ECAI 2016
G.A. Kaminka et al. (Eds.)

doi:10.3233/978-1-61499-672-9-524

An Improved State Filter Algorithm for SIR Epidemic Forecasting

Weipeng Huang [1] and **Gregory Provan** [1]

Abstract. In epidemic modeling, state filtering is an excellent tool for enhancing the performance of traditional epidemic models. We introduce a novel state filter algorithm to further improve the performance of state-of-the-art approaches based on Susceptible-Infected-Recovered (SIR) models. The proposed algorithm merges two techniques, which are typically used separately: linear correction, as seen in the Ensemble Kalman Filter (EnKF), and resampling, as used in the Particle Filter (PF). We compare the inferential accuracy of our approach against the EnKF and the Ensemble Adjustment Kalman Filter (EAKF), using algorithms employing both an uncentered covariance matrix (UCM) and the standard column-centered covariance matrix (CCM). Our algorithm requires $\mathcal{O}(DN)$ more time than EnKF does, where D is the ensemble dimension and N denotes the ensemble size. We demonstrate empirically that our algorithm with UCM achieves the lowest root-mean-square-error (RMSE) and the highest correlation coefficient (CORR) amongst the selected methods, in 11 out of 14 major real-world scenarios. We show that the EnKF with UCM outperforms the EnKF with CCM, while the EAKF gains better accuracy with CCM in most scenarios.

1 Introduction

Epidemic prediction has a long history, and an early model SIR model [27] has proved essential for accurate forecasting [4, 28]. The SIR model divides the population into three sub-populations: susceptible (S), infected (I) and recovered (R). During the outbreak of an infectious disease, some susceptible individuals will become infected by contact with the infected individuals, and some infected individuals will recover within a certain period of time.

Recently, Shaman and Karspeck, and Yang et al. [34, 44] showed that state filtering methods significantly improve the inferential accuracy of the SIRS-humidity model (which is a variant of SIR) [4]. Yang et al. also empirically demonstrated that EnKF, EAKF and PF have the lowest RMSE [44]. Although the difference in the performance of these three filters is small, EAKF comes out on top, while PF is in bottom place. They examine the model performance concerning 115 cities in the United States (U.S.), using only Google Flu Trends (GFT) data [16].

The EnKF is a Monte Carlo approximation of the Kalman filter, which represents the distribution of the system state using a collection of state vectors, called an ensemble, and replace the KF covariance matrix by the sample covariance computed from the ensemble. The EnKF assumes Gaussian-distributed models, while the PF does not impose that restriction. However, Kalman-type filters require fewer ensemble members (or particles) than the PF to guarantee good performance [36]. Moreover, the EnKF applies linear correction updates to the states to satisfy the Maximum Likelihood; the PF updates its ensemble members from the existing particles by sampling from their weight distributions, where the weights are assigned by the ensemble members' importance (a.k.a importance weighting) [12, 10, 30]. Our algorithm integrates all these techniques, correcting the states with Maximum Likelihood, and updating the ensemble by sampling from the best-performing particles. The algorithm imposes only slight additional time complexity to the EnKF; however, it requires the same ensemble size as the EnKF, provided the improved performance over the two single filters is attained.

[43] shows that the EnKF underestimates the state covariance matrix. Therefore, we compare the model accuracy of the Kalman-type filters empirically, with UCM and CCM. UCM [5] is mostly discussed in Principal Component Analysis (PCA), hence, our use of UCM in the EnKF and the EAKF is novel. Centering the data or keeping it uncentered remains an open question in pattern recognition [20]. There are only two theoretical papers [5, 20] analyzing UCM and CCM, and they both performed eigen-analysis of certain features of both types. We empirically compare these approaches using real-world infection data [7] from the U.S. Center for Disease Control and Prevention (CDC), which contains weekly influenza-like illnesses (ILI) statistics.

Our contributions are as follows. We compare our state filter algorithms with state-of-the-art filters on the nationwide ILI data of 2011-15 and the regional ILI data of 2014-15. The empirical results demonstrate that our approach obtains the optimal RMSE and CORR amongst the examined filters, in 11 out of 14 cases. It also shows that the EnKF and our approach create more accurate predictions with UCM rather than CCM, whereas the EAKF gains better performance with CCM, given the tested scenarios.

In the rest of the paper, Sect. 2 reviews the related work. Sect. 3 elaborates on the models. Following that, Sect. 4 discusses our approach, and then conducts the empirical analysis in Sect. 5. Finally, Sect. 6 concludes the paper.

2 Related Work

Kermack and Mckendrick [27] introduced a key early model for epidemic forecasting, the SIR model. Later, several models were derived from it, such as SI, SIRS, SEIR, SIS, etc. [4, 19]. Researchers have also developed other types of models for computational epidemiology, e.g., agent-based models, meta-population models, spatial models, and stochastic models [35].

Recently, data-driven solutions have shown great promise. Ginsberg et al. [15] used Google search data to build a logistic regression

[1] The Insight Centre for Data Analytics, Department of Computer Science, University College Cork, Cork, Ireland
Email: weipeng.huang@insight-centre.org, gprovan@cs.ucc.ie

model based on the odds ratio of the search-term frequency. Santillana et al. [33, 32] then used the term frequency directly with several machine learning techniques, such as Support Vector Machine, Least Absolute Shrinkage and Selection Operator, and AdaBoost regression etc., to outperform Google's solution. These innovative methods obtain great performance, but at a high computational cost.

Shaman and Karspeck [34] applied the EAKF to the SIRS-humidity model (SIRS-EAKF). SIRS-humidity adds the humidity data as a component to the standard SIRS model, given a correlation between the spread of the epidemic and the humidity levels. Yang et al. [44] then compared a few filtering methods, such as EnKF, EAKF, PF, Maximum Likelihood Filter etc., on the SIRS model. They reported that the EAKF, the EnKF and the PF were the top three performers. Later, [45] estimated a few SIRS-EAKF epidemiological parameter ranges for the seasonal flu and pandemics for a few seasons. We compare the EnKF and the EAKF, with CCM and UCM, while [44] only tested on the generic cEnKF and cEAKF[2].

There are related works connecting the EnKF and the PF. Hoteit et al. [22] introduced the method of combing Kalman correction and resampling, and later simplified the algorithm by removing the unnecessarily complex steps in the resampling circle [21]. [41] then extended Hoteit's method to mixture Gaussian models. Different from the above principle, [14, 9] suggested a strategy that adopts the weighted sum of the posterior states propagated with the EnKF and the PF. Slivinski et al. presented a hybrid filter EnKF-PF for Lagrangian data assimilation [40, 39]. The most complex step in most methods is the covariance matrix approximation; however we performed a comparison between CCM and UCM for deciding the simplest covariance approximation procedure. Our algorithm thereby addresses the overwhelming complexity in the existing approaches.

3 Models

3.1 Notation

In SIR modeling, the population has three sub-categories: susceptible (S), infected (I), and recovered (R). Given the total population M, the percentages of the three sub-groups are $\{s, i, r\}$, where $s = S/M$, likewise for i and r. Lastly, β and γ denote the mean contact rate and the mean recovery rate, respectively.

For the filtering approaches, we first denote the state vector by $x = \begin{bmatrix} s & i \end{bmatrix}^{\mathrm{T}}$. We denote the one-element estimate vector by $y = \begin{bmatrix} i \end{bmatrix}$, the observation by $z = \begin{bmatrix} \hat{i} \end{bmatrix}$, and parameter $\theta = \begin{bmatrix} \beta & \gamma \end{bmatrix}^{\mathrm{T}}$. We use a transition function $f(\cdot)$, and the state to observation mapping matrix, H, to define the following dynamical state space system

$$x_{t+1} = f(x_t, \theta_t) + u_t \tag{1a}$$
$$y_{t+1} = H x_{t+1} + v_t \tag{1b}$$

Moreover, we denote the observed data z_t for time t. In our formulation, $f(\cdot)$ is governed by the SIR dynamics and $H = \begin{bmatrix} 0 & 1 \end{bmatrix}$.

