Validating Forecasting Strategies of Simple Epidemic Models on the 2015-2016 Zika Epidemic

Nicolas L. Puglisi

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Mathematics

Omar Saucedo, Chair
Leah Johnson
Michael Robert

April 26, 2024
Blacksburg, Virginia

Keywords: Mathematical Modeling, Forecasting, Infectious Disease, Public Health

Validating Forecasting Strategies of Simple Epidemic Models on the 2015-2016 Zika Epidemic

Nicolas L. Puglisi

(ABSTRACT)

Accurate forecasting of infectious disease outbreaks is vital for safeguarding global health and the well-being of individuals. Model-based forecasts enable public health officials to test what-if scenarios, evaluate control strategies, and develop informed policies to allocate resources effectively. Model selection is a pivotal aspect of creating dependable forecasts for infectious diseases. This thesis delves into validating forecasts of simple epidemic models. We use incidence data from the 2015-2016 Zika virus outbreak in Antioquia, Colombia to assess what model features result in accurate forecasts. We employed the Parametric Bootstrapping and Ensemble Kalman Filter methods to assimilate data and then generated 14-day-ahead forecasts throughout the epidemic across five case studies. We visualized each forecast to show the training/testing split in data and associated prediction intervals. Forecasting accuracy was evaluated using five statistical performance metrics. Early into the epidemic, phenomenological models - like the generalized logistic model - resulted in more accurate forecasts. However, as the epidemic progressed, the mechanistic model incorporating disease latency outperformed its counterparts. While modeling disease transmission mechanisms is crucial for accurate Zika incidence forecasting, additional data is needed to make these models more reliable and precise.

Validating Forecasting Strategies of Simple Epidemic Models on the 2015-2016 Zika Epidemic

Nicolas L. Puglisi

(GENERAL AUDIENCE ABSTRACT)

Accurate forecasting of infectious disease outbreaks is vital for safeguarding global health and the well-being of individuals. Model-based forecasts enable public health officials to test what-if scenarios, evaluate control strategies, and develop informed policies to allocate resources effectively. Model selection is a pivotal aspect of creating dependable forecasts for infectious diseases. This thesis delves into validating forecasts of simple epidemic models. We use data from the 2015-2016 Zika virus outbreak in Antioquia, Colombia, to assess what model features result in accurate forecasts. We considered two techniques to generate 14-day-ahead forecasts throughout the epidemic across five case studies. We visualized each forecast and evaluated model accuracy. Early into the epidemic, simple growth models resulted in more accurate forecasts. However, as the epidemic progressed, the model incorporating disease-specific characteristics outperformed its counterparts. While modeling disease transmission is crucial for accurate epidemic forecasting, additional data is needed to make these models more reliable and precise.

Dedication

I dedicate this thesis to my family, who have supported me every step of the way.

Acknowledgments

Thank you to my advisor, Dr. Omar Saucedo, for providing mentorship throughout every step of this journey. Thank you to all the faculty and students of the MathBio lab for creating the most welcoming educational environment. Thank you to my friends for lifting my spirits when I struggled. Thank you to Alayna for being patient and understanding. Finally, thank you to my family for all the love and support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Accurate forecasting of infectious disease outbreaks, including events like the COVID-19 pandemic, is crucial in safeguarding global health and well-being. By anticipating the trajectory of disease transmission and monitoring real-time case numbers, we can empower public health authorities to mount rapid and effective responses during epidemics. Epidemic modeling and forecasting serve as a critical bellwether for allocating resources, making informed policy decisions, and implementing targeted preventive measures [13, 18, 25]. Ultimately, forecasting is a proactive approach that can mitigate the far-reaching consequences of infectious disease outbreaks.

Forecasting is a powerful tool for addressing essential public health questions; however, several challenges exist. First, accurate data availability remains critical for reliable forecasts. Second, human behavior during an epidemic can significantly impact transmission dynamics, deviating from assumed patterns [28]. Additionally, there is still a need for a unifying framework to validate and assess model-based forecasts of infectious diseases. Despite these hurdles, the COVID-19 pandemic spurred collaborative efforts in epidemic modeling, leading to the establishment of the Centers for Disease Control and Prevention (CDC) COVID-19 Forecasting Hub [9]. While forecasting will not result in a perfect prediction of an epidemic's future outcome, it can enable public health officials to test what-if scenarios, evaluate control strategies, and develop informed policies.

1.1 Zika Virus Forecasting Strategies

Zika virus (ZIKV) is a mosquito-borne disease characterized by "dengue-like" symptoms, consisting of fever, rash, and muscle and joint pain [5, 30]. On February 1, 2016, The World Health Organization (WHO) declared Zika-related microcephaly a Public Health Emergency of International Concern (PHEIC), lasting until November of 2016 [30]. Zika has posed significant control and epidemic modeling challenges due to the existence of multiple transmission pathways such as mosquito-to-humans through bites, human-to-human sexual transmission, and human-to-human vertical transmission [30]. In addition, most people who become infected will not show symptoms of the disease [14]. Current surveillance data suggests that Zika has affected 89 countries and territories, and there is no treatment or preventative medicine to combat infection [14, 30]. This thesis explores a range of Zika forecasting strategies - starting with simple epidemic models - to develop a cohesive framework to validate which model features and mechanisms result in more accurate forecasts of Zika incidence in a forward stepwise fashion.

The primary thesis question is: How do model-based forecasts of simple epidemic models compare under a Parametric Bootstrapping and Ensemble Kalman Filtering approach? Stemming from the need to evaluate model formulation and mechanism, we then ask: Does the top-performing model change as the epidemic progresses, and how do spikes in Zika incidence affect the forecasting performance of each model? This thesis will explore these questions.

Chapter 2

Review of Literature

Chapter 2 seeks to highlight previous studies that have investigated questions relating to methods and their applications for modeling infectious diseases and forecasting. As modeling relates to forecasting, there is a rich collection of studies on forecasting infectious disease [9, 15, 26, 32]. A fundamental step in forecasting emergent infectious diseases is to address parameter and model uncertainty, which a method known as Parametric Bootstrapping enables through repeated sampling from a best-fit model [2, 6, 31].

Another common method of assimilating data to quantify uncertainty within a system is the Ensemble Kalman Filter. The Ensemble Kalman Kalman Filter acts by propagating model states forward in time with added noise and then updates the estimated states by computing the Kalman gain matrix with observed data [12, 19]. Applications of the Ensemble Kalman filter have also been successfully used to explore a common issue for public health officials: under-reporting in case data [27]. While many robust frameworks for modeling and forecasting exist, every system is susceptible to errors; ensemble modeling aims to offset biases inherent to a given model's structure [36, 37]. Ensemble models allow for multiple types of data and techniques to be combined together, often resulting in improved forecasts [3, 13].

Another crucial step within the model forecasting workflow is validation. We use statistical metrics to measure prediction error and determine which models result in more accurate forecasts [11, 16, 24]. Furthermore, we will discuss particular modeling frameworks to bridge

knowledge into application on Zika.

2.1 Parametric Bootstrapping Literature

In the paper "Fitting dynamic models to epidemic outbreaks with quantified uncertainty: A primer for parameter uncertainty, identifiability, and forecasts," Gerardo Chowell describes a data-oriented modeling framework able to assimilate observations, estimate transmission parameters, and generate forecasts [2]. Discussion on this framework begins by classifying the types of mathematical models that can be applied to this framework, namely phenomenological and mechanistic models.

Phenomenological models are empirical, not relying on specific physical laws or mechanisms. Two common phenomenological models are the Generalized-Growth Model (GGM) and its extension, the Generalized Richards Model (GRM). The appeal of Phenomenological models is that they are easy to implement and usually require less data due to the fewer number of parameters compared to mechanistic techniques. Mechanistic models aim to explain and capture patterns within data by incorporating physical laws, such as population and transmission dynamics. A canonical example includes the Susceptible-Infected-Recovered (SIR) model, along with the numerous extensions, such as the Susceptible-Exposed-Infected-Recovered (SEIR) model and Ross-Macdonald model for host-vector dynamics [23].

Mathematical model parameter estimates are subject to uncertainty, which can lead to unrealistic representations of a system when not accounted for appropriately. Model uncertainty can arise from sources such as noise in the data and underlying assumptions of a model's structure. Quantifying parameter and model uncertainty is crucial for understanding disease dynamics and developing successful forecasts. The parametric bootstrapping framework begins by estimating parameter values through a least-squares fitting approach to the time-

series data to obtain a best-fit model $f(t_i, \hat{\Theta})$ for $i = 1, \cdots, n$ observations. Datasets are repeatedly generated from the best-fit model to re-estimate parameters and obtain a sampling distribution for the model parameters. Phenomenological models have often been selected for use under this framework because they can be implemented quickly to characterize disease dynamics and make short-term forecasts.

Under the parametric bootstrap framework, forecasting a model is a relatively straightforward procedure. First, model parameter sets, $\{\Theta_1, \Theta_2, ..., \Theta_S\}$ are sampled directly from the empirical distributions generated by the parametric bootstraps, and then a collection of models: $f(t, \Theta_1), f(t, \Theta_2), ..., f(t, \Theta_S)$ are pushed forward by a time horizon of h units to: $f(t + h, \Theta_1), f(t + h, \Theta_2), ..., f(t + h, \Theta_S)$ [2]. Chowell ends the discussion by arguing that researchers need to consider a host of different model types to bolster our efforts in forecasting. Ultimately, this notion is the springboard for the research done in this study - how do different forecasting techniques and models compare when validated on the 2015-2016 Zika epidemic?

Continuing with Parametric Bootstrapping literature, Chowell et al. follow up the work from the previous paper in "An ensemble n-sub-epidemic modeling framework for short-term forecasting epidemic trajectories: Application to the COVID-19 pandemic in the USA." The paper's primary goal was to construct an n-sub-epidemic model and define a weighting strategy to combine the best-performing models into an ensemble [3]. To achieve this, Chowell et al. applied the parametric bootstrapping technique on an extended, generalized-logistic growth model (GLM) and then ranked the performance from best to worst in order of ascending AIC_c [3]. Having ranked models from best to worst, individual model contributions are weighted to create an ensemble with J members. The contribution w_i for the i-th best model is computed as:

$$w_i = \frac{\frac{1}{AIC_c i}}{\frac{1}{AIC_c 1} + \frac{1}{AIC_c 2} + ... + \frac{1}{AIC_c J}}, \tag{2.1}$$

where $i = 1, ..., J$ [7]. Forecasting the ensemble model is carried out in the same way as in [2]; Chowell et al. assessed the model calibration and short-term forecasts on COVID-19 pandemic data ranging from April 2020 to February 2022.

2.2 Kalman Filtering Literature

Continuing with the discussion of forecasting literature, the Kalman Filter is another commonly used data assimilation framework. Ghostine et al. demonstrate the successful use of the Ensemble Kalman Filter in estimating states and constraining model output to boost forecasting performance [19]. Ghostine et al. analyzed an extended SEIR compartmental to demonstrate the non-negativity required for biological interpretation and the existence of a unique endemic equilibrium. From there, Ghostine et al. begin the discussion on their data assimilation regimen; data utilized for the study came from the Saudi Center for Disease Prevention and included information on the number of deaths, recovered, and confirmed cases from the COVID-19 pandemic. An augmented ensemble Kalman filter was used to estimate the state variables and parameters. The SEIR system was then evaluated for forecasting performance by making two-week-ahead forecasts across the time span of data. The model performed well, with the root mean absolute error (RMAE) values remaining within the margin of observational error [19].

As with the Parametric Bootstrap framework, the Ensemble Kalman Filter has many successful applications in forecasting many different infectious diseases. However, much of the literature presented focuses on a single model at a time but not on validation against other techniques.

2.3 Ensemble Modeling

Following the discussion of the Parametric Bootstrap and Kalman Filtering frameworks, a key point to note is both methodologies employ ensembles as a way to understand the uncertainty structure required for forecasting. However, this is not the only interpretation of an ensemble model. Yamana et al. begin the discussion of superensemble modeling, a type of ensemble combining multiple sources of data, by highlighting recent work that has demonstrated how infectious disease forecasts have become more accurate [36]. Using a single model often results in errors due to model misspecifications. Weather forecasting systems bypass these misspecification issues by implementing ensemble techniques, where the inherent biases of one system get balanced by another.

Yamana et al. built three distinct forecasting systems using weekly dengue incidence spanning from April 1990 to April 2023, sourced by the Puerto Rico Department of Health and the CDC [36]. The first model selected for the ensemble is a SIR model utilizing an ensemble adjustment Kalman filter; it is important to note the simplifying assumption on the model structure due to the lack of data on mosquito infection rates and patient immunological history [36]. The second model employs a statistical approach in which the outbreak to be forecasted is generated by weighting outbreak trajectories from prior dengue seasons. The third model utilized historical likelihood forecasts by fitting probability distributions to historical data on outbreak characteristics of past dengue seasons.

Yamana et al. generated forecasts from the three above methods in multiple combinations ranging from a pair of two to a full superensemble featuring all three forecasting techniques [36]. The superensemble forecasts performed well, with mean absolute error (MAE) equal to or lower than individual forecasts. While the superensembles perform well, such systems require a rich history of data to train - data that might not be available for emergent infectious

diseases.

2.4 Conclusion of Literature Review

As discussed throughout Chapter 2, extensive research centers on infectious disease modeling and forecasting. The Parametric Bootstrapping framework is flexible regarding the model and assumed error structure that can be applied to the framework. The Ensemble Kalman Filtering framework is another powerful forecasting tool with successful applications across many disciplines and is attractive due to the computational ease of incorporating large amounts of data. When applicable, ensemble modeling is a powerful method able to smooth over biases of individual models and improve forecasting performance. Each data assimilation framework presented has trade-offs with varying levels of complexity and strength of assumptions. Thus, there is a need to validate these techniques and determine which results in more accurate disease forecasting.

Validation is one of the most crucial steps in the forecasting workflow. It is necessary to understand better which modeling techniques and features result in high-quality forecasts to hone our attempts at predicting the future of epidemics. Zika is of particular interest to this study because there are multiple transmission pathways that public health officials could consider when developing a forecasting system. Previous Zika research has investigated questions of prediction using machine-learning techniques, Google Trends data, and mathematical models incorporating vector dynamics [1, 17, 29]. However, we must still determine which mathematical models and transmission pathways are best suited for forecasting the Zika virus and if there is a gain in performance over simple models. This study seeks to develop a validation framework that starts with simple epidemic models and builds up in complexity - like a forward stepwise approach - to better future attempts at forecasting Zika

epidemics.

Chapter 3

Methods

Chapter 3 provides extensive background information on the Parametric bootstrapping and Ensemble Kalman Filtering methods, epidemic models, and model validation strategies used for comparison in this study. To achieve this, we will first present the steps of the Parametric Bootstrapping framework in depth.

3.1 Description of Parametric Bootstrapping

The Parametric Bootstrap is a data assimilation framework where a best-fitted model, denoted $f(t_i, \hat{\Theta})$, is used to resample data. In this way, bootstrapping allows for quantifying parameter uncertainty through sampling distributions of model parameters. To start the algorithm, we first need to obtain parameter estimates $\hat{\Theta} = \{\hat{\theta}_1, \hat{\theta}_2, ..., \hat{\theta}_m\}$ by fitting the model $f(t_i, \Theta)$ to time series data in hand. The parametric bootstrap framework is flexible in model formulation, and two examples that have extensive use for the framework are the generalized growth model (GGM) and the generalized Richards model (GRM); the GGM describes incidence growth and is characterized by the following differential equation:

$$\frac{dC}{dt} = rC^p(t),$$

where $C(t)$ is the solution for the cumulative number of cases at time t, r is a growth rate parameter, and p is a "deceleration of growth" parameter bounded between 0 and 1 [2]. The GRM is an extension to the GGM with a "size of epidemic parameter" added and defined by the following differential equations:

$$\frac{dC}{dt} = rC^p\Big(1 - \Big(\frac{C}{K}\Big)^a\Big),$$

with r representing the intrinsic growth rate parameter, K is the parameter for the epidemic size, and the parameter a scales logistic growth [2, 35]. There are countless other models to explore for forecasting, so discussion on the models included for the thesis is provided in Section 3.3. As for data fitting techniques, the parametric bootstrap framework is also flexible, but for the purposes of this thesis, least-squares fitting techniques are used and defined as:

$$\hat{\Theta} = \texttt{argmin} \sum_{i=1}^{n} (f(t_i, \theta) - y_{t_i})^2,$$

where n denotes the number of data points, and t_i is an index of the time series data. Non-linear data fitting approaches are also applicable, including weighted least squares:

$$\hat{\Theta} = \texttt{argmin} \sum_{i=1}^{n} w_{t_i}(f(t_i, \theta) - y_{t_i})^2,$$

where w_{t_i} are non-negative weights for each data point in the time series [2]. Using this best-fit model, we can simulate datasets to begin the bootstrapping process. To achieve this, the cumulative curve function $F^*(t_j, \hat{\Theta})$ as:

$$F^*(t_j, \hat{\Theta}) = \sum_{k=1}^{j} f(t_k, \hat{\Theta}), \quad j = 1, 2, ..., n.$$

The cumulative curve is used to simulate S replicate data sets $\{f_1^*(t_j, \hat{\Theta}), f_2^*(t_j, \hat{\Theta}), ..., f_S^*(t_j, \hat{\Theta})\}$, where each data set $f_S^*(t_j, \hat{\Theta})$ is generated by assuming Poisson error structure such that:

$$f_l^*(t_j, \hat{\Theta}) = Po(F(t_j, \hat{\Theta}) - F(t_{j-1}, \hat{\Theta})), \quad j = 2, 3, ..., n, \quad l = 1, 2, ..., S.$$

The assumption of Poisson error can be relaxed to include negative binomial or normal error structures. From here, model parameters are re-estimated for each of the simulated data sets to obtain the set $\{\hat{\Theta}_1, \hat{\Theta}_2, ..., \hat{\Theta}_S\}$. Finally, the re-estimated parameters are used to characterize parameter sampling distributions. Now that we have quantified model uncertainty, short-term forecasting becomes possible by sampling parameter sets $\{\Theta_1, \Theta_2, ..., \Theta_S\}$ from the previously derived distributions. Then, an ensemble of models $f(t, \Theta_1), f(t, \Theta_2), ..., f(t, \Theta_S)$ is propagated forwards in time by h units to a future state $f(t + h, \Theta_1), f(t + h, \Theta_2), ..., f(t + h, \Theta_S)$.

3.2 Description of the Kalman Filter

Before providing an in-depth discussion on the Ensemble Kalman Filter, this section will give background on the classic Kalman Filter. The classic Kalman Filter is a data assimilation technique with a historical application in control theory and geophysical systems, like weather prediction [21, 22]. The Kalman Filter acts by considering the entire distribution of states for the evolution matrix F, and then relates states to observations through an observation matrix G [21, 22, 27]. This process is described by the following state-space system:

$$X_{j+1} = \mathrm{F}X_j + V_{j+1}, \; V_{j+1} \sim \mathcal{N}(0, \mathrm{C})$$

$$Y_{j+1} = \mathrm{G}X_{j+1} + W_{j+1}, \; W_{j+1} \sim \mathcal{N}(0, \mathrm{D})$$

where X_{j+1} is an n-dimensional vector containing information on unobserved states, Y_{j+1} is an m-dimensional vector of data, and V_{j+1} and W_{j+1} are normally distributed state and observation errors governed by covariances C and D, respectively, with notation following Mitchell and Arnold's work [27]. A key assumption for the classic Kalman Filter is linearity in the state operator, so direct applications of this method to infectious disease modeling are limited. However, the linearity assumption can be relaxed if we consider an ensemble Kalman Filter instead.

The Ensemble Kalman Filter (EnKF) is an extended version of the classic Kalman Filter, where the system is represented by an ensemble sample from state and observation distribution [27]; the addition of an ensemble sampling allows for non-linear models to be utilized. Assimilating data sequentially under this framework updates model uncertainty dynamically, informing state-covariance matrices. Sequential data assimilation is also "attractive" for applications within public health departments with regular disease incidence updates because these processes can be automated.

The EnKF algorithm is best described as a two-step process: the forecasting step, in which the forward model of interest estimates future states, and the analysis step, which corrects predictions using newly accessible data. Within the context of statistics, the EnKF is a Bayesian process where our posterior distributions of unknown states - and parameters in the augmented case - are sequentially updated through conditioning on observed data. The state-space model for the forecasting step of the EnKF is given by:

$$X_{j+1} = F(X_j) + V_{j+1},\ V_{j+1} \sim \mathcal{N}(0, \mathrm{C})$$
$$Y_{j+1} = G(X_{j+1}) + W_{j+1},\ W_{j+1} \sim \mathcal{N}(0, \mathrm{D}),$$

where F and G are nonlinear operators, X_j and Y_j are vectors of N ensemble members for

states and observations, and V_{j+1} and W_{j+1} are normally distributed state and observation errors with covariance C and D, respectively [22, 27].

