2023

Forecasting COVID-19 Cases Using Dynamic Time Warping and Incremental Machine Learning Methods

Luis Miralles-Pechuán
Technological University Dublin, Ireland, luis.miralles@tudublin.ie

Ankit Kumar
University College Dublin, Ireland

Andres L. Suare\-Cetrulo
University College Dublin, Ireland

Funder: This research received no external funding

Received: 20 April 2022 | Revised: 15 December 2022 | Accepted: 10 January 2023

DOI: 10.1111/exsy.13237

ORIGINAL ARTICLE

Expert Systems WILEY

Forecasting COVID-19 cases using dynamic time warping and incremental machine learning methods

Luis Miralles-Pechuán[1] | Ankit Kumar[2] | Andrés L. Suárez-Cetrulo[2]

[1]School of Computer Science, Technological University Dublin, Dublin, Ireland

[2]Centre for Applied Data Analytics Research (CeADAR), University College Dublin, Dublin, Ireland

Correspondence
Luis Miralles-Pechuán, School of Computer Science, Technological University Dublin, Dublin, Ireland.
Email: luis.miralles@tudublin.ie

Abstract

The investment of time and resources for developing better strategies is key to dealing with future pandemics. In this work, we recreated the situation of COVID-19 across the year 2020, when the pandemic started spreading worldwide. We conducted experiments to predict the coronavirus cases for the 50 countries with the most cases during 2020. We compared the performance of state-of-the-art machine learning algorithms, such as long-short-term memory networks, against that of online incremental machine learning algorithms. To find the best strategy, we performed experiments to test three different approaches. In the first approach (single-country), we trained each model using data only from the country we were predicting. In the second one (multiple-country), we trained a model using the data from the 50 countries, and we used that model to predict each of the 50 countries. In the third experiment, we first applied clustering to calculate the nine most similar countries to the country that we were predicting. We consider two countries to be similar if the differences between the curve that represents the COVID-19 time series are small. To do so, we used time series similarity measures (TSSM) such as Euclidean Distance (ED) and Dynamic Time Warping (DTW). TSSM return a real value that represents the distance between the points in two time series which can be interpreted as how similar they are. Then, we trained the models with the data from the nine more similar countries to the one that was predicted and the predicted one. We used the model ARIMA as a baseline for our results. Results show that the idea of using TSSM is a very effective approach. By using it with the ED, the obtained RMSE in the single-country and multiple-country approaches was reduced by 74.21% and 74.70%, respectively. And by using the DTW, the RMSE was reduced by 74.89% and 75.36%. The main advantage of our methodology is that it is very simple and fast to apply since it is only based on time series data, as opposed to more complex methodologies that require a deep and thorough study to consider the number of parameters

All authors have equally contributed to the development of this work.

involved in the spread of the virus and their corresponding values. We made our code public to allow other researchers to explore our proposed methodology.

KEYWORDS
COVID-19 prediction, dynamic time warping, epidemiology curve, incremental machine learning, time series similarity measures

1 | INTRODUCTION

The first cases of COVID-19 were reported to the global public in December 2019. Its origin has been placed in Wuhan, in the Hubei province of China (Shereen et al., 2020). Shortly after its announcement, on 30 January 2020, the World Health Organization (WHO) declared the virus a public health emergency of international concern (Zu et al., 2020).The COVID-19 virus has a high transmissibility level that causes severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). Due to its pathogenicity—the capacity for causing harm—and its ease of transmission across humans, COVID-19 has caused tremendous damage to public health, initially in China and subsequently in the rest of the world. According to Worldometers.info, which collects worldwide reported cases, until August 2022, around 582 million people contracted COVID-19, and 6.4 million died because of COVID-19. On top of that, the restrictive actions taken by governments to mitigate the virus, such as lockdowns and quarantines, have caused serious damage to the global economy. For instance, there was a lack of supply of many products and many jobs were lost in many countries (Bonaccorsi et al., 2020).

During 2021, the vaccines Pfizer-BioNTech, Moderna, Johnson & Johnson, Sputnik V and others were approved, and by July, more than 3.9 billion vaccines were massively administered around the world (Vashi & Coiado, 2021). Additionally, the original virus mutated into less severe variants such as Omicron, which lowered the hospitalization rates (Barnard et al., 2021). Because of this, the world is slowly recovering, and the situation in August 2022 looks much more favourable than the previous years. However, although the prospects in 2022 regarding the pandemic are much better, it is still possible that future pandemics could emerge. It is necessary to keep studying the best strategies and solutions to prevent their negative effects (Shereen et al., 2020).

The coronavirus pandemic has been combated by many disciplines. For example, the medical field has contributed with vaccination, public health with masking and social distancing and many sciences (biology, maths, statistics,...) have helped in building COVID-19 models to support governments in taking the best measures (Zeroual et al., 2020). There are a lot of disciplines that can positively contribute to mitigating the COVID-19 effects (Lalmuanawma et al., 2020). However, due to our computing background, we find particularly interesting the approaches based on developing Artificial Intelligence (AI) applications.

