Fast Mining and Forecasting of Co-evolving Epidemiological Data Streams

Tasuku Kimura
SANKEN, Osaka University
Osaka, Japan
tasuku@sanken.osaka-u.ac.jp

Yasuko Matsubara
SANKEN, Osaka University
Osaka, Japan
yasuko@sanken.osaka-u.ac.jp

Koki Kawabata
SANKEN, Osaka University
Osaka, Japan
koki@sanken.osaka-u.ac.jp

Yasushi Sakurai
SANKEN, Osaka University
Osaka, Japan
yasushi@sanken.osaka-u.ac.jp

ABSTRACT

Given a large, semi-infinite collection of co-evolving epidemiological data containing the daily counts of cases/deaths/recovered in multiple locations, how can we incrementally monitor current dynamical patterns and forecast future behavior? The world faces the rapid spread of infectious diseases such as SARS-CoV-2 (COVID-19), where a crucial goal is to predict potential future outbreaks and pandemics, as quickly as possible, using available data collected throughout the world. In this paper, we propose a new streaming algorithm, EpiCast, which is able to model, understand and forecast dynamical patterns in large co-evolving epidemiological data streams. Our proposed method is designed as a dynamic and flexible system, and is based on a unified non-linear differential equation. Our method has the following properties: (a) *Effective*: it operates on large co-evolving epidemiological data streams, and captures important world-wide trends, as well as location-specific patterns. It also performs real-time and long-term forecasting; (b) *Adaptive*: it incrementally monitors current dynamical patterns, and also identifies any abrupt changes in streams; (c) *Scalable*: our algorithm does not depend on data size, and thus is applicable to very large data streams. In extensive experiments on real datasets, we demonstrate that EpiCast outperforms the best existing state-of-the-art methods as regards accuracy and execution speed.

CCS CONCEPTS

• **Information systems** → **Data stream mining**; • **Mathematics of computing** → **Nonlinear equations**.

KEYWORDS

Data streams; Time series; Epidemics; Non-linear dynamical systems; Tensor Data analysis; Real-time forecasting

KDD '22, August 14–18, 2022, Washington, DC, USA

ACM ISBN 978-1-4503-9385-0/22/08…$15.00
https://doi.org/10.1145/3534678.3539078

ACM Reference Format:
Tasuku Kimura, Yasuko Matsubara, Koki Kawabata, and Yasushi Sakurai. 2022. Fast Mining and Forecasting of Co-evolving Epidemiological Data Streams. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '22), August 14–18, 2022, Washington, DC, USA.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3534678.3539078

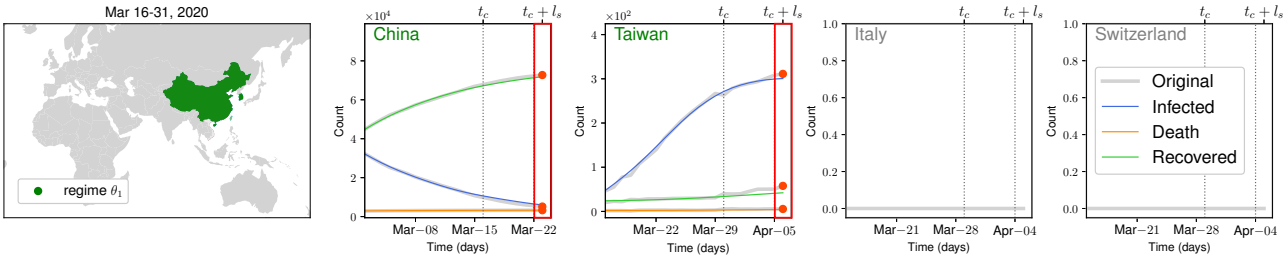
1 INTRODUCTION

A new viral epidemic, COVID-19, has spread across the world [1, 46], and it has strongly affected aspects of human life such as ways of working and communicating [8, 34]. A significant interest for us in relation to making better decisions is to obtain more accurate estimates regarding potential future outbreaks and pandemics since if we know the number of people who will be infected in the future, we can manage pandemic risks in advance by, for example, avoiding frequent social activities and controlling hospital occupancy rates [6]. In a specific location, however, the problem is that we must make estimates to design countermeasures without having sufficient data. So, how can we find useful patterns that help us to forecast future phenomena in co-evolving epidemics? As we receive new data, how can we efficiently incorporate dynamical patterns found in different locations/countries into forecasting?

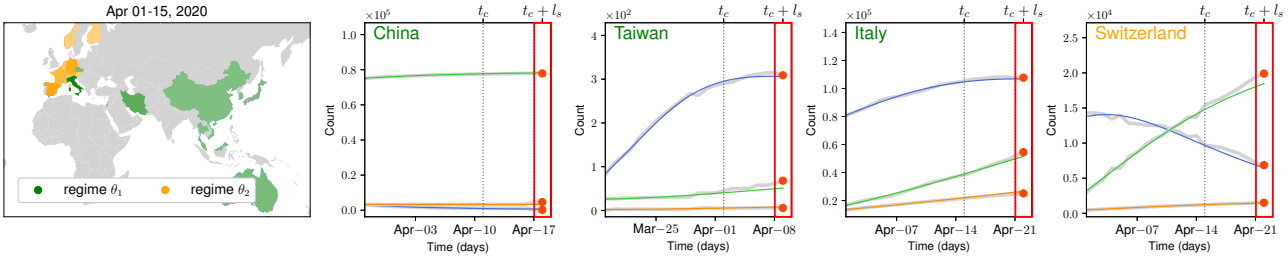
In this paper, we focus on an important problem, namely, the real-time modeling and forecasting of co-evolving epidemiological data streams, and specifically, we present EpiCast [4]. Intuitively, the problem we wish to solve is as follows:

Informal Problem 1. ***Given*** *an epidemiological data stream* $\mathcal{X} = \{X_1, \cdots, X_t, \cdots, X_{t_c}, \cdots\}$*, where* t_c *increases with every new time point, and each entry* $X_t = \{x_{ij}(t)\}_{i,j=1}^{d,r}$ *describes* d*-dimensional vectors/observations in* r *locations at time point* t*,* ***find*** *global and local-level representative patterns, and then* ***forecast*** l_s*-steps-ahead outbreaks, continuously, in a streaming fashion.*

Importance of streaming mining of co-evolving epidemics. Today, our connected and open society allows us to access public health surveillance reports and statistics, such as the daily number of cases, deaths and recovered in each location, and they are continuously updated and shared at a national/regional level [2, 3]. Given such a collection of open epidemic data from multiple locations, how can we find important patterns and rules, and forecast future outbreaks and pandemics? If we already have sufficient data, we



(a) Snapshots of EpiCast for the initial spread of COVID-19 in Asia (Mar 16-31, 2020)



(b) Snapshots of EpiCast for the spreading process of COVID-19 from Asia to Europe (April 1-15, 2020)

Figure 1: Modeling and forecasting power of EpiCast for real COVID-19 data streams: our method provides real-time forecasts based on non-linear equations that capture both global and location-specific epidemic patterns, e.g., (a) the initial spread of COVID-19 in Asia (e.g., China and Taiwan), and (b) the spreading process from Asia to Europe and other areas.

could try using basic time-series analytical tools, e.g., ARIMA, or some recent deep neural network (DNN) based approaches [14, 21]. However, these existing methods basically cannot handle streaming data collections, and thus cannot incrementally train/update model parameters. Besides, the DNN-based approaches require large amounts of input/training data in advance. So, can we do better than the existing methods?

This is precisely the question behind our work. We want a new, streaming method that can continuously monitor the changing situations of epidemics, and estimate/update current dynamical patterns, efficiently and adaptively. Here, most importantly, we want to handle *non-stationary* and *spatially co-evolving* time-series data. As we will discuss further in section 2, a new viral epidemic, COVID-19 is repeatedly mutating and changing its properties over time, and transforming from one country to another, which makes the process by which it spreads more complex. Thus, the ideal method should be able to monitor the current epidemic situation, and dynamically and adaptively capture the latent relationships and interactions between multiple locations.

Preview of our results. Here, we briefly show that our method is capable of modeling and forecasting epidemiological data streams. Figure 1 shows a running example of EpiCast with real COVID-19 data streams. The example consists of $d = 3$ dimensional sequences, which correspond to the daily number of current infections (blue line), the total number of confirmed recoveries (green line), and the total death toll (orange line). Here, the left column shows EpiCast-Map, which illustrates typical/representative epidemic patterns (i.e., $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2$) in each location/country (hereafter we refer to such patterns as "regimes"). The right columns show snapshots of seven-days-ahead forecasting in four different locations (i.e., China, Taiwan, Italy and Switzerland), at each time point t_c. Specifically, the black vertical line shows the current time point t_c, and the red rectangle shows the seven-days-ahead forecasted values, where the solid colored lines show the estimated values, and the gray lines and red points show the original/actual/observed values.

As shown in Figure 1, our method successfully captures current non-linear dynamical patterns of co-evolving epidemics and generates long-term forecasts. For example, EpiCast captured (a) the initial spread of COVID-19 in Asia from Mar 16 to 31, 2020 , and (b) the process by which the infection spread from Asia to Europe and other areas from April 1 to 15, 2020. In addition, the snapshot figures indicate that our method successfully forecast seven-days-ahead future values in different locations. This is because our method has certain desirable properties, namely, (**P1**) non-linear modeling over complex epidemic streams, and (**P2**) a model-sharing mechanism among multiple locations (both described in section 4).

Contributions. In this paper, we focus on an important problem, namely, the real-time modeling and forecasting of co-evolving epidemiological data streams, and we present EpiCast, which has the following desirable properties:

(1) **Effective**: it captures important world-wide dynamical epidemic trends, as well as location-specific patterns, in given data streams, and performs long-range forecasts.
(2) **Adaptive**: it can continuously and adaptively capture current dynamical patterns, and describe how emerging viruses would spread into neighbouring locations over time.
(3) **Scalable**: it is designed as a streaming algorithm that allows us to determine important non-linear epidemic patterns and generate future predictions, within a constant time.

