

# Evaluating ARIMA, LSTM, and simple benchmarks for short-term COVID-19 case forecasting

*Ethan Liu*

Columbus Academy, Gahanna, USA

ethanliu787@gmail.com

---

**Abstract.** This study compares Autoregressive Integrated Moving Average (ARIMA) and Long Short-Term Memory (LSTM) models for short-term forecasting of new confirmed COVID-19 cases in the United States. Using national-level data derived from the Johns Hopkins University CSSE COVID-19 repository, we evaluate both models under multiple experimental configurations that vary data frequency, training-test split, historical lookback length, in-put features, and model training strategy. Forecast performance is further compared with persistence and constant-growth benchmarks using rolling one-week-ahead evaluation. The results show that conclusions about model performance are highly sensitive to the experimental configuration. LSTM generally performed better than ARIMA in weekly forecasting, whereas ARIMA performed better under some daily-data settings. Additional epidemiological inputs and expanding-window retraining improved performance in several configurations. However, under the central weekly evaluation, both ARIMA and LSTM were outperformed by the simple persistence and constant-growth benchmarks. These findings demonstrate that increased model complexity does not necessarily lead to better short-term forecasting performance and highlight the importance of experimental design and strong baseline comparisons when evaluating time-series forecasting models.

**Keywords:** COVID-19, ARIMA, LSTM, Time Series Forecasting

---

## 1. Introduction

The COVID-19 pandemic created an urgent need for methods capable of accurately forecasting short-term changes in disease activity. Reliable forecasts of new cases can support situational awareness and help governments, public health agencies, and healthcare organizations prepare for increases in infections. For example, short-term forecasts can inform medical-resource allocation, hospital-capacity planning<sup>1</sup>, and timely public-health alerts when local transmission increases.

Forecasting COVID-19 cases is challenging because reported case counts rarely follow stable patterns. Transmission can be affected by new variants, vaccination, public behavior, government interventions, testing

---

<sup>1</sup> Centers for Disease Control and Prevention, "Behind the Model: Short-Term Forecasts to Estimate the Hospital Burden of COVID-19," January 8, 2025, <https://www.cdc.gov/cfa-behind-the-model/php/data-research/covid-shortterm-forecasts.html>.

availability, and reporting practices. As a result, epidemiological time series may contain rapid increases, sudden declines, changing trends, and substantial noise. Forecast performance may therefore depend not only on the forecasting model itself, but also on factors such as data frequency, training-set size, historical lookback length, and the inclusion of additional epidemiological variables [1, 2].

A range of statistical and machine-learning approaches have been applied to COVID-19 forecasting. Autoregressive Integrated Moving Average (ARIMA) is a traditional statistical time-series method that models relationships among past observations, differenced values, and forecast errors, whereas Long Short-Term Memory (LSTM) networks are designed to learn temporal dependencies from sequential data. Previous studies have compared ARIMA and LSTM, as well as related approaches such as Prophet, stacked LSTM, and nonlinear autoregressive neural networks. Some studies have reported better forecasting performance from LSTM-based approaches than from ARIMA for particular datasets and periods [3]. However, findings vary across studies because of differences in countries, time periods, forecast horizons, preprocessing procedures, model inputs, train-test splits, and evaluation metrics. This makes it difficult to attribute forecasting performance to model type alone.

A further limitation is that more complex forecasting models are not always evaluated against simple baseline methods under identical test conditions. A model may achieve apparently good predictive accuracy while offering little practical improvement over a forecast that simply repeats the most recently observed value. Comparing ARIMA and LSTM with simple persistence and constant-growth benchmarks can therefore provide a more informative assessment of their forecasting value.

This study addresses the following research questions:

- RQ1. How do different data and training configurations affect the forecasting performance of ARIMA and LSTM?
- RQ2. How do ARIMA and LSTM compare in one-week-ahead COVID-19 case forecasting, relative to persistence and constant-growth benchmarks?

To address these questions, we evaluate ARIMA and LSTM models for one-week-ahead forecasting of confirmed COVID-19 cases under multiple experimental configurations. We examine the effects of data frequency, train-test split, lookback length, and input features, and compare both models with persistence and constant-growth benchmarks under the same rolling forecasting procedure.

