression

Defining notation:

- observations from $i = 1, \dots, n$
- $\mathbf{x}_m$ is a predictor variable in the predictor matrix $\mathbf{X}$ with $m = 1, \dots, M$, measuring exposure variables in this case
- $\mathbf{x}_i$ is a vector of values for a single observation in $\mathbf{X}$ with $i = 1, \dots, n$, measuring the health outcome in this case
- x_{im} is an observation of $\mathbf{x}_m$
- Y_i is an observation of $\mathbf{Y}$
- ρ is the tuning parameter inside the exponential expression of the covariance, equals $2l$
- τ is the tuning parameter outside the expontential expression of the covariance, equals σ_f^2
- k is the smoothing function, the Gaussian in this case
- $\mathbf{K}$ is the $n \times n$kernel matrix, with (i, j)th element $k(\mathbf{x}_i, \mathbf{x}_j)$
- $h(\mathbf{x}_i)$ the flexible function relating X to Y
- $\boldsymbol{\epsilon}_i \overset{\text{iid}}{\sim} N(0, \sigma^2)$, the residuals of the response

Kernel machine regression

In this section, we introduce kernel machine regression, with attention to its specific implementation in BKMR. Kernel machine regression is a nonparametric regression technique that can be used to capture nonlinear effects and nonadditive interactions. In the typical linear regression setting,

$$Y_i = \mathbf{x}_i^\top \boldsymbol{\beta} + \epsilon_i$$

where Y_i measures a health outcome at a given point, $\mathbf{x}_i = (x_1, \ldots, x_M)^\top$ is a vector of M exposures, $\boldsymbol{\beta}$ is a vector of weights, and ϵ_i is a random variable from $\boldsymbol{\epsilon} \overset{\text{iid}}{\sim} N(0, \sigma^2)$. We can see that this function assumes that there is a linear relationship between the exposure and the response, and that the combined effects of multiple exposures are additive.

Kernel machine regression defines this relationship using a function $h : \mathbb{R}^M \rightarrow \mathbb{R}$, where

$$Y_i = h(\mathbf{x}_i) + \epsilon_i$$

Here, $h(\cdot)$ is represented by the function $k(\cdot, \cdot)$, a kernel. The kernel controls the covariance, or the similarity, between values of $h(\mathbf{x})$ and as such ensures that points near each other on the prediction surface will have similar values — or, in other words, that the prediction surface will be smooth. In the case of kernel machine regression, we define a positive definite kernel where $k : \mathbb{R}^M \times \mathbb{R}^M \rightarrow \mathbb{R}$. There are many choices of functions to represent k. BKMR uses the Gaussian kernel, also known as the radial basis function or, sometimes, the squared exponential kernel. The Gaussian kernel is defined

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\{-\frac{||\mathbf{x} - \mathbf{x}'||^2}{2l}\}$$

where $||\mathbf{x} - \mathbf{x}'||^2 = \sum_{m=1}^{M} (x_m - x'_m)^2$ for $\mathbf{x}$, a set of exposure values, and $\mathbf{x}$, the exposure profile of a second subject, σ_f^2 is a tuning parameter that controls how much the function is allowed to vary, and l is a tuning parameter that controls the relationship between the correlation between two points and their distance. Greater values of l will enforce more dependence between points and make the resulting function smoother. Note that BKMR uses $\rho = 2l$ and $\tau = \sigma_f^2$, so, henceforth, we will be referring to these parameters using BKMR's notation.

Now that we have defined h and k, we can think about how to characterize the relationship between our response and predictors. Kernel machine regression is a nonparametric technique because it does not specify a functional form for this relationship. Hence, we will think about estimating the response at a particular query point. Operationally, kernel machine regression uses a weighted average of all the observations in the dataset to estimate the response, as follows

$$\bar{Y} = \frac{\sum_{i=1}^{n} w_i Y_i}{\sum_{i=1}^{n} w_i}$$

with some set of weights $\{w_i\}_{i=1}^n$. Intuitively, we want to weight the observations that are closer to the query point more heavily. Using the Gaussian kernel as a weight allows us to achieve this. Replacing the weight with the Gaussian kernel, we get

$$\bar{Y} = \frac{\sum_{i=1}^{n} K(\mathbf{x}, \mathbf{x}_i) Y_i}{\sum_{i=1}^{n} K(\mathbf{x}, \mathbf{x}_i)}$$

As we move through the predictor space, we can think of the prediction as a continuous moving average of local points in the dataset. This gives us the following relationship

$$\text{cor}(h_i, h_j) = \exp\{-(\frac{1}{\rho})\sum_{m=1}^{M}(x_{im} - x_{jm})^2\}$$

which allows us to see that values h near each other will have a higher correlation and thus similar values. This is also why the resulting function is smooth.