Due to the implementation of an ensemble within the Kalman framework, the probability distributions are represented by discrete samples, where each ensemble member is propagated independently. To demonstrate this, let $S_{j|j}$ be a discrete sample from the posterior distribution at time j:

$$S_{j|j} = \{x^1_{j|j}, x^2_{j|j}, ..., x^N_{j|j}\},$$

where N is the ensemble size, and $x_{j|j} \in \mathbb{R}^d$ is a realized sample of states from the posterior distribution X_j at time j. The forecasting step acts on each of the N ensemble members by using the posterior distribution from the previous step, X_j, as a new prior and updates it by the state evolution:

$$x^n_{j+1|j} = F(x^n_{j|j}) + v^n_{j+1}, \quad n = 1, ..., N,$$

with each v^n_{j+1} being an i.i.d realization of the random variable $V_{j+1} \sim \mathcal{N}(0, C)$, and $x_{j+1|j}$ representing the state sample realizations evolved to time $j+1$, conditional to the previous step j. The benefit of using ensemble samples is now we can compute prediction mean $\bar{x}_{j+1|j}$ and sample covariance $\Gamma_{j+1|j}$ statistics to correct our state and observation prediction, defined as the following:

$$\bar{x}_{j+1|j} = \frac{1}{N}\sum_{n=1}^{N} x^n_{j+1|j} \in \mathbb{R}^d$$
$$\Gamma_{j+1|j} = \frac{1}{N-1}\sum_{n=1}^{N}(x_{j+1|j} - \bar{x}_{j+1|j})(x_{j+1|j} - \bar{x}_{j+1|j})^\intercal \in \mathbb{R}^{dxd},$$

where d is the dimension of the system [22, 27]. Using the ensemble statistics, each ensemble

member is then corrected during the analysis step:

$$x_{j+1|j+1} = x_{j+1|j} + K_{j+1}(y^n_{j+1} - \hat{y}^n_{j+1}), \quad n = 1, ..., N$$
$$y^n_{j+1} = y_{j+1} + w^n_{j+1}, \quad w_{j+1} \sim N(0, D),$$

with the collection of y^n_{j+1} being an artificially generated ensemble of observations centered around y_{j+1}, the datum being assimilated, and $\hat{y}^n_{j+1}$ being the predicted observation from the forecasting step. The Kalman gain K_{j+1} can be defined using cross-correlation information matrices as:

$$K_{j+1} = \Phi^{x\hat{y}}_{j+1}(\Phi^{\hat{y}\hat{y}}_{j+1} + D)^{-1},$$

where $\Phi^{x\hat{y}}_{j+1}$ is the cross-covariance matrix of the state and predicted observation, $\Phi^{\hat{y}\hat{y}}_{j+1}$ is the forecast error within the observation ensemble, and D is the observation covariance [22, 27]. After applying the Kalman gain, the analysis step results in the posterior ensemble

$$S_{j+1|j+1} = \{x^1_{j+1|j+1}, x^2_{j+1|j+1}, ..., x^N_{j+1|j+1}\}.$$

This whole framework continues until all time series data in hand are assimilated, and is selected in this study for comparisons against the parametric bootstrapping framework.

3.3 Mathematical Modeling

The general form of a dynamic model comprised of an n-dimensional system of ordinary differential equations is given by:

$$\begin{aligned}
\frac{dx_1}{dt} &= f_1(x_1, x_2, \cdots, x_n, \Theta) \\
\frac{dx_2}{dt} &= f_2(x_1, x_2, \cdots, x_n, \Theta) \\
&\vdots \\
\frac{dx_n}{dt} &= f_n(x_1, x_2, \cdots, x_n, \Theta),
\end{aligned}$$

where $\frac{dx_i}{dt}$ denotes the rate of change in the i-th system state and $\Theta = (\theta_1, \theta_2, \cdots, \theta_m)$ is the set of parameters. For forecast Zika incidence, we use simple epidemic models split into two classifications. First, phenomenological models describe growth without specifying physical laws or mechanisms. The first model of interest is the generalize-growth model (GGM), given by the following differential equation:

$$\frac{dC}{dt} = rC^p(t),$$

where, $\frac{dC}{dt}$ describes the growth in incidence at time t, $C(t)$ describes the cumulative incidence at time t, r is a growth parameter, and p is a growth-scaling parameter. The next forecasting model is the generalized-logistic model (GLM), given by the following differential equation:

$$\frac{dC}{dt} = rC^p\Big(1 - \Big(\frac{C}{K}\Big)\Big),$$

where, $\frac{dC}{dt}$ described the growth in incidence at time t, $C(t)$ describes the cumulative incidence at time t, r is a growth parameter, p is a growth-scaling parameter, and K is the size of the epidemic. The last phenomenological model of interest to the study is the generalized

Richards model (GRM), which is given by the following differential equation:

$$\frac{dC}{dt} = rC^p\Big(1 - \Big(\frac{C}{K}\Big)^a\Big),$$

where, $\frac{dC}{dt}$ describes the growth in incidence at time t, $C(t)$ describes the cumulative incidence at time t, r is a growth parameter, p is a growth-scaling parameter, K is the size of the epidemic, and a is a parameter that scales the sigmoidal dynamics from the GLM. The GGM, GLM, and GRM are included in this study to validate the forecasting performance of simple models and determine if increasing model complexity results in higher accuracy forecasts.

Mechanistic models are the other class of models included and incorporate specified population and disease transmission dynamics into the modeling structure. These models are often described as compartmental models, in which the host population is split into distinct epidemiological states, such as susceptible (S), infected (I), and recovered (R). The first compartmental model uses these states to form the SIR model, defined by the following system of differential equations:

$$\begin{aligned}
\frac{dS}{dt} &= -\frac{\beta SI}{N} \\
\frac{dI}{dt} &= \frac{\beta SI}{N} - \gamma I \\
\frac{dR}{dt} &= \gamma I \\
\frac{dC}{dt} &= \frac{\beta SI}{N},
\end{aligned}$$

where S, I, and R represent the susceptible, infected, and recovered population classes, respectively, C denotes an auxiliary state to track the cumulative incidence, β is a constant transmission parameter between susceptible and infected hosts, N is the population size, and

$1/\gamma$ represents the mean infectious period. The SEIR model is an extension of the SIR model that incorporates a disease latency mechanism. The SEIR model is given by the following system of ordinary differential equations:

Susceptible-Exposed-Infected-Recovered Model (SEIR):

$$\begin{aligned}
\frac{dS}{dt} &= -\frac{\beta SI}{N} \\
\frac{dE}{dt} &= \frac{\beta SI}{N} - \kappa E \\
\frac{dI}{dt} &= \kappa E - \gamma I \\
\frac{dR}{dt} &= \gamma I \\
\frac{dC}{dt} &= \kappa E,
\end{aligned}$$

where S, E, I, and R represent the susceptible, exposed, infected, and recovered population classes, respectively, C denotes an auxiliary state to track the cumulative incidence, β is a constant transmission parameter between susceptible and infected hosts, N is the population size, $1/\kappa$ represents the mean latent period, and $1/\gamma$ represents the mean infectious period.

3.3.1 Model Performance

Thus far, we have discussed the literature and methodologies behind forecasting under the parametric bootstrap and ensemble Kalman filter frameworks, but a key component in the forecasting workflow is validation. To achieve this, we will evaluate models based on a series of performance metrics to quantify the prediction error against observed incidence. Just as it is standard practice to consider multiple model performance metrics when determining which model is "best" equipped to describe dynamics present within data, it is important to have multiple metrics to evaluate a given forecasting period. The performance metrics

considered for this thesis are the Mean Squared Error (MSE), Root-Mean Squared Error (RMSE), and Mean Absolute Error (MAE). The MSE is given by:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(f(t_i,\Theta) - y_{t_i})^2.$$

The RMSE is given by:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(f(t_i,\Theta) - y_{t_i})^2}.$$

The MAE is given by:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|f(t_i,\Theta) - y_{t_i}|.$$

The above performance metrics are selected for their interpretability and frequent application to evaluating forecasting and have seen use within forecasting challenges [2, 3, 6, 19, 31]. However, there are many other performance metrics that investigate features, such as the coverage rate of a $(1-\alpha)*100\%$ prediction interval and the weighted interval score (WIS). The coverage rate of a $(1-\alpha)*100\%$ prediction interval for a forecast F is defined as the proportion of observed data that falls within the $(1-\alpha)*100\%$ prediction interval. The WIS is a proper score that provides quantiles of the predictive forecast distribution by weighting a set of Interval Scores (IS)[8]; that is, the WIS has a minimal cost when true forecasts are reported [20]. We define the IS of a $(1-\alpha)*100\%$ prediction interval as:

$$\text{IS}_\alpha(F,y) = (u-l) + \frac{2}{\alpha}*(l-y)*\mathbf{I}(y<l) + \frac{2}{\alpha}*(y-u)*\mathbf{I}(y>u),$$

where $\mathbf{I}$ is an indicator function, l and u represent the $\frac{2}{\alpha}$ and $1-\frac{2}{\alpha}$ quantiles of the forecast

F, and y is the observed data; this metric describes the proportion of observations captured within the $(1-\alpha)*100\%$ prediction interval [3, 20]. The WIS aims to describe the entire predictive distribution by considering many central predictive intervals at confidence levels $(1-\alpha_1) < (1-\alpha_2) < ... < (1-\alpha_K)$, and is defined as:

$$\text{WIS}_{\alpha_{0:K}}(F,y) = \frac{1}{K+\frac{1}{2}}\Big(\frac{1}{2}|y-m| + \sum_{k=1}^{K} w_k \text{IS}_{\alpha_k}(F,y)\Big),$$

with weights $w_k = \frac{\alpha_k}{2}$ for $k = 1, 2, ..., K$, and predictive mean m. So, the WIS is best interpreted as a measure of how close the entire predictive distribution is to the observations [3, 10].

3.4 Data

This study uses a time series of daily Zika disease incidence reported by the Secretary of Health of Antioquia, Colombia [5]. At the time of data collection, Antioquia, Colombia, had an estimated population of 6.3 million. The disease incidence reporting for this data set occurred between December 28, 2015, and April 10, 2016; Zika had been reported to be circulating throughout 150 municipalities of Colombia, with the epidemic reaching a total case count of 75,187 by April 23, 2016 [5]. Zika incidence reporting was based on the onset of symptoms. The data set is visualized in Figure 3.1:

Figure 3.1 represents the time series of Zika incidence from the 2015-2016 Zika epidemic in Antioquia, Colombia. The x-axis represents the time from the start of the epidemic to the end of data reporting, and the black dots represent the number of newly reported Zika cases in a day. The peak in disease incidence occurs 36 days into the epidemic, with secondary spikes in incidence observed at the beginning and end of March.

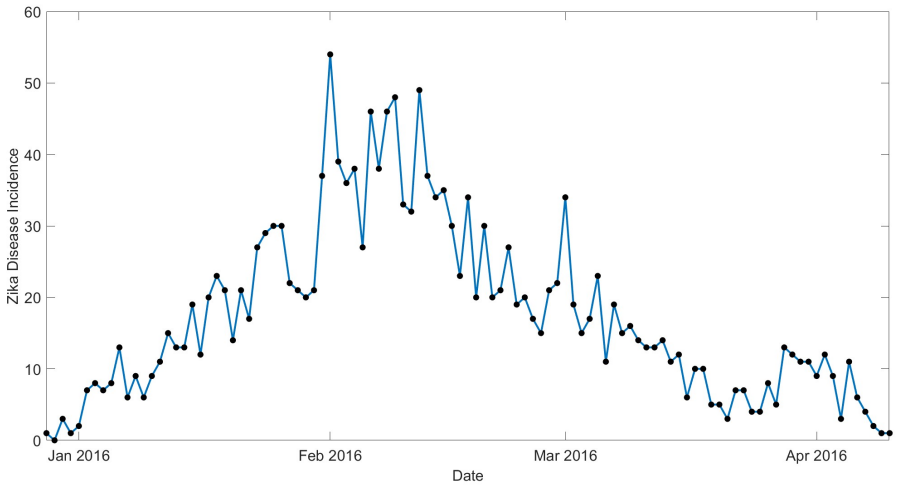

Figure 3.1: Zika disease incidence data reported daily by the Secretary of Health of Antioquia, Colombia. Surveillance data reporting occurred between December 28, 2015, and April 10, 2016, and is based on the onset of disease symptoms.

Chapter 4

Results

Chapter 4 focuses on the simulation and validation of simple epidemic model forecasts on the 2015-2016 Zika epidemic. The Zika incidence data 3.1 is used to fit model parameters for the GGM, GLM, GRM, SIR, and SEIR models described in Section 3.3. All data fitting routines are performed with QuantDiffForecast MATLAB toolbox [8]. In addition, the QuantDiffForecast toolbox is utilized for all forecasting under the Parametric Bootstrapping framework, with some modifications to the source script for figure outputs. For the Ensemble Kalman Filter analysis step, inspiration came from Mitchell and Arnold's original hard-coded script, with significant modification to allow for systems of higher order, forecast solution storage, and visualization of results [27]. The MATLAB scripts for the Ensemble Kalman Filtering are provided in A.2.2. We then simulate and visualize 14-day ahead forecasts for each model across five different case studies described in Section 4.1.

4.1 Simulations

To compare which forecasting model and assimilation method performs best on the Antioquia Zika epidemic data set, a series of 5 case studies are conducted. Case Study 1 trains all models on the first 35 days of data, fitting model parameters and quantifying uncertainty, and then forecasts Zika disease incidence 14 days ahead to compute performance against the observed data. Case Study 2 assimilates the data Case Study 1 predicted against, refitting

each system and forecasts 14 days ahead - that is, Case Study 2 assimilates the first 49 days of Zika incidence and then forecasts up to the 63rd day of the epidemic. Each successive case study follows the pattern of refitting on two more weeks of data and forecasting two weeks ahead until Cast Study 5, which is trained on 91 days of data and then forecasts to the 105th day of the epidemic. The series of case studies are designed to understand how the amount of available data influences forecasting performance and determine if Parametric bootstrapping or the Ensemble Kalman Filter is more effective in utilizing quantified uncertainty to produce better forecasts. In addition, the data cutoff points of each case study can assist with questions like: how resilient is each model for forecasting in the presence of spikes in observed incidence towards the end of an epidemic? Spikes in observed data can occur for many reasons, like data dumps at public health institutions, correction in reported data, or periods of heightened contact due to holidays.

All model-based forecasting under the Parametric Bootstrapping technique is simulated using the QuantDiffForecast toolbox [8]. To use the QuantDiffForecast toolbox, we first specify the data file, parameter estimation technique, error distribution, number of bootstrap realizations, and the model of interest within an options file. For the purposes of this study, we select the default option of non-linear least squares fitting, a Poisson error structure, 300 bootstrap realizations, and utilize MultiStart with ten initial guesses. Parameter search bounds and initial conditions for each model are listed in A.1 and A.2. The options files used for simulating each model are provided in A.2. To generate the 14-day-ahead forecasts, each options file is called in the **Run_Forecasting_ODEModel** function with a training period specified by the case study and a forecasting duration of 14 days. Each forecast is then visualized using the **plotForecast_ODEModel** function by calling the results from the **Run_Forecasting_ODEModel** function. Modifications were made to the source code of the **plotForecast_ODEModel** to prevent figure titles from overlapping.

We utilized the standard Ensemble Kalman Filter presented in Section 3.2, where X represents the true cumulative number of Zika cases, Y is the unobserved Zika incidence, F is the dynamical operator represented by the SIR or SEIR models described in Section 3.3, and G is a function recovering incidence from cumulative incidence. The optimal parameter set obtained through fitting the SIR and SEIR models using the **Run_Forecasting_ODEModel** function with non-linear least squares are used for all ensemble members of the SIR and SEIR models run under the Ensemble Kalman filter. Tables A.3 and A.4 contain parameter search bounds for fitting the SIR and SEIR models. The parameter bounds and initial values are selected to be constant across all five case studies to maintain consistency.

4.1.1 Case Study 1: Day 35

Case study 1 seeks to assess early forecasting performance and determine which model performs best in a scenario where the peak in Zika incidence has yet to be observed. So, the experiment is interpreted as early forecasting attempts of a newly developed epidemic with limited information. Each model is fit using the first 35 days of Zika incidence data and then forecasts 14 days ahead. The ensemble mean of each model is then used to compute forecast performance metrics contained in Table 4.1. First, consider the 14-day-ahead forecast of the GGM:

Figure 4.1 represents the 14-day forecast of the GGM under the parametric bootstrapping framework. The histograms on top depict sampling distributions of model parameters from the 300 bootstrap realizations. The bottom plot shows the calibration and forecasting periods split by the vertical line at day 34. The blue dots represent the Zika disease incidence, and the red solid line is the mean model fit of the ensemble. The grey lines surrounding the red line represent the bootstrap fits of the GGM model. The cyan lines represent the forecasting

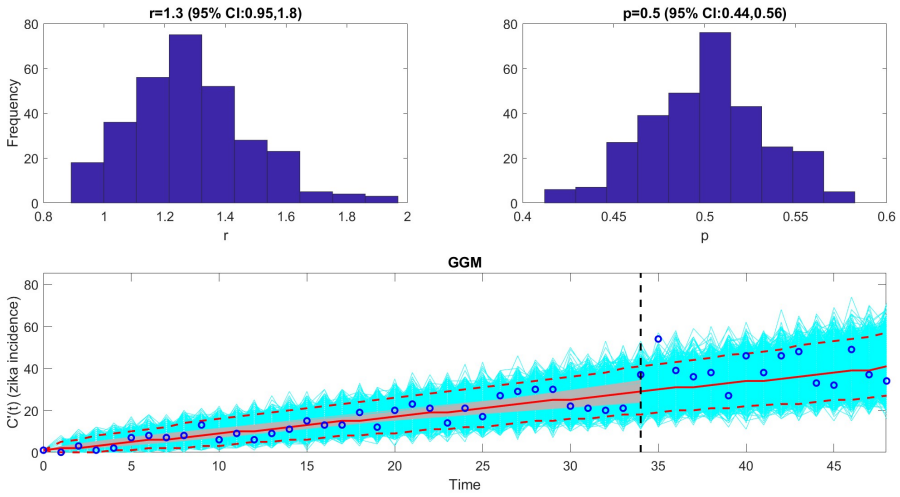

Figure 4.1: 14-day-ahead forecast of GGM trained on 35 days of Zika epidemic under the parametric bootstrapping method. The histograms on top depict sampling distributions of model parameters from the 300 bootstrap realizations. The bottom plot shows the calibration and forecasting periods split by the vertical line at day 34. The blue dots represent the Zika disease incidence, and the red solid line is the mean model fit of the ensemble. The grey lines surrounding the red line represent the bootstrap fits of the GGM model. The cyan lines represent the forecasting uncertainty surrounding the model and constitute a 95% prediction interval, represented as the red dashed lines. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

uncertainty surrounding the model and constitute a 95% prediction interval, represented as the red dashed lines.