## 2. Related work

### 2.1. COVID-19 spread prediction dataset

COVID-19 tracking datasets commonly organize confirmed cases, deaths, recoveries, and other epidemiological measurements by date and geographic region. The data in this study were obtained from the JHU CSSE COVID-19 Data Repository<sup>2</sup>. The JHU CSSE repository supported its public dashboard and combined information from public health agencies and other reporting organizations. Data collection ended in March 2023, making the repository a record of the earlier pandemic period<sup>3</sup>.

<sup>2</sup> Johns Hopkins University Center for Systems Science and Engineering, "COVID-19 Data Repository," GitHub, <https://github.com/CSSEGISandData/COVID-19>.

<sup>3</sup> See note 2 above.

## 2.2. ARIMA models for time-series forecasting

Time-series forecasting uses chronologically ordered observations to estimate future values. In disease forecasting, past case counts may contain information about transmission trends, reporting cycles, and infection waves. However, structural changes caused by variants, interventions, testing practices, and human behavior can make relationships learned during one period less reliable during another. Autoregressive Integrated Moving Average, also known as ARIMA, is a statistical forecasting framework combining autoregressive, differencing, and moving-average components [2]. The autoregressive component relates the target to previous observations; differencing can remove nonstationary trends; and the moving-average component incorporates previous forecast errors. ARIMA provides an interpretable statistical baseline and can operate with less training data than many deep-learning approaches.

## 2.3. LSTM models for time-series forecasting

Long Short-Term Memory (LSTM) is a recurrent neural-network architecture designed to retain and selectively update information across sequential observations. Its input, forget, and output gates regulate how information enters, remains in, and leaves the hidden state, enabling the model to learn longer-term temporal dependencies more effectively than conventional recurrent networks.

LSTM has been widely applied to COVID-19 forecasting for outcomes including cases, deaths, recoveries, and hospital demand [4]. Studies have used both daily and weekly data and have incorporated additional variables such as deaths, vaccination rates, mobility, and public-health restrictions. Some comparative studies have reported better forecasting performance for LSTM than ARIMA [3]. However, LSTM performance is sensitive to the amount and quality of training data, lookback length, model architecture, and training hyperparameters such as hidden-state size, learning rate, batch size, dropout, optimizer, and number of epochs. Forecast accuracy may also deteriorate during abrupt changes in transmission, testing, or reporting patterns. These variations suggest that findings from a single country, period, or model configuration should not be generalized, motivating evaluation across multiple data configurations and comparison with both statistical models and simple forecasting benchmarks [2-6].

# 3. Methodology

## 3.1. Forecasting model configurations

**ARIMA Configuration.** The ARIMA experiments used models of order  $\text{ARIMA}(p, 0, 0)$ , where  $p$  represented the historical lookback length. Weekly experiments tested  $p = 4, 6, 8$ , corresponding to the previous four, six, and eight weeks. The differencing parameter was fixed at  $d = 0$  because the cumulative confirmed-case series had already been transformed into daily new cases and subsequently aggregated into weekly totals. Additional differencing would therefore change the forecasting target from new-case counts to changes in new-case counts. The moving-average parameter was fixed at  $q = 0$  to maintain a consistent model structure across lookback settings and to avoid estimating additional error terms from the limited 40-week training set used in the central 25/75 evaluation. Each model generated a one-week-ahead forecast. During expanding-window evaluation, each observed weekly value became available before the subsequent forecast was produced.

**LSTM Configuration.** The LSTM model consisted of a single LSTM layer followed by a fully connected output layer producing one scalar prediction for confirmed cases in the target week. For weekly confirmed-only experiments, each input sample had shape  $(L, 1)$ , where  $L$  was the lookback length. For multivariate experiments using confirmed cases and deaths, the input shape was  $(L, 2)$ . Input features were standardized

before training, and predictions were transformed back to the original case-count scale. The baseline model used 16 hidden units. Hyperparameter tuning examined hidden-state size, learning rate, batch size, dropout, optimizer, and maximum training epochs, with separate grid searches conducted for the weekly and daily experiments. Early stopping was applied to reduce overfitting. The forecasting horizon was one week (7 days) ahead.

