The Method of Analogues

Evan L. Ray*, Krzysztof Sakrejda, Stephen A. Lauer, Alexandria C. Brown, Xi Meng, and Nicholas G. Reich

Department of Biostatistics and Epidemiology,
University of Massachusetts – Amherst, Amherst, MA

1 Introduction

In this document, we describe the method we use for predicting Dengue fever incidence in the 2015 Dengue Forecasting Project. Briefly, we employ a non-parametric approach to prediction in dynamical systems referred to as state space reconstruction or the method of analogues (MoA). Our implementation of MoA is similar in spirit to the simplex projection method of Sugihara and May [1990]; however, we frame the approach in terms of kernel regression and kernel density estimation [Silverman, 1986, Racine and Li, 2004].

The remainder of this document is organized as follows. We describe the general motivation for MoA in Section 2, give some details for our implementation based on kernel methods in Section 3, and discuss our strategy for estimating the model parameters in Section 4.

2 Overview of the Method of Analogues

The spread of an infectious disease within a population is often represented as a dynamical process. For example, the SIR model and its variants partition the population into disjoint groups of people according to their infection status, and describe changes in the size of each group over time with a system of differential equations. These models make several assumptions regarding the number of states (i.e., disease categories) and the functional forms describing the rates at which people move through these categories over time.

A central challenge in predicting infectious disease outbreaks is that we typically observe many fewer covariates than are important to the evolution of the process, and the functional forms describing how these covariates enter into the model are not known precisely. For example, even the most basic formulation of the SIR model involves the number of Susceptible and Infected individuals in the population at each time point. Most infectious disease data sets include observations of the number of infected individuals in the population, but do not include measures of the number of susceptible individuals. Furthermore, the SIR model is a very simplified representation of the evolution of the disease. In reality, there are likely to be many other important factors affecting disease dynamics such as weather, immigration, and the presence of other diseases. Many of these other factors may also be unobserved.

MoA is an alternate, non-parametric aproach to learning about dynamical systems that can be applied when we are uncertain of the functional forms describing evolution of the process and we do not have observations of all of the

*Email: elray@umass.edu; Corresponding author

state variables. The method can be motivated by Takens' theorem [Takens, 1981], which establishes a connection between the original dynamical process and the process that is obtained by forming vectors of lagged observations. Roughly, the theorem states that if the dynamical process satisfies certain conditions, there is a smooth, one-to-one and onto mapping with a smooth inverse between the limit sets of the original dynamical process and the limit sets of the lagged observation process. This provides some justification for studying the lagged observation process in order to learn about long-term dynamics of the system. The theorem also tells us that we must include at least $2D+1$ lags in the lagged observations in order to guarantee that the limit sets are equivalent in the sense described above, where D is the dimension of the state space of the underlying dynamical process.

Although this theorem can help motivate the study of the lagged observation process, it does not answer two key questions for a practical implementation:

1. If the dimension D of the underlying process is unknown, we do not know how many or which lags should be included when we form the lagged observation process.

2. It does not provide us with a specific method for studying the lagged observation process or using it to perform prediction.

To do: Insert some pictures and a brief description of the intuitive idea.

We use a non-parametric approach based on kernel regression (for point predictions) and kernel density estimation (for predictive distributions) to learn the relationship between the lagged observation process and its future values. We use cross-validation to select the lags that are included and estimate the bandwidth parameters for the kernel functions. We give a more formal statement of the method and our estimation procedures in Sections 3 and 4.

3 Method Description

Suppose we observe $\mathbf{z}_t = \{z_{t,1}, \ldots, z_{t,D}\} \in \mathbb{R}^D$ at each point in time $t = 1, \ldots, T$. Our goal is to obtain a predictive distribution for one of the observed variables, with index $d_{pred} \in \{1, \ldots, D\}$, over a range of prediction horizons contained in the set $\mathcal{P}$. For example, if we have weekly data and we are interested in obtaining predictions for a range between 4 and 6 weeks after the most recent observation then $\mathcal{P} = \{4, 5, 6\}$. Let P be the largest element of the set $\mathcal{P}$ of prediction horizons.