The GGM performs well in modeling the early growth dynamics of the Zika epidemic, with the resulting 14-day forecast capturing the majority of observations within the 95% prediction internal. The GGM's forecast also results in the lowest error metrics of Case Study 1. Next, consider the 14-day-ahead forecast of the GLM:

Figure 4.2 represents the 14-day forecast of the GLM under the parametric bootstrapping framework. The GLM captures the early growth dynamics well during the training period, but the forecast underestimates Zika incidence, resulting in higher errors. However, the GLM should outperform the GGM when more data is used for future forecasts due to the

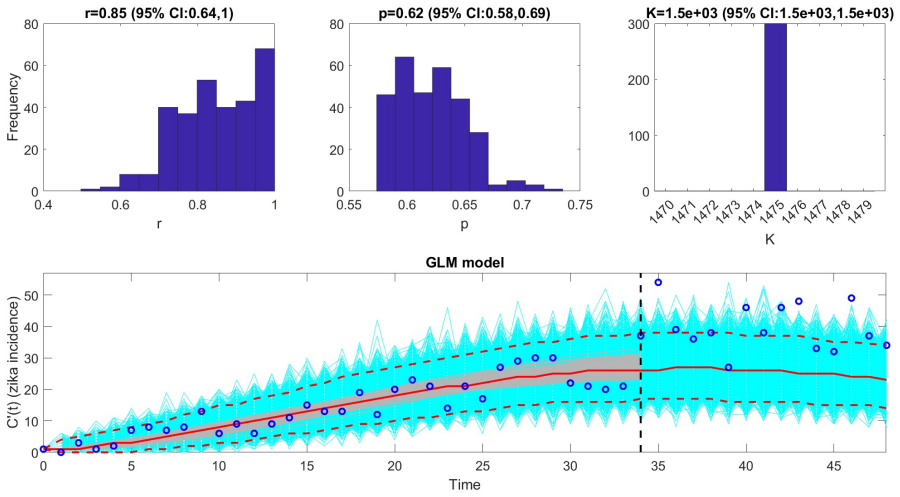

Figure 4.2: 14-day-ahead forecast of GLM Trained on 35 days of Zika epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

extra parameter K. Now, consider the 14-day-ahead forecast of the GRM:

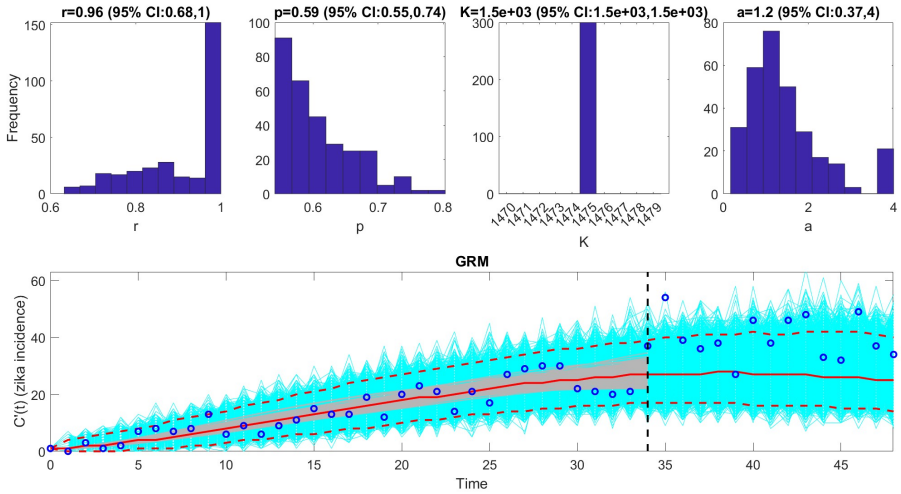

Figure 4.3: 14-day-ahead forecast of GRM trained on 35 days of Zika epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

Figure 4.3 represents the 14-day forecast of the GRM under the parametric bootstrapping framework. Similar to the GGM and GLM models, the GRM captures early growth in Zika

incidence well, but the resulting forecast from the model underestimates incidence. The parameter sampling distributions are skewed and also exhibit high variance, suggesting that parameters r and p are not practically identifiable within this calibration period [2]. Next, consider our first forecast of the SIR model:

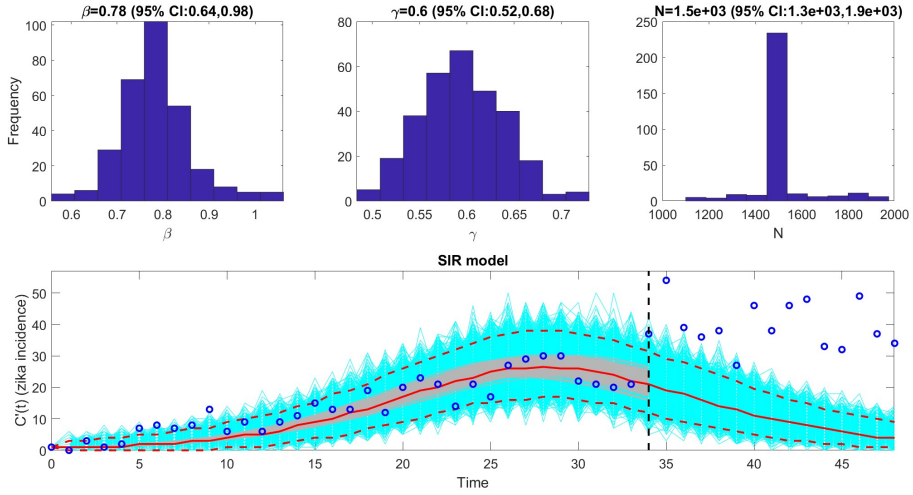

Figure 4.4: 14-day-ahead forecast of SIR model trained on 35 days of Zika epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

Figure 4.4 depicts the 14-day forecast of the SIR model fit using the parametric bootstrapping technique. The SIR model captures the early growth in Zika incidence. However, the 14-day forecast significantly underestimates incidence, the 95% prediction interval covers 0% of the observed data, and results in some of the highest reported errors out of the seven models tested. Given that the SIR model did not perform well in this first case study, we should also expect the SEIR model to perform similarly. To validate this, consider the 14-day forecast of the SEIR model:

Figure 4.5 describes the output from the 14-day forecast of the SEIR model under the parametric bootstrap. Like the SIR, the SEIR model fits well in the calibration period. Yet, the forecast results in the highest errors out of all seven tested models. The parameter

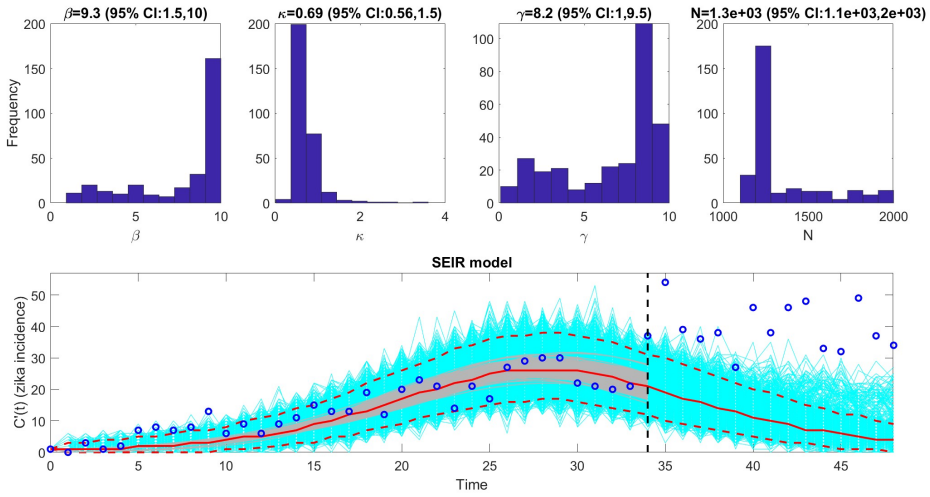

Figure 4.5: 14-day-ahead forecast of SEIR model trained on 35 days of Zika epidemic under the Parametric Bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

distributions suggest that β and γ are not practically identifiable in this case study. There are other measures that can be implemented to obtain a definite answer to the question of the practical identifiability of model parameters, like the Monte Carlo approach [33, 34], but this will be reserved for future work. As we have seen, the SIR and SEIR models do not produce high-quality 14-day forecasts in an early epidemic scenario. However, we will now consider these models under an Ensemble Kalman Filter to see if performance improves. Consider the 14-day ahead forecast of the SIR model:

Figure 4.6 represents the 14-day forecast of the SIR model under the EnKF framework. The figure is structured in the same way as the figures from the parametric bootstrap but with key differences in the interpretation of the 95% prediction intervals. Initial attempts to estimate covariance from the state variables resulted in numerical errors throughout the simulation. A remedy to this issue is assuming a fixed variance of four for the state noise and fixing observational noise as a white noise process for all case studies. This is a limitation of the study, but future work would look into variance inflation techniques to avoid ensemble

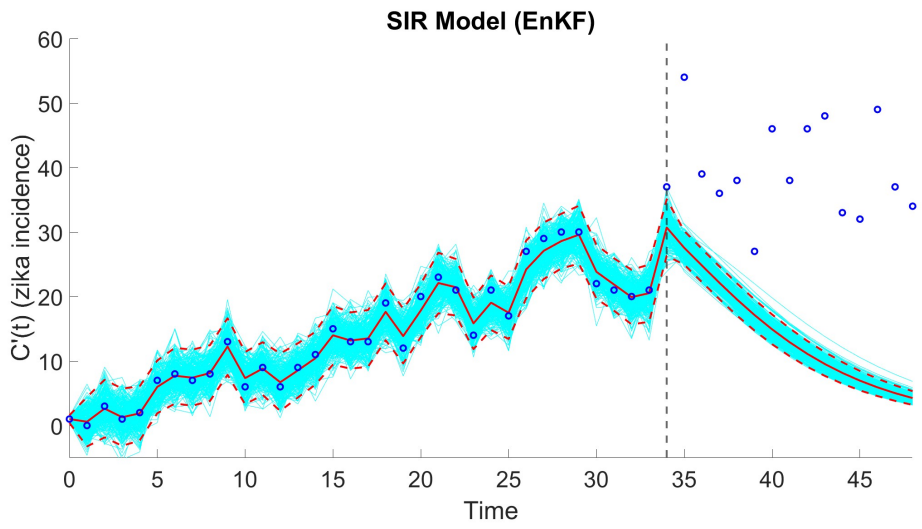

Figure 4.6: 14-day-ahead forecast of SIR model trained on 35 days of Zika epidemic under the Ensemble Kalman Filter. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

convergence. Due to sources of noise being fixed, the 95% prediction intervals are computed only for the visual effect, so the coverage and WIS metrics will not be computed. The 14-day forecast of the SIR under the EnKF underestimates the observed Zika incidence, but the EnKF yields lower errors compared to the SIR under the parametric bootstrap. Finally, consider the SEIR model under the EnKF:

Figure 4.7 depicts the 14-day forecast of the SEIR model calibrated on the EnKF. Similar to the SIR model, a fixed noise process is used for all case studies to avoid rapid convergence of ensemble members. The forecast underestimates disease incidence but also results in lower errors compared to the SEIR model under a parametric bootstrapping approach. This is the final forecast of Case Study 1, so now we will discuss the performance of each model in the following table:

Table 4.1 summarizes the performance of each model tested during Case Study 1. The GGM yields the best forecast of Zika incidence, resulting in the lowest MAE, MSE, RMSE, and WIS; forecasts with lower prediction errors are more accurate. The GGM also resulted in the

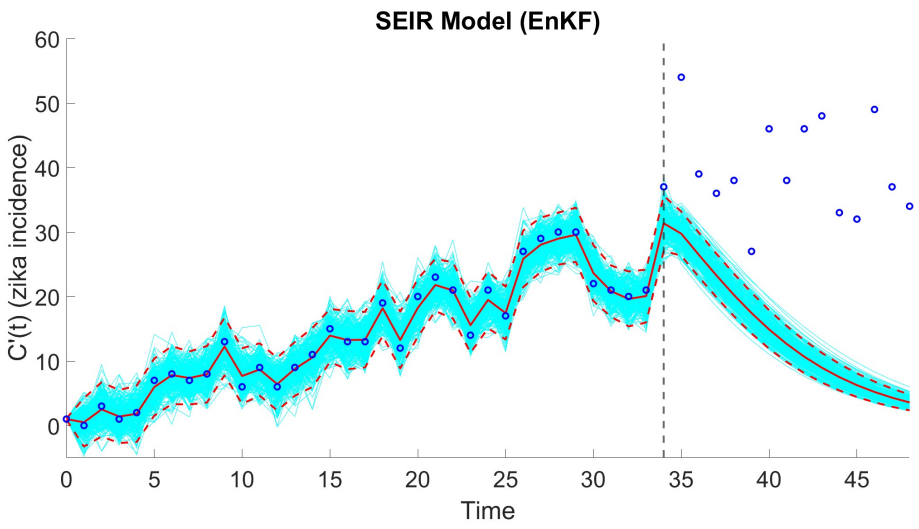

Figure 4.7: 14-day-ahead forecast of SEIR model trained on 35 days of Zika epidemic under the Ensemble Kalman Filter. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

Forecasting Performance Metrics					
Model	MAE	MSE	RMSE	Coverage	WIS
GGM	8.305	97.596	9.879	92.857	5.190
GLM	14.301	257.892	16.059	42.857	10.613
GRM	13.357	230.465	15.181	64.286	8.988
SIR	29.353	925.239	30.418	0	26.604
SEIR	29.716	947.150	30.776	0	26.523
SIR(EnKF)	26.218	773.657	27.815	*	*
SEIR(EnKF)	27.472	850.308	29.160	*	*

Table 4.1: Performance metrics of model-based forecasts from Case Study 1

highest coverage, as the GGM's 95% prediction interval captured 92.857% of the observed incidence data. The GLM and GRM performed similarly, where these models captured early growth dynamics well but resulted in larger prediction errors, reduced coverage, and a higher WIS when compared to the GGM. The mechanistic models systematically underestimated Zika incidence, resulting in the highest errors observed for Case Study 1. However, the Ensemble Kalman Filter's state estimation resulted in a reduction of prediction error over the Parametric Bootstrap approach for the SIR and SEIR models.

4.1.2 Case Study 2: Day 49

Case study 2 seeks to assess how model forecasts perform after a peak in incidence has been observed and the epidemic has begun to taper off. Each model is fit using the first 49 days of Zika incidence data and then forecasts 14 days ahead to day 63. Similarly to Case Study 1, the ensemble mean of each model is then used to compute forecast performance metrics reported in Table 4.2. Consider the 14-day-ahead forecast of the GGM:

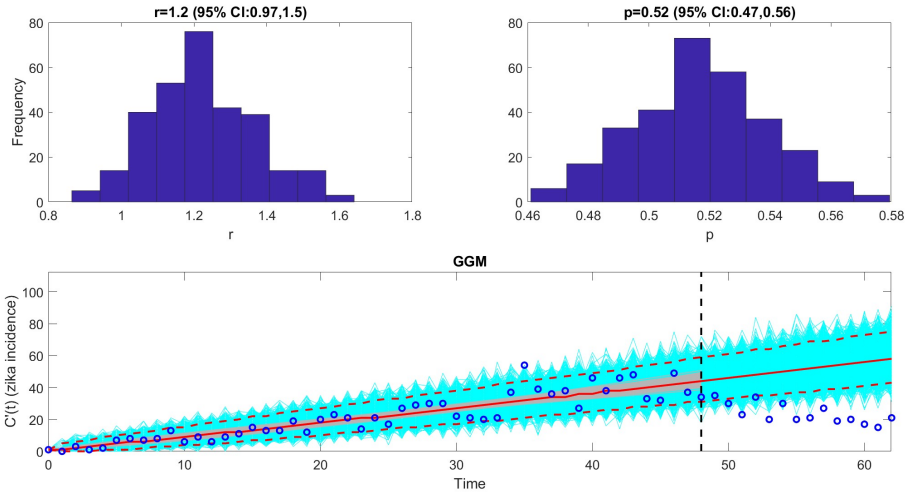

Figure 4.8: 14-day-ahead forecast of GGM Trained on 49 days of Zika epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

Figure 4.8 demonstrates how model-based forecasting can go wrong. Since the GGM does not have a carrying capacity parameter, the model will continue to predict increases in incidence - even after the epidemic begins to diminish. As a result, the 95% prediction interval only captures one of the observed values, and the errors in prediction become significantly higher when compared to the GGM under Case Study 1. Now, we will consider the 14-day forecast of the GLM:

Figure 4.9 shows the GLM performing significantly better when compared to the GLM from

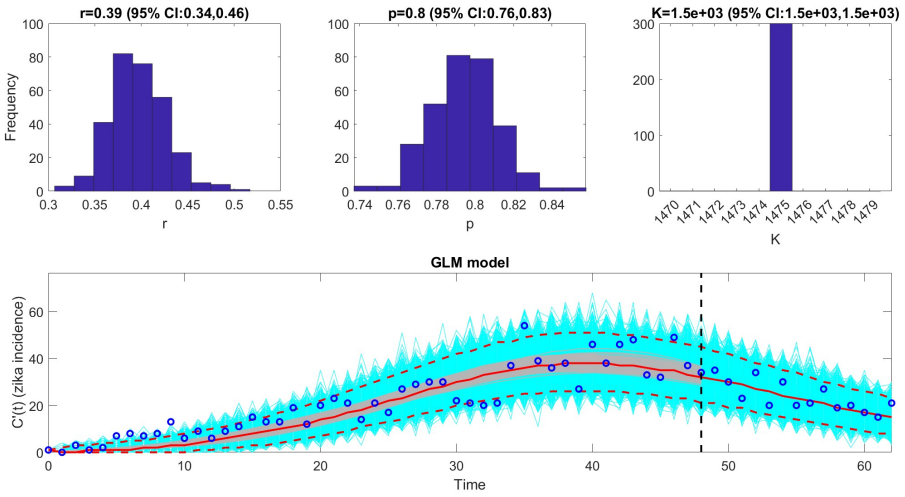

Figure 4.9: 14-day-ahead forecast of GLM trained on 49 days of Zika epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

Case Study 1. The ensemble mean fits incredibly well to the observed data, resulting in the lowest prediction errors in Case Study 2. The 95% prediction interval also covers 100% of the observed data.

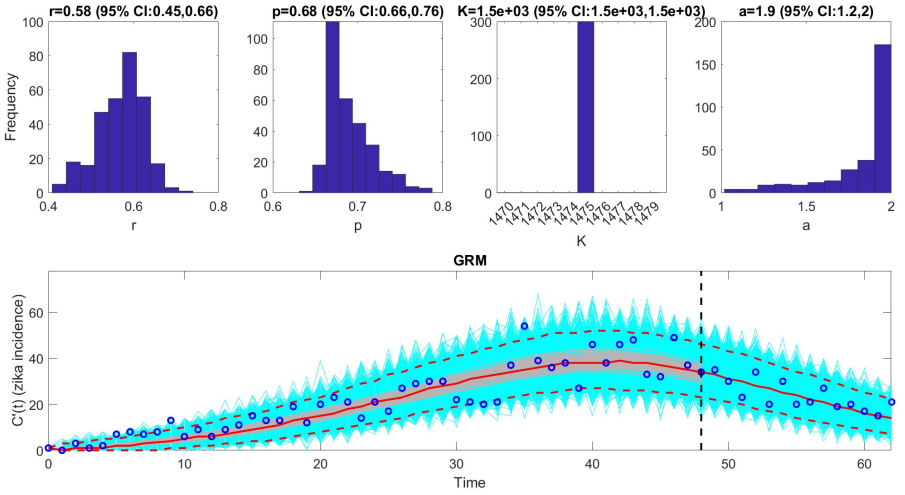

Figure 4.10: 14-day-ahead forecast of GRM trained on 49 days of Zika epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

Figure 4.10 demonstrates noticeable increases in forecasting performance for the GRM when compared to Case Study 1. Here, the GRM fits well to early growth in Zika incidence, and the resulting 14-day forecast yields 100% coverage with errors comparable to the GLM. The histograms of r and p are closer-knit, suggesting that these parameters are now practically identifiable at this point within the time series. In summary, the GRM has shown improvement with the inclusion of data past a first observed peak. Now, we will consider the next 14-day forecast of the SIR model:

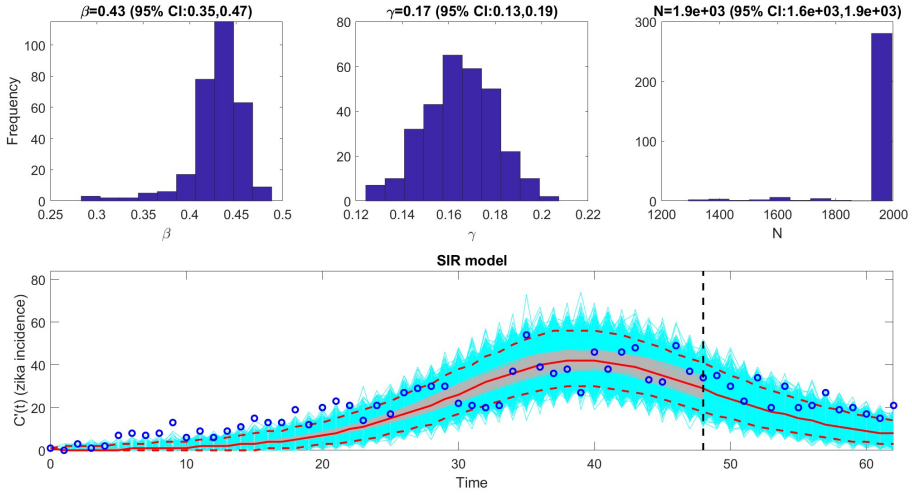

Figure 4.11: 14-day-ahead forecast of SIR model trained on 49 days of Zika epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

Figure 4.11 depicts the 14-day forecast of the SIR model under the parametric bootstrapping method. The SIR model shows a marked increase in forecasting performance over Case Study 1, with the ensemble mean falling closer to observed data and the 95% prediction interval capturing observed incidence. Next, consider the 14-day forecast of the SEIR model:

Figure 4.12 represents the 14-day forecast of the SEIR model under the parametric bootstrapping method. Like the SIR model, the SEIR shows improvements in the fit and forecasting performance compared to Case Study 1; in fact, the SEIR model previously had the highest

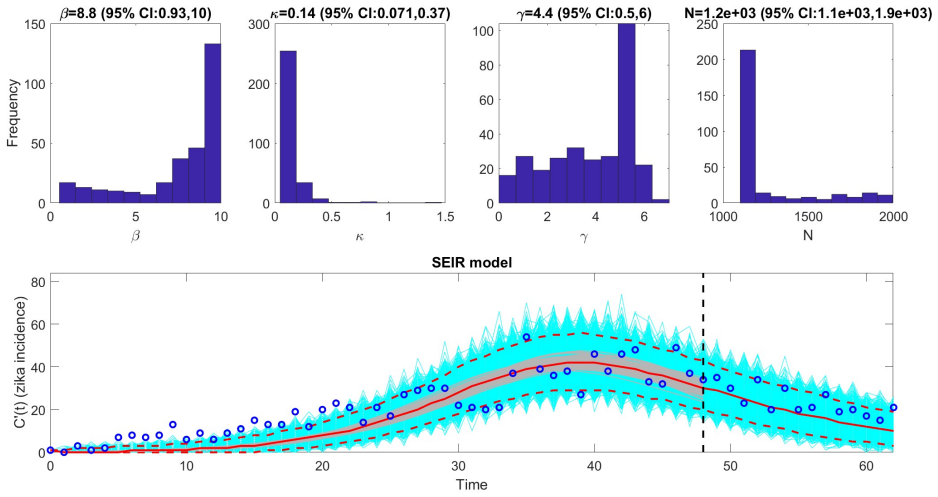

Figure 4.12: 14-day-ahead forecast of SEIR model trained on 49 days of Zika epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

reported errors in prediction and the highest WIS out of all models tested. A key observation is the SEIR results in better forecasting performance over the SIR as the exposed class induces a delay in Zika incidence, leading the model to project heightened incidence. Now, we will see if simulating the SIR model under the EnKF results in higher-quality forecasts when calibrated up to day 49:

Figure 4.13 shows the 14-day forecast of the SIR model calibrated using the EnKF. The ensemble mean underestimates Zika incidence, but the state estimation from the EnKF does yield a decrease in prediction errors over the SIR using the parametric bootstrap. Next, we will consider the 14-day forecast of the SEIR model simulated under the EnKF:

Figure 4.14 depicts the 14-day forecast of the SEIR model calibrated on the EnKF. Compared to Figure 4.13, the SEIR model projects higher Zika incidence over the forecasting period. However, the SEIR model under a parametric bootstrapping routine predicts disease incidence marginally better than the EnKF; the observed difference in prediction errors could be attributed to stochasticity in the EnKF. This concludes the final forecast of Case

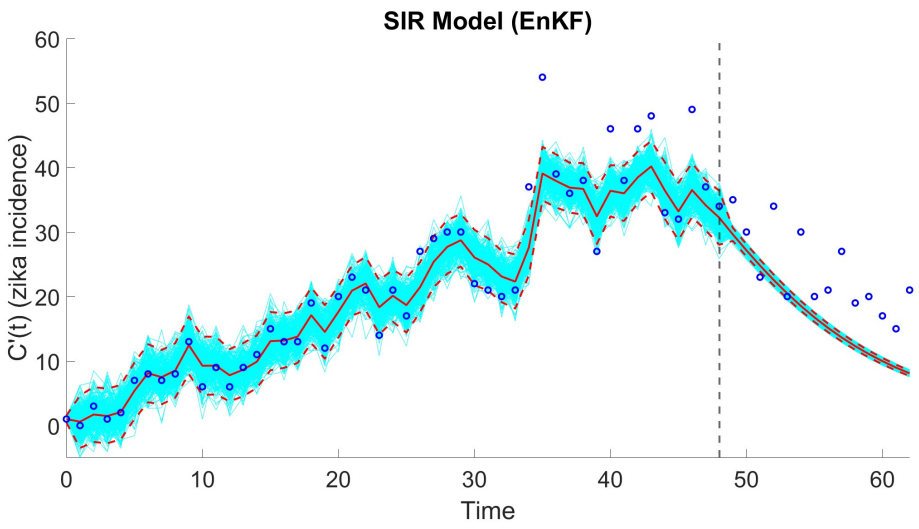

Figure 4.13: 14-day-ahead forecast of SIR model trained on 49 days of Zika epidemic under the Ensemble Kalman Filter. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

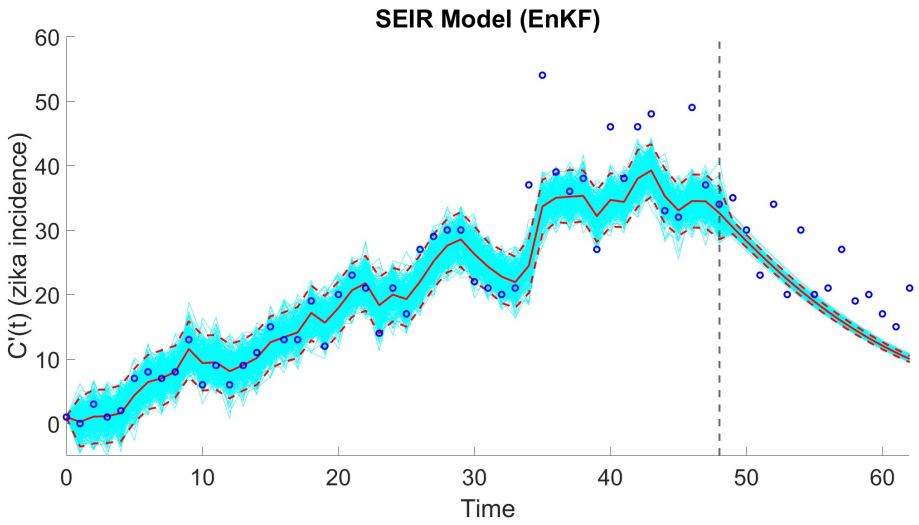

Figure 4.14: 14-day-ahead forecast of SEIR model trained on 49 days of Zika epidemic under the Ensemble Kalman Filter. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

Study 2, so we will now summarize out findings in the following table:

Table 4.2 summarizes the performance of all models tested throughout Case Study 2. The GLM produced the best forecast of Zika incidence, with the lowest prediction error, WIS,

Forecasting Performance Metrics					
Model	MAE	MSE	RMSE	Coverage	WIS
GGM	27.997	871.994	29.530	7.143	22.442
GLM	3.380	17.858	4.226	100	2.180
GRM	3.524	19.240	4.386	100	2.248
SIR	8.182	84.585	9.197	57.143	5.815
SEIR	5.559	41.893	6.472	85.714	3.588
SIR(EnKF)	6.956	64.424	8.027	*	*
SEIR(EnKF)	5.585	42.138	6.491	*	*

Table 4.2: Performance metrics of models-based forecasts from Case Study 2

and achieved 100% coverage of observed data. The GRM is a close contender for best performance, achieving 100% coverage but trails behind in all other measures. In addition, Figure 4.10 suggests that GRM model parameters are now practically identifiable. The SIR and SEIR models also demonstrate significant improvement in forecasting Zika incidence, with a reduction in MSE by over an order of magnitude when compared to Case Study 1. Finally, applying the GGM for Case Study 2 resulted in the worst-performing forecast, as the GGM is best applied to model early growth behavior for an epidemic.

4.1.3 Case Study 3: Day 63

Case study 3 seeks to assess how resilient model-based forecasts are in the presence of a secondary peak in incidence during the forecasting period. Each model is fit using the first 63 days of Zika incidence data and then forecasts 14 days ahead to day 77. Similarly to the previous case studies, the ensemble mean of each model is then used to compute forecast performance metrics reported in Table 4.3. Consider the 14-day-ahead forecast of the GGM:

Figure 4.15 represents the 14-day forecast of the GGM calibrated on the first 63 days of the Zika epidemic. The GGM accurately predicts the secondary peak in incidence observed on day 65 of the epidemic, but the remainder of the forecast overestimates incidence. Next,

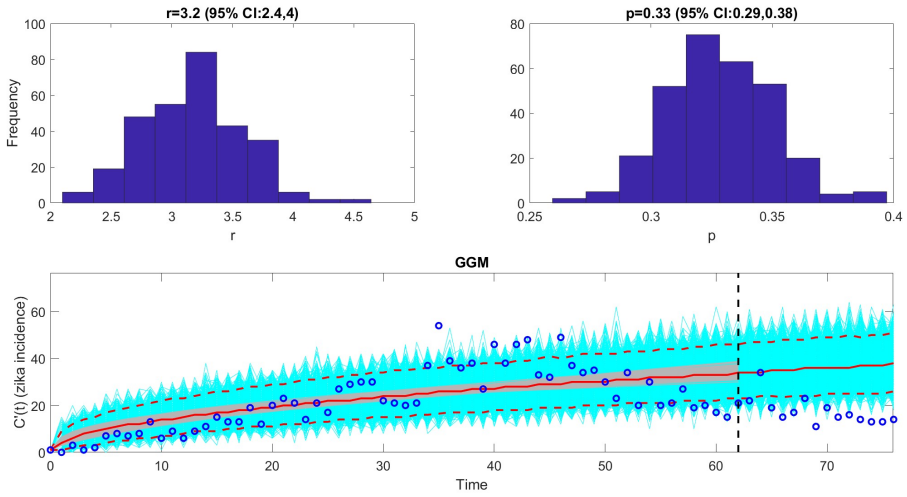

Figure 4.15: 14-day-ahead forecast of GGM trained on 63 days of Zika epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

consider the 14-day forecast of the GLM:

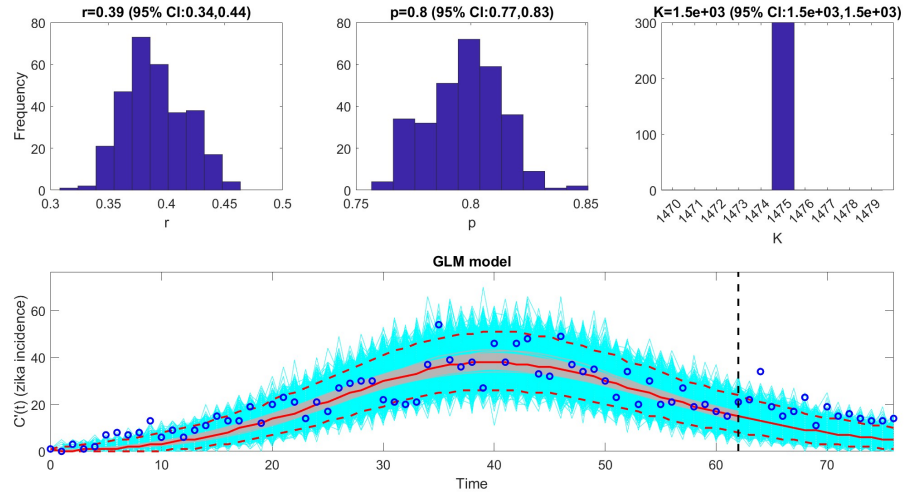

Figure 4.16: 14-day-ahead forecast of GLM trained on 63 days of Zika epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

Figure 4.16 represents the 14-day forecast of the GLM calibrated on the first 63 days of the Zika epidemic. The GLM maintains a great fit during the calibration period, but the resulting

forecast underestimates the observed incidence. The 95% prediction interval captures 35.71% of the observations.

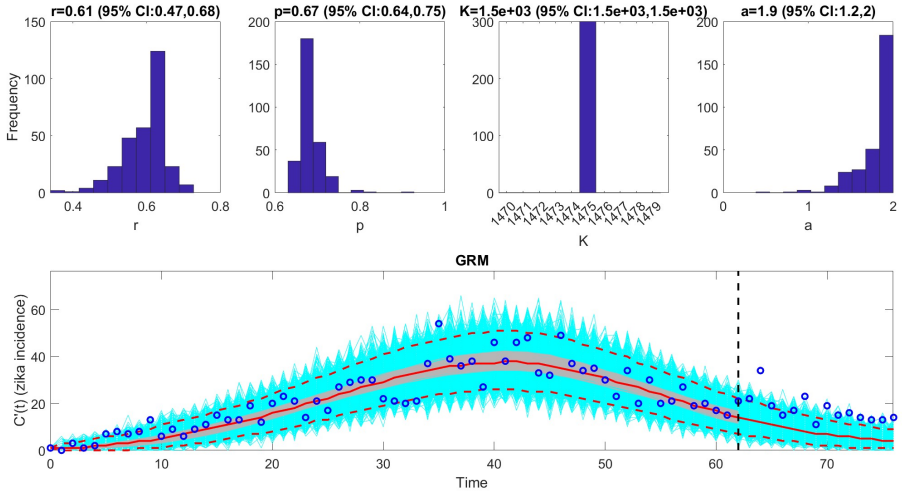

Figure 4.17: 14-day-ahead forecast of GRM trained on 63 days of Zika epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

Figure 4.17 depicts the 14-day forecast of the GRM calibrated using the parametric bootstrap. Like the GLM forecast from Figure 4.16, the GRM underestimates the observed Zika incidence. However, the GRM results in higher levels of prediction error and lower coverage.

Figure 4.18 represents the 14-day forecast of the SIR model calibrated using the parametric bootstrap. The SIR exhibits lower quality forecasts in Case Study 3, with the 95% prediction interval covering 0% of the observed incidence data. Now, we will compare how the SEIR model forecast performs against the SIR model in the presence of noisy observed data:

Figure 4.19 depicts the 14-day forecast of the SEIR model calibrated on the parametric bootstrap. The resulting forecast underestimates the observed incidence but yields lower prediction errors compared to the SIR model. In addition, the 95% prediction interval captures 64.28% of the observations. The histogram for β suggests that this parameter is not practically identifiable 63 days into the epidemic. However, investigating parameter

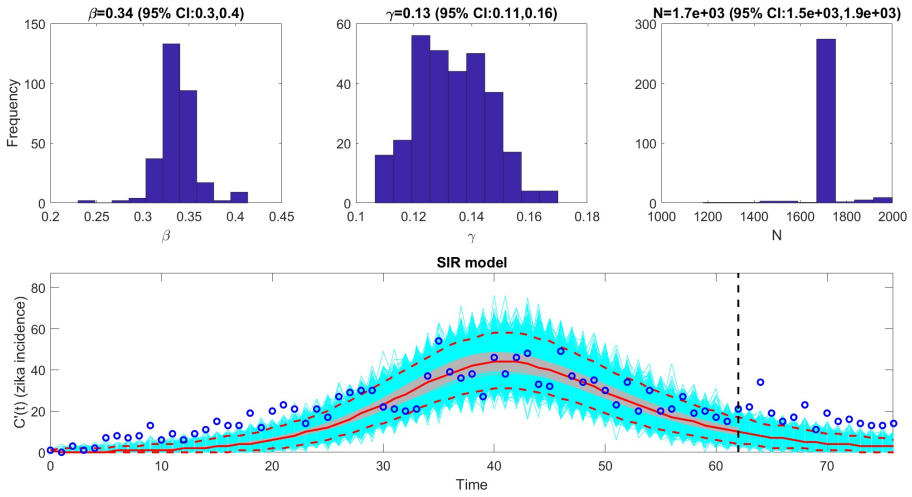

Figure 4.18: 14-day-ahead forecast of SIR model trained on 63 days of Zika epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

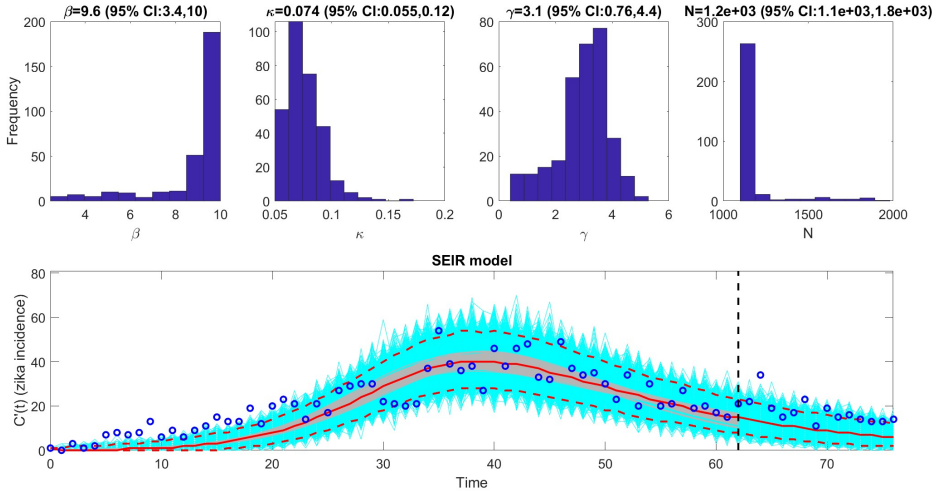

Figure 4.19: 14-day-ahead forecast of SEIR model trained on 63 days of Zika epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

distributions for sporadic behavior is only a diagnostic, so other techniques will need to be applied in the future to verify if β is practically identifiable or not.

Figure 4.20 represents the 14-day forecast of the SIR model calibrated using the EnKF.

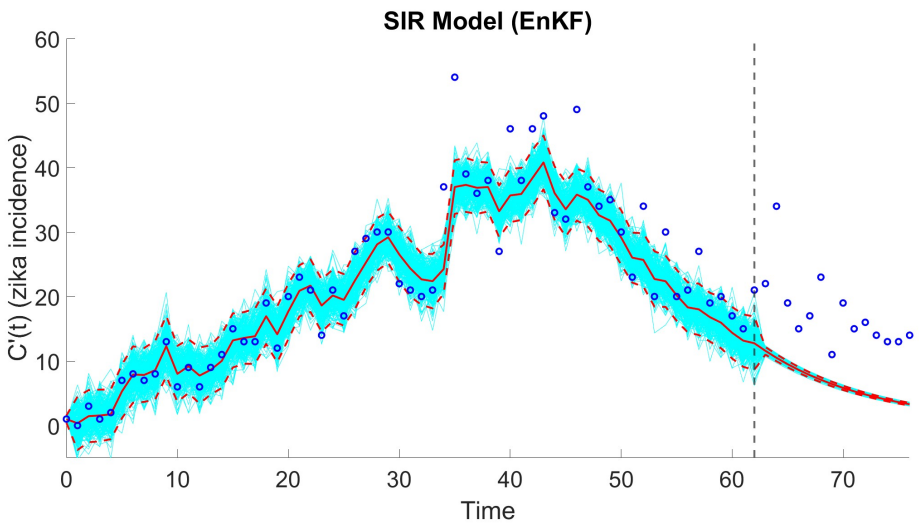

Figure 4.20: 14-day-ahead Forecast of SIR Model Trained on 63 days of Zika epidemic under the Ensemble Kalman Filter. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

Similar to the SIR under the parametric bootstrapping routine, the 14-day forecast mean underestimates the Zika disease incidence observed. However, the EnKF does provide a reduction in forecasting error; the SIR forecast from Figure 4.18 resulted in an MSE of 170.332, while the SIR under the EnKF reports an MSE of 137.063. Finally, we consider the SEIR model under the EnKF:

Figure 4.21 represents the 14-day forecast of the SEIR model calibrated under the EnKF framework. Compared to the forecast of the SIR under the EnKF, the SEIR predicts higher disease incidence over the prediction window and results in lower errors over the SIR and SEIR under bootstrapping.

Table 4.3 summarizes the results of all forecasts generated through Case Study 3. The SEIR model is the top performer across Case Study 3, resulting in the lowest prediction error, highest coverage of observed data, and lowest WIS. However, the SIR model is not resilient in predicting spikes in incidence later into an epidemic; modeling for disease latency within the mechanistic framework does yield improvements in forecast performance. The

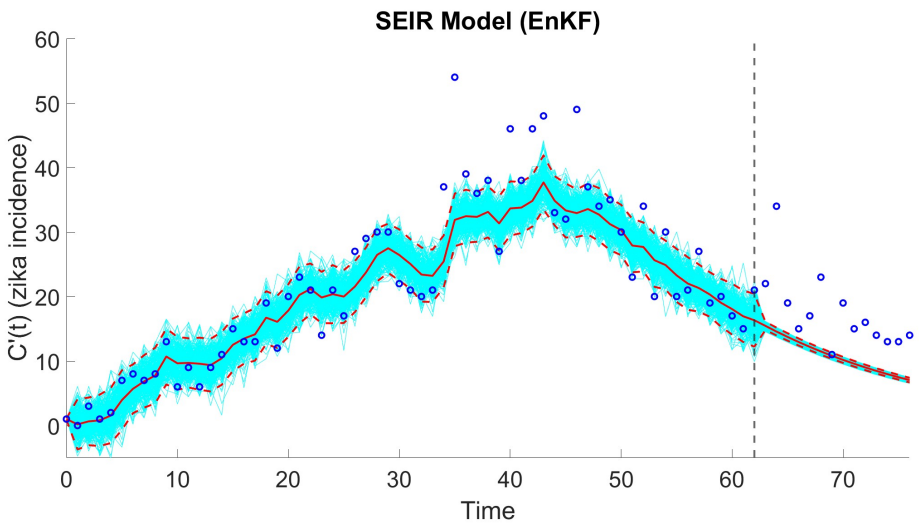

Figure 4.21: 14-day-ahead forecast of SEIR model trained on 63 days of Zika epidemic under the Ensemble Kalman Filter. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

Forecasting Performance Metrics					
Model	MAE	MSE	RMSE	Coverage	WIS
GGM	18.499	383.055	19.572	7.143	14.341
GLM	8.576	92.035	9.593	35.714	6.471
GRM	9.764	113.673	10.662	14.286	7.609
SIR	12.231	170.332	13.051	0	10.487
SEIR	7.813	80.574	8.976	64.286	5.757
SIR(EnKF)	10.873	137.063	11.707	*	*
SEIR(EnKF)	6.721	64.227	8.014	*	*

Table 4.3: Performance metrics of model-based forecasts from Case Study 3

GLM is the second-best performing model under bootstrapping, beating the SIR and GRM in prediction errors and coverage. The GGM continues the trend of overestimating Zika incidence as the model is not fit for describing long-term disease dynamics.