### 3.2. Persistence and constant growth benchmarks

The persistence benchmark assumes that the next value is equal to the most recently observed value:

$$\hat{y}_{t+1} = y_t \quad (1)$$

where  $y_t$  denotes the observed number of weekly confirmed COVID-19 cases in week  $t$ , and  $\hat{y}_{t+1}$  denotes the forecast for week  $t + 1$ .

The constant-growth benchmark extrapolates the most recent multiplicative growth rate to the following week. The growth factor is calculated as

$$g_t = \frac{y_t}{y_{t-1}} \quad (2)$$

and the forecast is then given by

$$\hat{y}_{t+1} = y_t g_t \quad (3)$$

where  $g_t$  represents the growth factor between weeks  $t - 1$  and  $t$ .

Both benchmarks were evaluated using the same rolling one-week-ahead procedure as the ARIMA and LSTM models. For each target week, forecasts were generated using only observations available at that time. After the actual value for the target week became available, it was added to the observed history and used in generating the subsequent forecast. This procedure was repeated across all 122 weekly targets in the 25/75 test period.

### 3.3. Evaluation metrics

Forecasting performance was evaluated using Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Weighted Absolute Percentage Error (WAPE). MAE measures the average absolute difference between predicted and observed values, while RMSE gives greater weight to larger forecasting errors. WAPE measures the total absolute forecasting error relative to the total observed case volume, providing a scale-relative measure of forecast accuracy. Lower values indicate better forecasting performance for all three metrics [7].

These metrics provide complementary measures of forecasting accuracy. MAE represents the typical magnitude of forecasting error, RMSE places greater emphasis on larger errors, and WAPE expresses the overall error relative to the observed case volume. WAPE was used as the primary metric when comparing experimental configurations. Because the weekly and daily experiments operate at different temporal scales, their MAE and RMSE values were not directly compared [8].

## 4. Experimental design

### 4.1. Data preparation

The JHU COVID-19 dataset used in this study contained 1,134 daily observations, corresponding to 162 complete weeks after preprocessing. The cumulative confirmed-case counts were converted into daily new

cases as

$$x_t = C_t - C_{t-1} \quad (4)$$

where  $C_t$  denotes the cumulative confirmed-case count on day  $t$ . The daily new-case counts were then aggregated into weekly totals. Confirmed cases were used as the primary input and forecasting target, while death counts were included as an additional input in selected multivariate experiments.

Two chronological training-test splits were evaluated. The 75%/25% (75/25) training-test split used the first 122 weeks for training and the final 40 weeks for testing, while the 25%/75% (25/75) training-test split used the first 40 weeks for training and the remaining 122 weeks for testing. Chronological splitting ensured that future observations were not used to predict earlier observations. The 25/75 split was used for the central comparison to evaluate performance with a more limited initial training history.

#### 4.2. Fixed-split experiments

The initial experiments used weekly confirmed-case data with four-, six-, and eight-week lookback windows. ARIMA and LSTM were evaluated using the 75/25 chronological split, with both models forecasting confirmed cases one week ahead.

The experiments were then repeated using the 25/75 split to examine model performance with only 40 initial training weeks and a substantially longer test period. As forecasting errors increased under the weekly 25/75 setting, additional experiments were conducted using daily observations. The four-, six-, and eight-week lookback windows were converted to equivalent windows of 28, 42, and 56 days, while the forecasting horizon remained seven days.

Additional experiments examined whether including death counts as an input improved confirmed-case forecasting. For LSTM, each input sample changed from  $(L, 1)$  to  $(L, 2)$ , where  $L$  denotes the selected lookback length and the two features represent confirmed cases and deaths. Confirmed cases remained the forecasting target. For ARIMA, death counts were incorporated as an exogenous variable, producing an ARIMAX-style model. Only information available before the forecast target was supplied to either model.

Separate hyperparameter searches were conducted for the weekly and daily LSTM models. The search evaluated 72 combinations of hidden-state size, learning rate, batch size, dropout, and optimizer. Each configuration was trained for a maximum of 500 epochs with early stopping. The weekly search selected 64 LSTM units, a learning rate of 0.005, batch size 1, dropout 0.2, and RMSprop. The daily search selected 64 units, a learning rate of 0.001, batch size 1, dropout 0.2, and Adam. Hyperparameter selection was based on chronological validation WAPE rather than test-set performance.