Connection with mixed models

It is useful to make connections between this definition of kernel machine regression and mixed models. To do this, we can represent $h(\mathbf{x})$ as following a Gaussian process probability distribution, defined

$$h(\mathbf{x}) \sim \mathcal{GP}(\mathbf{0}, k(\mathbf{x}, \mathbf{x}'))$$

with mean function m and covariance function k, where $\mathbf{x}$ is a vector of exposure values, and $\mathbf{x}'$ is the exposure profile of another subject. A Gaussian process is a collection of random variables, of which any finite number follow a multivariate normal distribution. Here, we assume that the expected value of the function with input $\mathbf{x}$ is $\mathbf{0}$. We use k for the covariance function, which represents the dependence between the function values with inputs $\mathbf{x}$ and $\mathbf{x}'$: $k(\mathbf{x}, \mathbf{x}') = \mathbb{E}[(h(\mathbf{x}) - m(\mathbf{x}))(h(\mathbf{x}') - m(\mathbf{x}'))]$.

Now, we can represent h as a collection of variables from a Gaussian process. h follows a multivariate normal distribution

$$h(\mathbf{x}) \sim N(\mathbf{0}, \mathbf{K})$$

where $h(\mathbf{x}) = h(\mathbf{x}_1), h(\mathbf{x}_2), \dots, h(\mathbf{x}_n)$, and $\mathbf{K}$ is the kernel matrix. The kernel matrix is an $n \times n$ matrix with (i, j)th element $k(\mathbf{x}_i, \mathbf{x}_j)$. Now, returning back to the regression view, we can think of each Y_i as following the distribution

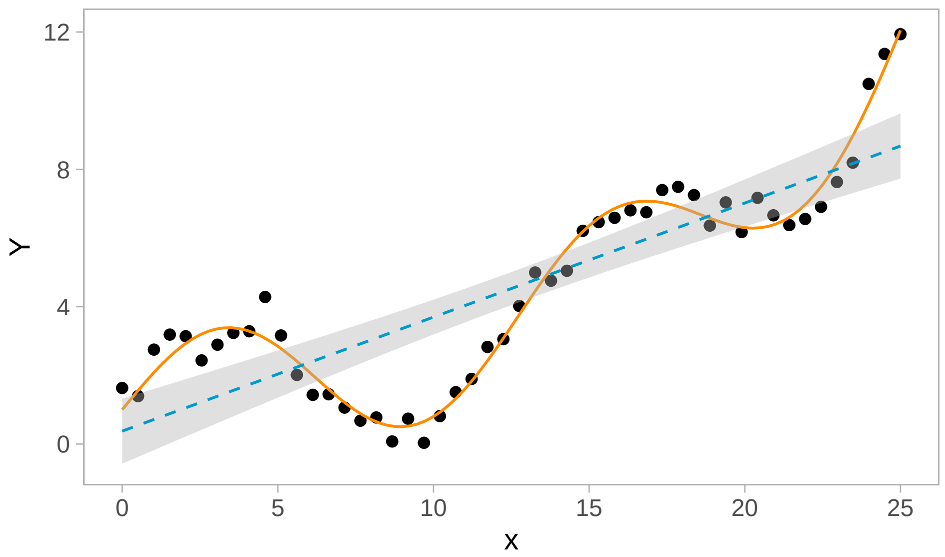
$$Y_i \overset{\text{ind}}{\sim} N(h(\mathbf{x}_i), \sigma^2) \text{ for } i = 1, \dots, n$$

where σ^2 comes from the variance of the residuals.

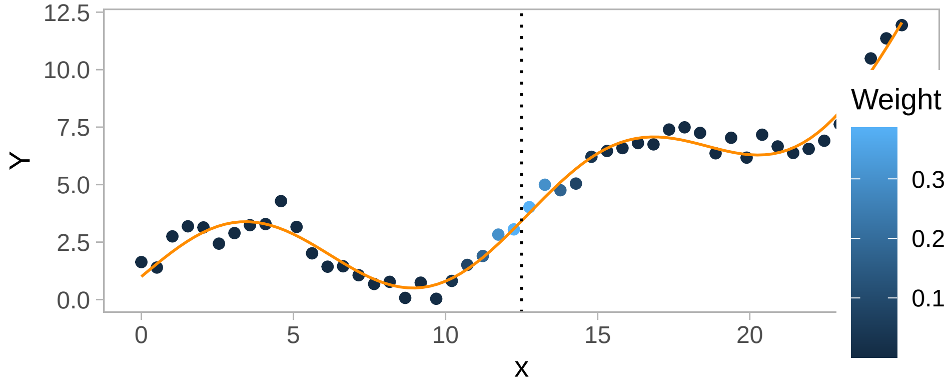
Toy example

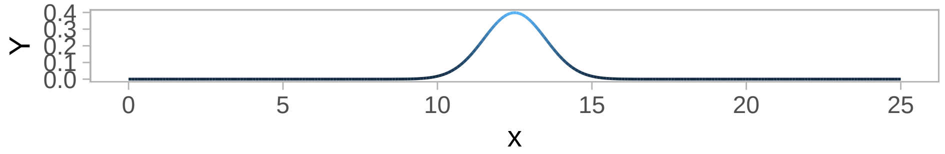
In the following section, we introduce the technique with a toy example.

