<table>
<thead>
<tr>
<th><strong>Title</strong></th>
<th>An improved state filter algorithm for SIR epidemic forecasting</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Author(s)</strong></td>
<td>Huang, Weipeng; Provan, Gregory</td>
</tr>
<tr>
<td><strong>Publication date</strong></td>
<td>2016-01</td>
</tr>
<tr>
<td><strong>Type of publication</strong></td>
<td>Conference item</td>
</tr>
</tbody>
</table>
[http://dx.doi.org/10.3233/978-1-61499-672-9-524](http://dx.doi.org/10.3233/978-1-61499-672-9-524) |
| **Rights** | © 2016, the Authors and IOS Press. This article is published online with Open Access by IOS Press and distributed under the terms of the Creative Commons Attribution Non-Commercial License 4.0 (CC BY-NC 4.0).  
[https://creativecommons.org/licenses/by-nc/4.0/](https://creativecommons.org/licenses/by-nc/4.0/) |
| **Item downloaded from** | [http://hdl.handle.net/10468/3769](http://hdl.handle.net/10468/3769) |

Downloaded on 2020-08-24T20:18:37Z
An Improved State Filter Algorithm for SIR Epidemic Forecasting

Weipeng Huang and Gregory Provan

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 $O(DN)$ 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
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.

There are related works connecting the EnKF and the PF. Hoteit et al. [22] introduced the method of combining 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, while [44] only tested on the generic cEnKF and cEAKF.

Hence, the ensemble version of the dynamical system is:

\[
X_{t+1} = f(X_t, \Theta_t) + \{u_t^{(n)}\}^N \tag{2a}
\]

\[
Y_{t+1} = HX_{t+1} + \{v_t^{(n)}\}^N \tag{2b}
\]

We denote the weights of the ensemble members, by \(w_t = [w_t^{(1)} \ldots w_t^{(N)}]^T\). Therefore, for the ensemble or particle based methods, \(i\) is the expected value of the infection rate (a.k.a. prevalence) at time \(t\), such that

\[
y_t = [i_t] = HE[X_t] \tag{3}
\]

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 \(\tilde{H} = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}\). 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 Susceptible-Infected-Recovered

We select the version of the SIR [28] that uses the ratios \(s, i, r\) 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 \(\beta\), and infected individuals can recover from the disease with recovery rate \(\gamma\).

\[
\frac{\partial s_t}{\partial t} = \beta s_t i_t - \gamma r_t s_t = \gamma i_t
\]

\[
\frac{\partial i_t}{\partial t} = 0
\]

\[
\frac{\partial r_t}{\partial t} = \gamma r_t
\]

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 \(f()\). We thus obtain

\[
x_{t|t-1} = Bx_{t-1|t-1} \tag{6}
\]

\[
P_{t|t-1} = BP_{t-1|t-1}B^T + U \tag{7}
\]

\[
K_t = P_{t|t-1}H^T \left( HP_{t|t-1}H^T + V \right)^{-1} \tag{8}
\]

\[
x_t|t = x_{t|t-1} + K_t (z_t - H x_{t|t-1}) \tag{9}
\]

\[
P_{t|t} = (I - K_t H) P_{t|t-1} \tag{10}
\]
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 $K$:

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

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

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

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

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:

$$C_c = \frac{1}{N-1} \left( XX^T - \bar{x}1_N^T X^T - X1_N \bar{x}^T + \bar{x}1_N 1_N X^T \right)$$

$$= \frac{1}{N-1} X X^T - \frac{N}{N-1} \bar{x} \bar{x}^T$$

$$= C_u - \frac{N}{N-1} \bar{x} \bar{x}^T \tag{13}$$

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 $\lim_{N\to \infty} N = \frac{1}{N} = N$.

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

$$x_{t|t}^{(n)} = A^T \left( x_{t|t-1}^{(n)} - \bar{x}_{t|t-1} \right) + \bar{x}_{t|t} \quad n = 1 \ldots N \tag{14}$$

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

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

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 $\theta_t$ and $x_t$ from the proposal distributions $q_0(\theta_t^{(n)} | x_{t-1}^{(n)}, z_t)$ and $q_x(x_t^{(n)} | x_{t-1}^{(n)}, z_t, \theta_t^{(n)})$ in sequence. Hence, it normalizes the weights such that for every $n$,

$$w_t^{(n)} \propto w_{t-1}^{(n)} p(\theta_t | s_t^{(n)}) p(z_t | x_t^{(n)}, \theta_t^{(n)}) p(x_t^{(n)} | x_{t-1}^{(n)}, \theta_t^{(n)}) p(x_{t-1}^{(n)} | \theta_t^{(n)}) \tag{16}$$