We demonstrate the effectiveness and efficiency of our method using real world datasets and show that it outperforms existing methods in terms of both accuracy and scalability (please see section 5).

Outline. The rest of the paper is organized in the conventional way: next we describe related work, before moving on to our proposed model, algorithms, experiments and conclusions.

2 RELATED WORK

We provide a survey of the related literature, which falls broadly into two categories: (1) epidemiology and (2) time series analysis and data stream mining.

Epidemiology. SARS-CoV-2 (COVID-19) is still spreading rapidly. One of the factors behind this is that COVID-19 has a long incubation period between infection and the onset of the disease and a short latent period, which is the time from infection to infectiousness [15, 20, 22]. If the latent period is shorter than the incubation period, the infected person can spread the virus without any signs or symptoms [31]. Research has been conducted into COVID-19 countermeasures. The entire COVID-19 genome was identified in February 2020 [48, 56]. Although the coronavirus is believed to be resistant to mutation because of its RNA proofreading function [29], it has been pointed out that it mutates when the number of replication attempts increases due to the expansion of the infection period [16]. COVID-19 repeatedly mutates and changes its properties, making the spreading process more complex. For example, D614G [19] , which started to spread in March 2020, is a missense mutation that affects the spike protein of COVID-19. As of January 2022, COVID-19 variants of concern (VOCs) continue to appear in the world [47] ; for example, a total of 139 countries have reported B.1.1.7 (first detected in the United Kingdom), 87 countries B.1.351 (first detected in South Africa), and 54 countries P.1 (first detected in Brazil and Japan). In addition, because the influence of a virus [5] and the efficacy of the vaccine [24] vary depending on the biological characteristics of the individual, such as the ABO blood group or the biochemistry of population groups resulting from ethnic or genetic factors, epidemic patterns may differ even in locations where the same variant of COVID-19 has been confirmed. The combination of these factors further complicates the epidemic patterns of COVID-19. As a consequence, COVID-19 has the characteristic of mutating over time thus altering the behavior of the epidemic patterns and changing the epidemic patterns depending on the combination of locations and variants.

Time series analysis and data stream mining. Time-series data analysis and engineering is an important topic that has attracted huge interest in many fields [13, 25, 33, 38, 45, 53]. Conventional methods include autoregression (AR) and the Kalman filter (KF) [12], and they have generated a wide range of extensions [37, 39, 40]. In addition to non-linear models [27, 42] for more general problem settings where we have fewer assumptions for a dataset, we can apply domain knowledge to a model by choosing an appropriate non-linear differential equation [28, 41], which enables us to forecast complex dynamics even when it has not been observed in recent/historical data. Online and streaming algorithms have become more important in terms of processing and analyzing large amounts of data under time/memory limitations [9, 23, 26, 51, 52, 55]. The emergence of novel viruses also requires real-time forecasting [43, 44], where the critical goal is to forecast the peaks of pandemics. But unfortunately, they did not discuss scalability in relation to learning their proposed models.

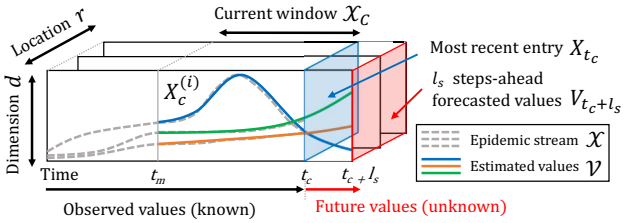


Figure 2: Illustration of real-time forecasting over an epidemiological data stream: Given an epidemic stream $\mathcal{X} = \{X_1, \cdots, X_{t_c}, \cdots\}$, where $X_{t_c} \in \mathbb{N}^{d \times r}$ is the most recent entry at time point t_c, it incrementally maintains the current window $\mathcal{X}_C = \{X_t\}_{t=t_m}^{t_c}$, and captures co-evolving epidemic patterns in $\mathcal{X}_C$. It then forecasts the l_s-steps-ahead future values, i.e., $V_{t_c+l_s} \in \mathbb{N}^{d \times r}$.

The deep neural network (DNN) has become an alternative way to obtain high dimensional time-domain features and forecast future phenomena in various contexts [10, 14, 17, 35, 50, 54]. TCN [21] is a temporal convolutional network, which can learn multi-level temporal causality, and train faster than RNNs. EpiDeep [7], which represents some of the most recent research in this regard, succeeded in applying DNN to modeling the dynamics of a seasonally occurring infection over the weighted influenza-like illness (wILI) datasets. However, these DNN-based methods are still insufficient for modeling atypical spreads of emerging viruses because they need appropriate data to regularize their large number of parameters. Please also see Appendix A for further discussion of COVID-related data analysis and forecasting.

As a consequence, none of them can handle all the requirements, namely capturing non-linear behavior in epidemic streams, mining infectious patterns between multiple locations and adaptive real-time forecasting.

3 PROPOSED MODEL

In this section, we propose our model for co-evolving epidemiological data streams. We begin by introducing our formal problem definition, and then describe our model in detail.

We assume that we have a semi-infinite collection of epidemiological data, namely an epidemic stream $\mathcal{X} = \{X_1, \cdots, X_t, \cdots, X_{t_c}, \cdots\}$, where t_c indicates the current time point (i.e., the total duration of the stream), and t_c increases with every new time point. We can obtain a new observation X_{t_c+1} at every time point and thus the total size of $\mathcal{X}$ increases. Each entry $X_t = \{x_{ij}(t)\}_{i,j=1}^{d,r}$ describes d-dimensional vectors/observations in r locations at time point t. In this paper, we set $d = 3$, corresponding to the daily number of current infections, the total number of confirmed recoveries, and the total death toll. Thus, we can treat this set of $d \times r$ epidemic sequences as a 3rd-order tensor stream, i.e., $\mathcal{X} \in \mathbb{N}^{d \times r \times t_c}$.

To make it possible to forecast future epidemic phenomena in a streaming fashion, where we are restricted from accessing all the observations due to time and memory limitations, we define a

small time window so that we have only recent data in $\mathcal{X}$, as described in Figure 2. More specifically, we define $\mathcal{X}_C = \{X_t\}_{t=t_m}^{t_c}$ as the current window that contains the most recent series of an epidemic stream, where t_m indicates the starting time point of the current window. Similarly, $V_{t_c+l_s} \in \mathbb{N}^{d\times r}$ denotes the l_s-steps-ahead future values that correspond to the period for which we want to generate estimates. In this respect, we formally describe our problem as follows.

Problem 1 (Real-time forecasting). **Given** *an epidemic stream* $\mathcal{X} \in \mathbb{N}^{d\times r\times t_c}$*, and the current window* $\mathcal{X}_C = \{X_t\}_{t=t_m}^{t_c}$, **forecast** l_s*-steps-ahead future values* $V_{t_c+l_s} \in \mathbb{N}^{d\times r}$ *in a streaming fashion.*

Here, there are two major questions when it comes to solving Problem 1: (1) How can we model complex, non-linear dynamics in epidemic streams? (2) How can we efficiently and effectively estimate such non-linear models in multiple locations? We provide the answers below.

3.1 EpiCast-base – with a single epidemic

The first problem is finding a way to model the complex dynamics of epidemic sequences. Here, for simplicity, we first focus on an epidemic stream in a single location. As shown in the preview of an epidemic stream in the introduction section, viruses tend to spread non-linearly. We thus propose using a non-linear dynamical system, and specifically, we introduce an SEIR-based model, whose mechanism we summarize below.

The model we propose assumes five classes – (1) **S**usceptible people, who potentially become infected, change over time while the number of (2) **E**xposed people increases with a infection rate β. Then, the people are (3) **I**nfected with a incidence rate σ. After that, those who are in this class will (4) **R**ecover at a recovery rate γ or (5) **D**ie with a mortality rate δ. The time dependency of these five classes is described by the following differential equations.

$$\frac{dS}{dt} = -\beta S(t)I(t), \quad \frac{dE}{dt} = \beta S(t)I(t) - \sigma E(t),$$
$$\frac{dI}{dt} = \sigma E(t) - \gamma I(t) - \delta I(t), \quad \frac{dR}{dt} = \gamma I(t), \quad \frac{dD}{dt} = \delta I(t). \qquad (1)$$

Note that the model also includes the initial populations of exposed (E_0), patients (I_0), confirmed recoveries (R_0) and death toll (D_0) at starting time point t_0, to generate subsequent estimated values. We can obtain the initial susceptible population S_0 by computing $S_0 = N - (E_0 + I_0 + R_0 + D_0)$ according to the notion of the total population.

Consequently, the entire parameter set we want to estimate can be summarized as follows.

Model 1 (EpiCast-base). *Let* $\boldsymbol{\theta}$ *be a set of EpiCast-base parameters, i.e.,* $\boldsymbol{\theta} = \{\beta, \sigma, \gamma, \delta, N, E_0, I_0, R_0, D_0, t_0\}$*, which consists of*

- β*: Infection rate of the epidemic* ($0 \le \beta \le 1$)
- σ*: Incidence rate of the epidemic* ($0 \le \sigma \le 1$)
- γ*: Recovery rate of the epidemic* ($0 \le \gamma \le 1$)
- δ*: Mortality rate of the epidemic* ($0 \le \delta \le 1$)
- N*: Potential population of an epidemic* ($0 \le N$)

where, E_0*,* I_0*,* R_0*,* D_0 (≥ 0) *show the initial populations of those exposed, patients, confirmed recoveries and the death toll at time point* t_0*.*

We consistently use the symbols $\mathcal{V} \in \mathbb{N}^{d\times r\times t}$, $V_t \in \mathbb{N}^{d\times r}$, $V^{(i)} \in \mathbb{N}^{d\times t}$ and $\mathbf{v}_i(t) \in \mathbb{N}^d$ ($i = 1, \cdots, r$) as estimated values for an original epidemic stream using Model 1. For example, an epidemic sequence $V^{(i)} \subset \mathcal{V}$ for the i-th location is given by the accumulated data starting from the values $\mathbf{v}_i(t_0) = \{E_0, I_0, R_0, D_0\}$ to the expected end point $\mathbf{v}_i(t)$ using Equation (1).