#### 4.3. Expanding-window experiment

An expanding-window experiment was conducted to examine forecasting performance when newly observed data were progressively incorporated into the training history. The experiment began with 40 weeks of training data. The model first forecast week 41. Once the actual value for week 41 became available, it was added to the training history before the model generated the forecast for week 42. This process was repeated sequentially throughout the remaining 122 weeks.

#### 4.4. Benchmark comparison

ARIMA and LSTM were also compared with the persistence and constant-growth benchmarks defined in Section 3.2 under a common evaluation setting. The comparison used weekly data, the 25/75 chronological split, confirmed cases as the only input, and a one-week-ahead forecasting horizon. ARIMA and LSTM used a

six-week lookahead, while persistence and constant growth used the most recent one and two weekly observations, respectively. All four forecasting methods were evaluated over the same 122 test weeks using the same rolling one-week-ahead procedure.

## 5. Results

### 5.1. ARIMA and LSTM performance under fixed-split configurations

Table 1 presents the performance of ARIMA and LSTM under the different fixed-split experimental configurations. In the initial weekly experiment using the 75/25 split, LSTM produced lower errors than ARIMA for all three lookahead lengths. The six-week LSTM achieved the lowest error in this group, with an MAE of 80,776.95, an RMSE of 110,553.36, and a WAPE of 16.528%. The four- and eight-week LSTM models produced WAPE values of 26.578% and 22.665%, respectively. In comparison, ARIMA produced WAPE values between 56.571% and 60.885%.

When the weekly experiment was repeated using the 25/75 split, errors increased for both models. ARIMA produced WAPE values ranging from 71.899% to 85.320%, while LSTM produced values ranging from 40.114% to 42.190%. The four-week LSTM achieved the lowest WAPE in this group at 40.114%, while the eight-week ARIMA achieved the lowest WAPE among the ARIMA configurations at 71.899%.

The 25/75 experiment was subsequently repeated using daily observations, with lookahead windows of 28, 42, and 56 days. In contrast to the weekly experiments, ARIMA produced lower errors than LSTM for all three lookahead lengths. The 28-day ARIMA achieved the lowest WAPE at 24.908%, followed by 27.089% for the 42-day configuration and 33.193% for the 56-day configuration. The corresponding LSTM models produced WAPE values of 60.822%, 60.926%, and 55.252%, respectively.

Hyperparameter tuning produced different effects in the weekly and daily LSTM experiments. The tuned weekly LSTM, using a six-week sequence of confirmed cases and deaths, achieved an MAE of 315,282.21, an RMSE of 806,936.42, and a WAPE of 40.749%. This was close to the 41.284% WAPE obtained by the untuned six-week LSTM using confirmed cases only. However, the two configurations also differed in their input features, so this comparison does not isolate the effect of hyperparameter tuning alone.

The tuned daily LSTM used a 42-day lookahead and achieved an MAE of 54,314.92, an RMSE of 131,997.55, and a WAPE of 49.155%. Compared with the untuned 42-day LSTM, which produced a WAPE of 60.926%, tuning reduced WAPE by 11.771 percentage points. Nevertheless, the tuned daily LSTM remained less accurate than the corresponding 42-day ARIMA, which achieved a WAPE of 27.089%.

**Table 1.** Performance of ARIMA and LSTM under different fixed-split configurations

| Setting      | Lookback | Horizon | Input     | Model | MAE        | RMSE       | WAPE    |
|--------------|----------|---------|-----------|-------|------------|------------|---------|
| Weekly 75/25 | 4 weeks  | 1 week  | Confirmed | ARIMA | 276,475.37 | 331,212.37 | 56.571% |
| Weekly 75/25 | 4 weeks  | 1 week  | Confirmed | LSTM  | 129,894.12 | 186,404.29 | 26.578% |
| Weekly 75/25 | 6 weeks  | 1 week  | Confirmed | ARIMA | 297,560.06 | 352,388.38 | 60.885% |
| Weekly 75/25 | 6 weeks  | 1 week  | Confirmed | LSTM  | 80,776.95  | 110,553.36 | 16.528% |
| Weekly 75/25 | 8 weeks  | 1 week  | Confirmed | ARIMA | 282,459.72 | 346,741.22 | 57.795% |
| Weekly 75/25 | 8 weeks  | 1 week  | Confirmed | LSTM  | 110,770.81 | 151,197.30 | 22.665% |
| Weekly 25/75 | 4 weeks  | 1 week  | Confirmed | ARIMA | 660,126.92 | 956,612.86 | 85.320% |
| Weekly 25/75 | 4 weeks  | 1 week  | Confirmed | LSTM  | 310,363.73 | 808,005.53 | 40.114% |
| Weekly 25/75 | 6 weeks  | 1 week  | Confirmed | ARIMA | 652,596.13 | 972,013.12 | 84.346% |