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

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

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 \( X_{t|t-1} \) 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(\( \mathcal{X}_0|t, \epsilon, H \))

1. Initialization \( w_0 \leftarrow \{ N^{-1} \}^N \)
2. for \( t \leftarrow 1 \ldots T \) do \( \triangleright T \leftarrow \infty \) for continuous forecasting
3. \( \left[ t \right] \leftarrow \text{SIR} \left( X_{t-1|t-1} \right) \)
4. \( z_t \leftarrow H \sum_{n=1}^N w_{t-1} x_{t-1|t-1}^{(n)} \) \( \triangleright \) the prediction
5. if \( t \neq T - 1 \) then
6. \( \text{if } z_t \text{ streams in } \) \( X_{t|t-1}, w_t-1, z_t, \epsilon \)
7. \( \text{return } \mathcal{X}_{t|t-1}, w_t \)

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, \( \epsilon \), and a weight score variable \( w_t^{(n)} \) for each particle \( n \). More specifically, the weight represents the normalized likelihood \( w_t^{(n)} = p \left( z_{1:t} \mid x_1^{(n)} \right) \). If the particle’s weight score is less than the threshold \( \epsilon \), we replace it with a randomly picked existing particle with large weight. Consider the system in a Hidden Markov representation, we have:

\[
p \left( z_1:t \mid x_1^{(n)} \right) = p \left( z_{1:t-1} \mid x_1^{(n)} \right) p \left( z_t \mid x_t^{(n)} \right) = p \left( z_{1:t-1} \mid x_1^{(n)} \right) p \left( z_t \mid x_t^{(n)} \right) \propto w_t^{(n)} p \left( z_t \mid x_t^{(n)} \right)
\]

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 \( \{ N^{-1} \}^N \) 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. \( \text{Norm}(\cdot) \) 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 \( w^* \) (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 \( \epsilon \), hence it resamples the particles from the survivors (according to their weights) to replace the non-survivors.

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

1. Function BASS(\( X_{t|t-1}, w_{t-1}, z_t, \epsilon \))
2. \( X_{t|t} \leftarrow \text{EnKF} \left( X_{t|t-1}, z_t \right) \)
3. for \( n \leftarrow 1 \ldots N \) do
4. \( w_t^{(n)} \leftarrow w_{t-1} p \left( z_t \mid x_t^{(n)} \right) \)
5. \( w_t \leftarrow \text{Norm} \left( w_t \right) \)
6. \( E \leftarrow \{ n : w_t^{(n)} < \epsilon, n = 1 \ldots N \} \)
7. \( w^* \leftarrow \{ w_t^{(n)} : w_t^{(n)} \geq \epsilon, n = 1 \ldots N \} \)
8. \( w^* \leftarrow \text{Norm} \left( w^* \right) \)
9. Sample \( |E| \) indices \( G \sim w^* \triangleright |E| \) returns the size of \( E \)
10. \( w_t^{(G)} \leftarrow w_t^{(G)} \)
11. \( X_t^{(G)} \leftarrow X_{t|t}^{(G)} + \{ u_t^{(g)} \}^G \)
12. \( w_t \leftarrow \text{Norm} \left( w_t \right) \)
13. \( \text{return } X_{t|t}, w_t \)

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 \( \mathcal{O}(6N + N + 2N + 2DN) = \mathcal{O}(DN) \) 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 nationwide 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.

Initialisation. Every initial infection percentage \( t_0^{(n)} \), 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 \( \mu \approx \frac{2b}{b} \approx 2z_0 \). Given \( s_0^{(n)} + r_0^{(n)} + i_0^{(n)} = 1 \) and \( r_0^{(n)} = 0 \) for the particle \( n \), we have \( s_0^{(n)} = 1 - i_0^{(n)} \), \( n = 1 \ldots N \). Hence, \( \beta \) and \( \gamma \) 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 \( R_0 = \frac{\beta}{\gamma} \), 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 \( s \sim \mathcal{N}(0, 10^{-4} I) \). 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:

\[
q_s \left( \theta_t^{(n)} \mid x_{t-1|t}^{(n)}, z_t \right) = p \left( \theta_t \mid s_t^{(n)} \right)
\]

\[
q_c \left( x_t^{(n)} \mid x_{t-1|t}^{(n)}, z_t, \theta_t^{(n)} \right) = p \left( x_t^{(n)} \mid x_{t-1|t}^{(n)}, \theta_t^{(n)} \right)
\]
We also assume that $p(z_i | x_i^{(n)})$ is distributed by Gaussian. The sufficient statistics for the Gaussian $\mathcal{N}(\mu, \sigma^2)$ are the sample mean for $\mu$ and sample variance for $\sigma^2$, or sample covariance matrix for $\Sigma$ in the multivariate Gaussian $\mathcal{N}(\mu, \Sigma)$. Finally, let $S_{x_f} = N/2$.

