Annals of Operations Research
https://doi.org/10.1007/s10479-025-06795-2

ORIGINAL RESEARCH

Novel methods for adaptive time-series forecasting and prediction-interval construction

Dhammika Amaratunga[1] · Javier Cabrera[2] · Nuria Diaz-Tena[3] · Michael N. Katehakis[3] · Chun-Pang Lin[2] · Jin Wang[3]

Received: 21 April 2024 / Accepted: 29 July 2025

Abstract
We propose novel methods for adaptive series forecasting and prediction-interval construction, illustrated with COVID-19 case and death counts. Our framework applies an automated transformation to reduce heteroscedasticity, then imposes a constrained smoothing near the forecast edge via robust quadratic regression, emphasizing recent data. A Long Short-Term Memory (LSTM) model combined with ARIMA-based noise correction further refines the forecast. Compared to conventional methods (e.g., ARIMA alone, unprocessed deep learning), this adaptive approach achieves superior metrics and reliable bootstrap-derived confidence and prediction intervals. We also highlight how reinforcement learning (RL) can offer promising avenues for real-time decision-making and further improvements in forecasting adaptability.

Keywords Adaptive learning · Pre-processing · Forecast · Confidence and prediction interval

1 Introduction

Machine learning (ML) techniques, including ARIMA (Box & Jenkins, 1970) and neural networks, are employed for time-series forecasting. However, short and volatile data series can lead to overfitting. By applying pre-processing and constrained smoothing, we enhance forecast accuracy and confidence, as demonstrated in the context of COVID-19 case counts predictions.

The global health landscape has been significantly impacted by the COVID-19 pandemic. Monitoring spatial and temporal disease progression and devising appropriate medical and socio-economic intervention strategies (Amaratunga et al., 2022) are paramount. For effective mitigation, forecasting the disease trajectory several weeks ahead is essential. This

Dhammika Amaratunga, Javier Cabrera, Nuria Diaz-Tena, Michael Katehakis, Chun Pang Lin, Jin Wang have contributed equally to this work.

Extended author information available on the last page of the article

Published online: 01 September 2025

paper introduces an adaptive machine learning methodology to project upcoming COVID-19 case numbers, accompanied by respective confidence and prediction intervals.

Unsurprisingly, extensive literature already exists on this topic. Broadly, the literature consists of models that can be categorized into epidemiological and machine learning models. Rahimi et al. (2021) offered a comprehensive review of COVID-19 forecasting models, including SIR (Susceptible-Infected-Removed), Moving Average, ARIMA, deep learning, et al. The foundational epidemiological model, the SIR model (Kermack & McKendrick, 1927), classifies a population into three distinct categories: susceptible (individuals at risk of infection), infected (transmitting the disease), and removed (recovered with immunity or no longer in the system). This model has been widely adopted, such as Rainisch et al. (2022), and extended to more complicated ones for various situations by adding more classes of individuals, e.g. SIRD (Susceptible-Infectious-Recovered-Dead)(Anastassopoulou et al., 2020) and SARIIqSq (Susceptible-Asymptomatic-Recovered-Infected Isolated-Infected (Iq)-Quarantined Susceptible) (Sarkar et al., 2020). La Gatta et al. (2021) integrated graph convolutional networks (GCNs) and sequential deep learning models to optimize the time-dependent parameters of an SIR model. However, SIR models were unable to face challenges in the long term COVID-19 pattern prediction (Moein et al., 2021). In the epidemiological models, parameters like the basic reproduction number (R0), infection rate and recovery rate required estimation, typically in large populations. The accuracy of these models could be significantly compromised by unforeseen interventions not incorporated in the model.

As for machine learning models for time-series analysis, ARIMA models works well for seasonal data but changes in observations and model specifications, such as mid-series shifts in variance and autocorrelations, make these models unstable. Cartus et al. (2022) applied ARIMA modeling to estimate drug overdose deaths in the US during the pandemic. Chintalapudi et al. (2020) applied ARIMA modeling to forecast the number of cases and recovered cases in Italy. The sophisticated deep learning models were widely used to analyze time-series data with many applications in the science (Rocha-Solache et al., 2022). The total number of COVID confirmed cases, trends and possible stopping times of the pandemic outbreak were predicted by applying deep learning models with Long Short-Term Memory (LSTM) layers (Chimmula and Zhang 2020; Shahid et al. 2020; Chandra et al. 2022). A deep learning model was compared with the Savitzky Golay Smoothing (Rasjid et al., 2021). Gal and Ghahramani (2016) and Lakshminarayanan et al. (2017) proposed methods for uncertainty estimations in deep learning-based forecasting. Reinforcement learning for time-Series forecasting and adaptive learning were discussed in Burnetas and Katehakis (1996) and Burnetas et al. (2025), and Multi-armed bandits were applied in dynamic forecasting (Auer et al., 2002; Slivkins, 2019). However, deep learning models do not always exhibit good performance (Zhao et al., 2022), especially when they were applied to predict COVID-19 daily cases. This was because they rely on large amounts of training data for obtaining high prediction accuracy. Additionally, the variance of the time-series data changed significantly over time. Moreover, these papers did not include methods to construct confidence and prediction intervals.

In this paper we developed the following algorithmic framework aimed at forecasting daily confirmed COVID cases effectively:

- **Part 1**: Pre-processing

(1) Transformation. Apply a nonlinear monotonic transformation to the data to reduce the skewness and stabilize the variance.
(2) Smoothing. Smooth the series with the smooth spline constrained to go through a robust estimator of the last point of the series (denoted as p) so as to emphasize the trend at that stage. The difference between the observed series and the smoothed nonlinear series represents the noise component, which itself forms a stationary series.

- **Part 2**: Forecasting by deep learning

(1) Model the smooth series by building a Recurrent Neural Network (RNN) with LSTM layers followed by a dense layer and use this model to forecast the trend.
(2) Fit an ARIMA model to the noise series and use it to forecast the noise.