Let $\sim$ denote "distributed according to"; henceforth we assume the noise is zero-mean Gaussian such that $u \sim \mathcal{N}(0, U)$ and $v \sim \mathcal{N}(0, V)$. We define an ensemble as a group of particles, where a particle is a random sample from a certain distribution. The N-ensemble of states, estimates, and parameters are respectively depicted as

$$X_t = \left\{ x_t^{(n)} \right\}^N \qquad Y_t = \left\{ y_t^{(n)} \right\}^N \qquad \Theta_t = \left\{ \theta_t^{(n)} \right\}^N .$$

[2] We add "c" in front of the filter names to indicate the filters using CCM, and use prefix "u" for those using UCM.

Hence, the ensemble version of the dynamical system is:

$$X_{t+1} = f(X_t, \Theta_t) + \left\{ u_t^{(n)} \right\}^N \tag{2a}$$
$$Y_{t+1} = H X_{t+1} + \left\{ v_t^{(n)} \right\}^N \tag{2b}$$

We denote the weights of the ensemble members, by $w_t = \begin{bmatrix} w_t^{(1)} & \dots & w_t^{(N)} \end{bmatrix}^{\mathrm{T}}$. Therefore, for the ensemble or particle based methods, i_t is the expected value of the infection rate (a.k.a. prevalence) at time t, such that

$$y_t = \begin{bmatrix} i_t \end{bmatrix} = H E[X_t] \tag{3}$$

where $E[X]$ returns the mean of X. Let I denote the identity matrix and $\propto$ denote "proportional to". $0_{D \times N}$ refers to a D-by-N matrix of zeros, and 1_N is a length-N vector of ones.

Parameter estimation with KFs. The parameter estimation with the EnKF and the EAKF proceeds by regarding the parameters as augmented states [29, 23, 13, 1]. Specifically, we denote the refined state vector by $\tilde{x}$ and its corresponding ensemble set $\tilde{X}$, such that

$$\tilde{x} = \begin{bmatrix} x \\ \theta \end{bmatrix} \implies \tilde{X} = \begin{bmatrix} X \\ \Theta \end{bmatrix} \tag{4}$$

Given Eq. (1) and (2), we get $\tilde{H} = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix}$. In implementing the KFs, we replace x, X and H by $\tilde{x}$, $\tilde{X}$ and $\tilde{H}$, respectively. In the PF, we use the original x, X and H.

3.2 Suceptible-Infected-Recovered

We select the version of the SIR [28] that uses the ratios s, i, and r. As SIR assumes that the birth and death are negligible to the whole population during a period, the population M is constant and $s_t + i_t + r_t \equiv 1$ holds at any time within a particular period. The model depicts the dynamics by assuming the susceptible individuals become infected with probability β, and infected individuals can recover from the disease with recovery rate γ.

$$\frac{\partial s}{\partial t} = \beta s_t i_t \qquad \frac{\partial i}{\partial t} = \beta s_t i_t - \gamma i_t \qquad \frac{\partial r}{\partial t} = \gamma i_t \tag{5}$$

Apparently, r does not contribute to computing the prevalence i. We thus only present s and i in the state vector x, and omit the equations related to r in the rest of this paper.

3.3 Kalman Filter

The KF is a method that computes the posterior states based on the Maximum Likelihood of a linear Gaussian dynamical system [8, 36, 30]. During each KF round, Eq. (6) and (7) execute a prediction phase, and Eq. (8) to (10) run a correction phase. P denotes the covariance of the states, and K the Kalman Gain matrix. Also, we use a transition matrix B to approximate the transition function $f(\cdot)$. We thus obtain

$$x_{t|t-1} = B x_{t-1|t-1} \tag{6}$$
$$P_{t|t-1} = B P_{t-1|t-1} B^{\mathrm{T}} + U \tag{7}$$
$$K_t = P_{t|t-1} H^{\mathrm{T}} \left(H P_{t|t-1} H^{\mathrm{T}} + V \right)^{-1} \tag{8}$$
$$x_{t|t} = x_{t|t-1} + K_t \left(z_t - H x_{t|t-1} \right) \tag{9}$$
$$P_{t|t} = (I - K_t H) P_{t|t-1} \tag{10}$$

The critical step of a KF is to compute the Kalman Gain K, then combine the observed data to calibrate the state vector x. The KF assumes that the *posteriori* $x_{t|t}$ is more probable as the input for the next estimation than the *priori* $x_{t|t-1}$. Eq. (9) is known as a Kalman-type (or linear) correction step.

3.3.1 Ensemble Kalman Filter

The KF is an optimal filter for linear Gaussian systems with Gaussian noise [8]. For nonlinear systems, researchers have developed the Extended Kalman Filter, the Unscented Kalman Filter, the EnKF and the EAKF, etc. [11, 36, 30]. We consider the EnKF and the EAKF as they require less parameter tuning than the other Kalman Filters.

The EnKF estimates the covariance matrix P, through the sample covariance of the ensemble [11, 25, 12, 13]. Therefore, Eq. (7) and (10) are not used in the EnKF. With the sample covariance denoted by C, we have the Kalman Gain K:

$$K_t = C_{t|t-1}H^{\mathrm{T}}\left(HC_{t|t-1}H^{\mathrm{T}} + V\right)^{-1} \tag{11}$$

We now show the UCM C_c and the CCM C_u are computed using:

$$C_u = \frac{1}{N-1}XX^{\mathrm{T}} \tag{12a}$$

$$C_c = \frac{1}{N-1}(X - \bar{x}1_N^{\mathrm{T}})(X - \bar{x}1_N^{\mathrm{T}})^{\mathrm{T}}, \tag{12b}$$

where the mean vector $\bar{x} = \frac{1}{N}X1_N$. The CCM is the UCM of the ensemble after being centered. It follows that:

$$\begin{aligned} C_c &= \frac{1}{N-1}\left(XX^T - \bar{x}1_N^{\mathrm{T}}X^{\mathrm{T}} - X1_N\bar{x}^{\mathrm{T}} + \bar{x}1_N^{\mathrm{T}}1_N\bar{x}^{\mathrm{T}}\right) \\ &= \frac{1}{N-1}XX^T - \frac{N}{N-1}\bar{x}\bar{x}^{\mathrm{T}} \\ &= C_u - \frac{N}{N-1}\bar{x}\bar{x}^{\mathrm{T}} \end{aligned} \tag{13}$$

Given any $X > 0_{D\times N}$, C_c is strictly smaller than C_u. The sample covariance matrix C_u and C_c both approach the corresponding population covariance matrix asymptotically as N grows, as $\lim_{N\to\infty} N - 1 = N$.

Finally, the EnKF executes the correction as in Eq. (9) to update every prior state particle $x_{t|t-1}^{(n)}$ to the posterior state $x_{t|t}^{(n)}$.