**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 for each case. In each season and region, we carry out the 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, 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 heatmap, the lighter color implies better accuracy. In the CORR heatmap, 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.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>cEAKF-cBass</td>
<td>4.483</td>
<td>0.000</td>
<td>0.132</td>
<td>0.099</td>
</tr>
<tr>
<td>cEnKF-cBass</td>
<td>1.891</td>
<td>0.000</td>
<td>8.768</td>
<td>0.000</td>
</tr>
<tr>
<td>PF-cBass</td>
<td>2.472</td>
<td>0.000</td>
<td>1.926</td>
<td>0.000</td>
</tr>
<tr>
<td>uBass-cBass</td>
<td>−0.394</td>
<td>0.000</td>
<td>−0.933</td>
<td>0.006</td>
</tr>
<tr>
<td>uEAKF-cBass</td>
<td>3.597</td>
<td>0.000</td>
<td>5.081</td>
<td>0.000</td>
</tr>
<tr>
<td>uEnKF-cBass</td>
<td>−0.566</td>
<td>0.000</td>
<td>−0.796</td>
<td>0.033</td>
</tr>
<tr>
<td>uBass-cEAKF</td>
<td>−4.878</td>
<td>0.000</td>
<td>−1.065</td>
<td>0.001</td>
</tr>
<tr>
<td>uBass-cEnKF</td>
<td>−2.286</td>
<td>0.000</td>
<td>−0.701</td>
<td>0.000</td>
</tr>
<tr>
<td>uBass-PF</td>
<td>−2.866</td>
<td>0.000</td>
<td>−2.859</td>
<td>0.000</td>
</tr>
<tr>
<td>uEAKF-uBass</td>
<td>3.992</td>
<td>0.000</td>
<td>6.014</td>
<td>0.000</td>
</tr>
<tr>
<td>uEnKF-uBass</td>
<td>−0.172</td>
<td>0.265</td>
<td>0.136</td>
<td>0.998</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>cEAKF-cBass</td>
<td>1.804</td>
<td>0.000</td>
<td>2.705</td>
<td>0.000</td>
<td>0.422</td>
</tr>
<tr>
<td>cEnKF-cBass</td>
<td>6.046</td>
<td>0.000</td>
<td>14.182</td>
<td>0.000</td>
<td>9.997</td>
</tr>
<tr>
<td>PF-cBass</td>
<td>5.041</td>
<td>0.000</td>
<td>2.681</td>
<td>0.000</td>
<td>4.275</td>
</tr>
<tr>
<td>uBass-cBass</td>
<td>−0.749</td>
<td>0.357</td>
<td>−0.721</td>
<td>0.000</td>
<td>−0.890</td>
</tr>
<tr>
<td>uEAKF-cBass</td>
<td>4.945</td>
<td>0.000</td>
<td>4.574</td>
<td>0.000</td>
<td>3.203</td>
</tr>
<tr>
<td>uEnKF-cBass</td>
<td>0.939</td>
<td>0.121</td>
<td>6.466</td>
<td>0.000</td>
<td>8.424</td>
</tr>
<tr>
<td>uBass-cEAKF</td>
<td>−2.554</td>
<td>0.000</td>
<td>−3.427</td>
<td>0.000</td>
<td>−0.468</td>
</tr>
<tr>
<td>uBass-cEnKF</td>
<td>−6.795</td>
<td>0.000</td>
<td>−14.904</td>
<td>0.000</td>
<td>−10.887</td>
</tr>
<tr>
<td>uBass-PF</td>
<td>−5.790</td>
<td>0.000</td>
<td>−3.402</td>
<td>0.000</td>
<td>−5.165</td>
</tr>
<tr>
<td>uEAKF-uBass</td>
<td>5.694</td>
<td>0.000</td>
<td>5.296</td>
<td>0.000</td>
<td>4.093</td>
</tr>
<tr>
<td>uEnKF-uBass</td>
<td>1.688</td>
<td>0.000</td>
<td>7.188</td>
<td>0.000</td>
<td>9.314</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>cEAKF-cBass</td>
<td>−1.325</td>
<td>0.992</td>
<td>−0.063</td>
<td>1.000</td>
<td>0.148</td>
</tr>
<tr>
<td>cEnKF-cBass</td>
<td>20.548</td>
<td>0.000</td>
<td>10.117</td>
<td>0.000</td>
<td>6.336</td>
</tr>
<tr>
<td>PF-cBass</td>
<td>2.949</td>
<td>0.702</td>
<td>15.509</td>
<td>0.000</td>
<td>3.553</td>
</tr>
<tr>
<td>uBass-cBass</td>
<td>−0.372</td>
<td>1.000</td>
<td>0.105</td>
<td>1.000</td>
<td>−1.314</td>
</tr>
<tr>
<td>uEAKF-cBass</td>
<td>4.612</td>
<td>0.180</td>
<td>6.448</td>
<td>0.000</td>
<td>5.231</td>
</tr>
<tr>
<td>uEnKF-cBass</td>
<td>24.718</td>
<td>0.000</td>
<td>5.922</td>
<td>0.000</td>
<td>2.799</td>
</tr>
<tr>
<td>uBass-cEAKF</td>
<td>0.952</td>
<td>0.999</td>
<td>0.168</td>
<td>1.000</td>
<td>−1.462</td>
</tr>
<tr>
<td>uBass-cEnKF</td>
<td>−20.920</td>
<td>0.000</td>
<td>−10.012</td>
<td>0.000</td>
<td>−7.649</td>
</tr>
<tr>
<td>uBass-PF</td>
<td>−3.322</td>
<td>0.571</td>
<td>−15.405</td>
<td>0.000</td>
<td>−4.867</td>
</tr>
<tr>
<td>uEAKF-uBass</td>
<td>4.984</td>
<td>0.114</td>
<td>6.343</td>
<td>0.000</td>
<td>6.544</td>
</tr>
<tr>
<td>uEnKF-uBass</td>
<td>25.090</td>
<td>0.000</td>
<td>5.817</td>
<td>0.000</td>
<td>4.113</td>
</tr>
</tbody>
</table>