4.1.4 Case Study 4: Day 77

Case study 4 seeks to assess the forecasting quality of each model late into the epidemic. Each model is fit using the first 77 days of Zika incidence data and then forecasts up to day 91. The ensemble mean of each model is then used to compute forecast performance metrics reported in Table 4.4. Consider the 14-day-ahead forecast of the GGM:

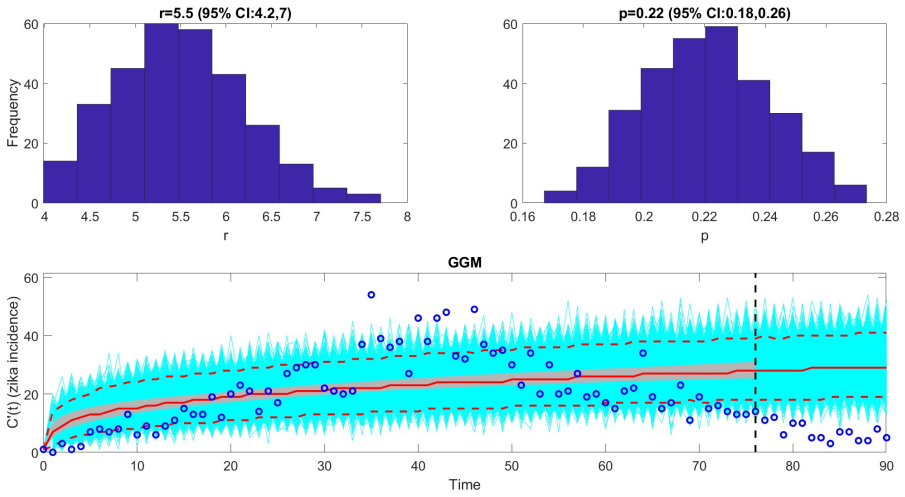

Figure 4.22: 14-day-ahead forecast of GGM trained on 77 days of Zika epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

Figure 4.22 represents the 14-day-ahead forecast of the GGM calibrated on the first 77 days of the Zika epidemic. The GGM fits straight through the training data, resulting in a forecast that systematically overestimates Zika incidence. The lack of any mechanism to modulate growth over time will always result in poor late-stage forecasting. However, there are interesting consequences of how bootstrapping propagates uncertainty of the GGM in Case Study 5. Next, consider the 14-day forecast of the GLM calibrated on the first 77 days of the Zika epidemic:

Figure 4.23 depicts the 14-day forecast of the GLM trained under the parametric bootstrap-

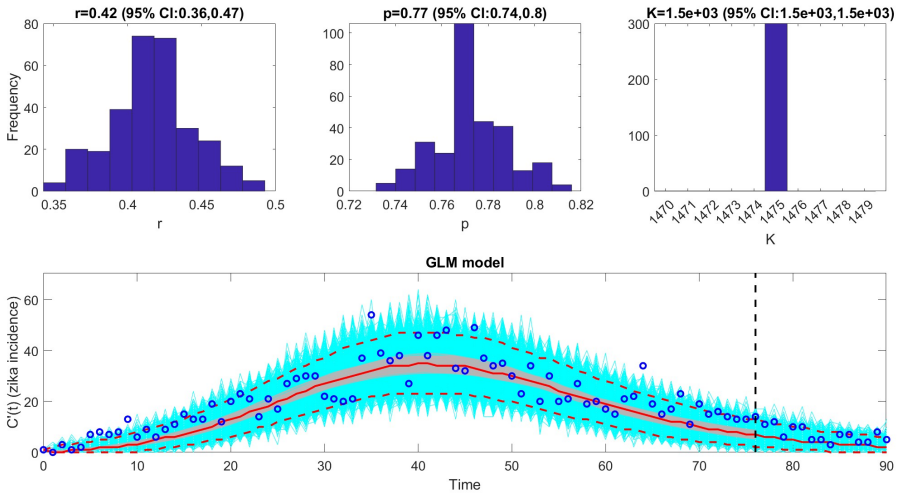

Figure 4.23: 14-day-ahead forecast of GLM trained on 77 days of Zika epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

ping method. The forecast fits well, resulting in low prediction error, and 85.71% of observed Zika incidence is covered by the 95% prediction interval.

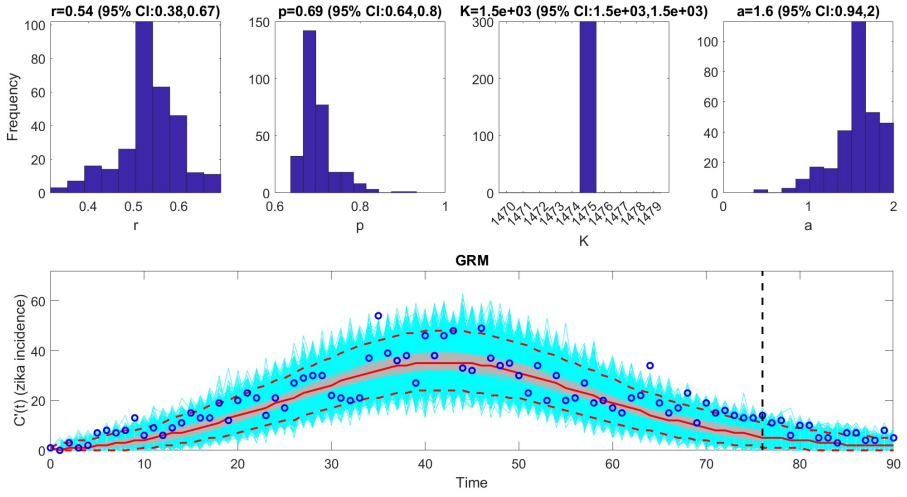

Figure 4.24: 14-day-ahead forecast of GRM trained on 77 days of Zika epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

Figure 4.24 represents the 14-day forecast of the GRM trained on the first 77 days of the Zika

epidemic. This forecast underestimates Zika incidence to a higher degree when compared to the GLM forecast from 4.23; prediction errors from the GRM are higher, and the resulting 95% prediction interval covers only 50% of the observed data.

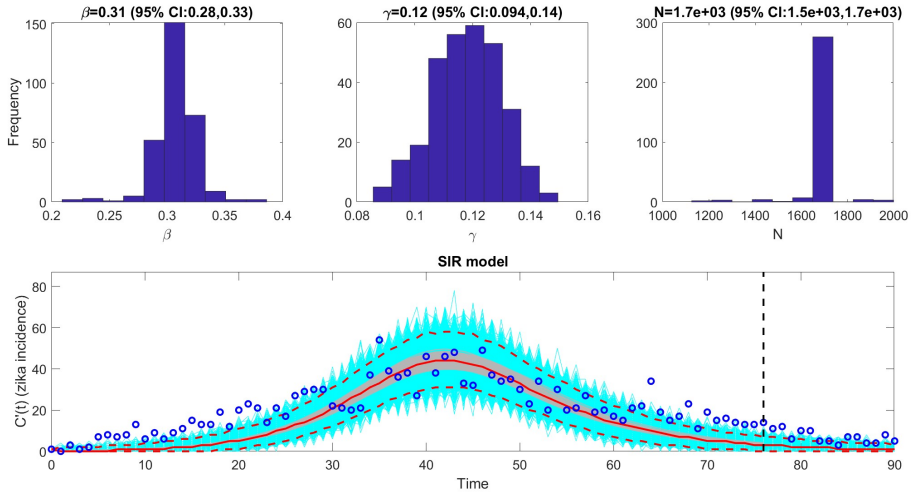

Figure 4.25: 14-day-ahead forecast of SIR model trained on 77 days of Zika Epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

Figure 4.25 depicts the 14-day forecast of the SIR model calibrated on the first 77 days of the Zika epidemic. From an initial glance, the SIR model fails to capture the growth dynamics around the early and later stages of the calibration period. As a result, the 14-day forecast underestimates Zika incidence and predicts that the epidemic is about to end, with the ensemble mean approaching 0. This behavior in fit and forecast suggests that the SIR model is not a great choice and is missing a key mechanism in predicting Zika incidence. The notion of the SIR model missing factors in Zika's epidemiology is supported as we consider the following 14-day forecast of the SEIR model:

Figure 4.26 represents the 14-day forecast of the SEIR model calibrated on the first 77 days of the Zika epidemic. This forecast performs incredibly well in predicting incidence, yielding the lowest MAE, MSE, RMSE, and WIS across all models fit using the parametric boot-

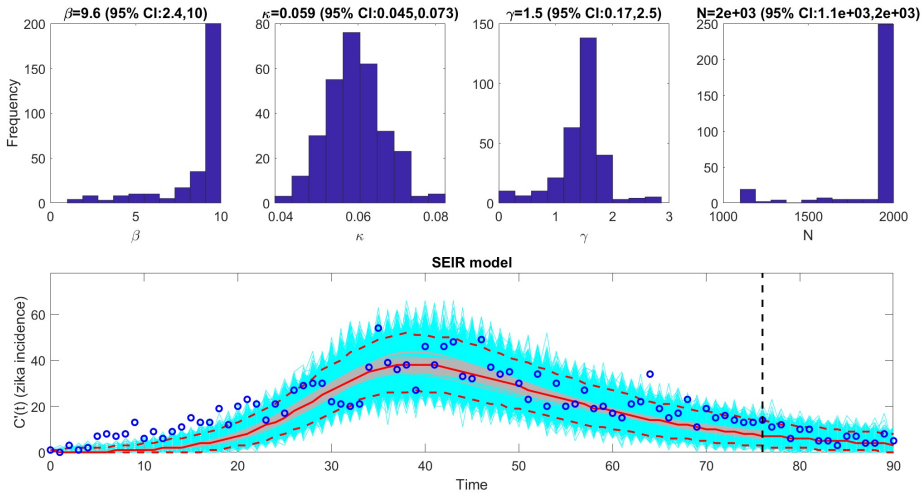

Figure 4.26: 14-day-ahead forecast of SEIR model trained on 77 days of Zika epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

strap. The 95% prediction interval covers 100% of observed Zika incidence - a pronounced increase over the 42.86% coverage from the SIR model. Drastic differences in the forecasting performance of the SIR and SEIR models support the inclusion of latency in Zika infections to produce accurate forecasts.

Figure 4.27 represents the 14-day forecast of the SIR model calibrated on the first 77 days of the Zika epidemic using the EnKF framework. The state estimation from the EnKF does correct the SIR model, resulting in lower prediction errors over the SIR model using a bootstrapping approach.

Figure 4.28 depicts the 14-day forecast of the SEIR model calibrated on the first 77 days of the Zika epidemic using the EnKF approach of state estimation. The resulting forecast yields lower prediction errors over the bootstrapping approach, with this technique projecting higher incidence. However, limitations in state error estimation do prevent coverage and WIS from being used as a point of comparison between methods.

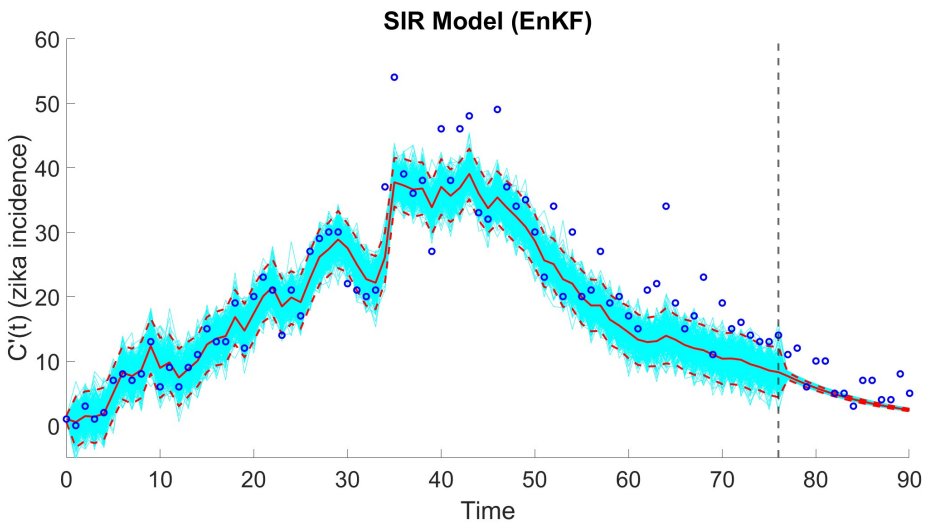

Figure 4.27: 14-day-ahead forecast of SIR model trained on 77 days of Zika epidemic under the Ensemble Kalman Filter. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

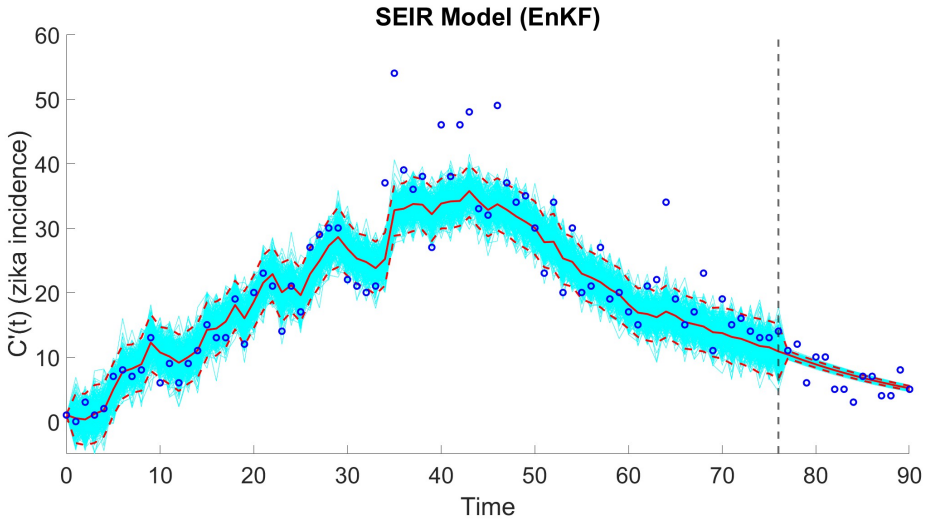

Figure 4.28: 14-day-ahead forecast of SEIR model trained on 77 days of Zika epidemic under the Ensemble Kalman Filter. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

Table 4.4 summarizes the results from all 14-day forecasts conducted over Case Study 4. Just as in Case Study 3, the SEIR model resulted in forecasts with the lowest MAE, MSE, RMSE, and WIS; There is a noticeable gain in forecasting performance over the SIR model,

Forecasting Performance Metrics					
Model	MAE	MSE	RMSE	Coverage	WIS
GGM	21.920	489.729	22.130	0	17.867
GLM	2.989	13.016	3.608	85.714	2.008
GRM	3.801	19.240	4.386	50	2.694
SIR	5.104	31.759	5.636	42.857	4.123
SEIR	2.208	7.775	2.788	100	1.459
SIR(EnKF)	2.589	9.919	3.149	*	*
SEIR(EnKF)	1.846	4.765	2.183	*	*

Table 4.4: Performance metrics of model-based forecasts from Case Study 4

which suggests that disease latency is key for making quality forecasts of Zika. The GLM maintains its position as the second-best-performing model, resulting in prediction errors similar to the SEIR model and higher coverage compared to the GRM and SIR models. The Ensemble Kalman Filter continues to reduce prediction error, but future work would include cross-validating these results with attention to the state error process.

4.1.5 Case Study 5: Day 91

Case study 5 seeks to assess the forecasting quality of each model at the end of the Zika epidemic. Each model is trained using the first 91 days of Zika incidence data and then forecasts up to day 105. This particular case study imposes an extra challenge due to the spike in reported disease incidence on day 92. The ensemble mean of each model is then used to compute forecast performance metrics reported in Table 4.5. Consider the 14-day-ahead forecast of the GGM:

Figure 4.29 represents the 14-day forecast of the GGM calibrated on 91 days of the Zika epidemic. This forecast is an extreme example of what can go wrong when an inappropriate model is selected for late-term forecasts of an epidemic. The GGM performed well when calibrated with 35 days of data, but each successive case study demonstrated a decline in

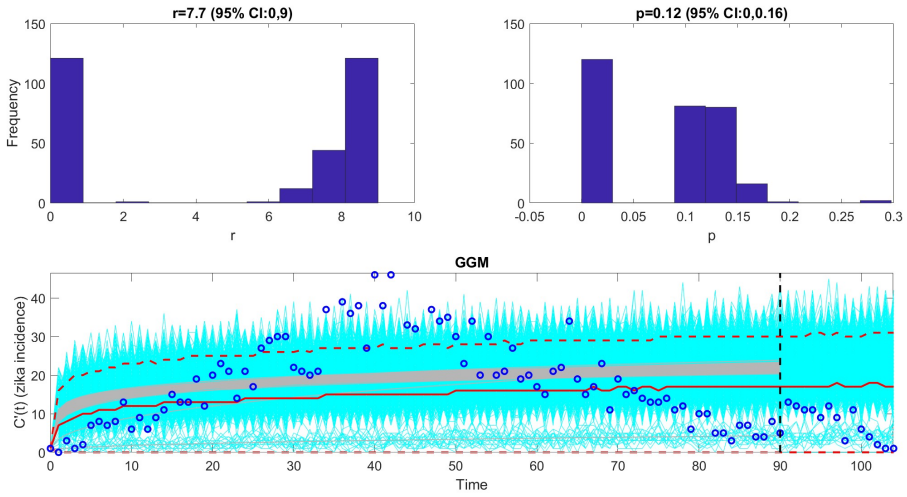

Figure 4.29: 14-day-ahead forecast of GGM trained on 91 days of Zika epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

performance. The 95% prediction interval attains 100% coverage with a WIS comparable to other models, but evaluating performance only with these metrics is incredibly misleading. It is a necessity to include measures of prediction error in addition to coverage of the 95% PI and WIS to evaluate forecasting performance fully. Next, we consider the 14-day-ahead forecast of the GLM:

Figure 4.30 depicts the 14-day forecast of the GLM calibrated using the parametric bootstrapping approach. The resulting forecast underestimates Zika incidence, with lower coverage and higher errors over Case Study 4. However, the reduced performance is attributed to the late-term spike in incidence that the GLM cannot account for with the current model formulation. Now, we consider the 14-day forecast of the GRM:

Figure 4.31 represents the 14-day forecast of the GRM calibrated using the bootstrapping approach. The training window maintains a good fit to the data, but the resulting forecast underestimates incidence. The GRM results in higher prediction errors when compared to the GLM, which has remained a consistent result across all 5 case studies conducted. So,

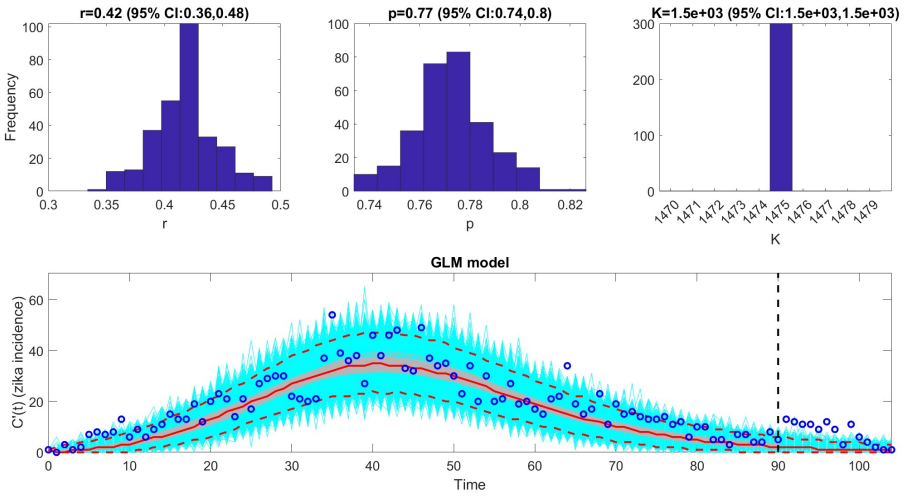

Figure 4.30: 14-day-ahead forecast of GLM trained on 91 days of Zika epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

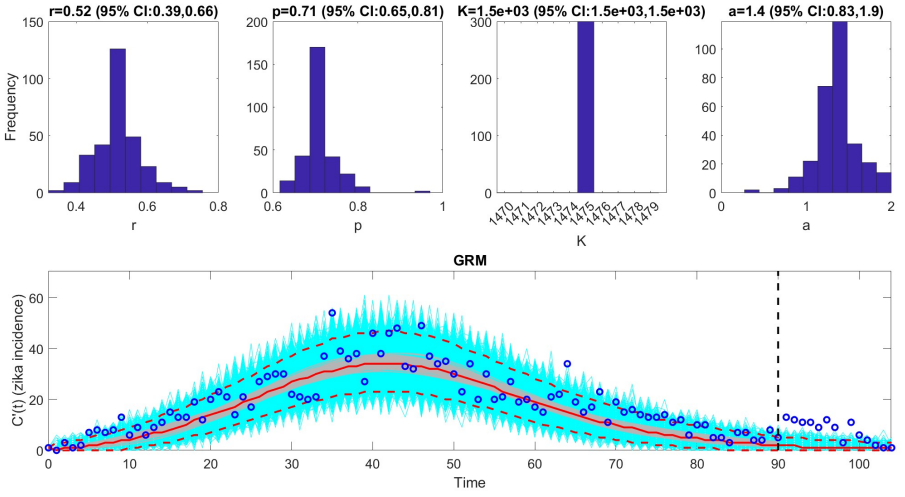

Figure 4.31: 14-day-ahead forecast of GRM trained on 91 days of Zika epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

extending the GLM with the scaling parameter a does not improve forecasting accuracy.