**Table 1.** Continued

|              |         |        |                    |                       |            |            |         |
|--------------|---------|--------|--------------------|-----------------------|------------|------------|---------|
| Weekly 25/75 | 6 weeks | 1 week | Confirmed          | LSTM                  | 319,421.69 | 845,183.28 | 41.284% |
| Weekly 25/75 | 8 weeks | 1 week | Confirmed          | ARIMA                 | 556,290.34 | 844,520.71 | 71.899% |
| Weekly 25/75 | 8 weeks | 1 week | Confirmed          | LSTM                  | 326,429.21 | 834,331.22 | 42.190% |
| Daily 25/75  | 28 days | 7 days | Confirmed          | ARIMA                 | 27,522.31  | 58,208.81  | 24.908% |
| Daily 25/75  | 28 days | 7 days | Confirmed          | LSTM                  | 67,206.35  | 143,554.02 | 60.822% |
| Daily 25/75  | 42 days | 7 days | Confirmed          | ARIMA                 | 29,932.18  | 61,540.50  | 27.089% |
| Daily 25/75  | 42 days | 7 days | Confirmed          | LSTM                  | 67,320.53  | 143,795.04 | 60.926% |
| Daily 25/75  | 56 days | 7 days | Confirmed          | ARIMA                 | 36,676.68  | 68,950.10  | 33.193% |
| Daily 25/75  | 56 days | 7 days | Confirmed          | LSTM                  | 61,051.91  | 138,017.55 | 55.252% |
| Weekly 25/75 | 6 weeks | 1 week | Confirmed + Deaths | LSTM <sub>Tuned</sub> | 315,282.21 | 806,936.42 | 40.749% |
| Daily 25/75  | 42 days | 7 days | Confirmed          | LSTM <sub>Tuned</sub> | 54,314.92  | 131,997.55 | 49.155% |

### 5.2. Expanding-window evaluation

Table 2 reports the performance of the tuned LSTM and ARIMAX-style model under the expanding-window evaluation. The tuned LSTM achieved an MAE of 153,788.59, an RMSE of 338,139.62, and a WAPE of 19.877%. The ARIMAX-style model produced an MAE of 174,612.18, an RMSE of 373,610.60, and a WAPE of 22.568%. The tuned LSTM therefore produced lower errors than the ARIMAX-style model across all three metrics.

**Table 2.** Performance under the expanding-window evaluation

| Model              | MAE        | RMSE       | WAPE    |
|--------------------|------------|------------|---------|
| Tuned LSTM         | 153,788.59 | 338,139.62 | 19.877% |
| ARIMAX-style model | 174,612.18 | 373,610.60 | 22.568% |

### 5.3. Comparison with simple benchmarks

Table 3 compares ARIMA and LSTM with the persistence and constant-growth benchmarks under the common weekly 25/75 evaluation described in Section 4.4. Persistence produced the lowest error across all three metrics, with an MAE of 160,496.37, an RMSE of 348,273.61, and a WAPE of 20.744%. Constant growth produced similar results, with an MAE of 163,494.78, an RMSE of 378,442.91, and a WAPE of 21.131%. Its WAPE was only 0.387 percentage points higher than that of persistence.

**Table 3.** Comparison of ARIMA and LSTM with simple forecasting benchmarks

| Method                    | MAE        | RMSE       | WAPE    |
|---------------------------|------------|------------|---------|
| Persistence Benchmark     | 160,496.37 | 348,273.61 | 20.744% |
| Constant Growth Benchmark | 163,494.78 | 378,442.91 | 21.131% |
| ARIMA                     | 652,596.13 | 972,013.12 | 84.346% |
| LSTM                      | 319,421.69 | 845,183.28 | 41.284% |

Among ARIMA and LSTM, LSTM produced lower errors, with a WAPE of 41.284% compared with 84.346% for ARIMA. However, both complex models produced higher errors than the persistence and constant-growth benchmarks under the central evaluation setting.