In order to perform prediction, we will use lagged observations. Let $\mathbf{l}^{max} = (l_1^{max}, \ldots, l_D^{max})$ specify the maximum number of lags for each observed variable that may be used for prediction, and let $L = \max_d l_d^{max}$ be the overall largest lag that may be used across all variables. In the estimation procedure we describe in Section 4, we will select a subset of these lags to actually use in the predictions. We capture which lags are actually used in the vector

$$\mathbf{u} = (u_{1,0}, \ldots, u_{1,l_1^{max}}, \ldots, u_{D,0}, \ldots, u_{D,l_D^{max}}), \text{ where}$$

$$u_{d,l} = \begin{cases} 0 \text{ if lag } l \text{ of variable } d \text{ is not used in forming predictions} \\ 1 \text{ if lag } l \text{ of variable } d \text{ is used in forming predictions.} \end{cases}$$

By analogy with the standard notation in autoregressive models, we define

$$\mathbf{y}_t = (z_{t,d_{pred}}, \ldots, B^{(P-1)} z_{t,d_{pred}}) \text{ and}$$
$$\mathbf{x}_t = (B^{(P)} z_{t,1}, \ldots, B^{(P+l_1^{max}-1)} z_{t,1}, \ldots, B^{(P)} z_{t,D}, \ldots, B^{(P+l_D^{max}-1)} z_{t,D})$$

Here, $B^{(a)}$ is the backshift operator defined by $B^{(a)}z_{t,d} = z_{t-a,d}$. Note that the lengths of $\mathbf{y}_t$ and $\mathbf{x}_t$, as well as exactly which lags are used to form them, depend on $\mathcal{P}$ and $\mathbf{l}^{max}$; we suppress this dependence in the notation for the sake of clarity. The vector $\mathbf{y}_t$ represents the prediction target when our most recent observation was made at time $t - P$: the vector of observed values at each prediction horizon $p \in \mathcal{P}$. The variable $\mathbf{x}_t$ represents the vector of all lagged covariates that are available for use in performing prediction.

To make the notation concrete, suppose that $\mathbf{z}_t$ contains the observed case count for week t in San Juan, the observed case count for week t in Iquitos, and the date on Monday of week t, and our goal is to predict the weekly case count in San Juan. Then $D = 3$ and $d_{pred} = 1$. If we want to predict the weekly case counts for the two weeks after the most recently observation, then $p = 2$. If we specify that the model may include the two most recent observations for the case counts in San Juan and Iquitos, but only the time index at the most recent observation then $\mathbf{l}^{max} = (1, 1, 0)$. If our current model uses only the most recently observed case counts for San Juan and Iquitos then $\mathbf{u} = (1, 0, 1, 0, 0)$, where the 1's are in the positions of the $\mathbf{u}$ vector representing lag 0 of the counts for San Juan and lag 0 of the counts for Iquitos. The variable $y_t^{(P)}$ is a vector containing the observed case counts for San Juan in weeks $t + 1$ and $t + 2$; $\mathbf{x}_t^{(\mathbf{l}^{max})}$ contains the observed case counts for San Juan and Iquitos in weeks t and $t - 1$ as well as the time index variable in week t.

In order to perform prediction, we regard $\{(\mathbf{y}_t, \mathbf{x}_t), t = 1 + P + L, \ldots, T\}$ as a sample from the joint distribution of $(\mathbf{Y}, \mathbf{X})$. We wish to estimate the conditional distribution of $\mathbf{Y}|\mathbf{X}$. In order to do this, we employ kernel density estimation. Let $K^{\mathbf{Y}}(\mathbf{y}, \mathbf{y}^*, H^{\mathbf{Y}})$ and $K^{\mathbf{X}}(\mathbf{x}, \mathbf{x}^*, H^{\mathbf{X}})$ be kernel functions centered at $\mathbf{y}^*$ and $\mathbf{x}^*$ respectively and with bandwidth matrices $H^{\mathbf{Y}}$ and $H^{\mathbf{X}}$. We estimate the conditional distribution of $\mathbf{Y}|\mathbf{X}$ as follows:

$$\widehat{f}_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}|\mathbf{X} = \mathbf{x}) = \frac{\widehat{f}_{\mathbf{Y},\mathbf{X}}(\mathbf{y}, \mathbf{x})}{\widehat{f}_{\mathbf{X}}(\mathbf{x})} \tag{1}$$

$$= \frac{\sum_{t \in \tau} K^{\mathbf{Y}}(\mathbf{y}, \mathbf{y}_t, H^{\mathbf{Y}}) K^{\mathbf{X}}(\mathbf{x}, \mathbf{x}_t, H^{\mathbf{X}})}{\sum_{t \in \tau} K^{\mathbf{X}}(\mathbf{x}, \mathbf{x}_t, H^{\mathbf{X}})} \tag{2}$$

$$= \sum_{t \in \tau} w_t K^{\mathbf{Y}}(\mathbf{y}, \mathbf{y}_t, H^{\mathbf{Y}}), \text{ where} \tag{3}$$

$$w_t = \frac{K^{\mathbf{X}}(\mathbf{x}, \mathbf{x}_t, H^{\mathbf{X}})}{\sum_{t^* \in \tau} K^{\mathbf{X}}(\mathbf{x}, \mathbf{x}_{t^*}, H^{\mathbf{X}})} \tag{4}$$

In Equation (1), we are simply making use of the fact that the conditional density for $\mathbf{Y}|\mathbf{X}$ can be written as the quotient of the joint density for $(\mathbf{Y}, \mathbf{X})$ and the marginal density for $\mathbf{X}$. In Equation (2), we obtain separate kernel density estimates for the joint and marginal densities in this quotient. In Equation (3), we rewrite this quotient by passing the denominator of Equation (2) into the summation in the numerator. We can interpret the result as a weighted kernel density estimate, where each observation $t \in \tau$ contributes a different amount to the final conditional density estimate. The amount of the contribution from observation t is given by the weight w_t, which effectively measures how similar $\mathbf{x}_t$ is to the point $\mathbf{x}$ at which we are estimating the conditional density. If $\mathbf{x}_t^{(\mathbf{l}^{max})}$ is similar to $\mathbf{x}_{t^*}^{(\mathbf{l}^{max})}$, a large weight is assigned to t; if $\mathbf{x}_t^{(\mathbf{l}^{max})}$ is different from $\mathbf{x}_{t^*}^{(\mathbf{l}^{max})}$, a small weight is assigned to t.

In kernel density estimation, it is generally required that the kernel functions integrate to 1 in order to obtain valid density estimates. However, after conditioning on $\mathbf{X}$, it is no longer necessary that $K^{\mathbf{X}}(\mathbf{x}, \mathbf{x}_t, H^{\mathbf{X}})$ integrate to 1. In fact, as can be seen from Equation (4), any multiplicative constants of proportionality will cancel out when we form the observation weights. We can therefore regard $K^{\mathbf{X}}(\mathbf{x}, \mathbf{x}_t, H^{\mathbf{X}})$ as a more general weighting function that measures the similarity between $\mathbf{x}$ and $\mathbf{x}_t$. As we will see, eliminating the constraint that $K^{\mathbf{X}}$ integrates to 1 is a useful expansion the space of functions that can be used in calculating the observation weights. However, we still

require that $K^{\mathbf{Y}}$ integrates to 1.

In Equations (1) through (4), τ is an index set of time points used in obtaining the density estimate. In most settings, we can take $\tau = \{1+P+L, \ldots, T\}$. These are the time points for which we can form the lagged observation vector $\mathbf{x}_t$ and the prediction target vector $\mathbf{y}_t$. However, we will place additional restrictions on the time points included in τ in the cross-validation procedure discussed in Section 4.