3.2 EpiCast – with co-evolving epidemics in multiple locations

A more important goal is to detect similar epidemic dynamical patterns (namely regimes), between multiple locations. It is a key concept that when we cannot access sufficient observations to estimate a model within a single location, we should share/apply a model obtained in another location for forecasting. With regard to the fact that there are differences between the basic features of locations such as population and culture (as we have seen in section 2), we propose separating the EpiCast parameters in $\boldsymbol{\theta}$ into two groups, namely, epidemic parameters $\boldsymbol{\theta}^{\boldsymbol{E}} = \{\beta, \sigma, \gamma, \delta\}$, and location parameters $\boldsymbol{\theta}^{\boldsymbol{L}} = \{N, E_0, I_0, R_0, D_0, t_0\}$, (that is, $\boldsymbol{\theta} = \boldsymbol{\theta}^{\boldsymbol{E}} \cup \boldsymbol{\theta}^{\boldsymbol{L}}$). Here, epidemic parameters $\boldsymbol{\theta}^{\boldsymbol{E}}$ can be shared with any location whereas location parameters $\boldsymbol{\theta}^{\boldsymbol{L}}$ are locally optimized for each country/region. For example, there are similar dynamics/regimes between countries where the same countermeasures are adopted but the timing can be different. Therefore, maintaining multiple EpiCast parameter sets allows us to take account of the knowledge obtained in any country/region for real-time forecasting.

We eventually define our full parameter set as follows.

Model 2 (EpiCast-full). *Let* $\boldsymbol{\Theta} = \{\boldsymbol{\Theta}^E, \boldsymbol{\Theta}^L\}$ *be a full parameter set of EpiCast, where,* $\boldsymbol{\Theta}^E$ *is a set of* g *representative epidemic parameters and,* $\boldsymbol{\Theta}^L$ *is a set of local parameters in* r *locations (here,* $g \ll r$*), i.e.,* $\boldsymbol{\Theta}^E = \{\boldsymbol{\theta}_1^{\boldsymbol{E}}, \cdots, \boldsymbol{\theta}_g^{\boldsymbol{E}}\}$, $\boldsymbol{\Theta}^L = \{\boldsymbol{\theta}_1^{\boldsymbol{L}}, \cdots, \boldsymbol{\theta}_r^{\boldsymbol{L}}\}$.

4 STREAMING ALGORITHMS

In this section, we propose a streaming algorithm, namely EpiCast, that incrementally captures multiple epidemic activities evolving over time. Our algorithm should have the following properties:

- **(P1)** Non-linear modeling over complex epidemic streams
- **(P2)** Model-sharing mechanism among multiple locations

We need to capture the complicated dynamics of real epidemic data that involve non-linear phenomena. To handle (**P1**), we propose a streaming algorithm that exploits non-linear differential equations (i.e., Model 1). We also want to detect similar dynamics/regimes in different locations, i.e., (**P2**), and thus we employ a model obtained in another location for forecasting (i.e., Model 2).

Figure 3 shows an overview of EpiCast, which consists of the following three algorithms.

- EpiEstimator: Given a single epidemic sequence at the i-th location, i.e., $X_C^{(i)} \subset \mathcal{X}_C$, it estimates a new model $\boldsymbol{\theta} = \{\boldsymbol{\theta}^{\boldsymbol{E}}, \boldsymbol{\theta}^{\boldsymbol{L}}\}$, from scratch.
- EpiFinder: Given a single epidemic sequence $X_C^{(i)} \subset \mathcal{X}_C$, and the current parameter set $\boldsymbol{\Theta}$, it searches for the best epidemic parameters $\boldsymbol{\theta}^{\boldsymbol{E}}$ in $\boldsymbol{\Theta}$, as well as estimates optimal local parameters $\boldsymbol{\theta}^{\boldsymbol{L}}$, and then generates estimated values $V_E^{(i)}$.

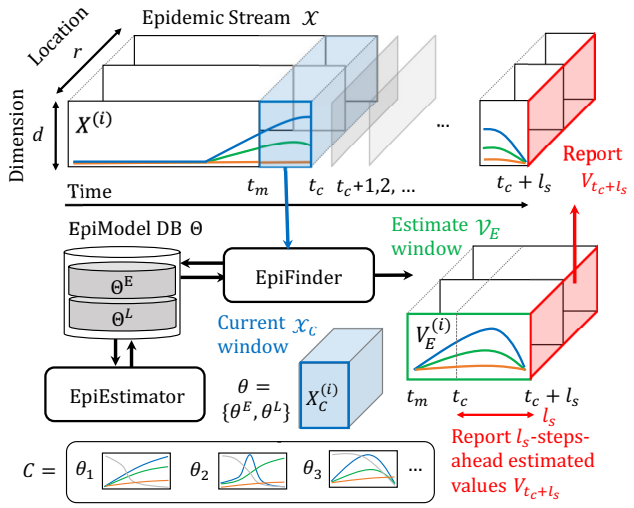


Figure 3: Overview of the EpiCast algorithm: Given an epidemic stream $\mathcal{X}$, it extracts the current window $\mathcal{X}_C$, and (a) finds the optimal patterns in the epidemic model database Θ. It then generates the l_s-steps-ahead values $V_{t_c+l_s}$. If there is a new (i.e., unknown) pattern in $\mathcal{X}_C$, (b) it also estimates the new model parameter set θ, and inserts it into Θ.

- EpiCast: Given an epidemic stream $\mathcal{X}$, it monitors a current window $\mathcal{X}_C$, runs the above two algorithms to obtain the estimated values $\mathcal{V}_E$, and reports the l_s-steps-ahead values $V_{t_c+l_s} \in \mathbb{N}^{d \times r}$, while maintaining the parameter set Θ.

4.1 Model estimation – EpiEstimator

For simplicity, we first describe how to estimate non-linear models effectively for a given epidemic stream. We specifically propose EpiEstimator, which can estimate model parameters $\boldsymbol{\theta} = \{\boldsymbol{\theta^E}, \boldsymbol{\theta^L}\}$, for a current window $\mathcal{X}_C$, focusing on a single location in an epidemic stream $\mathcal{X}$. Consider the epidemic sequence for a single location (i.e., $X_C^{(i)} \subset \mathcal{X}_C$, for the i-th location, where $i = 1, \ldots, r$). EpiEstimator finds the optimal $\boldsymbol{\theta} = \{\boldsymbol{\theta^E}, \boldsymbol{\theta^L}\}$, when given $X_C^{(i)}$ so that the model can minimize the following equation.

$$\{\boldsymbol{\theta^E}, \boldsymbol{\theta^L}\} = \arg\min_{\boldsymbol{\theta^{E\prime}}, \boldsymbol{\theta^{L\prime}}} \|X_C^{(i)} - V_C^{(i)}\|, \; V_C^{(i)} = f(\boldsymbol{\theta^{E\prime}}, \boldsymbol{\theta^{L\prime}}), \quad (2)$$

where $\|\cdot\|$ shows the mean square errors and $V_C^{(i)} = f(\boldsymbol{\theta^{E\prime}}, \boldsymbol{\theta^{L\prime}})$ shows the estimated sequence of $X_C^{(i)}$ given by Equation (1). Here, we use the fourth-order Runge-Kutta method [18] to generate the reconstructed data points. In the optimization of $\boldsymbol{\theta}$, we apply the Levenberg-Marquardt (LM) algorithm [30], which can solve the non-linear least squares minimization problem effectively.

4.2 Model selection – EpiFinder

The significant lack of information about infectious behavior within a single location makes it necessary to exploit multiple non-linear models that are obtained from any location for co-evolving epidemic streams, in which the dynamics in the current window $\mathcal{X}_C$ change over time and location. We thus propose the use of EpiFinder to determine dynamic patterns $\boldsymbol{\theta}$ by considering the adaption of known parameters in a full parameter set Θ.

Algorithm 1 EpiFinder $(X_C^{(i)}, \Theta)$

Input: (a) Current epidemic sequence $X_C^{(i)}$ in i-th location
(b) Current parameter set $\Theta = \{\Theta^E, \Theta^L\}$
Output: (a) Estimated values $V_E^{(i)}$ in i-th location
(b) Model parameter set $\boldsymbol{\theta} = \{\boldsymbol{\theta^E}, \boldsymbol{\theta^L}\}$
$C = \emptyset$; // Candidate parameter set
for $j = 1 : g$ **do**
  /* Model estimation with the j-th epidemic parameters and the i-th location parameters */
  Set $\{\boldsymbol{\theta^{E\prime}}, \boldsymbol{\theta^{L\prime}}\} = \{\boldsymbol{\theta^E}_j, \boldsymbol{\theta^L}_i\}$ as initial condition
  $\{\boldsymbol{\theta^E}, \boldsymbol{\theta^L}\} = \arg\min_{\boldsymbol{\theta^{E\prime}}, \boldsymbol{\theta^{L\prime}}} \|X_C^{(i)} - V_C^{(i)}\|$; // $V_C^{(i)} = f(\boldsymbol{\theta^{E\prime}}, \boldsymbol{\theta^{L\prime}})$
  $C = C \cup \{\boldsymbol{\theta^E}, \boldsymbol{\theta^L}\}$;
end for
/* Choose the best model in C */
$\{\boldsymbol{\theta^E}, \boldsymbol{\theta^L}\} = \arg\min_{\{\boldsymbol{\theta^{E\prime}}, \boldsymbol{\theta^{L\prime}}\} \in C} \|X_C^{(i)} - V_C^{(i)}\|$; // $V_C^{(i)} = f(\boldsymbol{\theta^{E\prime}}, \boldsymbol{\theta^{L\prime}})$
Compute $V_E^{(i)} = f(\boldsymbol{\theta^E}, \boldsymbol{\theta^L})$; $\boldsymbol{\theta} = \{\boldsymbol{\theta^E}, \boldsymbol{\theta^L}\}$;
return $\{V_E^{(i)}, \boldsymbol{\theta}\}$;

Algorithm 1 shows the details of EpiFinder. Here, the current sequence for the i-th location will be given (i.e., $X_C^{(i)} \subset \mathcal{X}_C$). Also, we assume that it already contains several model parameters in $\Theta = \{\Theta^E, \Theta^L\}$, which consists of epidemic parameters and location parameters, i.e., $\Theta^E = \{\boldsymbol{\theta}_1^E, \cdots, \boldsymbol{\theta}_g^E\}$, $\Theta^L = \{\boldsymbol{\theta}_1^L, \cdots, \boldsymbol{\theta}_r^L\}$, where g shows the number of typical/representative regime groups among r locations. Given $X_C^{(i)}$ and Θ, the algorithm searches for the best model $\boldsymbol{\theta} = \{\boldsymbol{\theta^E}, \boldsymbol{\theta^L}\}$ in terms of the reconstruction error between $X_C^{(i)}$ and $V_C^{(i)}$ that is generated by Equation (1).