3.3.2 Ensemble Adjustment Kalman Filter

The EAKF adds one more step at each round to improve the EnKF [2, 26]. This filter runs an EnKF round, and then employs a matrix A to further correct the ensemble members such that

$$\hat{x}_{t|t}^{(n)} = A^{\mathrm{T}}\left(x_{t|t-1}^{(n)} - \overline{x}_{t|t-1}\right) + \overline{x}_{t|t} \qquad n = 1\dots N \tag{14}$$

Anderson [2] stated that a number of values for A exist, raising a new problem of choosing A. [34, 44] used $A = 1.03I$ for the GFT data they examined. The research to date mostly selects A based on empirical tests [2, 34, 44, 45]. Ensemble adjustment in the EAKF is superior to the EnKF in preventing the filter divergence caused by the dubiously small prior covariances.

3.4 Particle Filter

A PF [3, 10, 30] is a sequential Monte Carlo method that can perform filtering for arbitrary models. It employs sequential importance sampling and resampling to draw samples from certain distributions, in order to approximate the "true" state variables by a weighted mean that satisfies

$$\mathrm{E}\left[X_t\right] = \sum_{n=1}^{N} w_t^{(n)} x_t^{(n)}. \tag{15}$$

These Monte Carlo methods approximate the true distribution of a state through sampling from a *proposal distributions*. For the simulations, we implement Storvik's PF algorithm [42], instead of the generic PF. At each timestamp, Storvik's PF samples θ_t and x_t from the *proposal distributions* $q_\theta\left(\theta_t^{(n)} \,\middle|\, x_{t-1}^{(n)}, z_t\right)$ and $q_x\left(x_t^{(n)} \,\middle|\, x_{t-1}^{(n)}, z_t, \theta_t^{(n)}\right)$ in sequence. Hence, it normalizes the weights such that for every n,

$$w_t^{(n)} \propto w_{t-1}^{(n)} \frac{p\left(\theta_t \,\middle|\, s_t^{(n)}\right) p\left(z_t \,\middle|\, x_t^{(n)}, \theta_t^{(n)}\right) p\left(x_t^{(n)} \,\middle|\, x_{t-1}^{(n)}, \theta_t^{(n)}\right)}{q_\theta\left(\theta_t^{(n)} \,\middle|\, x_{t-1}^{(n)}, z_t\right) q_x\left(x_t^{(n)} \,\middle|\, x_{t-1}^{(n)}, z_t, \theta_t^{(n)}\right)} \tag{16}$$

where s_t refers to the sufficient statistics for the parameters in the distribution. A sufficient statistic for an unknown parameter in a distribution, is the statistic that provides sufficient information for deciding that parameter. We approximate the required distributions by assuming some known distribution (e.g., Gaussian) rather than using the Markov Chain Monte Carlo, since [24, 6] suggested that a PF with appropriate assumption of distributions can yield better accuracy and far better computing efficiency.

The PF suffers from *degeneracy*, where the significant weights are occupied by a minor portion of the particles [3, 30]. It then uses the quantity of the *effective sample size* S_{eff} to control the resampling switch, where

$$S_{eff} \coloneqq \frac{N}{1 + \mathrm{Var}\left[w_t\right]} \approx \left[\sum_{n=1}^{N}\left(w_t^{(n)}\right)^2\right]^{-1}$$

If S_{eff} is smaller than a certain threshold, it is thought to be suffering from *degeneracy*. In such a case, the PF resamples X_t indirectly by sampling the indices of the states according to the weight distribution w_t. After resampling, all weights will be reset to N^{-1}.

4 Proposed State Filter

Our approach, ensemble adjustment using resampling (BASS), incorporates the Kalman-type (linear) correction, resampling and importance weighting, which prunes the worst-performing particles and weight the particles after every Kalman correction. The resampling helps the ensemble members converge more quickly to the true posterior distributions. The linear correction reduces the ensemble size, and makes the process more tractable. BASS ideally retains a sufficiently large proportion of the states, and the information of them. That is, we conduct partial resampling, in which the particles with negligible weight are replaced by those with large weight. It then naturally protects the process from *degeneracy* if there are sufficiently many ensemble members performing well. To mitigate against *sample impoverishment*, i.e., the loss of diversity amongst the ensemble population [3, 30], we also introduce noise when resampling the states.

4.1 BASS

BASS integrates the EnKF and the PF, and focuses on the state correction step. The correction phase prepares the posterior particles $X_{t|t}$ as input for the next prediction. Since the prior particles $X_{t|t-1}$ (e.g., generated by the SIR model) and the observations z_t are collected, it first runs one EnKF correction and fetches $X_{t|t}$. Next, it proceeds with weighting and partial resampling on $X_{t|t}$, to update $X_{t|t}$ and fetch the weights w_t. The forecasting procedure is shown in Algorithm 1. The EnKF execution is contained in the algorithm BASS (Algorithm 2).

Algorithm 1: FORECASTING$\left(X_{0|0},\ \epsilon,\ H\right)$

1 **Initialization** $w_0 \leftarrow \left\{N^{-1}\right\}^N$
2 **for** $t \leftarrow 1 \dots T$ **do** ▷ $T \leftarrow \infty$ for continuous forecasting
3 $X_{t|t-1} \leftarrow \text{SIR}\left(X_{t-1|t-1}\right)$
4 $\left[i_t\right] \leftarrow H \sum_{n=1}^{N} w_{t-1}^{(n)} x_{t|t-1}^{(n)}$ ▷ the prediction
5 **if** $t \neq T-1$ **then**
6 z_t streams in
7 $\left(X_{t|t},\ w_t\right) \leftarrow \text{BASS}\left(X_{t|t-1}, w_{t-1}, z_t, \epsilon\right)$

Compared with full resampling, partial resampling decreases the computational costs and removes particles with small weights.Partial resampling in BASS uses a global threshold variable, ϵ, and a weight score variable $w_t^{(n)}$ for each particle n. More specifically, the weight represents the normalized likelihood $w_t^{(n)} = p\left(z_{1:t} \,\middle|\, x_{1:t}^{(n)}\right)$. If the particle's weight score is less than the threshold ϵ, we replace it with a randomly picked existing particle with large weight. Consider the system in a Hidden Markov representation, we have

$$\begin{aligned} p\left(z_{1:t} \,\middle|\, x_{1:t}^{(n)}\right) &= p\left(z_{1:t-1} \,\middle|\, x_{1:t}^{(n)}\right) p\left(z_t \,\middle|\, x_{1:t}^{(n)}\right) \\ &= p\left(z_{1:t-1} \,\middle|\, x_{1:t-1}^{(n)}\right) p\left(z_t \,\middle|\, x_t^{(n)}\right) \\ &\propto w_{t-1}^{(n)} p\left(z_t \,\middle|\, x_t^{(n)}\right) \end{aligned}$$

The likelihood also coincides with the chained performance of the particular particle. From existing research [34, 44], we know that there must be state samples that consistently forecast well. Thus, the chained performance over time can be used for filtering out the worst-performing (with small likelihood) particles. Every newly resampled particle inherits the weight score from the root particle. As the weight is also the chained performance, we do not reset the weights back to $\left\{N^{-1}\right\}^N$ as in the PF, thus keeping the historical information of the particles.

BASS is detailed in Algorithm 2. The weights are initialized to a uniform one-sum vector. First, line 2 executes one EnKF execution and returns the posterior states, by ENKF$(\cdot)$. The normalization in line 5 prevents the likelihood from tending towards 0 as time grows. NORM$(\cdot)$ takes a non-negative weight vector and returns a normalized one-sum weight vector, such that the sum of the weights divides each weight. The particles fitting to the observations satisfactorily survive. Hence, we call E (in line 6) the non-survivor set, and w^s (in line 7) the survivors weight set. Normalizing the survivors' weights (in line 8) guarantees the under-performing particles are all replaced. Line 6 to 11 present the partial resampling procedure. It splits the particles into survivors and non-survivors depending on the threshold ϵ, hence it resamples the particles from the survivors (according to their weights) to replace the non-survivors.