- **Part 3**: Constructing bootstrap confidence and prediction intervals

The main contribution in this paper is to develop an adaptive learning methodology to forecast the time-series with heteroscedasticity and skewness as well as provide confidence and prediction intervals. Our methods showed excellent performances, especially at predicting changes in the trend. The detailed technical contributions are listed as follows:

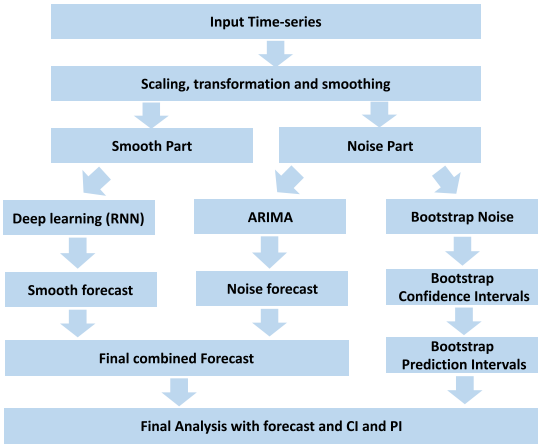
(1) Constrained smoothing of the data was proposed to emphasize the core trend pattern of the data, especially at the edge where forecasting is to be done.
(2) We combined deep learning to forecast the smooth trend of the data and ARIMA models to forecast the noise signal of the data and showed that this combination improved the overall forecasting — both the forecasts and the associated confidence intervals.
(3) An automated transformation of the data was used to reduce heteroscedasticity and skewness.
(4) We proposed a way to produce confidence and prediction intervals by implementing the bootstrap in combination with Deep Learning and using the smoothing constraint at the edge.
(5) Our methods showed good performance at predicting the changing trend.

In the following sections we will further describe the three parts of the methodology and assess the performance of the overall procedure for forecasting the evolution of COVID-19 in New Jersey. We found that the methods worked well for four weeks forecasting, and that the coverages of the confidence and prediction intervals constructed were very good. These findings were substantiated by applying the same methodology to COVID-19 data from California and flu data from Hong Kong.

2 Methods

Figure 1 shows the adaptive learning procedure. We specified the procedure step by step as follows.

Fig. 1 Adaptive learning procedure



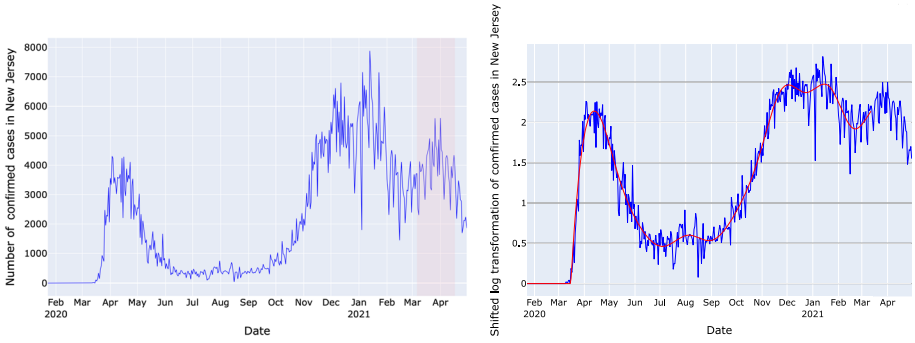


Fig. 2 Number of confirmed cases in New Jersey, transformation and smoothing. **a** The number of COVID-19 confirmed cases during the period 1/23/2020–5/1/2021. The period that we would forecast was marked in orange; **b** the time-series after the nonlinear monotonic transformation (blue line), an example of smoothing splines of the transformed time-series (red line)

2.1 Data source

The daily cumulative counts of confirmed COVID-19 cases in New Jersey during the period 1/23/2020–5/1/2021 were downloaded from the Johns Hopkins University Center for Systems Science and Engineering (JHU CSSE) (Dong et al., 2020) and the number of daily cases was calculated (see Fig. 2a).

Given data for a certain period, our goal was to forecast the number of daily confirmed cases during the following 28 days and construct corresponding confidence and prediction intervals. To test our methodology in different scenarios, the period from 3/6/2021–4/17/2021 (the orange highlighted part of Fig. 2a) was selected. This period was selected because it included an increasing trend, a change in trend, and a decreasing trend. We forecasted the number of cases within three intervals with a length of 28 days within this period: 3/6/2021–4/3/2021, 3/13/2021–4/10/2021, and 3/20/2021–4/17/2021. That is, data from 1/23/2020 to 3/5/2021 was used to forecast the case counts for the next 28 days (from

3/6/2021 to 4/3/2021). We repeated this process for the other two intervals by moving forward one week.

2.2 Nonlinear monotonic transformation

We denoted the raw counts as y_t, $t = 1, 2, 3, \cdots, T$. In Fig. 2a, the variances of the raw data changed over time. Specifically, the variance was large when the number of cases was large, and the variance was much smaller when the number of cases was small in the range 0–1000. In addition, the error distribution appeared to be right-skewed. We considered a large number of automated smoothing monotonic transformations, including shifted power transformations $z_t = (a + y_t)^p$, shifted-log transformation $z_t = \log(a + y_t)$, shifted arc-tangent and others. The reason for the shift was to avoid gaps around zero counts that were caused by optimizing the skewness alone. The statistic that minimized the skewness while avoiding gaps around the zero was called the zero-gap statistic (Cabrera & McDougall, 2002). Based on the zero-gap statistic we chose the shifted-log transformation for this data; for other data other transformations may work better. The good value that reduced the error skewness and made the error variance approximately homogeneous without creating a big gap around zero was $a = 500$. The plot of the time-series data after the transformation was shown as the blue line in Fig. 2b. The transformed series was denoted as z_t, $t = 1, 2, 3, \cdots, T$.

2.3 Smoothing with an edge constraint

One of characteristics of the data was that it was an aggregate of cases reported by all the counties of New Jersey. This reporting did not happen daily, it varied from county to county, hence this induced high variability which can impair the time-series modeling. For this reason, after applying the shifted-log transformation, the data was smoothed using splines (Wahba, 1975) to reduce the variability. The degrees of freedom for the spline were chosen to balance the goodness-of-fit and the smoothness of the estimated function. The elbow and the second derivative methods produced 14 as a reasonable value for the degrees of freedom for the data from 1/23/2020 to 3/14/2021 The corresponding smooth line is shown as a red line in Fig. 2b.

Another important issue with splines or any smoothing method is the high uncertainty at the edges. The most important point for forecasting is the edge at the right-hand side of the series. One possible way to moderate this issue is to estimate the right side edge point of the smooth series (denoted as ψ_T) by robustly fitting a quadratic model to the last k values of the time-series (in this case $k = 28$ was used) using MM-estimation and then to fit a smoothing spline to the entire time-series with the right side edge point constrained to be equal to the value that was obtained by the quadratic fit. We tried different values for the length k, e.g. 7, 14, 28, and 32; The value of $k = 28$ produced the best quadratic fitting that effectively captured the recent trend. The penalized regression problem for the smoothing splines with an edge constraint was presented as follows.

$$\min_f \sum_t (z_t - f(t))^2 + \lambda \int f''(t)^2 dt \quad \text{s.t.} \quad f(T) = \psi_T$$

In order to demonstrate the efficacy of our approach, we chose three consecutive edge points as illustrative examples: March 14th, 15th and 16th, 2021, for the time-series. As expected, the smoothed time-series without fixing the edge point fluctuated greatly at the right edge point. Compared to this, the variability of the splines with the edge point constraint was much less. Figure 3 illustrates the effect of smoothing splines for three consecutive time-series, comparing cases with and without an edge constraint. The short-term forecasts without fixing the edge constraint were sensitive to the last values, while our method was robust. The smoothed series was denoted as $\mu_t, t = 1, 2, 3, \cdots, T$. After smoothing the transformed series, the residuals, denoted as ε_t, can be obtained. Hence, we had the four resulting series:

y_t: raw counts, z_t: transformed series, μ_t: smoothed series, ε_t: residuals.