P-value

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 perceptively 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 ≤ s, i ≤ 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-
achieves better inferential accuracy. 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 $\mu_1$ and $\mu_2$ are the expectation of average resampling quantities of cBass and uBass respectively.

<table>
<thead>
<tr>
<th>average resampling size</th>
<th>$\mu_1 \leq \mu_2$</th>
<th>P-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>cBass</td>
<td>uBass</td>
<td></td>
</tr>
<tr>
<td>2011-12</td>
<td>34.07 (34.34, 35.00)</td>
<td>19.64 (18.44, 20.84)</td>
</tr>
<tr>
<td>2012-13</td>
<td>52.77 (51.20, 54.22)</td>
<td>35.19 (32.27, 38.11)</td>
</tr>
<tr>
<td>2013-14</td>
<td>47.98 (46.14, 49.81)</td>
<td>25.60 (23.96, 27.24)</td>
</tr>
<tr>
<td>2014-15</td>
<td>55.02 (53.15, 56.90)</td>
<td>32.94 (31.07, 34.81)</td>
</tr>
<tr>
<td>R 1</td>
<td>42.81 (41.75, 43.88)</td>
<td>34.55 (33.23, 35.87)</td>
</tr>
<tr>
<td>R 2</td>
<td>43.54 (41.62, 45.46)</td>
<td>35.54 (34.06, 37.01)</td>
</tr>
<tr>
<td>R 3</td>
<td>67.06 (64.08, 70.05)</td>
<td>53.62 (51.16, 56.08)</td>
</tr>
<tr>
<td>R 4</td>
<td>60.57 (58.27, 62.87)</td>
<td>51.54 (49.61, 53.45)</td>
</tr>
<tr>
<td>R 5</td>
<td>53.85 (51.33, 56.36)</td>
<td>53.07 (33.13, 37.01)</td>
</tr>
<tr>
<td>R 6</td>
<td>65.21 (62.17, 68.26)</td>
<td>63.54 (60.37, 66.72)</td>
</tr>
<tr>
<td>R 7</td>
<td>48.69 (46.85, 50.53)</td>
<td>62.43 (60.38, 64.48)</td>
</tr>
<tr>
<td>R 8</td>
<td>49.96 (48.28, 51.63)</td>
<td>34.04 (32.55, 35.54)</td>
</tr>
<tr>
<td>R 9</td>
<td>48.73 (47.23, 50.24)</td>
<td>40.01 (38.54, 41.48)</td>
</tr>
<tr>
<td>R 10</td>
<td>49.55 (48.45, 50.65)</td>
<td>50.02 (48.56, 51.47)</td>
</tr>
</tbody>
</table>

### 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 with UCM consistently outperforms that with CCM. 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

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

[43] Jeffrey S. Whitaker and Thomas M. Hamill, ‘Ensemble data assimila-