Algorithm 2: BASS$\left(X_{t|t-1},\ w_{t-1},\ z_t,\ \epsilon\right)$

1 **Function** BASS$\left(X_{t|t-1},\ w_{t-1},\ z_t,\ \epsilon\right)$
2 $X_{t|t} \leftarrow \text{ENKF}\left(X_{t|t-1},\ z_t\right)$
3 **for** $n \leftarrow 1 \dots N$ **do**
4 $w_t^{(n)} \leftarrow w_{t-1}^{(n)} p\left(z_t \,\middle|\, x_{t|t}^{(n)}\right)$
5 $w_t \leftarrow \text{NORM}\left(w_t\right)$
6 $E \leftarrow \left\{n:\ w_{t-1}^{(n)} < \epsilon,\ n = 1 \dots N\right\}$
7 $w^s \leftarrow \left\{w_{t-1}^{(n)}:\ w_{t-1}^{(n)} \geq \epsilon,\ n = 1 \dots N\right\}$
8 $w^s \leftarrow \text{NORM}\left(w^s\right)$
9 Sample $|E|$ indices $G \sim w^s$ ▷ $|E|$ returns the size of E
10 $w_t^{(E)} \leftarrow w_t^{(G)}$
11 $X_{t|t}^{(E)} \leftarrow X_{t|t}^{(G)} + \left\{u_t^{(g)}\right\}^G$
12 $w_t \leftarrow \text{NORM}\left(w_t\right)$
13 **return** $\left(X_{t|t},\ w_t\right)$

4.1.1 *Time Complexity*

BASS consists of the EnKF and the supplement (resampling and weighting). The time of the supplement for each iteration is in $\mathcal{O}(DN)$, where D is the state dimension and N is the ensemble quantity. In the algorithm, three normalizations in line 5, 8 and 12, force $\mathcal{O}(3 \times 2N)$ steps. Checking and computing the likelihood consumes time in $\mathcal{O}(N)$, from line 3 to 4. Next, drawing the survivors sets also takes time in $\mathcal{O}(N)$, in line 6 and 7. Ideally, resampling (from line 9 to 11) is just for a small portion of the particles, while the worst case costs the time $\mathcal{O}(2N + 2DN)$. Given the assumption D is at least close to 5, the extra time is bounded by $\mathcal{O}(6N + N + N + 2N + 2DN) = \mathcal{O}(DN)$ at each round.

5 Empirical Study

5.1 Experimental Setup

The ILI data [7] records the weekly infection statistics for the U.S., both nationwide and regionally. We select the data of 2011-15, and the 10 regions in 2014-15. Our simulations focus on the forecasts of the infection rate, for the national cases (4 cases) and the regional cases (10 cases) separately. We mainly focus on the RMSE between our predictions and the observations, and also present their CORR.

Initialization. Every initial infection percentage $i_0^{(n)}$, for the n-th particle, is sampled from a uniform distribution $\mathcal{U}(0, b)$, in which b approximately doubles the first observation of the ILI data z, such that the population mean $\mu = \frac{a+b}{2} \approx z_0$. Given $s_0^{(n)} + i_0^{(n)} + r_0^{(n)} = 1$ and $r_0^{(n)} = 0$ for the particle n, we have $s_0^{(n)} = 1 - i_0^{(n)},\ n = 1 \dots N$. Hence, β and γ are sampled from $\mathcal{U}(0, 1)$. The process will resample the state vector when $\beta \leq \gamma$ is detected. The reproduction number is given by $R_0 = \beta/\gamma$, and $R_0 \leq 1$ means there would not be an epidemic outbreak. With respect to the Kalman Filters, we set the noise $r \sim \mathcal{N}(0, 10^{-4} I)$ and $s \sim \mathcal{N}(0, 10^{-4})$. For the EAKF, we pick $A = 1.001I$ for A in Eq. (14). For uBass and cBass, we find that the resampling threshold $\epsilon = 10^{-5}$ is rather robust for all cases. For the PF solution, given Eq. (16), we select the *proposal distributions* according to the setting:

$$\begin{aligned} q_\theta\left(\theta_t^{(n)} \,\middle|\, x_{t-1}^{(n)}, z_t\right) &= p\left(\theta_t \,\middle|\, s_t^{(n)}\right) \\ q_x\left(x_t^{(n)} \,\middle|\, x_{t-1}^{(n)}, z_t, \theta_t^{(n)}\right) &= p\left(x_t^{(n)} \,\middle|\, x_{t-1}^{(n)}, \theta_t^{(n)}\right) \end{aligned}$$

Table 1. Mean RMSE% for the methodologies, with the 0.99 confidence interval within the parentheses. Every approach is repeated 50 times. In each tested situation, the best performer is labeled with bold face.

	2011-12	2012-13	2013-14	2014-15	
cBass	0.202 (0.200, 0.204)	0.469 (0.458, 0.480)	0.376 (0.369, 0.383)	0.594 (0.580, 0.608)	
cEAKF	0.651 (0.637, 0.664)	0.510 (0.498, 0.522)	0.522 (0.508, 0.536)	0.520 (0.512, 0.528)	
cEnKF	0.391 (0.391, 0.391)	1.373 (1.253, 1.494)	1.053 (0.990, 1.115)	1.372 (1.243, 1.501)	
PF	0.449 (0.413, 0.486)	0.689 (0.657, 0.721)	0.569 (0.523, 0.615)	0.677 (0.641, 0.714)	
uBass	0.163 (0.158, 0.168)	**0.406** (0.395, 0.417)	**0.269** (0.260, 0.279)	**0.427** (0.415, 0.440)	
uEAKF	0.562 (0.550, 0.574)	1.005 (0.989, 1.020)	0.847 (0.834, 0.860)	0.925 (0.910, 0.939)	
uEnKF	**0.146** (0.145, 0.146)	0.417 (0.417, 0.417)	0.295 (0.295, 0.296)	0.446 (0.446, 0.447)	
	2014-15 Region 1	2014-15 Region 2	2014-15 Region 3	2014-15 Region 4	2014-15 Region 5
cBass	0.351 (0.337, 0.365)	0.339 (0.330, 0.348)	0.898 (0.873, 0.922)	0.861 (0.833, 0.889)	0.630 (0.609, 0.651)
cEAKF	0.531 (0.520, 0.543)	0.610 (0.597, 0.623)	0.855 (0.845, 0.866)	0.734 (0.723, 0.744)	0.715 (0.697, 0.733)
cEnKF	0.955 (0.955, 0.956)	1.757 (1.732, 1.782)	1.897 (1.897, 1.898)	1.696 (1.695, 1.696)	2.207 (2.132, 2.283)
PF	0.855 (0.677, 1.033)	0.607 (0.569, 0.646)	1.325 (1.259, 1.392)	1.046 (0.825, 1.267)	1.589 (1.353, 1.825)
uBass	**0.276** (0.269, 0.283)	**0.267** (0.260, 0.274)	**0.809** (0.793, 0.825)	**0.718** (0.702, 0.735)	**0.467** (0.453, 0.481)
uEAKF	0.845 (0.833, 0.858)	0.797 (0.781, 0.812)	1.218 (1.206, 1.230)	1.219 (1.204, 1.234)	1.116 (1.102, 1.129)
uEnKF	0.445 (0.441, 0.449)	0.986 (0.934, 1.037)	1.740 (1.660, 1.821)	1.567 (1.392, 1.743)	0.874 (0.779, 0.969)
	2014-15 Region 6	2014-15 Region 7	2014-15 Region 8	2014-15 Region 9	2014-15 Region 10
cBass	0.639 (0.622, 0.655)	**0.462** (0.445, 0.478)	0.462(0.443, 0.481)	0.395 (0.386, 0.403)	0.436 (0.423, 0.449)
cEAKF	**0.506** (0.499, 0.513)	0.455 (0.445, 0.466)	0.477 (0.462, 0.492)	0.617 (0.607, 0.628)	0.402 (0.397, 0.407)
cEnKF	2.690 (2.670, 2.719)	1.473 (1.434, 1.512)	1.096 (1.096, 1.096)	1.157 (1.156, 1.159)	1.086 (1.086, 1.086)
PF	0.934 (0.880, 0.987)	2.012 (1.649, 2.376)	0.818 (0.620, 1.016)	0.681 (0.650, 0.711)	1.138 (0.915, 1.361)
uBass	0.601 (0.556, 0.647)	0.472 (0.461, 0.483)	**0.331** (0.320, 0.342)	**0.341** (0.330, 0.351)	**0.374** (0.366, 0.381)
uEAKF	1.100 (1.086, 1.114)	1.106 (1.094, 1.119)	0.985 (0.969, 1.002)	0.867 (0.848, 0.876)	0.808 (0.792, 0.820)
uEnKF	3.111 (2.171, 4.050)	1.054 (1.039, 1.069)	0.742 (0.721, 0.763)	1.155 (1.103, 1.207)	0.824 (0.795, 0.852)