2.4 Forecasting by a recurrent neural network

Recurrent neural network (RNN) models have been widely applied for time-series data prediction. We built a RNN model with three Long Short-Term Memory (LSTM) layers (with 300, 200, and 100 LSTM cells respectively) and one dense layer with one neuron, which was trained by using the smooth data. The time step was set as 7, that is, using the previous 7 days' data to predict the next day, which was used as a new input to predict the following value, and so on. We tested different settings of the parameters in the model. We tried larger time steps (e.g., 10 and 14) and more layers (e.g., 5 and 8), but this did not result in a better fit. Also, we tried fewer cells in the layers (e.g., 50 and 20), but the model fitting was worse. The data was divided into three parts: the first 80 percent for training, the next 10 percent for validation, and the remaining 10 percent for testing. During training, the learning rate was reduced by multiplying it by 0.1 if the training loss did not improve for five epochs. The threshold for measuring the new optimum was 1e−4. The learning rate started at 0.001 and the lower bound was 1e−10. We also applied early stopping mechanics. It would stop training when the loss stopped improving within 20 epochs. The hyper-parameters and training settings in the LSTM model were presented in Appendix A. After training the model, we performed forecasting for the future 28 days. Finally, the forecasted values were trans-

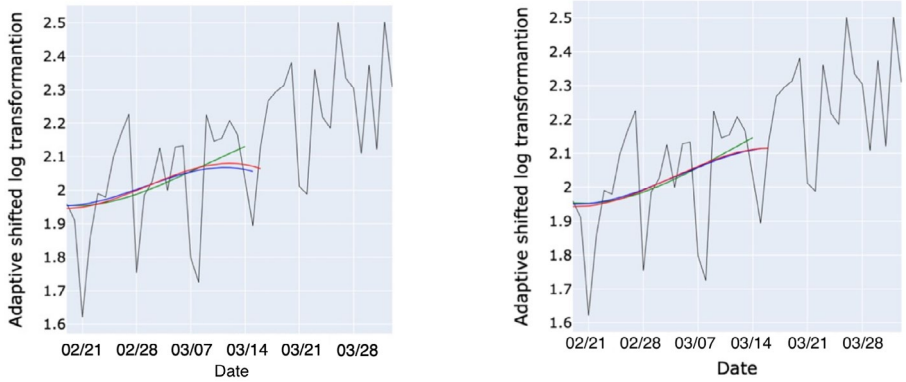


Fig. 3 Comparisons of smooth splines with and without an edge constraint. The black line was the shifted log transformed data. **a** Smoothing splines without an edge constraint; **b** Smoothing splines with an edge constraint. The three smooth lines in different colors indicated smoothing splines of the data sets ending on different dates, i.e., 3/15/2021, 3/16/2021, and 3/17/2021

formed from log scale back to the normal scale by using the inverse function of shifted log transformation. The forecast was combined with the noise forecast to produce the final forecast, $\hat{y}_t = \hat{\mu}_t + \hat{\varepsilon}_t$. The metrics applied to evaluate the predictions were mean absolute error (MAE) and median absolute error (MedAE).

$$\begin{aligned} \text{MAE} &= \frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y}_i| \\ \text{MedAE} &= \text{Med}_i(|y_i - \hat{y}_i|) \end{aligned}$$

2.5 Bootstrap algorithm for forecasting time-series

We propose the following bootstrap algorithm for forecasting time-series as follows.

(1) Conduct a block bootstrap of the residuals from the smoothed data and get *B* time-series of bootstrap residuals (Kunsch, 1989; Politis & White, 2004). In this work $B = 1000$ was used.
(2) Add the bootstrap residuals to the transformed smoothed series to obtain *B* bootstrap time-series.
(3) Smooth the *B* bootstrap time-series with the algorithm proposed in Sect. 2.3.
(4) Fit the LSTM model proposed in Sect. 2.4 to each of the *B* smoothed time-series to get *B* forecasts.
(5) Compute the bootstrap forecast by taking the median or the mean of the *B* forecasts.

This algorithm was applied to the time-series of transformed COVID-19 data from New Jersey. It required training the LSTM model *B* times, which was computationally intensive. However, the computation could be fully parallelized and with multi-processors would be quite fast.

2.6 Confidence and prediction intervals

We next introduce an algorithm aimed at computing confidence and prediction intervals for predictions generated through deep learning techniques. The heuristic basis of our approach is to view deep learning estimates from the RNN model as being the outcomes of averages of a large number of nonlinear functions, and therefore they should approximately follow the central limit theorem.

Let $\hat{y}$ be the location estimate calculated by an M-estimator of the 1000 bootstrap forecasts at time *t*. To estimate the scale or standard deviation, a simple method is to calculate the 2.5% and the 97.5% quantiles (*L*, *U*) of the bootstrap forecasts; the difference $(U - L)/3.92$ is an estimator of the scale. Alternatively, the scale could be estimated by a robust τ estimator.

The $(1-\alpha)100\%$ confidence interval for the forecast at time *t* is the interval $(L_{\alpha,i}, U_{\alpha,i})$, where $L_{\alpha,i}$ and $U_{\alpha,i}$ are the $\alpha/2$ and $1-\alpha/2$ quantiles of the 1000 bootstrap forecasts at time *i*.

A $(1-\alpha)100\%$ prediction interval for the forecast at time i can be constructed as follows. With the bootstrap forecasts we built a $(1-\alpha)100\%$ confident interval for the fore-

cast at time i. The same bootstrap forecasts may be used to estimate the standard error of the forecast at time i, denoted as s_i. On the other hand, we can also estimate the residuals standard deviation, s_ε. Since we use 1000 bootstraps and more than 500 residuals to estimate s_i and s_ε, the degrees of freedom of both estimates are very large and therefore we may use the normal distribution to generate an approximate the prediction interval instead of a t distribution. Therefore, we propose the classical $(1-\varepsilon)100\%$ prediction interval as follows.

$$\hat{y}_i \pm Z_{\alpha/2}\sqrt{s_i^2 + s_\varepsilon^2}$$

where $Z_{\alpha/2}$ is the $1-\alpha$ quantile of normal distribution.