Consider the following case where we want to model the relationship between a single predictor and a response variable. Suppose the true relationship between x and Y is defined $Y = e^{\frac{x}{10}} + 2\sin(\frac{x}{2})$

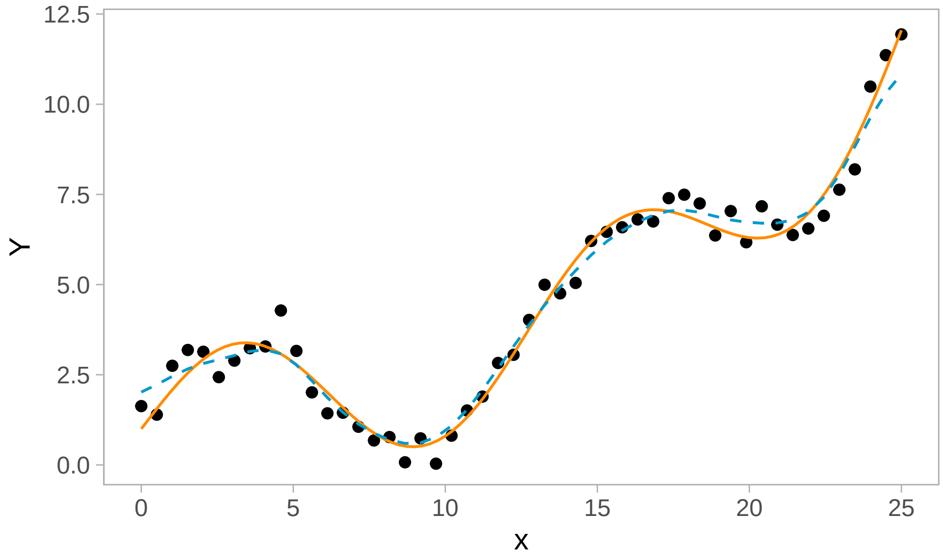


Define a query point





Note to self, bandwidth represent four times the quartiles of the probability density. In the case of a Gaussian kernel, the bandwidth represents $\frac{8}{3}\sigma$, so using an sd of 1 translates to bandwidth of 8/3



We can also use this example to understand how our choice of the scaling parameter influences model fit.

Variable selection

In this section, we discuss the two methods for variable selection in BKMR: hierarchical variable selectiona and component-wise variable selection.

Priors in a Bayesian framework

Discuss priors, what form do they take? what are the default settings?

The MCMC algorithm

Brief intro to MCMC

Detecting interactions

Discuss options for detecting interactions, i.e., can visualize relationship at various quantiles of other predictor, can conduct inference on inter-quantile difference

Potentially provide toy example

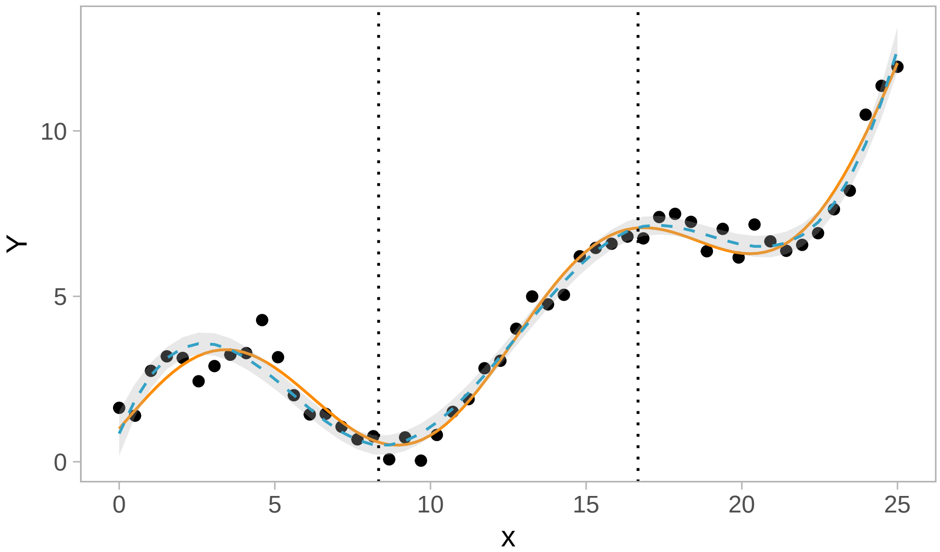
Bayesian semiparametric regression

differences w/ bkmr

- makes distributional assumptions about the dataset
- kernel regression is computationally intensive w/ large datasets but can handle many predictors
- bsr highly dependent on the choice of function

Connections to linear mixed model

Toy example



–>