We also assume that $p\left(x_t^{(n)} \,\middle|\, z_t\right)$ is distributed by Gaussian. The sufficient statistics for the Gaussian $\mathcal{N}(\mu, \sigma^2)$ are the sample mean for μ and sample variance for σ^2, or sample covariance matrix for Σ in the multivariate Gaussian $\mathcal{N}(\mu, \Sigma)$. Finally, let $S_{eff} = N/2$.

Implementation. The program is developed in Python, and is available at https://github.com/weipeng/pyepi.

5.2 Discussion

5.2.1 *Performance Result*

For this task, we find that the CORR is strongly correlated to the RMSE and therefore we focus on the RMSE. We average our results over 50 for each case. In each season and region, we carry out the Analysis of Variance (ANOVA) on the mean RMSE, with the null hypothesis that all methods gain equal RMSE. The data show that for all the scenarios, the null hypothesis is rejected with the P-values all less than $2e{-}16$. A statistical comparison is thought to be significant when its P-value is less than 0.05. We apply the Tukey test to conduct pairwise comparisons of the approaches. The Tukey test is commonly thought to be better than the pairwise t-test, as it embeds the protection to the rise in the risk of Type I error. The statistical analysis is carried out through the built-in functions in the statistical computing language R [31].

Mean RMSE. The mean RMSE result is shown in Table 1 and the Tukey result involving the Bass solutions is shown in Table 2. From 2011-15, uBass, uEnKF and cBass are the top 3 performers in order. The algorithm uBass achieves the optimal prediction accuracy in 3 seasons (2012-15), while uEnKF achieves the optimal accuracy for 2011-12. Also, cBass is the third best performer in the nationwide cases, gaining the third best performance in 3 cases. In 10 regional cases, uBass, cBass and cEAKF are the top three performers in order. The mean RMSE of uBass is significantly lower than that of both cBass and cEAKF in 5 regions, however, demonstrates better accuracy than both cBass and cEAKF in 4 regions (amongst the other 5 regions) although the differences between them are not significant. Our cBass achieves significantly lower RMSE than cEAKF in 2 regions, and lower (but not significantly) RMSE in 7 regions.

RMSE of the Mean Estimates. Fig. 1 displays the RMSE and the CORR of the mean estimates over 50 repetitions. In the RMSE heat-map, the lighter color implies better accuracy. In the CORR heat-map, the darker color implies higher correlation, and red indicates the positive direction while blue indicates the negative. In view of such a situation, uBass gains both the optimal RMSE and CORR in 11 out of 14 cases. Both heat-maps demonstrate uBass is clearly optimal, and cBass is the third across all cases. In conclusion, uBass and cBass consistently perform better in predicting the seasonal influenza level in the U.S.

Confidence Interval. Table 1 exhibits the mean RMSE, with the 99% confidence intervals. Our uBass approach consistently obtains high accuracy, and maintains a small/tight confidence interval gap (gap $< 0.03\%$), except for region 6. Besides, cBass yields the gaps smaller than 0.03% in 13 situations, and cEAKF achieves the tight gaps in all situations. Notice that, uEnKF has the smallest (compared with others) interval gaps for the 4 nationwide predictions, but gets large uncertainty in the regional predictions of 8 regions. However, as a derivation from uEnKF, uBass overcomes the adversity in the regional forecasting simulations.

Fig. 2 illustrates that most approaches have tight confidence intervals even at the 99% level. The PF and cEnKF show the visible intervals for all scenarios, and uEnKF illustrates strong uncertainty in regional scenarios.

Table 2. Tukey test of 0.99 confidence interval for measuring the mean of the RMSE of all approaches on the national influenza level amongst 2011-15 and the regional influenza level in 2014-15. Only the comparisons involving uBass and cBass are presented.

	2011-12		2012-13		2013-14		2014-15	
	◇	P-value	◇	P-value	◇	P-value	◇	P-value
cEAKF-cBass	4.483	0.000	0.132	0.999	1.462	0.000	−0.739	0.098
cEnKF-cBass	1.891	0.000	8.768	0.000	6.768	0.000	7.782	0.000
PF-cBass	2.472	0.000	1.926	0.000	1.932	0.000	0.837	0.037
uBass-cBass	−0.394	0.000	−0.933	0.006	−1.064	0.000	−1.666	0.000
uEAKF-cBass	3.597	0.000	5.081	0.000	4.717	0.000	3.309	0.000
uEnKF-cBass	−0.566	0.000	−0.796	0.033	−0.805	0.000	−1.476	0.000
uBass-cEAKF	−4.878	0.000	−1.065	0.001	−2.526	0.000	−0.927	0.013
uBass-cEnKF	−2.286	0.000	−9.701	0.000	−7.832	0.000	−9.449	0.000
uBass-PF	−2.866	0.000	−2.859	0.000	−2.996	0.000	−2.503	0.000
uEAKF-uBass	3.992	0.000	6.014	0.000	5.781	0.000	4.975	0.000
uEnKF-uBass	−0.172	0.265	0.136	0.998	0.259	0.599	0.190	0.993

	2014-15 Region 1		2014-15 Region 2		2014-15 Region 3		2014-15 Region 4		2014-15 Region 5	
	◇	P-value	◇	P-value	◇	P-value	◇	P-value	◇	P-value
cEAKF-cBass	1.804	0.000	2.705	0.000	−0.422	0.463	−1.273	0.277	0.851	0.685
cEnKF-cBass	6.046	0.000	14.182	0.000	9.997	0.000	8.346	0.000	15.770	0.000
PF-cBass	5.041	0.000	2.681	0.000	4.275	0.000	1.848	0.021	9.586	0.000
uBass-cBass	−0.749	0.357	−0.721	0.000	−0.890	0.001	−1.427	0.158	−1.632	0.038
uEAKF-cBass	4.945	0.000	4.574	0.000	3.203	0.000	3.579	0.000	4.855	0.000
uEnKF-cBass	0.939	0.121	6.466	0.000	8.424	0.000	7.065	0.000	2.436	0.000
uBass-cEAKF	−2.554	0.000	−3.427	0.000	−0.468	0.334	−0.154	1.000	−2.484	0.000
uBass-cEnKF	−6.795	0.000	−14.904	0.000	−10.887	0.000	−9.773	0.000	−17.402	0.000
uBass-PF	−5.790	0.000	−3.402	0.000	−5.165	0.000	−3.275	0.000	−11.218	0.000
uEAKF-uBass	5.694	0.000	5.296	0.000	4.093	0.000	5.006	0.000	6.488	0.000
uEnKF-uBass	1.688	0.000	7.188	0.000	9.314	0.000	8.492	0.000	4.068	0.000