So, how can we efficiently and effectively estimate the best model for the given $X_C^{(i)}$? We propose EpiFinder, which shares representative regimes obtained from multiple locations. Specifically, the algorithm searches for the best epidemic parameters $\boldsymbol{\theta}_j^E$ stored in Θ ($j = 1, \cdots, g$), as well as estimates optimal local parameters for the i-th location, by using the current location parameters $\boldsymbol{\theta}_i^L$. Here, location parameters $\boldsymbol{\theta^L} = \{N, E_0, I_0, R_0, D_0, t_0\}$ are fully estimated, while epidemic parameters $\boldsymbol{\theta^E} = \{\beta, \sigma, \gamma, \delta\}$ are updated within a limited range of values[1]. This enables $\boldsymbol{\theta^E}$ to preserve the base dynamics of epidemics.

4.3 Streaming algorithm – EpiCast

We now incorporate the two algorithms we have proposed into our streaming method, EpiCast, which realizes the real-time forecasting of epidemic streams, while considering time-evolving non-linear dynamics. Our final goal is to find both global and location-specific epidemic patterns, i.e., $\Theta = \{\Theta^E, \Theta^L\}$, so as to generate the l_s-steps-ahead future values $V_{t_c+l_s} \in \mathbb{N}^{d \times r}$ for all locations.

[1]We set the range of each parameter at ±0.1 (i.e., ±10%).

Algorithm 2 EpiCast ($\mathcal{X}_C, \mathbf{\Theta}$)

Input: (a) Current window $\mathcal{X}_C = \mathcal{X}[t_m : t_c]$
(b) Model parameter set $\mathbf{\Theta} = \{\mathbf{\Theta}^E, \mathbf{\Theta}^L\}$
Output: (a) l_s-steps-ahead values $V_{t_c+l_s} = \mathcal{V}[t_c + l_s]$
(b) Updated model parameter set $\mathbf{\Theta}' = \{\mathbf{\Theta}^{E\prime}, \mathbf{\Theta}^{L\prime}\}$
/*** (I) Estimate optimal parameters for each i-th location ***/
for $i = 1 : r$ **do**
  /* (1) Fitting by the previous best model at time $t_c - 1$ */
  $\{\boldsymbol{\theta}^E, \boldsymbol{\theta}^L\} = \boldsymbol{\theta}_i^{(t_c-1)}$; Compute $V_C^{(i)} = f(\boldsymbol{\theta}^E, \boldsymbol{\theta}^L)$;
  if $\|X_C^{(i)} - V_C^{(i)}\| > \epsilon$ **then**
    /* (2-1) Fitting by local area S for the i-th location */
    $\mathbf{\Theta}^{(S)} \subset \mathbf{\Theta}$; // Subset models of local area S
    $\{V_E^{(i)}, \boldsymbol{\theta}\}$ =EpiFinder $(X_C^{(i)}, \mathbf{\Theta}^{(S)})$;
    $V_C^{(i)} = V_E^{(i)}[t_m : t_c]$; // Estimated values from t_m to t_c
  end if
  if $\|X_C^{(i)} - V_C^{(i)}\| > \epsilon$ **then**
    /* (2-2) Fitting by other local areas */
    $\mathbf{\Theta}^{(\neg S)} \subset \mathbf{\Theta}$; // Subset models of local area, except S
    $\{V_E^{(i)}, \boldsymbol{\theta}\}$ =EpiFinder $(X_C^{(i)}, \mathbf{\Theta}^{(\neg S)})$;
    $V_C^{(i)} = V_E^{(i)}[t_m : t_c]$; // Estimated values from t_m to t_c
  end if
  /* (3) Estimate new regimes (if required) */
  if $\|X_C^{(i)} - V_C^{(i)}\| > \epsilon$ **then**
    $\{\boldsymbol{\theta}^E, \boldsymbol{\theta}^L\}$ =EpiEstimator $(X_C^{(i)})$;
    $\mathbf{\Theta}^E = \mathbf{\Theta}^E \cup \boldsymbol{\theta}^E$; $g = g + 1$; // Insert new model
    Compute $V_E^{(i)} = f(\boldsymbol{\theta}^E, \boldsymbol{\theta}^L)$;
  end if
  Update $\boldsymbol{\theta}_i^L$ in $\mathbf{\Theta}^L$;
end for
/*** (II) l_s-steps-ahead future value generation ***/
$V_{t_c+l_s} = \{V_E^{(i)}[t_c + l_s]\}_{i=1}^r$;
return $\{V_{t_c+l_s}, \mathbf{\Theta}'\}$;

Algorithm 2 summarizes the procedure with EpiCast. For the current window $\mathcal{X}_C$, it determines the best non-linear model for each location, i.e., $X_C^{(i)}$, through the following three steps:

First, it tries to use the previously estimated model parameters $\boldsymbol{\theta}_i^{(t_c-1)}$ for the i-th location at time point $t_c - 1$. This is based on the natural assumption that mutation happens occasionally, and thus the algorithm continues using $\boldsymbol{\theta}_i^{(t_c-1)}$ for efficiency.

Second, when the previous regime does not fit well, i.e., the reconstruction error is more than the required accuracy ϵ,[2] the candidate regime to represent the current sequence is selected by EpiFinder. However, the number of candidates in $\mathbf{\Theta}$ can increase as we find new dynamic patterns. To prevent a search of the full combination of regimes, we propose an efficient way of eliminating some of the search by building hierarchical groups of regimes in $\mathbf{\Theta}$. Specifically, we separate regimes in $\mathbf{\Theta}$ by using groups of neighboring locations, namely local areas (e.g., continents), which allows the search to consider the similarity of dynamics between near locations. The algorithm runs EpiEstimator in its own local area, i.e., $\mathbf{\Theta}^{(S)} \subset \mathbf{\Theta}$, when the i-th location belongs to the local area S. Unless it finds a regime that appropriately fits $X_C^{(i)}$, it extends the subset of candidates to the other local area, except for $\mathbf{\Theta}^{(S)}$, i.e., $\mathbf{\Theta}^{(\neg S)} \subset \mathbf{\Theta}$.

Third, if there is no appropriate model in $\mathbf{\Theta}$, it should estimate a new model $\{\boldsymbol{\theta}^E, \boldsymbol{\theta}^L\}$ using EpiEstimator. The new epidemic parameters $\boldsymbol{\theta}^E$ is then added into $\mathbf{\Theta}^E$, and also location parameters $\boldsymbol{\theta}^L$ are replaced into $\mathbf{\Theta}^L$.

Consequently, by iterating this procedure for each location, we eventually obtain all the estimated values $V_{t_c+l_s}$.

Lemma 1. *The time complexity of EpiCast is $O(g)$ at each time point.*

Proof. Please see Appendix C. □

5 EXPERIMENTS

In this section we demonstrate the effectiveness of EpiCast with the COVID-19 dataset [11], which can be obtained at our website[3]. The dataset consists of $d = 3$ dimensional vectors (infected, recovery and death), covering over 600 days on a daily basis. Due to a significant amount of missing data, we selected the top 50 countries in order of their GDP scores[4]. Our experiments were conducted on an Intel Core i7 2.8GHz quad core CPU with 16GB of memory. The experiments were designed to answer the following questions about EpiCast:

- Q1 Effectiveness: How well does it capture co-evolving epidemic patterns?
- Q2 Accuracy: How accurately does it forecast future outbreaks?
- Q3 Scalability: How does it scale in terms of computational time?

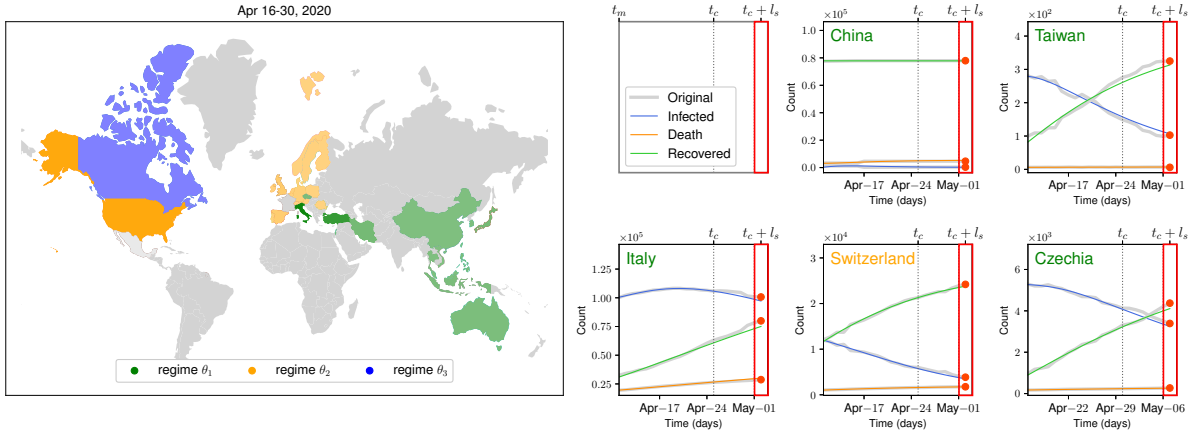
Q1. Effectiveness. We demonstrate the forecasting power of EpiCast in terms of capturing important patterns in epidemic streams and forecasting future values. We have already provided examples of EpiCast in Figure 1, which shows that the method effectively forecasts long-range future values. Here Figure 4 shows the additional results obtained when EpiCast continued to analyze the epidemic stream in Figure 1. For example, at the end of April (the top of Figure 4), we observe that the infection was under control in China. The epidemic states of other countries also changed. Taiwan's case was appearing to settle down, which was also observed for China. Meanwhile, Italy's case was similar to Taiwan's at the beginning of April. A pandemic started in Czechia, for which our method used the same regime as observed in Italy.