Figure 1 and Figure 2 show the actual and predicted weekly confirmed-case counts for the persistence and constant-growth benchmarks, respectively. In Figure 1, the persistence benchmark closely follows the observed series because each forecast simply uses the most recently observed value. This produces an approximately one-week lag. In Figure 2, the constant-growth benchmark responds more strongly to recent trends by extrapolating the latest growth rate. While its predictions remain close to the observed values during relatively stable periods, this approach can amplify sudden changes, leading to substantial overprediction around major peaks. These patterns are consistent with the results in Table 3, where both benchmarks achieved similar overall performance, although persistence produced slightly lower MAE, RMSE, and WAPE than constant growth.

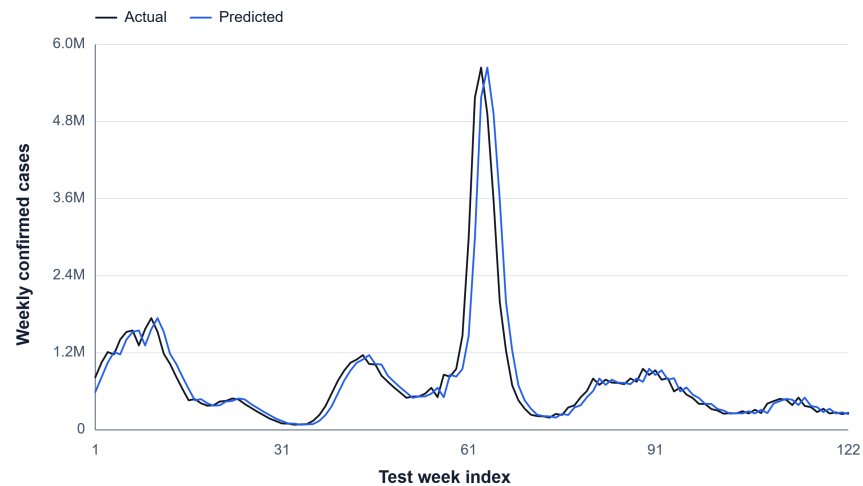


**Figure 1.** Actual and one-week-ahead predicted weekly confirmed cases using the persistence benchmark, which predicts each week using the most recently observed case count

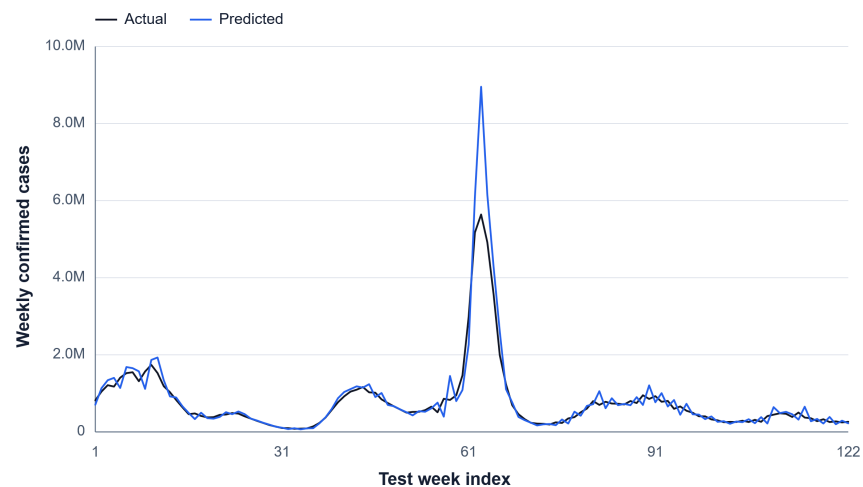


**Figure 2.** Actual and one-week-ahead predicted weekly confirmed cases using the constant-growth benchmark, which extrapolates the most recently observed weekly growth rate

## 6. Discussion

### 6.1. Interpretation of forecasting results

The results show that no single forecasting model performed best under every condition. Forecasting performance varied with the amount of training data, data frequency, lookback length, input variables, hyperparameter configuration, and evaluation method. This suggests that the experimental setup can substantially affect conclusions about the relative performance of ARIMA and LSTM.