	2014-15 Region 6		2014-15 Region 7		2014-15 Region 8		2014-15 Region 9		2014-15 Region 10	
	◇	P-value	◇	P-value	◇	P-value	◇	P-value	◇	P-value
cEAKF-cBass	−1.325	0.992	−0.063	1.000	0.148	1.000	2.229	0.000	−0.341	0.989
cEnKF-cBass	20.548	0.000	10.117	0.000	6.336	0.000	7.629	0.000	6.499	0.000
PF-cBass	2.949	0.702	15.509	0.000	3.553	0.000	2.863	0.000	7.018	0.000
uBass-cBass	−0.372	1.000	0.105	1.000	−1.314	0.020	−0.540	0.001	−0.624	0.809
uEAKF-cBass	4.612	0.180	6.448	0.000	5.231	0.000	4.672	0.000	3.698	0.000
uEnKF-cBass	24.718	0.000	5.922	0.000	2.799	0.000	7.608	0.000	3.875	0.000
uBass-cEAKF	0.952	0.999	0.168	1.000	−1.462	0.006	−2.769	0.000	−0.283	0.996
uBass-cEnKF	−20.920	0.000	−10.012	0.000	−7.649	0.000	−8.169	0.000	−7.123	0.000
uBass-PF	−3.322	0.571	−15.405	0.000	−4.867	0.000	−3.403	0.000	−7.642	0.000
uEAKF-uBass	4.984	0.114	6.343	0.000	6.544	0.000	5.211	0.000	4.322	0.000
uEnKF-uBass	25.090	0.000	5.817	0.000	4.113	0.000	8.148	0.000	4.500	0.000

◇ the difference ($\times 10^3$)

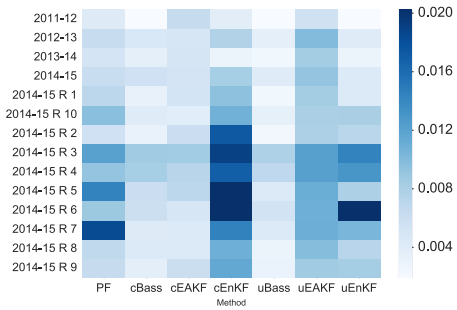
5.2.2 *Curve Fitting*

We describe the **calibration power/capacity** and discuss the curve fitting (in Fig. 2) for each filter in the following paragraphs.

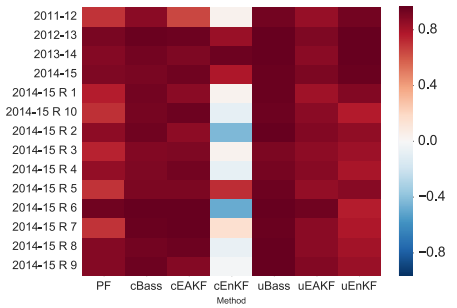
Fig. 2 demonstrates the curve fitting plots for the chosen nationwide and regional cases, including the 99% confidence interval. A Kalman-type filter is thought to have strong calibration power if it raises a big numerical change when correcting the prior states to the posterior states (Eq. (9)). The calibration power is decided by the Kalman Gain and thus decided by the covariance matrix. A calibration that is too strong introduces oscillations in the predicting curve, while a calibration capacity that is too weak fails to make the predictions close to the observed data. Hence a suitable calibration helps the filter forecasts more accurately.

We find that uEnKF works well in the nationwide cases, but achieves relatively large RMSE in 9/10 regional cases. The method cEnKF perceptibly fails to predict the epidemic. In comparison, the capacity of calibration in uEnKF is stronger than that in cEnKF. The predicting curves of uEnKF are oscillating, whereas cEnKF fails to approach the observations, particularly in the regional cases. According to Eq. (13), CCM of the non-negative state ensemble is numerically less than or equal to its UCM. We also find that a numerically small matrix will be influenced by noise easily. However, an excessively strong calibration makes the methods hit the boundaries of the states (e.g. $0 \leq s, i \leq 1$). For instance, in regions 6 and 8, it shows that, the states hitting the bounds will be corrected by the hard limits (Fig. 2), rather than by the filter, distorting the nature of the filtering methods. This problem falls into the category of constrained Kalman Filter [17, 18, 38, 37].

It shows that cEAKF have higher RMSE than uEAKF in all situations. In contrast to EnKF, uEAKF holds a weaker calibration ca-



(a) RMSE



(b) CORR

Figure 1. Heat map of the performance metrics, with the mean estimates of the 50 repeated forecasting simulations. R is short for Region.

pacity. For all scenarios after the epidemic peak, the predictions of uEAKF never manage to approach the observations until the very end of the flu seasons. Additionally, cEAKF converges to the observations slower than uEAKF before the epidemic peak, however outperforms uEAKF after the peak. It implies that the calibration capacity of uEAKF gradually declines over time.

The plots show that the performance of the PF strongly relies on the underlying models. The curve fitting for region 3 and 5 illustrates that the PF is not able to accommodate the fluctuations in the observation curves. [34, 44] availed of the SIRS model with the humidity data as extra features, which is superior to the SIR model. Their model addresses the epidemic peak, or multiple peaks, through the filter or humidity component. For a standalone PF, it is not capable of handling multiple peaks, since the SIR model foremost decides the shape of the prediction curve.

BASS with UCM consistently outperforms that with CCM (in 13 scenarios). The comparison between uBass and cBass follows the similar pattern of that in the EnKF. Both methods fit the observation curves well, but uBass has a more suitable calibration capacity and achieves better inferential accuracy.

5.2.3 *Resampling Size Analysis*

Table 3 exhibits the expected value of the average resampling size over time for the two Bass candidates of the 50 simulations. The two algorithms both resample only a small portion of their particles on average, even when providing accurate forecasts. Amongst all cases, the maximum percentage is merely 13.04% generated by cBass for the region 6 in 2014-15. It also shows that uBass consumes significantly smaller average resampling size per round, compared with cBass (P-values smaller than $10e{-}11$ for 10 cases). Only in region 6 and 10, does cBass resample less (with P-values 1 and 0.75). In our 500-particle simulations, the mean resampling size interval is $[34.67, 67.06]$ for cBass, and $[19.64, 63.54]$ for uBass.