Figure 5 shows the details of the EpiCast outputs, focusing on a single location, i.e., Czechia. The top left figures are the original data stream, for which our method forecasted seven-days-ahead values as shown in the figures bottom left, where the forecasts are visually well fitted without any divergence through the data. Here, we plotted the figure in both linear (left column) and log (right column) scales. The right side of Figure 5 shows the relative characteristics of regimes: specifically, the figure shows a scatter plot of the model parameter sets of multiple regimes, where

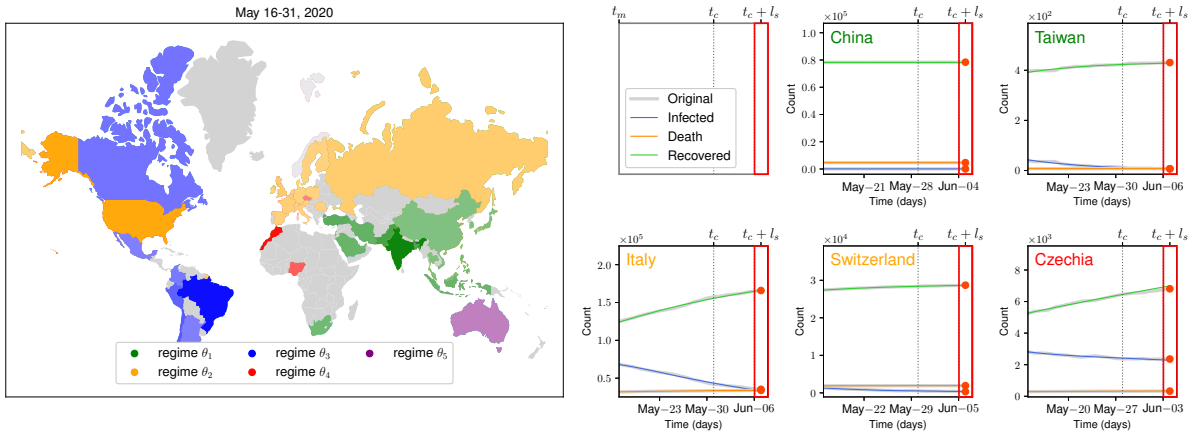
[2] In this paper, we set $\epsilon = 1/2\|X_C^{(i)}\|$.

[3] https://www.worldometers.info/coronavirus/

[4] https://www.imf.org/external/index.htm



(a) Snapshots of EpiCast-Map and seven-days-ahead future predictions for April 16-30, 2020



(b) Snapshots of EpiCast-Map and seven-days-ahead future predictions for May 16-31, 2020

Figure 4: EpiCast is effective: Snapshots of our method at different time points (continued from Figure 1). It successfully captures current dynamical patterns (i.e., regimes) in each location, and forecasts future co-evolving epidemics.

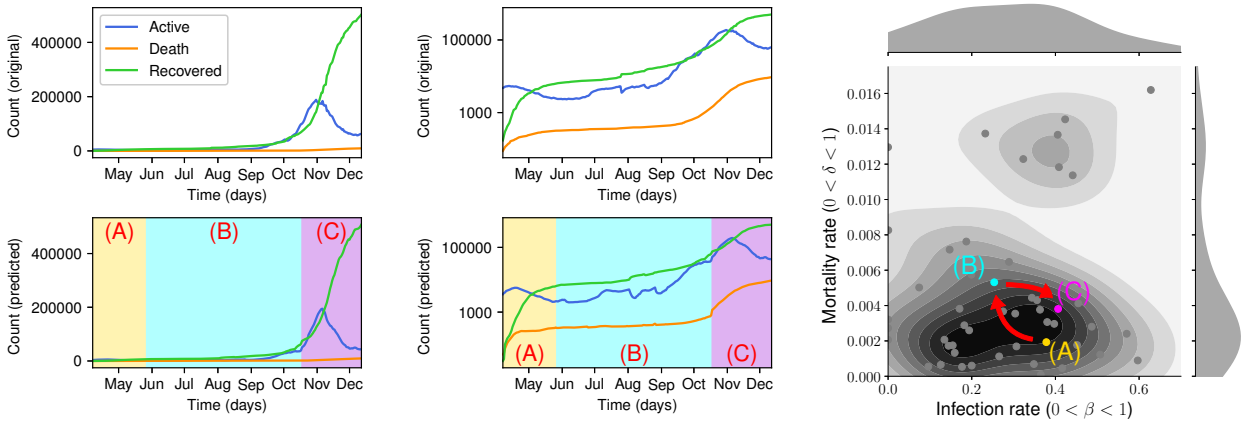


Figure 5: EpiCast is adaptive: The four figures on the left show the original data (top) and our seven-days-ahead predictions (bottom) for one location (Czechia), shown in linear (left) and log (right) scales. The figure on the right shows a scatter plot of the model parameter sets of regimes (here, it shows infection rate β vs. mortality rate γ), where each black dot corresponds to each regime, stored in Θ. EpiCast automatically and incrementally identifies the current epidemic patterns by switching models/regimes (i.e., (A) → (B) → (C)), and forecasts future events, efficiently and effectively.

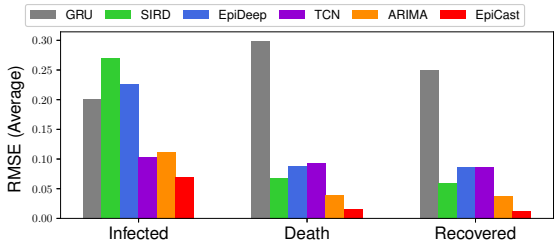


Figure 6: RMSE between original and forecast values: EpiCast consistently outperforms its competitors in all three dimensions and in terms of metrics (lower is better).

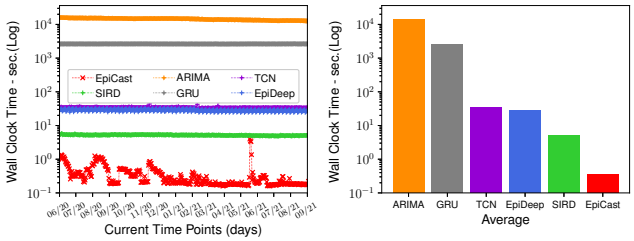


Figure 7: Scalability of EpiCast: (left) Wall clock time vs. data stream length t_c and (right) average time consumption. Our method is consistently superior to its competitors even when it estimates new models including non-linear parameters. Here, it is up to 37,500x faster than its competitors.

each black dot corresponds to each regime stored in Θ, which are estimated by EpiCast. Here, we observe three kinds of regimes: (A)→(B)→(C), where regimes (A) and (C) have higher infection rates β than regime (B). Regime (B) has a higher mortality rate γ than the other two regimes. Indeed, Czechia experienced its first pandemic during the yellow-shaded period, followed by the blue-shaded period when the pandemic settled down, and the purple-shaded period when the viruses spread more rapidly than in the first pandemic.

As shown in the figures, our proposed method successfully captures a wide variety of regimes, i.e., non-linear dynamical patterns, for multiple locations at different time points. The regimes summarize complex, time-evolving dynamical features, and thus allow us to understand and forecast the diffusion process of epidemics.

Q2. Accuracy. Next, we discuss the quality of EpiCast in terms of forecasting accuracy. We compared EpiCast with the following baselines: SIRD (a non-linear equation for modeling epidemics), ARIMA (a linear forecasting method). We also compared our method with GRU (a recurrent neural network model with gated recurrent units), TCN [21] (a temporal convolutional network, which can learn multi-level temporal causality, and constitutes another choice for analyzing sequential patterns), EpiDeep [7] (one of the most recent neural network models, which successfully applied DNN to modeling the dynamics of a seasonally occurring infection). Please also see a description of our experimental settings in Appendix D.

Figure 6 shows the average forecasting errors (the root mean square error (RMSE)) between the original values and the seven-days-ahead forecast values. Here, a lower value indicates a better forecasting accuracy. Our approach achieved the lowest fitting error for all three kinds of statistics (i.e., infected, death, recovered), which means that our method is capable of modeling both the rise and fall-part patterns at any time in epidemic streams. Please also see the additional experimental results in Appendix D.

More importantly, regime switching contributes to both accuracy and scalability, as described in the next experimental result.

Q3. Scalability. We also evaluate the efficiency of our forecasting algorithm. Figure 7 compares EpiCast with its competitors in terms of computation time at each time point t_c. Note that the figures are shown in linear-log scales. On the left in Figure 7, the upper range of the running time in EpiCast corresponds to the EpiEstimator process, which creates a new model. Despite the increasing number of regimes, the computation time remains constant until the end of the data stream. This result suggests that EpiFinder works effectively for the stream mining of multiple regimes because it allows the algorithm to maintain a large number of regimes. The right side of Figure 7 shows the average computation time for the entire epidemic stream. As we expected, EpiCast generates long-range future values significantly faster than its competitors for large streams (i.e., up to four orders of magnitude).

6 CONCLUSION

In this paper, we proposed EpiCast, which is designed for modeling and forecasting co-evolving epidemiological data streams. Our method has the following desirable properties:

(1) It is **Effective**: it captures important dynamical epidemic patterns (i.e., regimes) in data streams and provides long-range forecasting at any time.
(2) It is **Adaptive**: it can dynamically and adaptively capture current regimes, by sharing non-linear models among multiple locations.
(3) It is **Scalable**: we proposed an efficient algorithm that is constant in terms of input data size.

Using a real public COVID-19 dataset, we demonstrated that our proposed method outperforms existing methods in terms of forecasting accuracy with a significant reduction in computational time.