If we wish to obtain point predictions, we can use a summary of the predictive density. For example, if we take the expected value, we obtain kernel regression:

$$(\widehat{\mathbf{Y}}|\mathbf{X} = \mathbf{x}) = \mathbb{E}_{\widehat{f}_{\mathbf{Y}|\mathbf{X}}}\{\mathbf{Y}|\mathbf{X} = \mathbf{x}\} \tag{5}$$

$$= \int \sum_{t\in\tau} w_t K^{\mathbf{Y}}(\mathbf{y}, \mathbf{y}_t, H^{\mathbf{Y}})\mathbf{y}\, d\mathbf{y} \tag{6}$$

$$= \sum_{t\in\tau} w_t \mathbf{y}_t \tag{7}$$

The equality in Equation (7) holds if the kernel function $K^{\mathbf{Y}}(\mathbf{y}, \mathbf{y}_t, H^{\mathbf{Y}})$ is symmetric about $\mathbf{y}_t$, or more generally if it is the pdf of a random variable with expected value $\mathbf{y}_t$.

In our work, we have used product kernel functions for $K^{\mathbf{Y}}(\mathbf{y}, \mathbf{y}_t, H^{\mathbf{Y}})$ and $K^{\mathbf{X}}(\mathbf{x}, \mathbf{x}_t, H^{\mathbf{X}})$:

$$K^{\mathbf{Y}}(\mathbf{y}, \mathbf{y}_t, H^{\mathbf{Y}}) = \prod_{p\in\mathcal{P}} K_p^{\mathbf{Y}}(\mathbf{y}_p, \mathbf{y}_{t,p}], h_p^{\mathbf{Y}})$$

$$K^{\mathbf{X}}(\mathbf{x}, \mathbf{x}_t, H^{\mathbf{X}}) = \prod_{v=1}^{V} K_v^{\mathbf{X}}(\mathbf{x}_v, \mathbf{x}_{t,v}, H_v^{\mathbf{X}}), \text{ where } V = \sum_{d=1}^{D} \mathbf{l}_d^{max} \text{ is the total length of } \mathbf{X}.$$

The use of product kernels may not be optimal; this aspect of our approach could be revised in future work.

We have employed two functional forms for the component univariate kernel functions in our work. We use the squared exponential kernel for most of the component kernel functions in both $K^{\mathbf{Y}}$ and $K^{\mathbf{X}}$. This function has the following form:

$$K(x, x^*, h) \propto \exp\left\{-\frac{1}{2}\left(\frac{x - x^*}{h}\right)^2\right\}$$

The constant of proportionality can be ignored when this is used as a component kernel function in $K^{\mathbf{X}}$; when used as a component function in $K^{\mathbf{Y}}$, is is resolved by the constraint that the kernel function integrate to 1. It should be noted that the use of this kernel function in $K^{\mathbf{Y}}$ is technically inappropriatefor our application, since the prediction target (case counts) is a discrete random variable but the squared exponential kernel is defined for continuous values. We have adopted the squared exponential kernel for this purpose due to time constraints, and we plan to pursue the use of discrete kernels in future work.

The second kernel we have employed in the weighting kernel K^{wt} is the periodic kernel, which is commonly used in the literature on Gaussian Processes [Rasmussen and Williams, 2006]. This kernel is defined as follows:

$$K(x, x^*, h) \propto \exp\left[-\frac{1}{2} sin^2\left\{\rho(x - x^*)\right\}/h^2\right]$$

The parameter ρ determines the period of the kernel function. In our work, we have fixed ρ so that the kernel has a period of 1 year and estimated only the bandwidth h. We have employed the periodic kernel only for the component of the weighting kernel corresponding to the time index variable. Effectively, this means that observations are scored

as being similar to each other if they were recorded at a similar time of the year.