Next, we consider the 14-day forecast of the SIR model:

Figure 4.32 depicts the 14-day-ahead forecast of the SIR model trained with a bootstrapping

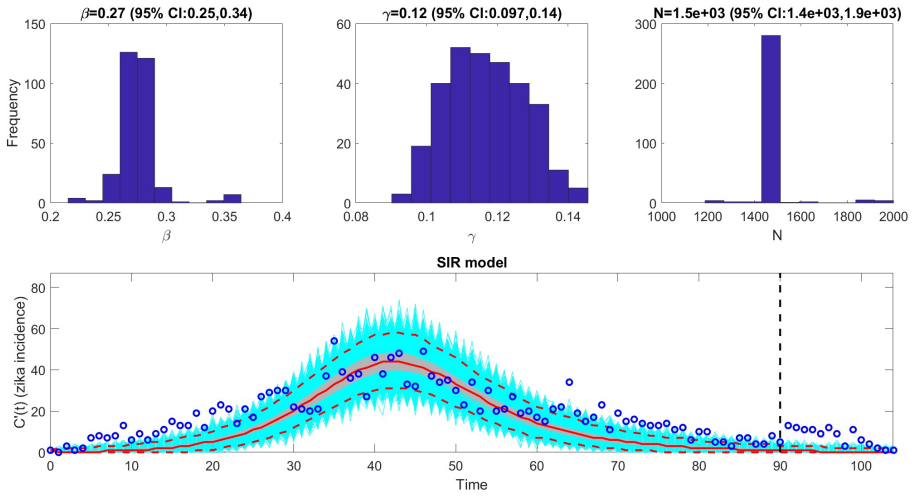

Figure 4.32: 14-day-ahead forecast of SIR model trained on 91 days of Zika epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

approach. The calibration window shows the SIR model does not fit well with the early and late growth dynamics of the Zika epidemic. As a result, the ensemble mean of the 14-day forecast predicts that the epidemic is over. Thus, the SIR model yields the highest prediction error out of all models tested in Case Study 5 - barring the GGM - and is a poor choice to make late-term forecasts in this scenario. We will now consider how the SEIR model fares with late-term forecasting of Zika incidence:

Figure 4.33 represents the 14-day forecast of the SEIR model trained with a bootstrapping approach. The calibration window shows a better fit to the data as the epidemic wanes, and the resulting forecast yields the lowest prediction errors across all models tested in Case Study 5. This case study showcases how modeling the latent period is key in reducing prediction error, even if the 14-day forecast misses the spike in incidence at the end of the epidemic. Finally, we consider how the Ensemble Kalman Filter affects the forecasting performance of the SIR and SEIR models:

Figure 4.34 depicts the 14-day forecast of the SIR model calibrated using the EnKF. The

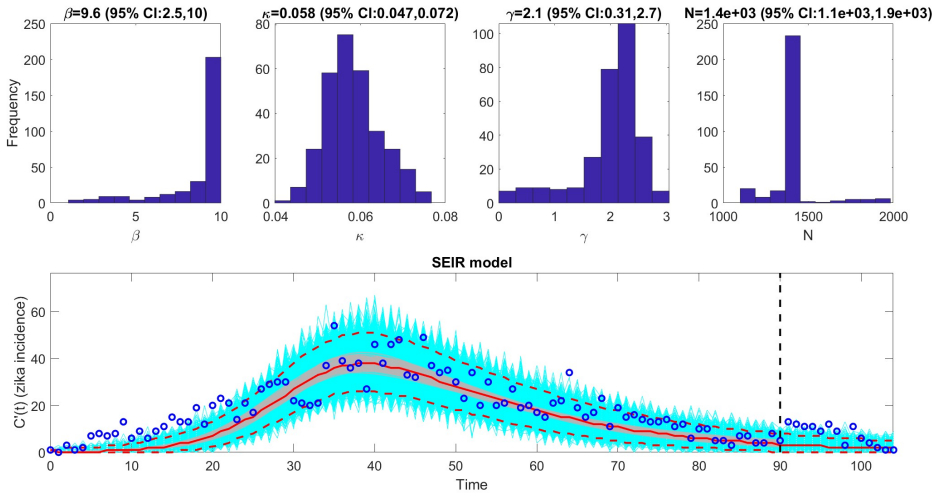

Figure 4.33: 14-day-ahead forecast of SEIR model trained on 91 days of Zika epidemic under the parametric bootstrapping method. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

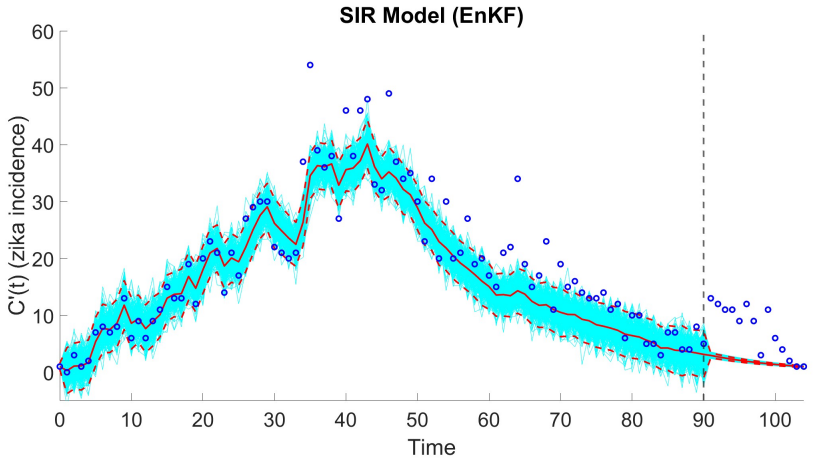

Figure 4.34: 14-day-ahead forecast of SIR model trained on 91 days of Zika epidemic under the Ensemble Kalman Filter. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

resulting forecast still fails to capture the spike in incidence on the 92nd day of the epidemic, but the EnKF yields a reduction in prediction error from the ensemble mean.

Figure 4.35 represents the 14-day forecast of the SEIR model calibrated using the EnKF. Similar to Figure 4.34, the EnKF does improve prediction in terms of error. However, future

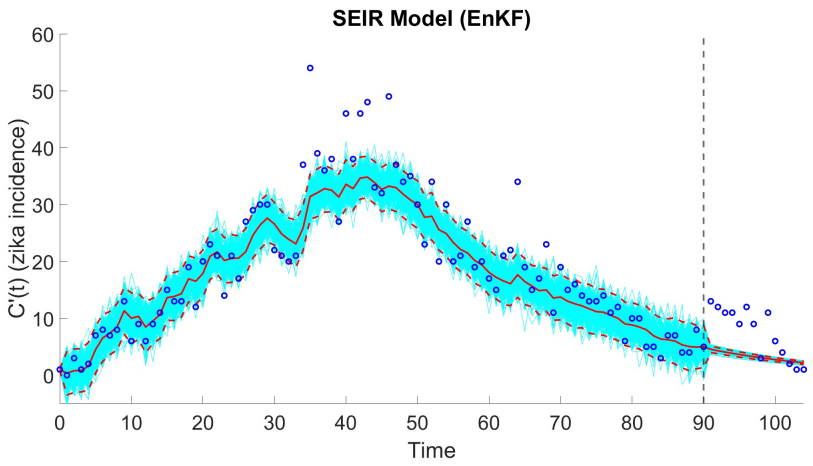

Figure 4.35: 14-day-ahead forecast of SEIR model trained on 91 days of Zika epidemic under the Ensemble Kalman Filter. The x-axis of the forecasting plot represents time in days, and the y-axis represents Zika disease incidence.

work is required to cross-validate these results to determine which technique results in better forecasts.

Forecasting Performance Metrics					
Model	MAE	MSE	RMSE	Coverage	WIS
GGM	13.961	214.476	14.645	100	4.841
GLM	6.079	52.778	7.265	28.571	5.300
GRM	6.265	55.329	7.438	28.571	5.515
SIR	6.922	65.236	8.077	21.429	6.393
SEIR	5.120	39.638	6.296	42.857	4.236
SIR(EnKF)	5.709	47.053	6.860	*	*
SEIR(EnKF)	4.691	31.860	5.644	*	*

Table 4.5: Performance metrics of model-based forecasts from Case Study 5

Table 4.5 summarizes the results from all 14-day-ahead forecasts conducted over Case Study 5. The SEIR model maintains its position as the top-performing model in late-term forecasting of Zika incidence. The bootstrapping and EnKF approaches yielded the lowest prediction errors for the SEIR model. In contrast, the SIR model failed to capture early and late dynamics within the calibration window, and the resulting forecast suggests that the epidemic had ended before day 105. The GLM maintained consistency in outperforming the GRM in

terms of error. An important takeaway from Case Study 5 is the importance of including multiple performance metrics; if one were to only evaluate the models in terms of coverage of the 95% prediction interval and WIS, then the GGM would be the clear winner. However, the GGM resulted in significantly higher prediction errors in comparison to the rest of the models.

Chapter 5

Discussion

The purpose of this retrospective study is to apply forecasting techniques to the 2015-2016 Zika epidemic to identify which simple epidemic models result in accurate forecasts of disease incidence. This chapter provides a summary of key results from each case study, connects our findings to our primary research questions, and then concludes with a discussion on limitations and future work. Our primary research question is: How do model-based forecasts of simple epidemic models compare under a Parametric Bootstrapping and Ensemble Kalman Filtering approach? By analyzing the results from each case study, we can then answer: does the top-performing model change as the epidemic progresses, and how do spikes in Zika incidence affect the forecasting performance of each model?

In Case Study 1, we calibrated the GGM, GLM, GRM, SIR, and SEIR models on the first 35 days of Zika incidence to assess forecasting performance early in an epidemic. The mean absolute error, mean squared error and root-mean-squared error is calculated using the forecast ensemble mean, and the resulting 95% prediction interval is used to compute coverage and the weighted interval score. Since data is limited in this first case study, the majority of models tested failed to capture the increase in Zika incidence. However, the GGM resulted in the best forecast with lower errors and 92.86% coverage of the prediction interval. The GLM and GRM performed similarly, with both models underestimating Zika incidence throughout the forecasting period; the GRM resulted in lower prediction errors and higher coverage over the GLM. The SIR and SEIR models resulted in high prediction errors,

with an MSE of 925.239 and 947.150, respectively. However, the state estimation offered by the EnKF approach did yield a reduction in prediction error for the SIR and SEIR models.

In Case Study 2, we observed an improvement in forecasting performance for all models tested - barring the GGM. The GLM resulted in the lowest prediction errors of 14-day forecasts, with an MSE of 17.858, and achieved 100% coverage of the 95% prediction interval. The GRM trailed behind the GLM in terms of prediction error, resulting in an MSE of 19.240. The SIR and SEIR models resulted in higher forecasting performance over the previous case study, with the SEIR model yielding the third best-performing forecast with an MSE of 64.424 and 85.71% coverage of the 95% prediction interval. The GGM showcased a departure from its position as the best model to forecast the start of the Zika epidemic and highlights the necessity of including growth scaling parameters or disease transmission mechanisms into the model structure as an epidemic progresses past the peak.

In Case Study 3, we observed a change in the top-performing model: the SEIR model resulted in the lowest prediction errors with an MSE of 80.574 and achieved 64.29% coverage of the 95% prediction interval. The GLM provided the second-best forecast, resulting in an MSE of 92.035, but only covered 35.71% of observations within the 95% prediction interval. The GRM performed similarly to the GLM in terms of prediction errors but with reduced coverage at 14.29%. The GGM and SIR models resulted in the poorest quality forecasts during Case Study 3, but the comparison of these two models showcases the necessity of reporting multiple types of performance metrics. In terms of prediction error, the SIR resulted in a lower MSE of 170.332 compared to the GGM at 383.055. However, we would arrive at a different conclusion the coverage of the 95% prediction interval is the only metric reported. The SIR resulted in 0% coverage, but the GGM resulted in 7.14% coverage, which might lead us to believe that the GGM results in a more accurate forecast. Therefore, an effective framework for comparing forecasting performance should report multiple measures

of performance.

In Case Study 4, we observe the SEIR model result in the best-performing forecast, with the lowest prediction errors reported across all case studies, and achieved 100% coverage of observed incidence within the 95% prediction interval. Contrasting the performance of the SIR model with the top-performing SEIR model, the SIR model resulted in higher prediction error and lower coverage. From the second case study onward, the SEIR model has continued to provide more accurate forecasts than the SIR model, with the notable difference between models being the addition of the disease latency. The EnKF approach provided a reduction in MSE for the SEIR model from 7.775 to 4.765. The GLM provided the second-best forecast, reporting an MSE of 13.016 and achieved 85.71% coverage. The GRM continues its pattern of trailing behind the GLM in forecasting performance, resulting in a higher reported MSE of 19.240 and only 50% coverage. Finally, the GGM continues the systematic overestimation of incidence, resulting in highly inaccurate forecasts with a reported MSE of 489.729 and 0% coverage.

In Case Study 5, the SEIR model achieved the highest-performing forecast with an MSE of 39.638 and 42.86% coverage of observed incidence. The GLM and GRM provided the second and third-best forecasts, resulting in an MSE of 52.778 and 55.329, respectively, and 28.57% coverage from both models. The SIR model continued the trend of falling behind the rest of the model in late-term forecasting, resulting in an MSE of 65.236 and 21.43% coverage. However, the EnKF approach did reduce the prediction errors of the SIR model below the GLM and GRM. Finally, the GGM demonstrates the importance of reporting multiple performance metrics to provide an accurate evaluation; the GGM achieved 100% coverage in the 95% prediction interval and the second lowest WIS but has the highest prediction errors of all models tested.

5.1 Summary

In summary, this retrospective study seeks to assess the forecasting performance of simple epidemic models and compare the Parametric Bootstrapping and Ensemble Kalman Filtering techniques through a series of five case studies. Each case study is designed to evaluate the models of interest on calibration windows of increasing lengths to identify when model features, or mechanisms result in best-performing forecasts and at what time these mechanisms result in improved accuracy. At the onset of the Zika epidemic, the GGM provided the most accurate forecast of incidence. The GLM resulted in the highest prediction accuracy and coverage throughout the second case study, for the mechanistic models do not have enough data to inform transmission and latency parameters. From the third case study onward, the SEIR model results in the highest quality forecasts, as the transmission and latency mechanisms are able to capture the slow decline in observed Zika incidence. The results from this study align with the forecasting literature on the usefulness of phenomenological models in characterizing and forecasting early epidemic growth [2, 4]. Yet, we still do not understand the exact time during an epidemic when mechanistic models begin to outperform phenomenological models in forecasting incidence. A result consistent across all case studies is that the SIR and SEIR models simulated under the Ensemble Kalman Filter showed increases in prediction accuracy over the bootstrapping technique. However, future work is required to cross-validate these results due to simplifying assumptions in state and observational noise processes.

5.2 Future Work

An important result of this thesis is that models incorporating disease mechanisms result in higher accuracy forecasts of Zika incidence later into an epidemic. To continue this disease forecasting framework, future research topics would include considerations of vector-borne disease models to see if additional mechanisms bolster forecasting performance. Vector-borne disease models consider the interactions between a vector and disease host populations, and in the case of Zika, the interactions between human and mosquito populations would be modeled via a biting rate. The models included in this study did not consider human-mosquito dynamics, so including this extra mechanism is important for future forecast validation because biting is the primary transmission mechanism of Zika. Another result from this study is the Ensemble Kalman Filter providing lower prediction error of the SIR and SEIR forecasts over the parametric bootstrap. However, the study is limited by the simplifying assumption of fixed state and observational noise, so future work is needed to cross-validate this result. Initial attempts to estimate the state errors through the ensemble mean resulted in numerical errors throughout the forecasting step, so future research includes exploration into other methods of quantifying uncertainty.

Another limitation of this study is that all models fit under the parametric bootstrap assumed a Poisson error distribution. The QuantDiffForecast toolbox supports multiple error structures for bootstrapping and maximum likelihood estimation techniques [8]. So, future work includes investigating how forecasting performance changes under another error structure or estimation technique. Throughout the results section, the GRM and SEIR parameter distributions exhibited consistent, sporadic behavior from the bootstrapping process. This behavior suggests that these model parameters are not identifiable, so future work using a Monte Carlo approach would be required to confirm these findings. Another limitation of this study is that we only mathematical models, so additional future work includes consid-

eration of statistical models like the Autoregressive Integrated Moving Average (ARIMA) models or Bayesian techniques using Markov Chain Monte Carlo (MCMC) sampling. Epidemic forecasting is still a developing field of research, so expanding the scope of forecasting validation to other techniques would greatly benefit our understanding of the subject, assisting epidemic modelers and public health officials to make the most informed decisions possible.

Bibliography

[1] Mahmood Akhtar, Moritz UG Kraemer, and Lauren M Gardner. A dynamic neural network model for predicting risk of zika in real time. *BMC medicine*, 17:1–16, 2019.

[2] Gerardo Chowell. Fitting dynamic models to epidemic outbreaks with quantified uncertainty: A primer for parameter uncertainty, identifiability, and forecasts. *Infectious Disease Modelling*, 2(3):379–398, 2017.

[3] Gerardo Chowell and Ruiyan Luo. Ensemble bootstrap methodology for forecasting dynamic growth processes using differential equations: application to epidemic outbreaks. *BMC Medical Research Methodology*, 21(1):1–18, 2021.

[4] Gerardo Chowell and Cécile Viboud. Is it growing exponentially fast?–impact of assuming exponential growth for characterizing and forecasting epidemics with initial near-exponential growth dynamics. *Infectious disease modelling*, 1(1):71–78, 2016.

[5] Gerardo Chowell, Doracelly Hincapie-Palacio, Juan Ospina, Bruce Pell, Amna Tariq, Sushma Dahal, Seyed Moghadas, Alexandra Smirnova, Lone Simonsen, and Cécile Viboud. Using phenomenological models to characterize transmissibility and forecast patterns and final burden of zika epidemics. *PLoS currents*, 8, 2016.

[6] Gerardo Chowell, Amna Tariq, and James M Hyman. A novel sub-epidemic modeling framework for short-term forecasting epidemic waves. *BMC Medicine*, 17:1–18, 2019.

[7] Gerardo Chowell, Sushma Dahal, Amna Tariq, Kimberlyn Roosa, James M Hyman, and Ruiyan Luo. An ensemble n-sub-epidemic modeling framework for short-term fore-

casting epidemic trajectories: Application to the covid-19 pandemic in the usa. *PLoS Computational Biology*, 18(10):e1010602, 2022.

[8] Gerardo Chowell, Amanda Bleichrodt, and Ruiyan Luo. Parameter estimation and forecasting with quantified uncertainty for ordinary differential equation models using quantdiffforecast: A matlab toolbox and tutorial. *Statistics in Medicine*, 2024.

[9] Estee Y Cramer, Yuxin Huang, Yijin Wang, Evan L Ray, Matthew Cornell, Johannes Bracher, Andrea Brennen, Alvaro J Castro Rivadeneira, Aaron Gerding, Katie House, Dasuni Jayawardena, Abdul H Kanji, Ayush Khandelwal, Khoa Le, Jarad Niemi, Ariane Stark, Apurv Shah, Nutcha Wattanachit, Martha W Zorn, Nicholas G Reich, and US COVID-19 Forecast Hub Consortium. The united states covid-19 forecast hub dataset. *medRxiv*, 2021. doi: 10.1101/2021.11.04.21265886. URL https://www.medrxiv.org/content/10.1101/2021.11.04.21265886v1.

[10] Estee Y Cramer, Evan L Ray, Velma K Lopez, Johannes Bracher, Andrea Brennen, Alvaro J Castro Rivadeneira, Aaron Gerding, Tilmann Gneiting, Katie H House, Yuxin Huang, et al. Evaluation of individual and ensemble probabilistic forecasts of covid-19 mortality in the united states. *Proceedings of the National Academy of Sciences*, 119 (15):e2113561119, 2022.

[11] Samuel Dixon, Ravikiran Keshavamurthy, Daniel H Farber, Andrew Stevens, Karl T Pazdernik, and Lauren E Charles. A comparison of infectious disease forecasting methods across locations, diseases, and time. *Pathogens*, 11(2):185, 2022.

[12] Ralf Engbert, Maximilian M Rabe, Reinhold Kliegl, and Sebastian Reich. Sequential data assimilation of the stochastic seir epidemic model for regional covid-19 dynamics. *Bulletin of mathematical biology*, 83(1):1, 2021.

[13] Alessio Farcomeni, Antonello Maruotti, Fabio Divino, Giovanna Jona-Lasinio, and Gianfranco Lovison. An ensemble approach to short-term forecast of covid-19 intensive care occupancy in italian regions. *Biometrical Journal*, 63(3):503–513, 2021.