Table 3. Average resampling size in the BASS algorithm of 500 particles, with the error threshold $\epsilon = 10^{-5}$. The 0.99 confidence intervals are shown in the parentheses. R is short for region in the first column, for 2014-15. The rightmost column shows the t-test on the null hypothesis $\mu_1 \leq \mu_2$, where μ_1 and μ_2 are the expectation of average resampling quantities of cBass and uBass respectively.

	average resampling size		$\mu_1 \leq \mu_2$
	cBass	uBass	P-value
2011-12	34.67 (34.34, 35.00)	19.64 (18.44, 20.84)	$< 2e{-}16$
2012-13	52.77 (51.20, 54.22)	35.19 (32.27, 38.11)	$< 2e{-}16$
2013-14	47.98 (46.14, 49.81)	25.60 (23.96, 27.24)	$< 2e{-}16$
2014-15	55.02 (53.15, 56.90)	32.94 (31.07, 34.81)	$< 2e{-}16$
R 1	42.81 (41.75, 43.88)	34.55 (33.23, 35.87)	$< 2e{-}16$
R 2	43.54 (41.62, 45.46)	35.54 (34.06, 37.01)	$2.75e{-}14$
R 3	67.06 (64.08, 70.05)	53.62 (51.16, 56.08)	$2.4e{-}15$
R 4	60.57 (58.27, 62.87)	51.54 (49.61, 53.45)	$1.13e{-}12$
R 5	53.85 (51.33, 56.36)	35.07 (33.13, 37.01)	$< 2e{-}16$
R 6	65.21 (62.17, 68.26)	63.54 (60.37, 66.72)	0.16
R 7	48.69 (46.85, 50.53)	62.43 (60.38, 64.48)	1
R 8	49.96 (48.28, 51.63)	34.04 (32.55, 35.54)	$< 2e{-}16$
R 9	48.73 (47.23, 50.24)	40.01 (38.54, 41.48)	$< 2e{-}16$
R 10	49.55 (48.45, 50.65)	50.02 (48.56, 51.47)	0.75

5.2.4 *Drawback of the Filtering Approach*

The shortcoming of the SIR-filter approaches is a one-week lag for predicting the epidemic peak, although the Kalman filters address the micro details to a certain extent. Shaman et al. [34] and Yang et al. [44] employed the SIRS assimilated with the humidity component. More orthogonal features may add value to the standalone SIR predictions. In future work, we plan to investigate how social content (web searches, Tweets, etc.) could help improve the model by allowing proactive predictions to address the peak timing.

6 Conclusion

This paper proposed an improved SIR-based filter algorithm, BASS, for predicting the seasonal influenza level. It empirically achieves the optimal RMSE and CORR in 11 out of 14 major real-world cases. We also examined UCM and CCM in the BASS, EnKF and EAKF. The experimental results indicate that, in our formulation, the BASS and EnKF perform better with UCM, and it is ideal for the EAKF to utilize CCM. Our future work includes combining social data to further enhance the approach of model and filters. We are also interested in combining the SIRS model with the filters to predict seasonal flu continuously. In addition, we would also like to assess whether our filtering algorithm is applicable to other general problems.

ACKNOWLEDGEMENTS

This research was supported by Science Foundation Ireland (SFI) under Grant Number SFI/12/RC/2289. We thank the reviewers for their insightful comments that helped us improve the paper considerably.

REFERENCES

[1] Saadettin Aksoy, Aydin Muhurcu, and Hakan Kizmaz, 'State and parameter estimation in induction motor using the Extended Kalman Filtering algorithm', *2010 Modern Electric Power Systems*, (3), 1–5, (2010).

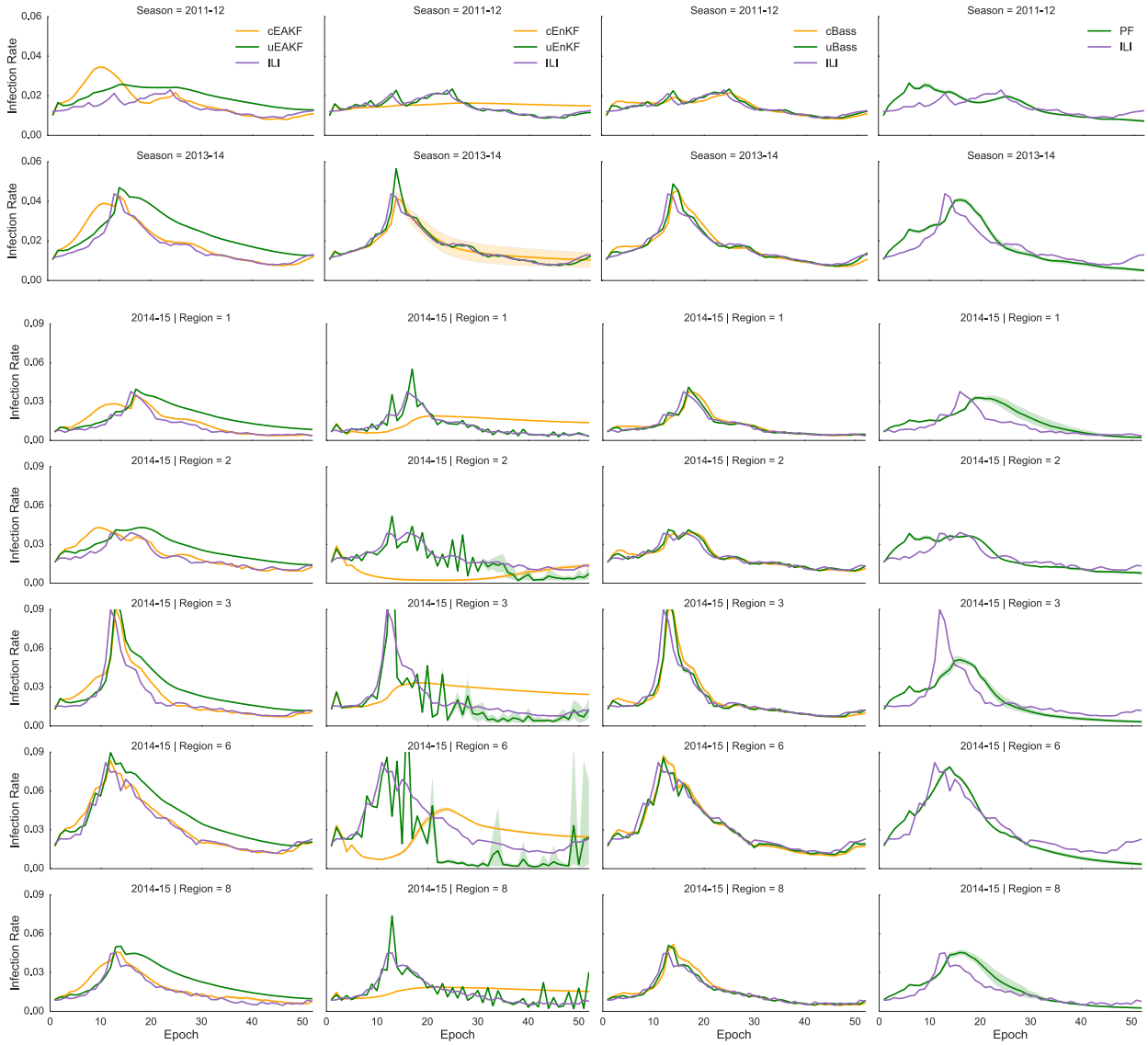


Figure 2. The curve fitting plots for selected national and regional cases. All predictions are with 0.99 confidence interval.

[2] Jeffrey L. Anderson, 'An Ensemble Adjustment Kalman Filter for Data Assimilation', *Monthly Weather Review*, **129**(12), 2884–2903, (2001).

[3] M Sanjeev Arulampalam, Simon Maskell, Neil Gordon, and Tim Clapp, 'A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking', *IEEE Transactions on Signal Processing*, **50**(2), 174–188, (2002).

[4] Norman T.J. Bailey, *The Mathematical Theory of Infectious Diseases and Its Applications*, Books on cognate subjects, Griffin, 1975.

[5] Jorge Cadima and Ian Jolliffe, 'On relationships between uncentred and column-centred principal component analysis', *Pak J Statist*, **25**(4), 473–503, (2009).

[6] Carlos M. Carvalho, Michael S. Johannes, Hedibert F. Lopes, and Nicholas G. Polson, 'Particle Learning and Smoothing', *Statistical Science*, **25**(1), 88–106, (2010).