There are several details specific to our application of the SSR method to the Dengue challenge data which we now discuss. First, it can be seen from the plots in Figure **??** that the weekly case counts are not smooth over time. The plot in panel (a) of the figure shows that the weekly case counts are jagged, with many small peaks around a larger trend. In panel (b), we see that this induces some "loops" in the lagged covariate space when a lag of one week is used. These loops can cause difficulties for the SSR method, because lagged covariate vectors representing different points in the disease's evolution may be near each other. This problem can be addressed in several ways. One possibility is the use of a different lag; for example, these loops might be reduced if a lag of three weeks were used instead of one week.

Another alternative that we pursue is the use of smoothed observations in forming the lagged observation vectors. We use smoothed case counts on a log scale for the weighting kernels, and the unsmoothed case counts on the original scale for the prediction kernels.

To do:

- more detail about smoothing.
- edge effects – for each x_t, smooth up to time t
- probably also smooth y_t
- what smoothing method to use?

4 Parameter Estimation

We use cross-validation to select the variables that are used in the model and estimate the corresponding bandwidth parameters by (approximately) minimizing a cross-validation measure of the quality of the predictions obtained from the model. Formally,

$$(\widehat{\mathbf{u}}, \widehat{H}^{\mathbf{X}}, \widehat{H}^{\mathbf{Y}}) \approx \underset{(\mathbf{u}, H^{\mathbf{X}}, H^{\mathbf{Y}})}{\operatorname{argmin}} \sum_{t^*=1+P+L}^{T} Q[\mathbf{y}_{t^*}, \widehat{f}(\mathbf{y}|\mathbf{X} = \mathbf{x}_{t^*}; \mathbf{u}, H^{\mathbf{X}}, H^{\mathbf{Y}}, \{(\mathbf{y}_t, \mathbf{x}_t) : t \in \tau_{t^*}\})] \tag{8}$$

Here, Q is a loss function that measures the quality of the estimated density $\widehat{f}$ given an observation $\mathbf{y}_{t^*}$. We have made the dependence of this estimated density on the the parameters $\mathbf{u}$, $H^{\mathbf{x}}$, and $H^{\mathbf{Y}}$, as well as on the data $\{(\mathbf{y}_t, \mathbf{x}_t) : t \in \tau_{t^*}\}$, explicit in the notation. In order to reduce the potential for our parameter estimates to be affected by local correlation in the time series, we eliminate all time points that fall within one year of t^* from the index set τ_{t^*} used to form the conditional density estimate $\widehat{f}(\mathbf{y}|\mathbf{X} = \mathbf{x}_{t^*}; \mathbf{u}, H^{\mathbf{X}}, H^{\mathbf{Y}}, \{(\mathbf{y}_t, \mathbf{x}_t) : t \in \tau_{t^*}\})$.

Talk about proper scoring rules and our particular choice of Q.

We use a forward/backward stagewise procedure to obtain the set of combinations of variables and lags that are included in the final model (represented by $\mathbf{u}$). For each candidate model, we use the limited memory box constrained optimization procedure of Byrd et al. [1995] to estimate the bandwidth parameters. The approximation in Equation (8) is due to the fact that this optimization procedure may not find a global minimum.

5 Predictions for Challenge

6 Variables

7 Computational Resources

We implemented our variation on SSR in the `R` statistical programming language [R Core Team, 2013].

8 Publications

Richard H Byrd, Peihuang Lu, Jorge Nocedal, and Ciyou Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16(5):1190–1208, 1995.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2013. URL `http://www.R-project.org/`.

Jeff Racine and Qi Li. Nonparametric estimation of regression functions with both categorical and continuous data. *Journal of Econometrics*, 119(1):99–130, 2004.

Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian processes for machine learning*. The MIT Press, 2006.

B.W. Silverman. *Density Estimation for Statistics and Data Analysis*. Monographs on Statistics and Applied Probability. Chapman and Hall, London, UK, 1986.

George Sugihara and Robert M. May. Nonlinear forecasting as a way of distinguishing chaos from measurement error in time series, April 1990.

Floris Takens. *Detecting strange attractors in turbulence*. Springer, 1981.