Acknowledgment. The authors would like to thank the anonymous referees for their valuable comments and helpful suggestions. This work was supported by JSPS KAKENHI Grant-in-Aid for Scientific Research Number JP20H00585, JP21H03446, JP22K17896, NICT 21481014, MIC/SCOPE 192107004, JST-Mirai JPMJMI19B3, JST-AIP JPMJCR21U4.

REFERENCES

[1] https://www.who.int/emergencies/diseases/novel-coronavirus-2019.
[2] https://protect-public.hhs.gov/.
[3] https://www.rki.de/EN/Home/homepage_node.html.
[4] *EpiCast.* https://sites.google.com/view/epicast-demo/home.
[5] Genomewide association study of severe covid-19 with respiratory failure. *New England Journal of Medicine*, 383(16):1522–1534, 2020.
[6] B. Adhikari, B. L. Lewis, A. Vullikanti, J. M. Jiménez, and B. A. Prakash. Fast and near-optimal monitoring for healthcare acquired infection outbreaks. *PLoS Comput. Biol.*, 15(9), 2019.
[7] B. Adhikari, X. Xu, N. Ramakrishnan, and B. A. Prakash. Epideep: Exploiting embeddings for epidemic forecasting. In *KDD*, pages 577–586, 2019.
[8] Andreadis, Georgios and Quirós Gámez, Ana Isabel. Prospective analysis of the impact of a pandemic in industry 4.0. *MATEC Web Conf.*, 318:01037, 2020.
[9] E. Beyazit, J. Alagurajah, and X. Wu. Online learning from data streams with varying feature spaces. In *AAAI/IAAI*, pages 3232–3239, 2019.

[10] P. Chen, S. Liu, C. Shi, B. Hooi, B. Wang, and X. Cheng. Neucast: Seasonal neural forecast of power grid time series. In *IJCAI*, pages 3315–3321, 2018.

[11] E. Dong, H. Du, and L. Gardner. An interactive web-based dashboard to track covid-19 in real time. *The Lancet Infectious Diseases*, 20(5), May 2020.

[12] J. Durbin and S. J. Koopman. *Time Series Analysis by State Space Methods*. Oxford University Press, 2 edition, 2012.

[13] C. Faloutsos, J. Gasthaus, T. Januschowski, and Y. Wang. Classical and contemporary approaches to big time series forecasting. In *SIGMOD*, pages 2042–2047, 2019.

[14] V. Flunkert, D. Salinas, and J. Gasthaus. Deepar: Probabilistic forecasting with autoregressive recurrent networks. *CoRR*, abs/1704.04110, 2017.

[15] W.-j. Guan, Z.-y. Ni, Y. Hu, W.-h. Liang, C.-q. Ou, J.-x. He, L. Liu, H. Shan, C.-l. Lei, D. S. Hui, B. Du, L.-j. Li, G. Zeng, K.-Y. Yuen, R.-c. Chen, C.-l. Tang, T. Wang, P.-y. Chen, J. Xiang, S.-y. Li, J.-l. Wang, Z.-j. Liang, Y.-x. Peng, L. Wei, Y. Liu, Y.-h. Hu, P. Peng, J.-m. Wang, J.-y. Liu, Z. Chen, G. Li, Z.-j. Zheng, S.-q. Qiu, J. Luo, C.-j. Ye, S.-y. Zhu, and N.-s. Zhong. Clinical characteristics of coronavirus disease 2019 in china. *New England Journal of Medicine*, 382(18):1708–1720, 2020.

[16] E. C. Holmes, G. Dudas, A. Rambaut, and K. G. Andersen. The evolution of ebola virus: Insights from the 2013–2016 epidemic. *Nature*, 538(7624):193–200, Oct 2016.

[17] M. R. Islam, S. Muthiah, B. Adhikari, B. A. Prakash, and N. Ramakrishnan. Deepdiffuse: Predicting the 'who' and 'when' in cascades. In *ICDM*, pages 1055–1060, 2018.

[18] E. A. Jackson. *Perspectives of Nonlinear Dynamics*, volume 1. Cambridge University Press, 1989.

[19] B. Korber, W. M. Fischer, S. Gnanakaran, H. Yoon, J. Theiler, W. Abfalterer, N. Hengartner, E. E. Giorgi, T. Bhattacharya, B. Foley, K. M. Hastie, M. D. Parker, D. G. Partridge, C. M. Evans, T. M. Freeman, T. I. de Silva, A. Angyal, R. L. Brown, L. Carrilero, L. R. Green, D. C. Groves, K. J. Johnson, A. J. Keeley, B. B. Lindsey, P. J. Parsons, M. Raza, S. Rowland-Jones, N. Smith, R. M. Tucker, D. Wang, M. D. Wyles, C. McDanal, L. G. Perez, H. Tang, A. Moon-Walker, S. P. Whelan, C. C. LaBranche, E. O. Saphire, and D. C. Montefiori. Tracking changes in sars-cov-2 spike: Evidence that d614g increases infectivity of the covid-19 virus. *Cell*, 182(4):812–827.e19, 2020.

[20] S. A. Lauer, K. H. Grantz, Q. Bi, F. K. Jones, Q. Zheng, H. R. Meredith, A. S. Azman, N. G. Reich, and J. Lessler. The incubation period of coronavirus disease 2019 (covid-19) from publicly reported confirmed cases: Estimation and application. *Annals of Internal Medicine*, 172(9):577–582, 2020.

[21] C. Lea, M. D. Flynn, R. Vidal, A. Reiter, and G. D. Hager. Temporal convolutional networks for action segmentation and detection. In *CVPR*, pages 1003–1012, 2017.

[22] Q. Li, X. Guan, P. Wu, X. Wang, L. Zhou, Y. Tong, R. Ren, K. S. Leung, E. H. Lau, J. Y. Wong, X. Xing, N. Xiang, Y. Wu, C. Li, Q. Chen, D. Li, T. Liu, J. Zhao, M. Liu, W. Tu, C. Chen, L. Jin, R. Yang, Q. Wang, S. Zhou, R. Wang, H. Liu, Y. Luo, Y. Liu, G. Shao, H. Li, Z. Tao, Y. Yang, Z. Deng, B. Liu, Z. Ma, Y. Zhang, G. Shi, T. T. Lam, J. T. Wu, G. F. Gao, B. J. Cowling, B. Yang, G. M. Leung, and Z. Feng. Early transmission dynamics in wuhan, china, of novel coronavirus–infected pneumonia. *New England Journal of Medicine*, 382(13):1199–1207, 2020.

[23] C. Liu, S. C. H. Hoi, P. Zhao, and J. Sun. Online ARIMA algorithms for time series prediction. In *AAAI*, pages 1867–1873, 2016.

[24] G. Liu, B. Carter, and D. K. Gifford. Predicted cellular immunity population coverage gaps for sars-cov-2 subunit vaccines and their augmentation by compact peptide sets. *Cell systems*, 12(1):102–107.e4, Jan 2021.

[25] L. Ma, D. V. Aken, A. Hefny, G. Mezerhane, A. Pavlo, and G. J. Gordon. Query-based workload forecasting for self-driving database management systems. In *SIGMOD*, pages 631–645, 2018.

[26] Y. Matsubara and Y. Sakurai. Dynamic modeling and forecasting of time-evolving data streams. In A. Teredesai, V. Kumar, Y. Li, R. Rosales, E. Terzi, and G. Karypis, editors, *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, pages 458–468. ACM, 2019.

[27] Y. Matsubara, Y. Sakurai, B. A. Prakash, L. Li, and C. Faloutsos. Rise and fall patterns of information diffusion: model and implications. In Q. Yang, D. Agarwal, and J. Pei, editors, *The 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '12, Beijing, China, August 12-16, 2012*, pages 6–14. ACM, 2012.

[28] Y. Matsubara, Y. Sakurai, W. G. van Panhuis, and C. Faloutsos. FUNNEL: automatic mining of spatially coevolving epidemics. In S. A. Macskassy, C. Perlich, J. Leskovec, W. Wang, and R. Ghani, editors, *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, pages 105–114. ACM, 2014.

[29] E. Minskaia, T. Hertzig, A. E. Gorbalenya, V. Campanacci, C. Cambillau, B. Canard, and J. Ziebuhr. Discovery of an rna virus 3'→5' exoribonuclease that is critically involved in coronavirus rna synthesis. *Proceedings of the National Academy of Sciences*, 103(13):5108–5113, 2006.

[30] J. J. Moré. The levenberg-marquardt algorithm: Implementation and theory. In *Numerical Analysis*, pages 105–116, 1978.

[31] H. Nishiura, N. M. Linton, and A. R. Akhmetzhanov. Serial interval of novel coronavirus (covid-19) infections. *International Journal of Infectious Diseases*, 93:284–286, 2020.

[32] G. Panagopoulos, G. Nikolentzos, and M. Vazirgiannis. Transfer graph neural networks for pandemic forecasting. In *AAAI/IAAI*, pages 4838–4845, 2021.

[33] J. Paparrizos, C. Liu, A. J. Elmore, and M. J. Franklin. Debunking four long-standing misconceptions of time-series distance measures. In *SIGMOD*, pages 1887–1905, 2020.

[34] K. Prem, Y. Liu, T. Russell, A. Kucharski, R. Eggo, N. Davies, M. Jit, P. Klepac, S. Flasche, S. Clifford, C. Pearson, J. Munday, S. Abbott, H. Gibbs, A. Rosello, B. Quilty, T. Jombart, F. Sun, C. Diamond, and J. Hellewell. The effect of control strategies to reduce social mixing on outcomes of the covid-19 epidemic in wuhan, china: a modelling study. *The Lancet Public Health*, 5, 03 2020.

[35] Y. Qin, D. Song, H. Chen, W. Cheng, G. Jiang, and G. W. Cottrell. A dual-stage attention-based recurrent neural network for time series prediction. In *IJCAI*, pages 2627–2633, 2017.

[36] A. Rodríguez, N. Muralidhar, B. Adhikari, A. Tabassum, N. Ramakrishnan, and B. A. Prakash. Steering a historical disease forecasting model under a pandemic: Case of flu and COVID-19. In *AAAI/IAAI*, pages 4855–4863, 2021.