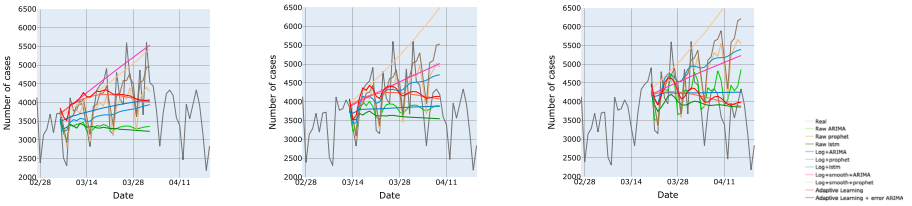
Alternatively, we can construct the bootstrap prediction intervals by adding bootstrap residuals to the bootstrap predictions and taking an envelope that contains $(1-\alpha)100\%$ of the points at each time point t in the forecasting period. From our experience with a few examples of COVID-19 cases time-series data, both approaches produced reasonable prediction intervals.

On rare occasions, due to sampling fluctuations, we may observe a temporary minor shrinking over time of the bootstrap confidence interval and/or the prediction interval. To prevent this, we set the distance between the bounds of the interval to be nondecreasing as a function of time.

We also examined the coverage of the confidence and prediction intervals as follows. We smoothed the series (the training period together with the forecasting period) with the smooth spline coerced to go through the fixed edge point p obtained in Sect. 2.3. Then, the bootstrap residuals were put onto the smoothed data to obtain 1000 sample series. We checked the coverage of the prediction interval by calculating what percentage of the sample series within the forecasting period were located within the prediction interval. Additionally, we smoothed the 1000 sample series with the smoothing splines coerced to go through the point ψ_T. The coverage of the confidence interval was obtained by calculating what percentage of the smoothed sample data in the forecasting period were located within the confidence interval.

3 Results

The above methods were applied to the time-series of COVID-19 cases in New Jersey from 1/1/2020 to 4/17/2021. Table 2 in Appendix B presents the forecasting comparison in the three intervals within the period from 3/6/2021 to 4/17/2021 (as defined in Sect. 2.1) by implementing different methods. Figure 4 illustrates the forecasts on the log scale. In the first three methods, we modeled the raw data by fitting an ARIMA model, PROPHET model (Taylor & Letham, 2017) and a deep learning model with LSTM cells respectively. In fitting the ARIMA model, we adopted the auto.arima function in the R forecast package (Hyndman & Khandakar, 2008), in which an optimal ARIMA model was selected by minimizing the corrected Akaike Information Criterion based on a stepwise search over different ARIMA models. In fitting the PROPHET model, we used a grid search to select the hyper-parameters. From the fourth through sixth models, we fitted the ARIMA and the deep learning models after performing the shifted log transformation. From seventh through ninth methods,



(a) 3/6/2021-4/3/2021 (b) 3/13/2021-4/10/2021 (c) 3/20/2021-4/17/2021

Fig. 4 28-day forecasting for number of confirmed cases in New Jersey in three intervals within the period from 3/6/2021 to 4/17/2021 by applying different methods

we applied a spline smooth to smooth the transformed data and fitted the smoothed data by ARIMA and deep learning models respectively. In the last model, in addition to the adaptive learning, we fitted an ARIMA model for the stationary noise series obtained after smoothing the transformed data. Then the last and final forecasting was the aggregate of the smooth forecast and the noise forecast. The ten methods were compared in terms of the different metrics, i.e., MAE and MedAE, using the daily number of COVID cases in New Jersey.

Table 2 shows 14-day and 28-day forecasting comparisons for the ten methods in three separate time intervals for New Jersey series, prior to the peak around 3/28/2021. The intervals were 3/6/2021–4/3/2021, 3/13/2021–4/10/2021, and 3/20/2021–4/17/2021, respectively. Figure 4 shows the forecast results compared to the original time-series in the raw scale. The adaptive learning method proposed in this paper showed the best performance in the majority of the periods. In a few of the short term (14-day) forecasts, the PROPHET models performed slightly better possibly because it was near peak where the time-series was less nonlinear. Furthermore, our method was able to forecast the time-series in the interval where there was a nonlinear trend, while the ARIMA and PROPHET models for both the log data and the smoothed data only gave reasonable forecasts for time periods without much nonlinearity.

To assess the general applicability of our approach, we also applied the same methods to predict the number of confirmed cases in California. The data from 1/23/2020 to 8/8/2021, was used to train the models. We forecasted the number of cases within three intervals with a length of 28 days after 8/8/2021. $a = 2000$ was in the shift-log transformation. Table 3 shows 14-day and 28-day forecasting comparisons for the ten methods in three separate time intervals for California near the peak around 8/21/2021. The intervals were 8/9/2021–9/5/2021, 8/16/2021–9/12/2021, and 8/23/2021–9/19/2021, respectively. Figure 6 shows the forecast results compared to the original time-series for California in the raw scale. The results show that adaptive learning with and without ARIMA of noise series outperformed the other methods. There were a few cases where the addition of the ARIMA noise component made the performance a little worse, but never by much. In general, the adaptive learning with the ARIMA noise component was the best performer.

To further assess the generality of our approach, we applied it to predict Hong Kong flu data. The weekly counts of flu cases in Hong Kong from 12/29/2013 to 3/10/2019 were downloaded from the Department of Health, HKSAR (Department of Health, 2023) and illustrated in Fig. 7a. This time-series showed different characteristics from COVID data in terms of variance and skewness. $a = 0$ was used in the shift-log transformation. Like COVID predictions, periods with increasing, change, and decreasing trends, from 12/16/2018 to

2/17/2019 were selected for forecasting (the orange highlighted part of Fig. 7a). We forecasted the counts of cases within three intervals each with a length of 8 weeks within this period: week of 12/23/2018-week of 2/10/2019, week of 12/30/2018-week of 2/17/2019, and week of 1/6/2019-week of 2/24/2019. The plot of the time-series data after the transformation and the smoothing splines were shown as the blue and red lines in Fig. 7b.

Table 4 shows 8-week forecasting for the number of flu cases in three intervals. Figure 8 illustrates the short term (4-week) and long term (8-week) forecasting comparisons in terms of MAE and MedAE. The adaptive learning method proposed in this paper showed the best performance in most of the periods, especially for long-term forecasting. It managed to forecast the changes in the trend again. The PROPHET models performed slightly better in the short-term forecasting in log scale, but it failed to forecast changes in the trend.

Figure 5 shows the 28-day 90% bootstrap and classical confidence and prediction intervals in the three time intervals proposed in Sect. 2.6. The prediction intervals covered most of the data in the forecasting period. Table 1 demonstrates the coverages of the 90% intervals, which are very close to 90%. The confidence interval also covered smooth data 90% of the time in prediction periods. The classical and bootstrap prediction intervals came out almost the same.