[14] Centers for Disease Control and Prevention, May 2019. URL https://www.cdc.gov/zika/about/index.html.

[15] Sebastian Funk, Anton Camacho, Adam J Kucharski, Rosalind M Eggo, and W John Edmunds. Real-time forecasting of infectious disease dynamics with a stochastic semi-mechanistic model. *Epidemics*, 22:56–61, 2018.

[16] Sebastian Funk, Anton Camacho, Adam J Kucharski, Rachel Lowe, Rosalind M Eggo, and W John Edmunds. Assessing the performance of real-time epidemic forecasts: A case study of ebola in the western area region of sierra leone, 2014-15. *PLoS computational biology*, 15(2):e1006785, 2019.

[17] Daozhou Gao, Yijun Lou, Daihai He, Travis C Porco, Yang Kuang, Gerardo Chowell, and Shigui Ruan. Prevention and control of zika as a mosquito-borne and sexually transmitted disease: a mathematical modeling analysis. *Scientific reports*, 6(1):28070, 2016.

[18] Timothy C Germann, Kai Kadau, Ira M Longini Jr, and Catherine A Macken. Mitigation strategies for pandemic influenza in the united states. *Proceedings of the National Academy of Sciences*, 103(15):5935–5940, 2006.

[19] Rabih Ghostine, Mohamad Gharamti, Sally Hassrouny, and Ibrahim Hoteit. An extended seir model with vaccination for forecasting the covid-19 pandemic in saudi arabia using an ensemble kalman filter. *Mathematics*, 9(6):636, 2021.

[20] Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, 102(477):359–378, 2007.

[21] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. 1960.

[22] Matthias Katzfuss, Jonathan R Stroud, and Christopher K Wikle. Understanding the ensemble kalman filter. *The American Statistician*, 70(4):350–357, 2016.

[23] William Ogilvy Kermack and Anderson G McKendrick. A contribution to the mathematical theory of epidemics. *Proceedings of the royal society of london. Series A, Containing papers of a mathematical and physical character*, 115(772):700–721, 1927.

[24] Max Kuhn, Kjell Johnson, et al. *Applied predictive modeling*, volume 26. Springer.

[25] Chelsea S Lutz, Mimi P Huynh, Monica Schroeder, Sophia Anyatonwu, F Scott Dahlgren, Gregory Danyluk, Danielle Fernandez, Sharon K Greene, Nodar Kipshidze, Leann Liu, et al. Applying infectious disease forecasting to public health: a path forward using influenza forecasting examples. *BMC Public Health*, 19:1–12, 2019.

[26] Sarah F McGough, John S Brownstein, Jared B Hawkins, and Mauricio Santillana. Forecasting zika incidence in the 2016 latin america outbreak combining traditional disease surveillance with search, social media, and news report data. *PLoS neglected tropical diseases*, 11(1):e0005295, 2017.

[27] Leah Mitchell and Andrea Arnold. Analyzing the effects of observation function selection in ensemble kalman filtering for epidemic models. *Mathematical Biosciences*, 339: 108655, 2021.

[28] Kelly R Moran, Geoffrey Fairchild, Nicholas Generous, Kyle Hickmann, Dave Osthus, Reid Priedhorsky, James Hyman, and Sara Y Del Valle. Epidemic forecasting is messier

than weather forecasting: The role of human behavior and internet data streams in epidemic forecast. *The Journal of infectious diseases*, 214(suppl_4):S404–S408, 2016.

[29] S Morsy, TN Dang, MG Kamel, AH Zayan, OM Makram, M Elhady, K Hirayama, and NT Huy. Prediction of zika-confirmed cases in brazil and colombia using google trends. *Epidemiology & Infection*, 146(13):1625–1627, 2018.

[30] World Health Organization, Dec 2022. URL https://www.who.int/news-room/fact-sheets/detail/zika-virus.

[31] Bruce Pell, Yang Kuang, Cecile Viboud, and Gerardo Chowell. Using phenomenological models for forecasting the 2015 ebola challenge. *Epidemics*, 22:62–70, 2018.

[32] Hazhir Rahmandad, Ran Xu, and Navid Ghaffarzadegan. Enhancing long-term forecasting: Learning from covid-19 models. *PLOS Computational Biology*, 18(5):e1010100, 2022.

[33] Omar Saucedo, Amanda Laubmeier, Tingting Tang, Benjamin Levy, Lale Asik, Tim Pollington, and Olivia Prosper. Comparative analysis of practical identifiability methods for an seir model. *arXiv preprint arXiv:2401.15076*, 2024.

[34] Necibe Tuncer and Trang T Le. Structural and practical identifiability analysis of outbreak models. *Mathematical biosciences*, 299:1–18, 2018.

[35] Malcolm E Turner Jr, Edwin L Bradley Jr, Katherine A Kirk, and Kenneth M Pruitt. A theory of growth. *Mathematical Biosciences*, 29(3-4):367–373, 1976.

[36] Teresa K Yamana, Sasikiran Kandula, and Jeffrey Shaman. Superensemble forecasts of dengue outbreaks. *Journal of The Royal Society Interface*, 13(123):20160410, 2016.

[37] WT Yun, Lydia Stefanova, and TN Krishnamurti. Improvement of the multimodel superensemble technique for seasonal forecasts. *Journal of Climate*, 16(22):3834–3840, 2003.

Appendices

Appendix A

Appendix

A.1 Supplemental Tables for 4

GGM Parameters		
Parameter	Search Interval	Initial Value
r	[0,9]	0.9
p	[0,1]	0.5
GLM Parameters		
Parameter	Search Interval	Initial Value
r	[0,1]	0.5
p	[0,1]	0.5
K	Fixed	1475
GRM Parameters		
Parameter	Search Interval	Initial Value
r	[0,9]	0.9
p	[0,1]	0.5
a	[0,2]	1
K	Fixed	1475

Table A.1: Parameter Search Bounds and Initial Values for GGM, GLM, and GRM Model Fitting

A.2 Simulation Script

A.2.1 Parametric Bootstrapping Options Script

QuantDiffForecast Options file for GGM:

SIR Parameters		
Parameter	Search Interval	Initial Value
β	[0,10]	4
γ	[0,10]	4
N	[1100, 2000]	1475
SEIR Parameters		
Parameter	Search Interval	Initial Value
β	[0,10]	4
κ	[0,4]	1
γ	[0,10]	4
N	[1100, 2000]	1475

Table A.2: Parameter Search Bounds and Initial Conditions for SIR and SEIR Model Fitting

SIR Optimal Parameter Sets			
Case Study	β	γ	N
1	0.78	0.6	1500
2	0.43	0.17	1960
3	0.34	0.13	1689
4	0.31	0.12	1653
5	0.27	0.12	1477

Table A.3: SIR optimal parameter sets for EnKF under each case study

SEIR Optimal Parameter Sets				
Case Study	β	κ	γ	N
1	9.35	0.69	8.17	1260
2	8.79	0.14	4.35	1171
3	9.62	0.07	3.14	1171
4	9.64	0.06	1.51	1987
5	9.62	0.06	2.14	1405

Table A.4: SEIR optimal parameter Sets for EnKF under each case study

```
1 function [cadfilename1,caddisease,datatype, dist1, numstartpoints
    ,B, model, params,vars,getperformance, forecastingperiod,
    windowsize1,tstart1,tend1,printscreen1]=options_forecast
2
3 % Declare global variables
```

```
4
5  global method1 % Parameter est. method
6
7  % Datasets properties
8
9  cadfilename1 = 'zika-incidence';
10
11 caddisease='zika'; % name of the disease
12
13 datatype='incidence'; % (cases, deaths, hospitalizations, etc)
14
15 % Parameter estimation
16
17 method1 = 0; % Type of estimation method
18
19 % Nonlinear least squares (LSQ)=0
20
21 dist1 = 1; % Define dist
22
23 % dist1 = 0; % Normal dist
24 % dist1 = 1; % Poisson error
25
26 switch method1
27     case 1
28         dist1=1;
```

```
29      case 3
30          dist1=3;
31      case 4
32          dist1=4;
33      case 5
34          dist1=5;
35  end
36
37  numstartpoints=10; % MultiStart
38
39  B=300; % Num. of Bootstraps
40
41  % ODE model
42
43  model.fc=@GGM;  % call model function
44  model.name='GGM'; % name of model
45
46  params.num=1;
47  params.label={'r','p'}; % model parameter labels
48  params.LB=[0 0];  % parameter est. LB
49  params.UB=[9 1]; % parameter est. UB
50  params.initial=[0.9 0.5]; % initial guess
51  params.fixed=[0 0]; % (0) est parameter (1) fix parameter
52  params.fixI0=1; % (1) fix initial value (0) not
53  params.composite='';  % composite parameter est
```

```
%params.extra0=[];

vars.label={'C'}; % state variables
vars.initial=1; % IC
vars.fit_index=1; % vector index for fitting
vars.fit_diff=1; % (1) fit derivative of state var (0) not

% Forecasting parameters

getperformance=1; % (1/0) forecast performance computation

forecastingperiod=14; % horizon of forecast

% Rolling window
windowsize1=17;  % moving window size

tstart1=1; % RW analysis start point

tend1=1;  % RW analysis end point

printscreen1=1; % (0/1) print screen
```

QuantDiffForecast Options file for GLM:

```
function [cadfilename1,caddisease,datatype, dist1, numstartpoints
   ,B, model, params,vars,getperformance, forecastingperiod,
   windowsize1,tstart1,tend1,printscreen1]=options_forecast
```

```
2
3 % Declare global variables
4
5 global method1 % Parameter est. method
6
7 % Datasets properties
8
9 cadfilename1='zika-incidence';
10
11 caddisease='zika'; % name of the disease
12
13 datatype='incidence'; % (cases, deaths, hospitalizations, etc)
14
15 % Parameter estimation
16
17 method1=0; % Est. method
18
19 % Nonlinear least squares (LSQ)=0
20
21 dist1=1; % Def distribution
22
23 %dist1=0; % Normal dist
24 %dist1=1; % Poisson error
25
26 switch method1
```

```
    case 1
        dist1=1;
    case 3
        dist1=3;
    case 4
        dist1=4;
    case 5
        dist1=5;
end

numstartpoints=10; % MultiStart

B=300; % Num. of bootstraps

% ODE model

model.fc=@GLM;  % call model function
model.name='GLM model'; % name of model

params.num=3;
params.label={'r','p','K'}; % model parameter labels
params.LB=[0 0 1000];  % parameter est. LB
params.UB=[1 1 1500]; % parameter est. UB
params.initial=[0.5 0.5 1475]; % initial guess
params.fixed=[0 0 1]; % (0) est parameter (1) fix parameter
```

```
52 params.fixI0=1; % (1) fix initial value (0) not
53 params.composite='';  % composite parameter est
54 %params.extra0=[]; %
55
56 vars.label={'C'}; % state variables
57 vars.initial=1; % IC
58 vars.fit_index=1; % vector index for fitting
59 vars.fit_diff=1; % (1) fit derivative of state var (0) not
60
61 % Forecasting parameters
62
63 getperformance=1; % (1/0) forecast performance computation
64
65 forecastingperiod=14; % horizon of forecast
66
67 % Rolling window
68
69 windowsize1=17;  % moving window size
70
71 tstart1=1; % RW analysis start point
72
73 tend1=1;  % RW analysis end point
74
75 printscreen1=1; % (0/1) print screen
```

QuantDiffForecast Options file for GRM:

```
function [cadfilename1,caddisease,datatype, dist1, numstartpoints
   ,B, model, params,vars,getperformance, forecastingperiod,
   windowsize1,tstart1,tend1,printscreen1]=options_forecast

% Declare global variables

global method1 % Parameter est. method

% Datasets properties

cadfilename1='zika-incidence';

caddisease='zika'; % name of the disease

datatype='incidence'; % (cases, deaths, hospitalizations, etc)

%
% Parameter estimation =

method1=0; % Type of estimation method

% Nonlinear least squares (LSQ)=0

dist1=1; % Define distribution

```

```
%dist1=0; % Normal dist
%dist1=1; % Poisson error

switch method1
    case 1
        dist1=1;
    case 3
        dist1=3;
    case 4
        dist1=4;
    case 5
        dist1=5;
end

numstartpoints=10; % MultiStart

B=300; % Num. of bootstraps

% ODE model

model.fc=@GRM;  % call model function
model.name='GRM'; % name of model

params.num=4;
params.label={'r','p','K','a'}; % model parameter labels
```

```
49 params.LB=[0 0 1000 0];  % parameter est. LB
50 params.UB=[1 1 1500 2]; % parameter est. UB
51 params.initial=[0.5 0.5 1475 1]; % initial guesses
52 params.fixed=[0 0 1 0]; % (0) est parameter (1) fix parameter
53 params.fixI0=1; % (1) fix initial value (0) not
54 params.composite='';  % composite parameter est
55 %params.extra0=[];
56
57 vars.label={'C'}; % state variables
58 vars.initial=1; % IC
59 vars.fit_index=1; % vector index for fitting
60 vars.fit_diff=1; % (1) fit derivative of state var (0) not
61
62 % Forecasting parameters
63
64 getperformance=1; % (1/0) forecast performance computation
65
66 forecastingperiod=14; % horizon of forecast
67
68 % Rolling window
69
70 windowsize1=17;  % moving window size
71
72 tstart1=1; % RW analysis start point
73
```

```
tend1=1;  % RW analysis end point

printscreen1=1; % (0/1) print screen
```

QuantDiffForecast Options file for SIR:

```
function [cadfilename1,caddisease,datatype, dist1, numstartpoints
   ,B, model, params,vars,getperformance, forecastingperiod,
   windowsize1,tstart1,tend1,printscreen1]=options_forecast

% Declare global variables

global method1 % Parameter est.

% Datasets properties

cadfilename1='zika-incidence';

caddisease='zika'; % name of the disease

datatype='incidence'; % (cases, deaths, hospitalizations, etc)

% Parameter estimation

method1=0; % Type of estimation method

% Nonlinear least squares (LSQ)=0,
```

```
20
21 dist1=1; % Def distribution
22
23 %dist1=0; % Normal dist
24 %dist1 = 1; % Poisson error
25
26 switch method1
27     case 1
28         dist1=1;
29     case 3
30         dist1=3;
31     case 4
32         dist1=4;
33     case 5
34         dist1=5;
35 end
36
37 numstartpoints=10; % MultiStart
38
39 B=300; % Num. of bootstraps
40
41 % ODE model
42
43 model.fc=@SIR1; % call model function
44 model.name='SIR model';   % name of model
```

```
45
46 params.num=3; % num of model parameters
47 params.label={'\beta','\gamma','N'};  % model parameter labels
48 params.LB=[0 0 1100]; % parameter est. LB
49 params.UB=[10 10 2000]; % parameter est. UB
50 params.initial=[4 4 1475]; % initial guess
51 params.fixed=[0 0 0]; % (0) est parameter (1) fix parameter
52 params.fixI0=1; % (1) fix initial value (0) not
53 params.composite='';  % composite parameter est
54 %params.composite_name=''
55
56 vars.num=4; % num of state variables
57 vars.label={'S','I','R','C'}; % state variables
58 vars.initial=[params.initial(3)-1 1 0 1];  % IC
59 vars.fit_index=4; % vector index for fitting
60 vars.fit_diff=1; % (1) fit derivative of state var (0) not
61
62 % Forecasting parameters
63
64 getperformance=1; % (1/0) forecast performance computation
65
66 forecastingperiod=14; % horizon of forecast
67
68 % Rolling window
69
```

```
windowsize1=17;  % moving window size

tstart1=1; % RW analysis start point

tend1=1;  % RW analysis end point

printscreen1=1; % (0/1) print screen
```

QuantDiffForecast Options file for SEIR:

```
function [cadfilename1,caddisease,datatype, dist1, numstartpoints
   ,B, model, params,vars,getperformance, forecastingperiod,
   windowsize1,tstart1,tend1,printscreen1]=options_forecast

% Declare global variables

global method1 % Parameter est. method

% Datasets properties

cadfilename1='zika-incidence';

caddisease='zika'; % name of the disease

datatype='incidence'; % (cases, deaths, hospitalizations, etc)

% Parameter estimation
```

```
16
17 method1=0; % Type of estimation method
18
19 % Nonlinear least squares (LSQ)=0,
20
21 dist1=1; % Def dist
22
23 %dist1=0; % Normal dist
24 %dist1=1; % Poisson error
25
26 switch method1
27     case 1
28         dist1=1;
29     case 3
30         dist1=3;
31     case 4
32         dist1=4;
33     case 5
34         dist1=5;
35 end
36
37 numstartpoints=10; % MultiStart
38
39 B=300; % Num. of bootstraps
40
```

```
41 % ODE model
42
43 model.fc=@SEIR1; % call model function
44 model.name='SEIR model';   % name of model
45
46 params.num=4; %
47 params.label={'\beta','\kappa','\gamma','N'};  % model parameter
      labels
48 params.LB=[0 0 0 1100]; % parameter est. LB
49 params.UB=[10 4 10 2000]; % parameter est. UB
50 params.initial=[4 1 4 1475]; % initial guess
51 params.fixed=[0 0 0 0]; % (0) est parameter (1) fix parameter
52 params.fixI0=1; % (1) fix initial value (0) not
53 params.composite='';  % composite parameter est
54 %params.composite_name='';
55
56 vars.num=5; % number of state vars
57 vars.label={'S','E','I','R','C'}; % state variables
58 vars.initial=[params.initial(4)-1 0 1 0 1];  % IC
59 vars.fit_index=5; % vector index for fitting
60 vars.fit_diff=1; % (1) fit derivative of state var (0) not
61
62 % Forecasting parameters
63
64 getperformance=1; % (1/0) forecast performance computation
```

```
65
66 forecastingperiod=14; % horizon of forecast
67
68 % Rolling window
69
70 windowsize1=17;  % moving window size
71
72 tstart1=1; % RW analysis start point
73
74 tend1=1;  % RW analysis end point
75
76 printscreen1=1; % (0/1) print screen
```