[37] M. Rogers, L. Li, and S. J. Russell. Multilinear dynamical systems for tensor time series. In *NIPS*, pages 2634–2642, 2013.

[38] Y. Sakurai, Y. Matsubara, and C. Faloutsos. Mining and forecasting of big time-series data. In T. K. Sellis, S. B. Davidson, and Z. G. Ives, editors, *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, Melbourne, Victoria, Australia, May 31 - June 4, 2015*, pages 919–922. ACM, 2015.

[39] J. Shaman and A. Karspeck. Forecasting seasonal outbreaks of influenza. *Proceedings of the National Academy of Sciences*, 109(50):20425–20430, 2012.

[40] Q. Shi, J. Yin, J. Cai, A. Cichocki, T. Yokota, L. Chen, M. Yuan, and J. Zeng. Block hankel tensor ARIMA for multiple short time series forecasting. In *AAAI/IAAI*, pages 5758–5766, 2020.

[41] H. A. Song, B. Hooi, M. Jereminov, A. Pandey, L. T. Pileggi, and C. Faloutsos. Powercast: Mining and forecasting power grid sequences. In *PKDD*, volume 10535 of *Lecture Notes in Computer Science*, pages 606–621, 2017.

[42] A. Taghvaei, J. de Wiljes, P. G. Mehta, and S. Reich. Kalman filter and its modern extensions for the continuous-time nonlinear filtering problem. *CoRR*, abs/1702.07241, 2017.

[43] M. Tizzoni, P. Bajardi, C. Poletto, J. J. Ramasco, D. Balcan, B. Gonçalves, N. Perra, V. Colizza, and A. Vespignani. Real-time numerical forecast of global epidemic spreading: Case study of 2009 a/h1n1pdm. *BMC medicine*, 10:165, 12 2012.

[44] S. R. Venna, A. Tavanaei, R. N. Gottumukkala, V. V. Raghavan, A. S. Maida, and S. Nichols. A novel data-driven model for real-time influenza forecasting. *IEEE Access*, 7:7691–7701, 2019.

[45] Q. Wen, K. He, L. Sun, Y. Zhang, M. Ke, and H. Xu. Robustperiod: Robust time-frequency mining for multiple periodicity detection. In *SIGMOD*, pages 2328–2337, 2021.

[46] WHO. Coronavirus disease 2019 (covid-19): situation report, 72. 2020.

[47] WHO. Covid-19 weekly epidemiological update, 27 april 2021. 2021.

[48] F. Wu, S. Zhao, B. Yu, Y.-M. Chen, W. Wang, Z.-G. Song, Y. Hu, Z.-W. Tao, J.-H. Tian, Y.-Y. Pei, M.-L. Yuan, Y.-L. Zhang, F.-H. Dai, Y. Liu, Q.-M. Wang, J.-J. Zheng, L. Xu, E. C. Holmes, and Y.-Z. Zhang. A new coronavirus associated with human respiratory disease in china. *Nature*, 579(7798):265–269, Mar 2020.

[49] C. Xiao, J. Zhou, J. Huang, A. Zhuo, J. Liu, H. Xiong, and D. Dou. C-watcher: A framework for early detection of high-risk neighborhoods ahead of COVID-19 outbreak. In *AAAI/IAAI*, pages 4892–4900, 2021.

[50] J. Ye, L. Sun, B. Du, Y. Fu, X. Tong, and H. Xiong. Co-prediction of multiple transportation demands based on deep spatio-temporal neural network. In *KDD*, pages 305–313, 2019.

[51] S. Yoon, J. Lee, and B. S. Lee. Ultrafast local outlier detection from a data stream with stationary region skipping. In *KDD*, pages 1181–1191, 2020.

[52] S. Yoon, Y. Shin, J. Lee, and B. S. Lee. Multiple dynamic outlier-detection from a data stream by exploiting duality of data and queries. In *SIGMOD*, pages 2063–2075, 2021.

[53] H. Yuan, G. Li, Z. Bao, and L. Feng. Effective travel time estimation: When historical trajectories over road networks matter. In *SIGMOD*, pages 2135–2149, 2020.

[54] C. Zhang, D. Song, Y. Chen, X. Feng, C. Lumezanu, W. Cheng, J. Ni, B. Zong, H. Chen, and N. V. Chawla. A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data. In *AAAI/IAAI*, pages 1409–1416, 2019.

[55] Z. Zhong, S. Yan, Z. Li, D. Tan, T. Yang, and B. Cui. Burstsketch: Finding bursts in data streams. In *SIGMOD*, pages 2375–2383, 2021.

[56] P. Zhou, X.-L. Yang, X.-G. Wang, B. Hu, L. Zhang, W. Zhang, H.-R. Si, Y. Zhu, B. Li, C.-L. Huang, H.-D. Chen, J. Chen, Y. Luo, H. Guo, R.-D. Jiang, M.-Q. Liu, Y. Chen, X.-R. Shen, X. Wang, X.-S. Zheng, K. Zhao, Q.-J. Chen, F. Deng, L.-L. Liu, B. Yan, F.-X. Zhan, Y.-Y. Wang, G.-F. Xiao, and Z.-L. Shi. A pneumonia outbreak associated with a new coronavirus of probable bat origin. *Nature*, 579(7798):270–273, Mar 2020.

APPENDIX

A ADDITIONAL RELATED WORK

Very recently, there has been a lot of work on COVID-related data analysis and forecasting. Here, CALI-NET [36] adjusts the training of DNN by using the COVID-related exogenous data (hospitalization rate, people tested, health related tweets, etc). C-Watcher [49] learns city-invariant representations from the mobility-related features using an adversarial encoder for the detection of high infection risk neighborhoods. MPNN+TL [32] is a graph neural network, which can generate representations for the regions based on their interactions and history from mobility data. However, these methods that use a wide variety of data about COVID-19 (human mobility, urban mobility, social media activity, etc) can only work in specific locations because they require a large number of data about the target location. Please also see section 2 for further discussion.

B PROPOSED MODEL

Table 1 gives an overview of the symbols used and their definitions.

Table 1: Symbols and definitions.

Symbol	Definition
d, r	Number of dimensions and locations
t_c	Current time point
$\mathcal{X}$	Epidemiological data stream, i.e., $\mathcal{X} \in \mathbb{N}^{d \times r \times t_c}$
$\mathcal{X}_C$	Current window, i.e., $\mathcal{X}_C = \{X_t\}_{t=t_m}^{t_c}, X_t \in \mathbb{N}^{d \times r}$
$V_{t_c+l_s}$	l_s-steps-ahead future values, i.e., $V_{t_c+l_s} \in \mathbb{N}^{d \times r}$
$\boldsymbol{\theta}$	Model parameter set, i.e., $\boldsymbol{\theta} = \boldsymbol{\theta^E} \cup \boldsymbol{\theta^L}$
$\boldsymbol{\theta^E}$	Epidemic parameters, i.e., $\boldsymbol{\theta^E} = \{\beta, \sigma, \gamma, \delta\}$
$\boldsymbol{\theta^L}$	Location parameters, i.e., $\boldsymbol{\theta^L} = \{N, E_0, I_0, R_0, D_0, t_0\}$
$\boldsymbol{\Theta}$	Model parameter set, i.e., $\boldsymbol{\Theta} = \{\boldsymbol{\Theta^E}, \boldsymbol{\Theta^L}\}$
$\boldsymbol{\Theta^E}$	Epidemic parameter set, i.e., $\boldsymbol{\Theta^E} = \{\boldsymbol{\theta}_1^{\boldsymbol{E}}, \cdots, \boldsymbol{\theta}_g^{\boldsymbol{E}}\}$
$\boldsymbol{\Theta^L}$	Location parameter set, i.e., $\boldsymbol{\Theta^L} = \{\boldsymbol{\theta}_1^{\boldsymbol{L}}, \cdots, \boldsymbol{\theta}_r^{\boldsymbol{L}}\}$

C STREAMING ALGORITHMS

Proof of Lemma 1. As we mentioned in section 4, EpiCast requires $O(g)$ time at each time point, where g shows the number of epidemic parameters stored in $\boldsymbol{\Theta^E}$.

Proof. Let l_c be the current window length, $l_c = t_c - t_m + 1$, and g be the number of epidemic parameters in $\boldsymbol{\Theta^E}$. EpiCast executes EpiFinder and EpiEstimator, each of which requires $O(g \cdot d \cdot r \cdot l_c)$ time for the parameter estimation. Since the sizes of the current window X_C given by d,r,l_c are negligibly small constant values, the total time complexity of EpiCast is $O(g)$ time per time point. □

D EXPERIMENTS

Experimental setup. We describe here our experimental setup and experimental parameters for the competitors. Here, we compare our method with the following methods:

- SIRD: a non-linear equation for modeling epidemics, where we optimized its parameters with the LM algorithm.
- ARIMA: a linear forecasting method, where we determined the optimal number of parameters, including autocorrelation coefficients, in the candidates $\{1, 2, 4, 8, 16\}$.
- GRU: a recurrent neural network (RNN) model with gated recurrent units (GRUs), where we stacked two GRU layers and four fully-connected layers with 30 hidden units.
- TCN [21]: a temporal convolutional network, which can learn multi-level temporal causality, where we built a $\{1, 2\}$-stacked TCN with a dilation set: $\{1, 2, 4, 8, 16, 32\}$, and searched for the best convolutional network structure in $\{16, 32, 64\}$ filters with one of the $\{3, 7\}$-length kernels.
- EpiDeep [7]: one of the most recent neural network models, which successfully applied DNN to modeling the dynamics of a seasonally occurring infection.

These offline baselines learn all historical data until the time when they perform forecasting. The DNN-based models are optimized based on Adam with a learning rate 0.01 and run 5000, 2000, 1000 epochs in GRU, TCN, EpiDeep, respectively.

Additional experiments: l_s-days-ahead epidemic prediction. As we mentioned in the introduction section, one of our motivations is the long-range forecasting over epidemiological data streams. So, how long ahead can our method forecast future epidemic patterns? Is there any difference between, say, 7-days-ahead and 28-steps-ahead forecasting results? We thus examined the forecasting power of EpiCast in terms of the future sequence length l_s.