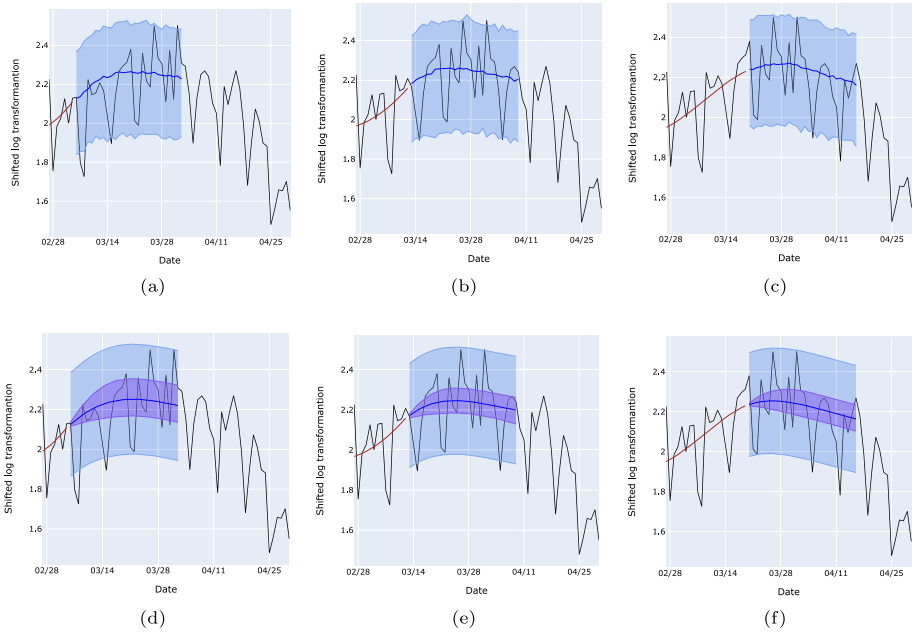


Fig. 5 90% Bootstrap prediction intervals and bootstrap and classical confidence intervals. **a**–**c** Are bootstrap prediction interval for the periods, 3/6/2021–4/3/2021, 3/13/2021–4/10/2021, and 3/20/2021–4/17/2021. **d**–**f** Are bootstrap confidence interval (purple area) and classical prediction interval (blue area) for the three periods. The blue line represents the median of the forecasts, while shaded areas indicate the bootstrap confidence and prediction intervals

Table 1 The coverage of the 90% confidence and prediction interval in the three intervals within the period from 3/6/2021 to 4/17/2021

Time intervals	3/6-4/3	3/13-4/10	3/20-4/17
Bootstrap confidence interval	89.40%	89.18%	89.06%
Bootstrap prediction interval	89.82%	90.52%	90.49%
Classical prediction interval	86.09%	89.10%	89.02%

4 Discussion and conclusions

In this paper, we developed a novel adaptive learning procedure for time-series forecasting and confidence-and-prediction-interval construction. The procedure consists of the following steps: (1) A non-linear transformation was used to address data irregularities that could impact the effectiveness of the predictive models. (2) This was followed by smoothing the data using a spline smooth with the right end point fixed to emphasize the importance of the trend during the last several time points for forecasting. The fixed point value was obtained by running robust quadratic regression for the last four weeks of the available data. (3) LSTM deep learning modeling was applied to train the smoothed data. (4) Lastly, the noise term after smoothing was forecasted using ARIMA and this was then combined with the smoothing forecast to obtain the final forecast.

To test our method, we analyzed daily COVID-19 cases for New Jersey, California and weekly flu cases for Hong Kong, and performed forecasting for three 28-day and 8-week intervals that showed different trend patterns, i.e., an increasing trend, a change in trend, and a decreasing trend. The results demonstrated the effectiveness and applicability of our method. Compared with traditional methods, e.g., ARIMA, PROPHET, and LSTM models, the adaptive learning method proposed in this paper generated forecasts with the best performance in terms of various metrics, including MAE and MedAE, especially when there were changes in the trend, while other models like PROPHET can give results were way off even with constrained smoothing when there were changes in trend. In monotonic periods, our method always can give decent forecasts (MAEs were not too far from the best among other methods).

We also proposed a novel procedure to obtain confidence and prediction intervals for the forecasts. The main idea is to first incorporate bootstrap errors into the smoothed data, resulting in the creation of 1000 sample series. Subsequently, we execute our proposed adaptive learning method to conduct forecasting for each of these sample series. It was established that these intervals were consistent with the trends in the data and showed strong coverage performance. Overall, the adaptive learning procedure proved effective for forecasting COVID-19 daily cases.

Further insights derived from the research presented in this paper encompass the following. It is a prerequisite to do extensive preprocessing when employing deep learning methods. This includes shifted-log transformation and smoothing. In this paper, the time-series was smoothed to avoid the high variability. It would have been easier to use larger time intervals such as weekly or monthly data (rather than daily data) but this would render the data insufficient for deep learning modeling. Besides pre-processing, another useful point is that combining deep learning with classical methods like ARIMA for the noise did improve the forecasting. Finally, the confidence and prediction intervals provide a way to assess the degree of certainty of the forecasts.

Future research direction could be integration of reinforcement learning into time-series forecasting and uncertainty estimation. Adaptive forecasting framework proposed in this

paper could be extended via Multi-Armed Bandits (MABs). A MAB framework could dynamically select the best forecasting model (e.g., ARIMA, LSTM, PROPHET) based on real-time performance. Upper Confidence Bound and Thompson Sampling could allocate exploration-exploitation trade-offs for different forecasting models.

Appendix A: LSTM hyper-parameters

Architecture:

Layer	Units	Return sequences
LSTM_1	300	Yes
LSTM_2	200	Yes
LSTM_3	100	No
Dense (linear)	1	–

- **Additional details:**Input window: 7×1
- Kernel initializer: glorot_uniform
- Recurrent activation: sigmoid
- Activation: tanh
- Activity regularizer: l2

- Total trainable parameters: 883,701**Training set-up:**Loss: mean-squared error (MSE).
- Optimizer: Adam
- LR schedule: ReduceLROnPlateau (factor 0.1, patience 5, min lr 1e-10, min delta 0.0001).
- Early stopping: patience 20, monitor training loss, restore best weights.
- Batch size: 10; epochs: max 1000 (typically stops after 400–800)

- Validation split: 10% of the chronological training data, no shuffling.**Pre-/post-processing**Standardize the transformed + smoothed series (zero mean, unit variance) using training data only.
- Form samples: $X_t = [z_{t-6}, \cdots, z_t] \mapsto z_{t+1}$ (stride 1).
- After prediction, inverse the scaling and the shifted-log transform.