Providing more training data generally improved both models, particularly LSTM. ARIMA performed better with daily data, which may be related to the larger number of observations available for estimating short-term relationships. In contrast, the daily LSTM may have been more affected by reporting noise while forecasting seven days ahead. Including weekly death counts generally improved forecasting accuracy, although the effect was not consistent across configurations. Hyperparameter tuning also produced a clearer improvement for the daily LSTM than for the weekly LSTM.

The expanding-window evaluation reduced forecasting error by allowing the models to incorporate newly observed weeks. However, under the central weekly 25/75 comparison, persistence and constant growth outperformed both ARIMA and LSTM. This indicates that greater model complexity does not necessarily lead to better forecasting performance, particularly when recent case levels provide strong information about the following week.

Forecasting errors were concentrated around sudden infection waves, when historical patterns became less reliable. Overall, the findings highlight the importance of evaluating forecasting models under multiple configurations and comparing them with simple benchmarks rather than assuming that either ARIMA or LSTM will consistently provide the best forecasts.

### 6.2. Limitations

This study has several limitations related to the data used for forecasting. The analysis was based on approximately three years of national-level U.S. COVID-19 data. Aggregating cases at the national level may obscure substantial differences in transmission patterns across states and regions. In addition, reported case counts were influenced by changes in testing availability and reporting practices over the course of the pandemic, which may have affected the consistency of the time series.

The scope of the model evaluation was also limited. The experiments examined a single-layer LSTM architecture, a restricted set of model configurations, and a small number of external variables. The findings therefore should not be assumed to generalize to other countries, regions, time periods, datasets, or alternative LSTM architectures and forecasting models.

## 7. Conclusions

This study compared ARIMA and LSTM for short-term COVID-19 case forecasting under a range of experimental configurations and evaluated both models against simple forecasting benchmarks. The findings show that forecasting performance was highly sensitive to the experimental setup. LSTM performed better in the weekly experiments, whereas ARIMA performed better when daily data were used under the 25/75 split. Including death counts generally improved weekly forecasting performance, although the effect was not consistent across all configurations. Hyperparameter tuning improved the daily LSTM, while expanding-window retraining reduced forecasting error by incorporating newly observed data. Importantly, neither ARIMA nor LSTM consistently outperformed the simple forecasting benchmarks. Under the central weekly

25/75 evaluation, persistence and constant growth achieved better forecasting performance than both complex models. These findings show that greater model complexity does not necessarily lead to higher forecasting accuracy and highlight the importance of considering experimental configuration and simple benchmark comparisons when evaluating forecasting models.

## References

- [1] Petropoulos, F., Makridakis, S., & Stylianou, N. (2022). COVID-19: Forecasting confirmed cases and deaths with a simple time series model. *International Journal of Forecasting*, 38(2), 439–452.
- [2] Sehrawat, P. K., & Vishwakarma, D. K. (2022, April). Comparative analysis of time series models on COVID-19 predictions. In *2022 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS)* (pp. 710–715). IEEE.
- [3] Uppalaiah, B., & Mallikarjuna Reddy, D. (2026). Evaluating ARIMA and LSTM models for forecasting COVID-19 case trends: A comparative study. *Model Assisted Statistics and Applications*, 21(2), 161–170.
- [4] Absar, N., Uddin, N., Khandaker, M. U., & Ullah, H. (2022). The efficacy of deep learning-based LSTM model in forecasting the outbreak of contagious diseases. *Infectious Disease Modelling*, 7(1), 170–183.
- [5] Alassafi, M. O., Jarrah, M., & Alotaibi, R. (2022). Time series predicting of COVID-19 based on deep learning. *Neurocomputing*, 468, 335–344.
- [6] Cumbane, S. P., & Gidófalvi, G. (2025). Deep learning-based approach for COVID-19 spread prediction. *International Journal of Data Science and Analytics*, 20(3), 1933–1949.
- [7] Balli, S. (2021). Data analysis of COVID-19 pandemic and short-term cumulative case forecasting using machine learning time series methods. *Chaos, Solitons & Fractals*, 142, Article 110512.
- [8] de Oliveira, L. S., Gruetzmacher, S. B., & Teixeira, J. P. (2021). COVID-19 time series prediction. *Procedia Computer Science*, 181, 973–980.