[7] Centers for Disease Control and Prevention. United States Influenza-like Illnesses Data. http://gis.cdc.gov/grasp/fluview/fluportaldashboard.html.

[8] Zhe Chen, 'Bayesian Filtering: From Kalman Filters to Particle Filters, and Beyond', *Statistics*, **182**(1), 1–69, (2003).

[9] Nawinda Chustagulprom, Sebastian Reich, and Maria Reinhardt, 'A hybrid ensemble transform filter for nonlinear and spatially extended dynamical systems', *ArXiv e-prints*, (sep 2015).

[10] Arnaud Doucet and Am Johansen, 'A tutorial on particle filtering and smoothing: fifteen years later', *Handbook of Nonlinear Filtering*, (December), 656–704, (2011).

[11] Geir Evensen, 'The Ensemble Kalman Filter: Theoretical formulation and practical implementation', *Ocean Dynamics*, **53**(4), 343–367, (2003).

[12] Geir Evensen, *Data assimilation: the ensemble Kalman filter*, Springer Science & Business Media, 2009.

[13] Geir Evensen, 'The ensemble Kalman filter for combined state and parameter estimation', *IEEE Control Systems*, **29**(3), 83–104, (2009).

[14] Marco Frei and Hans R Künsch, 'Bridging the ensemble kalman and particle filters', *Biometrika*, **100**(4), 781–800, (2013).

[15] Jeremy Ginsberg, Matthew H. Mohebbi, Rajan S. Patel, Lynnette Brammer, Mark S. Smolinski, and Larry Brilliant, 'Detecting influenza epidemics using search engine query data.', *Nature*, **457**(7232), 1012–1014, (2009).

[16] Google Inc. Google Flu Trends. https://www.google.org/flutrends.

[17] Nachi Gupta, 'Kalman Filtering in the Presence of State Space Equality', in *Control Conference, 2007. CCC 2007. Chinese*, number 07, pp. 107 – 113. IEEE, (2007).

[18] Nachi Gupta and Raphael Hauser, 'Kalman Filtering with Equality and Inequality State Constraints', *arXiv preprint arXiv:0709.2791*, (07), 26, (2007).

[19] Herbert W. Hethcote, 'The Mathematics of Infectious Diseases', *SIAM Review*, **42**(4), 599–653, (2000).
[20] Paul Honeine, 'An eigenanalysis of data centering in machine learning', *arXiv preprint arXiv:1407.2904*, (2014).
[21] Ibrahim Hoteit, Xiaodong Luo, and Dinh-Tuan Pham, 'Particle Kalman Filtering: A Nonlinear Bayesian Framework for Ensemble Kalman Filters', *Monthly weather review*, **140**(2), 528–542, (2012).
[22] Ibrahim Hoteit, Dinh-Tuan Pham, George Triantafyllou, and Gerasimos Korres, 'A new approximate solution of the optimal nonlinear filter for data assimilation in meteorology and oceanography', *Monthly Weather Review*, **136**(1), 317–334, (2008).
[23] John P. Jensen, *Ensemble Kalman filtering for state and parameter estimation on a reservoir model*, Master's thesis, Norges teknisk-naturvitenskapelige universitet, 2007.
[24] Michael Johannes and Nicholas Polson, 'Particle Filtering and Parameter Learning', *Available at SSRN 983646*, (April 2006), 1–42, (2008).
[25] Craig J. Johns and Jan Mandel, 'A two-stage ensemble kalman filter for smooth data assimilation', *Environmental and Ecological Statistics*, **15**(1), 101–110, (2008).
[26] Alicia R. Karspeck and Jeffrey L. Anderson, 'Experimental implementation of an ensemble adjustment filter for an intermediate ENSO model', *Journal of Climate*, **20**(18), 4638–4658, (2007).
[27] William O. Kermack and Anderson G. McKendrick, 'A contribution to the mathematical theory of epidemics', in *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, volume 115, pp. 700–721. The Royal Society, JSTOR, (1927).
[28] Madhav Marathe and Anil Kumar S. Vullikanti, 'Computational epidemiology', *Communications of the ACM*, **56**(7), 88–96, (July 2013).
[29] Hamid Moradkhani, Soroosh Sorooshian, Hoshin V Gupta, and Paul R. Houser, 'Dual state-parameter estimation of hydrological models using ensemble Kalman filter', *Advances in Water Resources*, **28**(2), 135–147, (2005).
[30] Kevin P. Murphy, *Machine learning: a probabilistic perspective*, MIT press, 2012.
[31] R Development Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2008. ISBN 3-900051-07-0.
[32] Mauricio Santillana, André T. Nguyen, Mark Dredze, Michael J. Paul, Elaine O. Nsoesie, and John S. Brownstein, 'Combining Search, Social Media, and Traditional Data Sources to Improve Influenza Surveillance', *PLOS Computational Biology*, **11**(10), e1004513, (2015).
[33] Mauricio Santillana, Elaine O. Nsoesie, Sumiko R. Mekaru, David Scales, and John S. Brownstein, 'Using Clinicians' Search Query Data to Monitor Influenza Epidemics', *Clinical Infectious Diseases*, **59**(10), 1446–1450, (2014).
[34] Jeffrey Shaman and Alicia Karspeck, 'Forecasting seasonal outbreaks of influenza', *Proceedings of the National Academy of Sciences of the United States of America*, **109**(50), 20425–20430, (2012).
[35] Constantinos I. Siettos and Lucia Russo, 'Mathematical modeling of infectious disease dynamics.', *Virulence*, **4**(4), 295–306, (2013).
[36] Dan Simon, *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*, John Wiley & Sons, 2006.
[37] Dan Simon, 'Kalman filtering with state constraints: a survey of linear and nonlinear algorithms', *IET Control Theory & Applications*, **4**(8), 1303, (2010).
[38] Vincent Sircoulomb, Ghaleb Hoblos, Houcine Chafouk, and José Ragot, 'State estimation under nonlinear state inequality constraints. A tracking application', *2008 16th Mediterranean Conference on Control and Automation*, 1669–1674, (2008).
[39] Laura Slivinski, Elaine Spiller, and Amit Apte, 'A hybrid particle-ensemble kalman filter for high dimensional lagrangian data assimilation', in *Dynamic Data-Driven Environmental Systems Science*, 263–273, Springer, (2015).
[40] Laura Slivinski, Elaine Spiller, Amit Apte, and Björn Sandstede, 'A hybrid particle–ensemble kalman filter for lagrangian data assimilation', *Monthly Weather Review*, **143**(1), 195–211, (2015).
[41] Andreas S. Stordal, Hans A. Karlsen, Geir Nævdal, Hans J. Skaug, and Brice Vallès, 'Bridging the ensemble Kalman filter and particle filters: The adaptive Gaussian mixture filter', *Computational Geosciences*, **15**(2), 293–305, (2011).
[42] Geir Storvik, 'Particle filters for state-space models with the presence of unknown static parameters', *IEEE Transactions on Signal Processing*, **50**(2), 281–289, (2002).
[43] Jeffrey S. Whitaker and Thomas M. Hamill, 'Ensemble data assimilation without perturbed observations', *Monthly Weather Review*, **130**(7), 1913–1924, (2002).
[44] Wan Yang, Alicia Karspeck, and Jeffrey Shaman, 'Comparison of Filtering Methods for the Modeling and Retrospective Forecasting of Influenza Epidemics', *PLoS Computational Biology*, **10**(4), (2014).
[45] Wan Yang, Marc Lipsitch, and Jeffrey Shaman, 'Inference of seasonal and pandemic influenza transmission dynamics', *Proceedings of the National Academy of Sciences*, **112**(9), 201415012, (2015).