Appendix B: Table of forecasting comparison for NJ

Sec Appendix Table 2

Table 2 Forecasting comparison with forecast horizon (14–28 days) in three intervals within the period from 3/6/2021 to 4/17/2021 by applying different methods

Starting date	3/6/2021			
Length	14-day		28-day	
	MAE	MedAE	MAE	MedAE
Raw data ARIMA	583.47	522.33	823.01	673.68
Raw data Prophet	317.95	279.35	443.13	405.86
Raw data LSTM	637.87	566.87	884.41	742.09
Log+ARIMA	493.20	566.05	621.94	622.03
Log+Prophet	**312.06**	266.65	**390.10**	**324.47**
Log+LSTM	488.33	472.51	589.29	515.29
Log+smooth+ARIMA	566.76	226.31	735.51	501.61
Log+smooth+Prophet	452.91	**150.51**	618.61	471.50
Adaptive learning	521.98	244.17	605.65	469.96
Adaptive learning with ARIMA for noise	501.43	216.92	590.60	477.41
Starting date	3/13/2021			
Length	14-day		28-day	
	MAE	MedAE	MAE	MedAE
Raw data ARIMA	475.30	370.56	523.67	416.62
Raw data Prophet	412.78	325.28	486.07	481.24
Raw data LSTM	662.18	577.78	700.77	698.04
Log+ARIMA	489.64	395.38	546.18	442.29
Log+Prophet	**394.25**	**285.99**	565.61	508.98
Log+LSTM	637.21	614.11	615.27	623.34
Log+smooth+ARIMA	592.34	471.54	660.45	574.63
Log+smooth+Prophet	608.24	516.81	1093.87	880.87
Adaptive learning	589.57	453.56	548.23	**397.26**
Adaptive learning with ARIMA for noise	486.21	453.58	**478.22**	422.43
Starting date	3/20/2021			
Length	14-day		28-day	
	MAE	MedAE	MAE	MedAE
Raw data ARIMA	492.63	404.65	524.06	496.33
Raw data Prophet	484.97	**390.63**	876.40	821.70
Raw data LSTM	666.23	565.83	553.51	462.63
Log+ARIMA	598.74	588.36	987.06	860.95
Log+Prophet	**475.90**	471.89	1016.02	908.47
Log+LSTM	658.21	600.65	593.50	934.16
Log+smooth+ARIMA	620.71	588.18	921.75	551.59
Log+smooth+Prophet	781.57	659.43	1685.55	1574.66
Adaptive learning	667.56	592.37	559.57	**405.30**
Adaptive learning with ARIMA for noise	598.29	627.21	**504.45**	454.09

"Log" means the shifted log transformation; "ARIMA" means the ARIMA model; "LSTM" means deep learning model with LSTM cells; "smooth" means smooth splines

The bold font represents the best results

Appendix C: Table and figures of forecasting comparison for CA

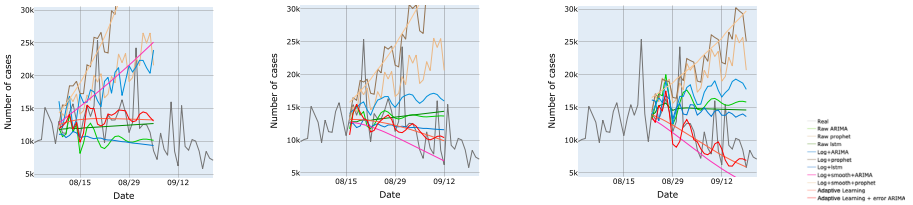
Sec Appendix Table 3 and Fig. 6.

Table 3 Forecasting comparison with forecast horizon (14–28 days) for the number of confirmed Covid cases in California in three intervals within the period from 8/9/2021 to 9/19/2021 by applying different methods

Starting date	8/9/2021			
Length	14-day		28-day	
	MAE	MedAE	MAE	MedAE
Raw data ARIMA	2988.55	1799.65	3336.40	2535.41
Raw data Prophet	3305.88	2722.59	5907.27	5644.57
Raw data LSTM	2724.68	1757.34	2476.84	1502.99
Log+ARIMA	3562.00	3038.17	5469.80	4748.25
Log+Prophet	5949.08	5121.11	14976.10	13212.08
Log+LSTM	3350.41	2787.78	3668.62	2932.77
Log+smooth+ARIMA	2888.86	2340.05	5679.72	4909.44
Log+smooth+Prophet	5832.92	5087.74	14402.65	13704.24
Adaptive learning	2195.66	1668.02	**2133.70**	**1354.62**
Adaptive learning with ARIMA for noise	**2159.57**	**1302.72**	2149.45	1404.89
Starting date	8/16/2021			
Length	14-day		28-day	
	MAE	MedAE	MAE	MedAE
Raw data ARIMA	**2015.92**	1564.63	3053.82	1945.47
Raw data Prophet	3524.49	3543.23	7065.02	5968.08
Raw data LSTM	2244.30	1120.53	3281.53	2254.12
Log+ARIMA	2241.04	1400.95	4202.45	4046.99
Log+Prophet	5807.33	6767.07	12536.68	9830.32
Log+LSTM	2653.16	1774.66	2861.30	2112.53
Log+smooth+ARIMA	3073.16	2227.10	3111.18	2170.53
Log+smooth+Prophet	6299.59	6957.62	13287.75	11042.55
Adaptive learning	2322.05	1436.17	2493.36	1762.69
Adaptive learning with ARIMA for noise	2132.73	**1052.73**	**2466.96**	**1429.31**
Starting date	8/23/2021			
Length	14-day		28-day	
	MAE	MedAE	MAE	MedAE
Raw data ARIMA	3598.07	3740.76	4860.42	5210.45
Raw data Prophet	5269.49	4933.75	8869.59	8945.87
Raw data LSTM	2669.11	2181.14	4051.81	3501.34
Log+ARIMA	3600.01	3197.67	5992.98	6233.37
Log+Prophet	5721.65	4631.11	10604.34	10179.79
Log+LSTM	2627.72	2265.00	3798.60	3675.60
Log+smooth+ARIMA	2789.48	2036.05	3451.24	2474.33
Log+smooth+Prophet	6109.27	5480.61	11294.60	10481.33
Adaptive learning	**2030.80**	1172.00	**2324.09**	1575.97
Adaptive learning with ARIMA for noise	2051.66	**931.56**	2438.69	**1381.40**

"Log" means the shifted log transformation; "ARIMA" means the ARIMA model; "LSTM" means deep learning model with LSTM cells; "smooth" means smooth splines