A.2.2 Ensemble Kalman Filter Script

Script for SIR model under EnKF:

```
1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2 %%%% Script for ENKF of ZIKV Data %%%%
3 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
4 clear all, close all
5 data_frame = importdata('zika-incidence.txt');
6 training_period = 49; % Set desired training window
7 zikv_data = data_frame(1:training_period,2);
8 time_data = data_frame(1:training_period,1);
9 train_test_data = data_frame(1:(training_period+14),2);
```

```
10 time_vec = data_frame(:,1);
11 data_vec = data_frame(:,2);
12
13 %Select case type ('case 1' through 'case 4')
14 casetype = 'case 3';
15
16 %Set ensemble size
17 N = 300;
18
19 %Set SIR model parameters
20 % Make randomized?
21 global params0
22 % beta gamma N is the order
23 % remove comment to select which parameter set to use:
24 % params0 = [0.78 0.6 1500]; % SIR day 35
25  params0 = [0.43 0.17 1960]; % SIR day 49
26 % params0 = [0.34 0.13 1689]; % SIR day 63
27 % params0 = [0.31 0.12 1653]; % SIR day 77
28 % params0 = [0.27 0.12 1477]; % SIR day 91
29 Np = params0(3);
30
31 tdata = time_data;
32
33 % Generate ensemble for INIT
34
```

```
35 x0 = [Np-1 1];
36 true_init = x0;
37
38 aS = 0.5*true_init(1);
39 bS = 1.5*true_init(1);
40
41 uniS = aS+(bS-aS).*rand(1,N);
42
43 aI = 0.5*true_init(2);
44 bI = 1.5*true_init(2);
45
46 uniI = aI+(bI-aI).*rand(1,N);
47
48 % Ensemble Storage
49 S = zeros(2,N);
50 S(1,:) = uniS;
51 S(2,:) = uniI;
52
53 S = [S; zeros(1,N);S(2,:)]; %Gives the 4th state variable, C'
      will be used for Nu to recover the incidence data
54
55 % Ensemble statis
56
57 % verify gamma0 matrix
58 xbar0 = (1/N)*sum(S,2);
```

```
59 gamma0 = ((S-xbar0)*transpose(S-xbar0))/(N-1);
60
61 %Store ensemble mean, +/- 2 standard deviation curves
62 fmean = zeros(length(zikv_data),4);
63 fplus = zeros(length(zikv_data),4);
64 fminus = zeros(length(zikv_data),4);
65
66 fmean(1,:) = xbar0;
67 fplus(1,:) = transpose(abs(1.96*sqrt(diag(gamma0))))+fmean(1,:);
68 fminus(1,:) = transpose(-abs(1.96*sqrt(diag(gamma0))))+fmean(1,:)
      ;
69
70 % Set up sample covariances
71 stdC = 2;
72 stdD = 1;
73 D = stdD^2;
74 nu = zeros(1,N);
75
76 options = odeset('RelTol',1e-8,'AbsTol',1e-8);
77
78 % Test Model before
79
80 % init = [x0 0 x0(2)]
81 % [~,Y] = ode45(@SIR1,1:1:49,init,options)
82 %
```

```
% plot(Y(:,4))

% Kalman Filter Loop:
% For training purposes, length of loop will depend on the
   training period
% 35,49, ect...

% Store EnKF solution trajectories for states
s_sol = zeros(N,length(zikv_data));
i_sol = zeros(N,length(zikv_data));
r_sol = zeros(N,length(zikv_data));
c_sol = zeros(N,length(zikv_data));
% Update with initial condition of ensemble:
s_sol(:,1) = S(1,:);
i_sol(:,1) = S(2,:);
r_sol(:,1) = S(3,:);
c_sol(:,1) = S(4,:);
%

for j = 2:(length(zikv_data))

    %Prediction step
    for n = 1:N

        ts = [tdata(j-1),tdata(j)];
```

```
107
108         x0 = [S(1,n);S(2,n);S(3,n);0]; %C is set to zero each
                solution to track incidence
109
110         [~,Y] = ode15s(@SIR1,ts,x0,options);
111
112         S(:,n) = transpose(Y(end,:))+stdC*randn(4,1);
113         nu(n) = Y(end,4);
114     end
115
116     % Ensemble mean
117     xbar = (1/N)*sum(S,2);
118
119     % Kalman analysis step
120
121     if strcmp(casetype,'case 1')== 1
122         yhat = IofT(S(1:2,:),1);
123     elseif strcmp(casetype,'case 2')== 1
124         yhat = IofT(S(1:2,:),0.7);
125     elseif strcmp(casetype, 'case 3')== 1
126         yhat = Integral(nu,1);
127     elseif strcmp(casetype, 'case 4')== 1
128         yhat = Integral(nu,0.7);
129     else
130         error('Case not found, try again');
```

```
131     end
132
133     % Kalman Gain and ensemble stats
134
135     yhatbar = (1/N)*sum(yhat,2);
136     cross = ((S-xbar)*transpose(yhat-yhatbar))/(N-1);
137     forecast = ((yhat-yhatbar)*transpose(yhat-yhatbar))/(N-1);
138     K = cross/(forecast + D);
139     ydata = zikv_data(j)+stdD*randn(1,N);
140
141     %Update ensemble
142     S = S+ K*(ydata-yhat);
143     s_sol(:,j) = S(1,:);
144     i_sol(:,j) = S(2,:);
145     r_sol(:,j) = S(3,:);
146     c_sol(:,j) = S(4,:);
147     %Ensemble statistics
148     fmean(j,:) = (1/N)*sum(S,2);
149     gamma = ((S-transpose(fmean(j,:)))*transpose(S-transpose(
          fmean(j,:))))/(N-1);
150
151     fplus(j,:) = transpose(abs(1.96*sqrt(diag(gamma))))+fmean(j
          ,:);
152     fminus(j,:) = transpose(-abs(1.96*sqrt(diag(gamma))))+fmean(j
          ,:);
```

```
153
154 end
155
156 %% Forecast Loop:
157 fmean_fore = zeros(14,4);
158 fplus_fore = zeros(14,4);
159 fminus_fore = zeros(14,4);
160 S_fore = S;
161 s_sol_forecast = zeros(N,14);
162 i_sol_forecast = zeros(N,14);
163 r_sol_forecast = zeros(N,14);
164 c_sol_forecast = zeros(N,14);
165 for k = training_period+1:(training_period + 14)
166
167     for n = 1:N
168
169         ts_fore = [time_vec(k-1),time_vec(k)];
170
171         x0_fore = [S_fore(1,n);S_fore(2,n);S_fore(3,n);0];
172
173         [~,Y_fore] = ode45(@SIR1,ts_fore,x0_fore,options);
174
175         S_fore(:,n) = transpose(Y_fore(end,:));%+ stdC*randn(4,1)
              ; % Addition needs to be changed for sample,covariance
176         nu_fore(n) = Y_fore(end,4);
```

```
177
178     end
179
180         s_sol_forecast(:,k-training_period) = S_fore(1,:);
181         i_sol_forecast(:,k-training_period) = S_fore(2,:);
182         r_sol_forecast(:,k-training_period) = S_fore(3,:);
183         c_sol_forecast(:,k-training_period) = S_fore(4,:);
184     % ensemble mean for each iteration:
185     fmean_fore(k-training_period,:) = (1/N)*sum(S_fore,2);
186     gamma_fore = ((S_fore-transpose(fmean_fore(k-training_period
            ,:)))*transpose(S_fore-transpose(fmean_fore(k-
            training_period,:))))/(N-1);
187     fplus_fore(k-training_period,:) = fmean_fore(k-
            training_period,:) + transpose(abs(1.96*sqrt(diag(
            gamma_fore))));
188     fminus_fore(k-training_period,:) = fmean_fore(k-
            training_period,:) + transpose(-abs(1.96*sqrt(diag(
            gamma_fore))));
189
190     if strcmp(casetype,'case 1')== 1
191         yhat_fore = IofT(S(1:2,:),1);
192     elseif strcmp(casetype,'case 2')== 1
193         yhat_fore = IofT(S(1:2,:),0.7);
194     elseif strcmp(casetype, 'case 3')== 1
195         yhat_fore = Integral(nu_fore,1);
```

```
196     elseif strcmp(casetype, 'case 4')== 1
197         yhat_fore = Integral(nu_fore,0.7);
198     else
199         error('Case not found, try again');
200     end
201
202     % Use the above 'case 3' yhat to generate predicted data and
            then
203     % compute metrics:
204
205    % End of Forecast!
206 end
207
208 % Compute performance metrics:
209 RMSEc=sqrt(mean((data_frame(training_period+1:training_period
       +14,2)-fmean_fore(:,4)).^2));
210 MSEc=mean((data_frame(training_period+1:training_period+14,2)-
       fmean_fore(:,4)).^2);
211 MAEc=mean(abs(data_frame(training_period+1:training_period+14,2)-
       fmean_fore(:,4)));
212
213 % define times for plotting
214 time = tdata;
215 time_pred = training_period+1:(training_period + 14);
216 % Test plot of saved solution trajectories of forecasted ODE
```

```
217 % for i = 1:300
218 % plot(c_sol_forecast(i,:))
219 % hold on
220 % end
221
222 % Visualize results
223
224 % Combine solutions from main loop and forecast?
225 c_total_sol = [c_sol c_sol_forecast];
226
227 total_time = [time; time_pred'];
228 total_fmean = [fmean;fmean_fore];
229 %% Main figure output from Kalman filter:
230
231 % Concat to make plots better:
232 c_sol_total = [c_sol c_sol_forecast];
233 fmean_total = [fmean; fmean_fore];
234 fminus_total = [fminus;fminus_fore];
235 fplus_total = [fplus;fplus_fore];
236
237 if ( strcmp(casetype,'case 3')== 1 || strcmp(casetype,'case 4')==
        1 )
238     Graph_incidence=figure;
239     hold on
240     for i = 1:300
```

```
241         plot(data_frame(1:training_period+14,1),c_sol_total(i,:),
               'c')
242     end
243     plot(data_frame(1:training_period+14,1),fmean_total(:,4),'r-'
           ,'LineWidth',2);
244     plot(data_frame(1:training_period+14,1),fplus_total(:,4),'r--
           ','LineWidth',2);
245     plot(data_frame(1:training_period+14,1),fminus_total(:,4),'r
           --','LineWidth',2);
246     xline(training_period-1, '--','LineWidth',2)
247     p=plot(data_frame(1:training_period+14,1),data_frame(1:
           training_period+14,2),'bo');
248     set(p,'LineWidth',2)
249     hold off
250     title('SIR Model (EnKF)');
251     xlabel('Time');
252     ylabel('C''(t) (zika incidence)');
253     set(gca,'FontSize',24);
254     xlim([0,training_period+13])
255     ylim([-5,60])
256 end
257
258 %% Figure I care about for plotting incidence forecast
259 GraphC_pred = figure
260     for i = 1:300
```

```
261     plot(data_frame(training_period+1:training_period+14,1),
           c_sol_forecast(i,:),'c')
262     hold on
263     end
264     plot(data_frame(training_period+1:training_period+14,1),
           fmean_fore(:,4),'r-','LineWidth',2)
265     plot(data_frame(training_period+1:training_period+14,1),
           fplus_fore(:,4),'r--','LineWidth',1);
266     plot(data_frame(training_period+1:training_period+14,1),
           fminus_fore(:,4),'r--','LineWidth',1);
267     p_fore = plot(data_frame(training_period+1:training_period
           +14,1),data_frame(training_period+1:training_period+14,2),
           'bo');
268     set(p_fore,'LineWidth',2)
269     %set(p_fore,'LineWidth',2)
270     xlim([data_frame(training_period+1,1) data_frame(
           training_period+14,1)])
271     xlabel('Time')
272     ylabel('C''(t) (zika incidence)')
273     title('SIR Model (EnKF)')
274     set(gca,'FontSize',16);
275     hold off
276
277 % Above plot has a difference of 1 day in time plotted
278 % Functions to compute observation model predictions
```

```
MAEc
MSEc
RMSEc
%% Functions
function I = IofT(S,rho)
    G = rho*[0 1];
    I = G*S;
end

function I = Integral(nu,rho)
    I = rho*nu;
end
```

Script for SEIR model under EnKF:

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% Script for ENKF of ZIKV Data %%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all, close all
data_frame = importdata('zika-incidence.txt');
training_period = 35; % Set desired training window
zikv_data = data_frame(1:training_period,2);
time_data = data_frame(1:training_period,1);
train_test_data = data_frame(1:(training_period+14),2);
time_vec = data_frame(:,1);
data_vec = data_frame(:,2);

```

```
%Select case type ('case 1' through 'case 4')
casetype = 'case 3';

%Set ensemble size
N = 300;

%Set SIR model parameters
% Make randomized?
global params0
% beta kappa gamma N is the order
% remove comment to select which parameter set to use:
 params0 = [9.35 0.69 8.17 1260]; % SIR day 35
% params0 = [8.79 0.14 4.35 1171]; % SIR day 49
% params0 = [9.62 0.07 3.14 1171]; % SIR day 63
% params0 = [9.64 0.06 1.51 1987]; % SIR day 77
% params0 = [9.62 0.06 2.14 1405]; % SIR day 91
Np = params0(4);

% Use time points from data_frame col 1
tdata = time_data;

% Generate ensemble for INIT
x0 = [Np-1 1];
true_init = x0;

```

```
aS = 0.9*true_init(1);
bS = 1.1*true_init(1);

uniS = aS+(bS-aS).*rand(1,N);

aI = 0.9*true_init(2);
bI = 1.1*true_init(2);

uniI = aI+(bI-aI).*rand(1,N);

% Ensemble Storage
S = zeros(3,N);
S(1,:) = uniS; % for S
S(2,:) = zeros(1,N); % for E
S(3,:) = uniI; % for I

S = [S; zeros(1,N);S(3,:)];

% Ensemble stats

% verify gamma0 matrix
xbar0 = (1/N)*sum(S,2);
gamma0 = ((S-xbar0)*transpose(S-xbar0))/(N-1);

fmean = zeros(length(zikv_data),5);
```

```
63 fplus = zeros(length(zikv_data),5);
64 fminus = zeros(length(zikv_data),5);
65
66 fmean(1,:) = xbar0;
67 fplus(1,:) = transpose(abs(1.96*sqrt(diag(gamma0))))+fmean(1,:);
68 fminus(1,:) = transpose(-abs(1.96*sqrt(diag(gamma0))))+fmean(1,:)
      ;
69
70 % Set up sample covariances
71 stdC = 2;
72 stdD = 1;    % 1
73
74 D = stdD^2;
75 nu = zeros(1,N);
76
77 options = odeset('RelTol',1e-8,'AbsTol',1e-8);
78
79 % Test Model before
80
81 % init = [x0 0 x0(2)]
82 % [~,Y] = ode45(@SIR1,1:1:49,init,options)
83 %
84 % plot(Y(:,4))
85
86 % Kalman Filter loop:
```

```
87 % For training purposes, length of loop will depend on the
       training period
88 % 35,49, ect...
89
90 % Store EnKF solution trajectories for states
91 s_sol = zeros(N,length(zikv_data));
92 e_sol = zeros(N,length(zikv_data));
93 i_sol = zeros(N,length(zikv_data));
94 r_sol = zeros(N,length(zikv_data));
95 c_sol = zeros(N,length(zikv_data));
96 % Update with initial condition of ensemble:
97 s_sol(:,1) = S(1,:);
98 e_sol(:,1) = S(2,:);
99 i_sol(:,1) = S(3,:);
100 r_sol(:,1) = S(4,:);
101 c_sol(:,1) = S(5,:);
102 %
103
104 for j = 2:(length(zikv_data))
105
106     % Prediction step
107     for n = 1:N
108
109         % Timeframe should just based on time index
110         ts = [tdata(j-1),tdata(j)];
```

```
111
112         x0 = [S(1,n);S(2,n);S(3,n);S(4,n);0]; %C is set to zero
                each solution to track incidence
113
114         [~,Y] = ode15s(@SEIR1,ts,x0,options);
115
116         S(:,n) = transpose(Y(end,:))+stdC*randn(5,1);
117         nu(n) = Y(end,5);
118     end
119
120     % Ensemble mean
121     xbar = (1/N)*sum(S,2);
122
123     % Kalman analysis step
124     if strcmp(casetype,'case 1')== 1
125         yhat = IofT(S(1:2,:),1);
126     elseif strcmp(casetype,'case 2')== 1
127         yhat = IofT(S(1:2,:),0.7);
128     elseif strcmp(casetype, 'case 3')== 1
129         yhat = Integral(nu,1);
130     elseif strcmp(casetype, 'case 4')== 1
131         yhat = Integral(nu,0.7);
132     else
133         error('Case not found, try again');
134     end
```

```
135
136     % Kalman Gain and ensemble statistics
137     yhatbar = (1/N)*sum(yhat,2);
138     cross = ((S-xbar)*transpose(yhat-yhatbar))/(N-1);
139     forecast = ((yhat-yhatbar)*transpose(yhat-yhatbar))/(N-1);
140     K = cross/(forecast + D);
141     ydata = zikv_data(j)+stdD*randn(1,N);
142
143     % Update ensemble
144     S = S+ K*(ydata-yhat);
145     s_sol(:,j) = S(1,:);
146     e_sol(:,j) = S(2,:);
147     i_sol(:,j) = S(3,:);
148     r_sol(:,j) = S(4,:);
149     c_sol(:,j) = S(5,:);
150     % Ensemble statistics
151     fmean(j,:) = (1/N)*sum(S,2);
152     gamma = ((S-transpose(fmean(j,:)))*transpose(S-transpose(
            fmean(j,:))))/(N-1);
153
154     fplus(j,:) = transpose(abs(1.96*sqrt(diag(gamma))))+fmean(j
            ,:);
155     fminus(j,:) = transpose(-abs(1.96*sqrt(diag(gamma))))+fmean(j
            ,:);
156
```

```
157 end
158
159 %% Forecast Loop:
160 % INIT forecast from previous step
161 fmean_fore = zeros(14,5);
162 fplus_fore = zeros(14,5);
163 fminus_fore = zeros(14,5);
164 S_fore = S;
165 s_sol_forecast = zeros(N,14);
166 e_sol_forecast = zeros(N,14);
167 i_sol_forecast = zeros(N,14);
168 r_sol_forecast = zeros(N,14);
169 c_sol_forecast = zeros(N,14);
170 for k = training_period+1:(training_period + 14)
171
172     for n = 1:N
173
174         ts_fore = [time_vec(k-1),time_vec(k)];
175
176         x0_fore = [S_fore(1,n);S_fore(2,n);S_fore(3,n);S_fore(4,n
              );0];
177
178         [~,Y_fore] = ode15s(@SEIR1,ts_fore,x0_fore,options);
179
180         S_fore(:,n) = transpose(Y_fore(end,:));%+ stdC*randn(4,1)
```

```
          ; % Error does not need to be added into forecast
181         nu_fore(n) = Y_fore(end,5);
182
183     end
184     % Store forecast solutions
185         s_sol_forecast(:,k-training_period) = S_fore(1,:);
186         e_sol_forecast(:,k-training_period) = S_fore(2,:);
187         i_sol_forecast(:,k-training_period) = S_fore(3,:);
188         r_sol_forecast(:,k-training_period) = S_fore(4,:);
189         c_sol_forecast(:,k-training_period) = S_fore(5,:);
190     % ensemble mean for each iteration:
191     fmean_fore(k-training_period,:) = (1/N)*sum(S_fore,2);
192     gamma_fore = ((S_fore-transpose(fmean_fore(k-training_period
           ,:)))*transpose(S_fore-transpose(fmean_fore(k-
           training_period,:))))/(N-1);
193     fplus_fore(k-training_period,:) = fmean_fore(k-
           training_period,:) + transpose(abs(1.96*sqrt(diag(
           gamma_fore))));
194     fminus_fore(k-training_period,:) = fmean_fore(k-
           training_period,:) + transpose(-abs(1.96*sqrt(diag(
           gamma_fore))));
195
196     if strcmp(casetype,'case 1')== 1
197         yhat_fore = IofT(S(1:2,:),1);
198     elseif strcmp(casetype,'case 2')== 1
```

```
199         yhat_fore = IofT(S(1:2,:),0.7);
200     elseif strcmp(casetype, 'case 3')== 1
201         yhat_fore = Integral(nu_fore,1);
202     elseif strcmp(casetype, 'case 4')== 1
203         yhat_fore = Integral(nu_fore,0.7);
204     else
205         error('Case not found, try again');
206     end
207
208     % Use the above 'case 3' yhat to generate predicted data and
          then
209     % compute metrics:
210
211    % End of Forecast!
212 end
213
214 % Compute performance metrics:
215 RMSEc=sqrt(mean((data_frame(training_period+1:training_period
    +14,2)-fmean_fore(:,5)).^2));
216 MSEc=mean((data_frame(training_period+1:training_period+14,2)-
    fmean_fore(:,5)).^2);
217 MAEc=mean(abs(data_frame(training_period+1:training_period+14,2)-
    fmean_fore(:,5)));
218
219 % define times for plotting
```

```
220 time = tdata;
221 time_pred = training_period+1:(training_period + 14);
222 % Test plot of saved solution trajectories of forecasted ODE
223 % for i = 1:300
224 % plot(c_sol_forecast(i,:))
225 % hold on
226 % end
227
228 % Visualize Results:
229
230 % Combine solutions from main loop and forecast?
231 c_total_sol = [c_sol c_sol_forecast];
232
233 total_time = [time; time_pred'];
234 total_fmean = [fmean;fmean_fore];
235 %% Main figure output from Kalman filter:
236
237 % Concat to make plots better:
238 c_sol_total = [c_sol c_sol_forecast];
239 fmean_total = [fmean; fmean_fore];
240 fminus_total = [fminus;fminus_fore];
241 fplus_total = [fplus;fplus_fore];
242
243 if ( strcmp(casetype,'case 3')== 1 || strcmp(casetype,'case 4')==
        1 )
```

```
244     Graph_incidence=figure;
245     hold on
246     for i = 1:300
247         plot(data_frame(1:training_period+14,1),c_sol_total(i,:),
                'c')
248     end
249     plot(data_frame(1:training_period+14,1),fmean_total(:,5),'r-'
            ,'LineWidth',2);
250     plot(data_frame(1:training_period+14,1),fplus_total(:,5),'r--
            ','LineWidth',2);
251     plot(data_frame(1:training_period+14,1),fminus_total(:,5),'r
            --','LineWidth',2);
252     xline(training_period-1, '--','LineWidth',2)
253     p=plot(data_frame(1:training_period+14,1),data_frame(1:
            training_period+14,2),'bo');
254     set(p,'LineWidth',2)
255     hold off
256     title('SEIR Model (EnKF)');
257     xlabel('Time');
258     ylabel('C''(t) (zika incidence)');
259     set(gca,'FontSize',24);
260     xlim([0,training_period+13])
261     ylim([-5,60])
262 end
263
```

```
%% Figure I care about for plotting incidence forecast
GraphC_pred = figure
    for i = 1:300
    plot(data_frame(training_period+1:training_period+14,1),
       c_sol_forecast(i,:),'c')
    hold on
    end
    plot(data_frame(training_period+1:training_period+14,1),
       fmean_fore(:,5),'r-','LineWidth',2)
    plot(data_frame(training_period+1:training_period+14,1),
       fplus_fore(:,5),'r--','LineWidth',1);
    plot(data_frame(training_period+1:training_period+14,1),
       fminus_fore(:,5),'r--','LineWidth',1);
    p_fore = plot(data_frame(training_period+1:training_period
       +14,1),data_frame(training_period+1:training_period+14,2),
       'bo');
    set(p_fore,'LineWidth',2)
    %set(p_fore,'LineWidth',2)
    xlim([data_frame(training_period+1,1) data_frame(
       training_period+14,1)])
    xlabel('Time')
    ylabel('C''(t) (zika incidence)')
    title('SEIR Model (EnKF)')
    set(gca,'FontSize',16);
    hold off
```

```
282
283 % Above plot has a difference of 1 day in time plotted
284 % Functions to compute observation model predictions
285 MAEc
286 MSEc
287 RMSEc
288 %% Functions
289 function I = IofT(S,rho)
290     G = rho*[0 1];
291     I = G*S;
292 end
293 x
294 function I = Integral(nu,rho)
295     I = rho*nu;
296 end
```