Table 2 shows the forecasting errors of EpiCast and its competitors for varying the forecasting steps: $l_s = 7, 14, 21, 28$. More specifically, Table 2 shows the average forecasting errors (the root mean square error (RMSE)) between the original values (i.e., infected/death/recovered cases) and the l_s-days-ahead forecast values. We also compared multiple current window length $l_c = 14$, 21, 28 (i.e., $l_c = t_c - t_m + 1$), where l_c corresponds to the training length of each model. Here, a lower value indicates a better forecasting accuracy. In Table 2, the underlines show approaches providing the best performance. Our approach achieves a high forecasting accuracy for every combination of each l_s and each l_c, which means that our method is capable of modeling both the rise and fall-part patterns at any time in epidemic streams.

Similarly, Table 3 shows the individual forecasting errors between the original values and the l_s-days-ahead forecast values for infected, death and recovered cases, respectively. Here, we provide some observations. ARIMA missed location information since it only models a univariate sequence individually. Although its performance as regards the number of deaths and recovered is relatively better, this is because there are no complex dynamics in the these sequences, which grow slowly over time. The SIRD model can capture dynamics regarding these who recovered and how their medical states will change but cannot handle location information. The deep neural network models can take all the features into account for forecasting. Here, we show the best scores of the models; however, these models are still poor at capturing epidemic dynamics. Specifically, TCN can capture the long-term trend by convoluting the input sequence in the time direction; EpiDeep learns seasonality from a large amount of historical data; however, they are unsuitable for capturing the rapid rise/fall-part patterns. Furthermore, DNN is difficult to use for decision making because it is a black box and cannot represent the regime shifting of infectious diseases and similarity among locations.

In summary, EpiCast provides better forecasts than these baselines because it explicitly handles multiple regimes, and so the forecasts can be interpreted in terms of location and experience to date.

Table 2: Comparison of marginal/average forecasting error: RMSE between original and l_s-steps-ahead values of EpiCast and its competitors at each current window length l_c (lower is better).

l_c	l_s	EpiCast	SIRD	ARIMA	GRU	TCN	EpiDeep
14	7	.0330 ± .0327	.1325 ± .1166	.0630 ± .0398	.2498 ± .0925	.0942 ± .0171	.1333 ± .0826
	14	.0583 ± .0587	.1977 ± .1716	.0912 ± .0551	.2760 ± .1080	.0856 ± .0442	.1586 ± .1141
	21	.0841 ± .0833	.2541 ± .2100	.1172 ± .0682	.2947 ± .1267	.1035 ± .0622	.1845 ± .1827
	28	.1082 ± .1040	.2992 ± .2335	.1408 ± .0801	.3123 ± .1542	.1235 ± .0739	.2187 ± .3827
21	7	.0376 ± .0364	.1313 ± .0969	.0761 ± .0463	.2855 ± .1092	.0900 ± .0259	.1332 ± .0827
	14	.0632 ± .0642	.1844 ± .1383	.1032 ± .0603	.3093 ± .1292	.0902 ± .0520	.1584 ± .1144
	21	.0882 ± .0858	.2323 ± .1685	.1285 ± .0736	.3274 ± .1540	.1102 ± .0668	.1844 ± .1834
	28	.1114 ± .1040	.2725 ± .1886	.1517 ± .0893	.3432 ± .1849	.1306 ± .0768	.2187 ± .3846
28	7	.0426 ± .0419	.1288 ± .0922	.0884 ± .0519	.3038 ± .1347	.0927 ± .0312	.1361 ± .0981
	14	.0679 ± .0700	.1671 ± .1180	.1149 ± .0660	.3294 ± .1635	.0950 ± .0543	.1634 ± .1579
	21	.0926 ± .0927	.2080 ± .1465	.1397 ± .0828	.3480 ± .1963	.1152 ± .0684	.1936 ± .2859
	28	.1160 ± .1177	.2462 ± .1716	.1630 ± .1101	.3660 ± .2515	.1352 ± .0781	.2191 ± .3948

Table 3: Comparison of individual forecasting error: RMSE between original and l_s-steps-ahead values of EpiCast and its competitors at each current window length l_c (lower is better). Please also see text for more observations.

l_c	l_s	dimension	EpiCast	SIRD	ARIMA	GRU	TCN	EpiDeep
14	7	Infected	.0703 ± .0266	.2695 ± .1038	.1118 ± .0301	.2006 ± .0506	.1032 ± .0220	.2256 ± .0691
		Death	.0165 ± .0173	.0686 ± .0379	.0398 ± .0130	.2993 ± .1085	.0923 ± .0120	.0885 ± .0340
		Recovered	.0123 ± .0108	.0593 ± .0188	.0374 ± .0102	.2496 ± .0821	.0869 ± .0110	.0858 ± .0421
	14	Infected	.1247 ± .0412	.3852 ± .1670	.1576 ± .0421	.2357 ± .0549	.1371 ± .0414	.2639 ± .1311
		Death	.0305 ± .0418	.1311 ± .0763	.0602 ± .0217	.3271 ± .1382	.0596 ± .0099	.1072 ± .0386
		Recovered	.0197 ± .0149	.0768 ± .0229	.0557 ± .0157	.2652 ± .0944	.0601 ± .0087	.1046 ± .0628
	21	Infected	.1790 ± .0569	.4723 ± .2081	.1972 ± .0521	.2685 ± .0655	.1772 ± .0546	.3058 ± .2660
		Death	.0449 ± .0597	.1974 ± .1076	.0802 ± .0315	.3424 ± .1664	.0658 ± .0164	.1252 ± .0426
		Recovered	.0283 ± .0192	.0926 ± .0271	.0741 ± .0248	.2732 ± .1157	.0674 ± .0141	.1225 ± .0800
	28	Infected	.2272 ± .0701	.5315 ± .2306	.2301 ± .0636	.2996 ± .0988	.2109 ± .0640	.3742 ± .6305
		Death	.0599 ± .0742	.2590 ± .1319	.0993 ± .0392	.3535 ± .1854	.0788 ± .0217	.1424 ± .0462
		Recovered	.0374 ± .0225	.1072 ± .0311	.0930 ± .0413	.2838 ± .1598	.0807 ± .0184	.1393 ± .0940
21	7	Infected	.0801 ± .0247	.2503 ± .0670	.1320 ± .0351	.2305 ± .0627	.1158 ± .0299	.2256 ± .0693
		Death	.0194 ± .0227	.0769 ± .0449	.0497 ± .0176	.3378 ± .1309	.0783 ± .0081	.0883 ± .0335
		Recovered	.0133 ± .0112	.0667 ± .0180	.0465 ± .0131	.2882 ± .0966	.0760 ± .0076	.0856 ± .0421
	14	Infected	.1332 ± .0373	.3436 ± .1143	.1745 ± .0456	.2665 ± .0728	.1510 ± .0472	.2639 ± .1317
		Death	.0352 ± .0575	.1280 ± .0691	.0701 ± .0273	.3613 ± .1656	.0590 ± .0138	.1070 ± .0381
		Recovered	.0213 ± .0157	.0816 ± .0209	.0649 ± .0208	.3000 ± .1158	.0607 ± .0116	.1044 ± .0629
	21	Infected	.1856 ± .0488	.4179 ± .1467	.2122 ± .0570	.3019 ± .1089	.1889 ± .0590	.3059 ± .2675
		Death	.0490 ± .0704	.1833 ± .0870	.0895 ± .0348	.3727 ± .1867	.0696 ± .0194	.1250 ± .0420
		Recovered	.0301 ± .0196	.0956 ± .0240	.0840 ± .0363	.3077 ± .1495	.0721 ± .0158	.1223 ± .0801
	28	Infected	.2312 ± .0643	.4720 ± .1677	.2441 ± .0802	.3329 ± .1634	.2208 ± .0672	.3747 ± .6338
		Death	.0631 ± .0771	.2361 ± .1006	.1081 ± .0410	.3813 ± .2014	.0840 ± .0242	.1422 ± .0456
		Recovered	.0398 ± .0229	.1095 ± .0285	.1028 ± .0551	.3154 ± .1854	.0870 ± .0194	.1391 ± .0941
28	7	Infected	.0905 ± .0285	.2468 ± .0603	.1500 ± .0390	.2529 ± .0732	.1261 ± .0336	.2332 ± .1078
		Death	.0220 ± .0291	.0679 ± .0258	.0595 ± .0230	.3640 ± .1746	.0767 ± .0083	.0884 ± .0308
		Recovered	.0153 ± .0121	.0716 ± .0151	.0558 ± .0181	.2944 ± .1138	.0753 ± .0081	.0868 ± .0478
	14	Infected	.1425 ± .0429	.3130 ± .0836	.1905 ± .0503	.2931 ± .1078	.1582 ± .0495	.2783 ± .2245
		Death	.0375 ± .0647	.1030 ± .0475	.0792 ± .0302	.3852 ± .2004	.0625 ± .0152	.1070 ± .0349
		Recovered	.0237 ± .0165	.0853 ± .0176	.0749 ± .0324	.3100 ± .1569	.0645 ± .0130	.1048 ± .0630
	21	Infected	.1943 ± .0634	.3815 ± .1163	.2271 ± .0732	.3295 ± .1685	.1953 ± .0606	.3341 ± .4599
		Death	.0507 ± .0754	.1441 ± .0644	.0982 ± .0363	.3959 ± .2150	.0736 ± .0206	.1249 ± .0386
		Recovered	.0327 ± .0200	.0983 ± .0203	.0938 ± .0496	.3185 ± .1976	.0766 ± .0169	.1218 ± .0742
	28	Infected	.2413 ± .1032	.4391 ± .1498	.2600 ± .1248	.3663 ± .2823	.2266 ± .0686	.3776 ± .6538
		Death	.0642 ± .0817	.1880 ± .0809	.1163 ± .0409	.4039 ± .2225	.0876 ± .0254	.1422 ± .0425
		Recovered	.0427 ± .0235	.1115 ± .0243	.1129 ± .0722	.3277 ± .2455	.0914 ± .0204	.1374 ± .0801