The bold font represents the best results

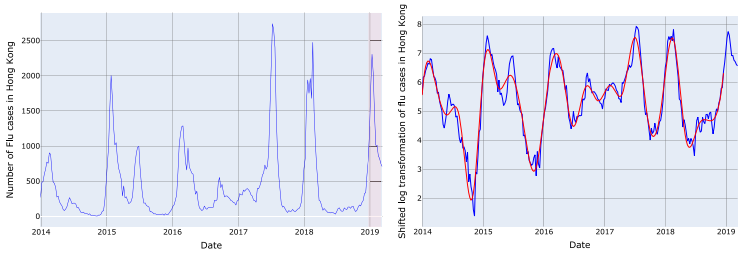


(a) 8/9/2021-9/5/2021 (b) 8/16/2021-9/12/2021 (c) 8/23/2021-9/19/2021

Fig. 6 28-day forecasting for number of confirmed cases in California in three intervals within the period from 8/9/2021 to 9/19/2021 by applying different methods

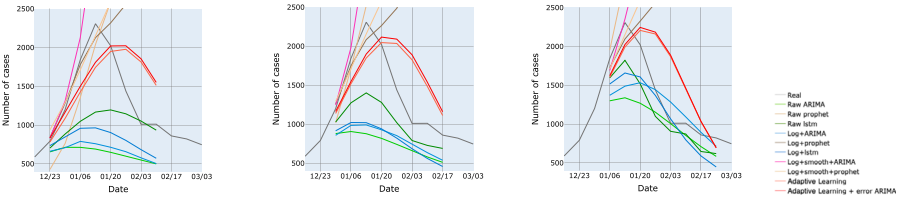
Appendix D: Table and figures of forecasting comparison for Hong Kong flu data

Sec Appendix Table 4 and Figs. 7, 8.



(a) (b)

Fig. 7 The weekly counts of flu cases in Hong Kong, transformation and smoothing. **a** The weekly counts of flu cases in Hong Kong from 12/29/2013 to 3/10/2019. The period that we would forecast was marked in orange; **b** the time series after the nonlinear monatomic transformation (blue line), an example of smoothed time series of the transformed time series (red line)



(a) 12/23/2018-2/10/2019 (b) 12/30/2018-2/17/2019 (c) 1/6/2019-2/24/2019

Fig. 8 8-week forecasting for number of flu cases in Hong Kong in three intervals within the period from 12/23/2018 through 2/24/2019 by applying different methods

Table 4 Forecasting comparison with forecast horizon (4–8 weeks) in three intervals within the period from 12/23/2018 to 2/24/2019 by applying different methods

Starting date	12/23/2018			
Length	4-week		8-week	
	MAE	MedAE	MAE	MedAE
Raw data ARIMA	842.57	808.36	819.93	676.12
Raw data Prophet	389.24	400.54	999.13	528.53
Raw data LSTM	584.21	554.92	446.99	304.51
Log+ARIMA	805.44	772.38	781.71	645.16
Log+Prophet	**93.40**	**73.21**	780.13	**238.05**
Log+LSTM	659.00	618.16	644.84	538.22
Log+smooth+ARIMA	403.18	199.42	5004.57	2378.80
Log+smooth+Prophet	130.28	138.84	987.46	359.82
Adaptive learning	275.35	264.46	377.14	458.58
Adaptive learning with ARIMA for noise	228.94	192.15	**360.64**	420.26
Starting date	12/30/2018			
Length	4-week		8-week	
	MAE	MedAE	MAE	MedAE
Raw data ARIMA	972.18	1070.27	714.67	564.29
Raw data Prophet	423.88	196.20	1670.23	1681.06
Raw data LSTM	597.54	656.39	435.29	**351.18**
Log+ARIMA	902.70	976.24	645.01	480.70
Log+Prophet	**155.39**	**159.72**	1088.94	643.95
Log+LSTM	868.55	951.57	658.58	534.00
Log+smooth+ARIMA	1062.08	520.61	7372.04	4938.55
Log+smooth+Prophet	235.91	141.15	1383.46	1070.32
Adaptive learning	217.53	193.61	**375.79**	392.48
Adaptive learning with ARIMA for noise	208.82	195.71	399.11	353.85
Starting date	1/6/2019			
Length	40-week		8-week	
	MAE	MedAE	MAE	MedAE
Raw data ARIMA	638.11	648.68	387.81	261.93
Raw data Prophet	1236.89	1084.99	2441.64	3021.85
Raw data LSTM	394.47	413.16	279.41	230.75
Log+ARIMA	446.34	480.95	**285.22**	**191.36**
Log+Prophet	438.46	257.66	1401.50	1503.83
Log+LSTM	**365.41**	371.06	297.31	298.85
Log+smooth+ARIMA	951.48	634.10	2299.30	2918.40
Log+smooth+Prophet	605.74	359.82	1819.11	1994.76
Adaptive learning	**365.43**	277.94	382.62	277.94
Adaptive learning with ARIMA for noise	367.62	**252.71**	387.69	252.71

"Log" means the shifted log transformation; "ARIMA" means the ARIMA model; "LSTM" means deep learning model with LSTM cells; "smooth" means smooth splines

The bold font represents the best results

Funding Dr. Michael Katehakis was funded in part by a grant from National Science Foundation (CMMI-1662629). Dr. Javier Cabrera was funded in part from a grant from the National Heart, Lung, and Blood Institute, National Institutes of Health (R01-HL150065). Dr. Javier Cabrera and Dr. Chun Pang Lin were funded in part by a grant from Johnson & Johnson, award no. AWD00008043-MOD001.

Declarations

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

Amaratunga, D., Cabrera, J., Ghosh, D., Katehakis, M. N., Wang, J., & Wang, W. (2022). Socio-economic impact on COVID-19 cases and deaths and its evolution in New Jersey. *Annals of Operations Research, 317*, 5–18. https://doi.org/10.1007/s10479-021-03941-4

Anastassopoulou, C., Russo, L., Tsakris, A., & Siettos, C. (2020). Data-based analysis, modelling and forecasting of the COVID-19 outbreak. *PLoS ONE, 15*(3), Article e0230405. https://doi.org/10.1371/journal.pone.0230405

Auer, P., Cesa-Bianchi, N., & Fischer, P. (2002). Finite-time analysis of the multiarmed bandit problem. *Machine Learning, 47*(2), 235–256. https://doi.org/10.1023/A:1013689704352

Box, G. E. P., & Jenkins, G. M. (1970). *Time series analysis: Forecasting and control*. Holden-Day.

Burnetas, A. N., Kanavetas, O., & Katehakis, M. N. (2025). Optimal data driven resource allocation under multi-armed bandit observations. *Annals of Operations Research*. https://doi.org/10.1007/s10479-025-06554-3

Burnetas, A. N., & Katehakis, M. N. (1996). Optimal adaptive policies for sequential allocation problems. *Advances in Applied Mathematics, 17*(2), 122–142. https://doi.org/10.1006/aama.1996.0007

Cabrera, J., & McDougall, A. (2002). *Statistical consulting*. Springer. https://doi.org/10.1007/978-1-4757-3663-2

Cartus, A. R., Li, Y., Macmadu, A., Goedel, W. C., Allen, B., Cerd, M., & Marshall, B. D. L. (2022). Forecasted and observed drug overdose deaths in the US during the COVID-19 pandemic in 2020. *JAMA Network Open, 5*(3), Article e223418. https://doi.org/10.1001/jamanetworkopen.2022.3418

Chandra, R., Jain, A., & Chauhan, D. S. (2022). Deep learning via LSTM models for COVID-19 infection forecasting in India. *PLoS ONE, 17*(1), Article e0262708. https://doi.org/10.1371/journal.pone.0262708

Chimmula, V. K. R., & Zhang, L. (2020). Time series forecasting of COVID-19 transmission in Canada using LSTM networks. *Chaos Solitons & Fractals, 135*, Article 109864. https://doi.org/10.1016/J.CHAOS.2020.109864

Chintalapudi, N., Battineni, G., & Amenta, F. (2020). COVID-19 virus outbreak forecasting of registered and recovered cases after sixty day lockdown in Italy: A data driven model approach. *Journal of Microbiology, Immunology, and Infection, 53*(3), 396–403. https://doi.org/10.1016/j.jmii.2020.04.004

Department of Health, H. (2023). *Flu express*. Retrieved from https://www.chp.gov.hk/en/resources/29/304.html

Dong, E., Du, H., & Gardner, L. (2020). An interactive web-based dashboard to track COVID-19 in real time. *The Lancet. Infectious Diseases, 20*(5), 533–534. https://doi.org/10.1016/S1473-3099(20)30120-1

Gal, Y., & Ghahramani, Z. (2016). Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of The 33rd international conference on machine learning* (pp. 1050–1059). PMLR. (ISSN: 1938-7228). Retrieved June 14 2025, from https://proceedings.mlr.press/v48/gal16.html

Hyndman, R. J., & Khandakar, Y. (2008). Automatic time series forecasting: The forecast package for R. *Journal of Statistical Software, 27*, 1–22. https://doi.org/10.18637/jss.v027.i03

Kermack, W. O., & McKendrick, A. G. (1927). A contribution to the mathematical theory of epidemics. *Proceedings of the Royal Society of London, 115*(772), 700–721. https://doi.org/10.1098/rspa.1927.0118
Kunsch, H. R. (1989). The jackknife and the bootstrap for general stationary observations. *The Annals of Statistics, 17*(3), 1217. https://doi.org/10.1214/aos/1176347265
La Gatta, V., Moscato, V., Postiglione, M., & Sperlì, G. (2021). An epidemiological neural network exploiting dynamic graph structured data applied to the COVID-19 outbreak. *IEEE Transactions on Big Data, 7*(1), 45–55. https://doi.org/10.1109/TBDATA.2020.3032755
Lakshminarayanan, B., Pritzel, A., & Blundell, C. (2017). Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems.* (Vol. 30). Curran Associates Inc.
Moein, S., Nickaeen, N., Roointan, A., Borhani, N., Heidary, Z., Javanmard, S. H., & Gheisari, Y. (2021). Inefficiency of SIR models in forecasting COVID-19 epidemic: A case study of Isfahan. *Scientific Reports, 11*(1), 1–9. https://doi.org/10.1038/s41598-021-84055-6
Politis, D. N., & White, H. (2004). Automatic block-length selection for the dependent bootstrap. *Econometric Reviews, 23*(1), 53–70. https://doi.org/10.1081/ETC-120028836
Rahimi, I., Chen, F., & Gandomi, A. H. (2021). A review on COVID-19 forecasting models. *Neural Computing and Applications, 35*, 1–11. https://doi.org/10.1007/s00521-020-05626-8
Rainisch, G., Jeon, S., Pappas, D., Spencer, K. D., Fischer, L. S., Adhikari, B. B., & Meltzer, M. I. (2022). Estimated COVID-19 cases and hospitalizations averted by case investigation and contact tracing in the US. *JAMA Network Open, 5*(3), Article e224042. https://doi.org/10.1001/jamanetworkopen.2022.4042
Rasjid, Z. E., Setiawan, R., & Effendi, A. (2021). A comparison: Prediction of death and infected COVID-19 cases in Indonesia using time series smoothing and LSTM neural network. *Procedia Computer Science, 179*, 982–988. https://doi.org/10.1016/j.procs.2021.01.102
Rocha-Solache, A., Rodríguez-Montoya, I., Sánchez-Argüelles, D., & Aretxaga, I. (2022). Time-domain deep-learning filtering of structured atmospheric noise for ground-based millimeter astronomy. *The Astrophysical Journal Supplement Series, 260*(1), 15. https://doi.org/10.3847/1538-4365/ac5259
Sarkar, K., Khajanchi, S., & Nieto, J. J. (2020). Modeling and forecasting the COVID-19 pandemic in India. *Chaos Solitons & Fractals, 139*, Article 110049. https://doi.org/10.1016/J.CHAOS.2020.110049
Shahid, F., Zameer, A., & Muneeb, M. (2020). Predictions for COVID-19 with deep learning models of LSTM, GRU and bi-LSTM. *Chaos Solitons & Fractals, 140*, Article 110212. https://doi.org/10.1016/J.CHAOS.2020.110212
Slivkins, A. (2019). *Introduction to multi-armed bandits*. Foundations and Trends in Machine Learning.
Taylor, S. J., & Letham, B. (2017). *Forecasting at scale (no. e3190v2)*. PeerJ Preprints.
Wahba, G. (1975). Smoothing noisy data with spline functions. *Numerische Mathematik, 24*(5), 383–393. https://doi.org/10.1007/BF01437407
Zhao, D., Zhang, R., Zhang, H., & He, S. (2022). Prediction of global omicron pandemic using ARIMA, MLR, and prophet models. *Scientific Reports, 12*(1), 18138. https://doi.org/10.1038/s41598-022-23154-4

Authors and Affiliations

Dhammika Amaratunga[1] · Javier Cabrera[2] · Nuria Diaz-Tena[3] · Michael N. Katehakis[3] · Chun-Pang Lin[2] · Jin Wang[3]

✉ Jin Wang
jin.w@rutgers.edu

Dhammika Amaratunga
damaratung@yahoo.com

Javier Cabrera
cabrera@stat.rutgers.edu

Nuria Diaz-Tena
nd796@newark.rutgers.edu

Michael N. Katehakis
mnk@rutgers.edu

Chun-Pang Lin
cp.lin@rutgers.edu

[1] Princeton Data Analytics LLC, Princeton, USA

[2] Department of Statistics, Rutgers University, 110 Frelinghuysen Rd, Piscataway, NJ 08854, USA

[3] Department of Management Science and Information Systems, Rutgers University, 100 Rockafeller Rd, Piscataway, NJ 08854, USA