AI has been used to fight COVID-19 in many ways, such as fast diagnosis and screening processes, contact tracing, vaccine development and forecasting cases (Lalmuanawma et al., 2020; Wynants et al., 2020). Thanks to predictive models able to make accurate estimations; resources in hospitals can be managed more intelligently, saving more lives; universities can develop strategies to organize the students' academic year in a better way and more effective planning to save the economy while guaranteeing satisfactory public health can be made (Nicola et al., 2020). Imposing restrictive measures, such as lockdown or self-isolation, can provoke catastrophic effects on the economy, and the mental health of the population can be undermined, but not controlling the virus can cause a high number of deaths. Therefore, it is essential to develop accurate models to help governments take the best possible actions to balance out the negative effects of potential pandemics.

Traditionally, forecasting methods were based on statistical methods such as ARIMA (which we also used in our experiments). Still, in recent years, researchers have used more advanced methods like Machine Learning (ML) algorithms. ML is a branch of AI algorithms to make machines able to learn automatically from the data, requiring very little or no human intervention. ML methods can capture non-linear relationships in the input data without prior knowledge. According to the literature, ML algorithms have replaced statistical models as a de-facto standard to make predictions (Hsu et al., 2016).

Inside ML, there is a category of methods called Incremental Learning Methods (ILMs), in which the input is a continuous stream of information, and the model is permanently updated with this data. ILMs are widely used in non-stationary domains like financial transactions, telecommunications, weather forecasts and the internet of things, to name a few, adapting to shifts and drifts that may occur implicitly in the data (Gama, Žliobaite, et al., 2013). Their main advantage is that models can be continuously updated with new instances and do not need to be trained from scratch, unlike classical ML methods. To our knowledge, ILMs have not yet been applied to predict coronavirus disease cases. We believe that our work will help to fill that gap in the literature.

In this piece of research, we put ourselves in the shoes of a country facing a pandemic that only has the information available at a certain date. Our work falls in the category of predicting and forecasting COVID-19 cases using ML. It is very important to study how to generate reliable and accurate predictive models in the shortest possible time to be ready in case we face a pandemic again. Accurately predicting the number of

cases a few weeks in advance allows governments to plan ahead, develop better strategies and take preventive actions before it is too late (Miralles-Pechuán, Jiménez, et al., 2020).

In this paper, as shown in Figure 1, we proposed three approaches to train the ML models; training them only with data from the predicted country (single-country approach); training them using the 50 countries in the dataset (multiple-countries approach) and lastly, training only with the nine most similar countries according to time series similarity measures (TSSM) in the third approach (time-series-filtering approach). We consider the degree of similarity between countries using time similarity measures such as DTW and ED. Time measures return a real number from the time series of COVID-19 cases in two countries. Time series similarity measures are an important research area and the main ingredient for time series classification and clustering (Serra & Arcos, 2014). There are many of them, such as Minkowski, Manhattan, Chebyshev, Frechet-DISC, Levenshtein, SAX, Hausdorff, or Bray-Curtis.

Our contribution is different to previous works for the following reasons. On the one hand, we compare the performance of a recent branch of ML algorithm called ILMs with popular ML methods in forecasting the number of COVID-19 cases over the 50 countries with the most COVID cases in 2020. We compared their performance against state-of-the-art methods such as Gradient Boosting and long-short-term memory (LSTM). On the other hand, we present a novel methodology that applies a simple but powerful idea: this is the use of a clustering-based approach based on TSSM to select the most similar countries to the predicted one. And then, we trained models with the data from those countries to predict future cases. We used ARIMA to have a baseline with which to compare our results. By using these strategies, we improved the accuracy of regressions ML methods. Results show that our approach is very efficient, and it is also very suitable for predicting quickly, like in the case when a pandemic erupts.

We have made our code available so that other researchers could explore our proposal further and compare their methods with our approach. We created the framework[1] to encourage the scientific community to develop new models and strategies to design more accurate algorithms to predict COVID-19 cases. To our knowledge, no other publication applies time series similarity measures to train models with data from multiple countries, uses ILMs, and conducts a thorough study of over 50 countries at eight points in time using RMSE and MAE. Our proposal has the advantage that it is straightforward and only needs the time series of COVID-19 cases.

The rest of the paper is organized as follows: Section 2 presents an overview of the epidemiological models to represent viruses and an overview of the supervised ML models. Particularly a subset of them called ILMs. Section 3 details the implemented methodology to apply the proposed approaches. Section 4 presents the conducted experiments to compare the performance of the ILMs against that of other popular methods, such as the popular deep learning method for time series called LSTM. Then, it compares the performance of the methods under different scenarios and training schemes to find out the optimal configuration. It also discusses the obtained results for each of the models and presents an analysis of both the static and the ILMs. Finally, Section 5 presents the main findings of our investigation and recommends some interesting lines of research for future work.

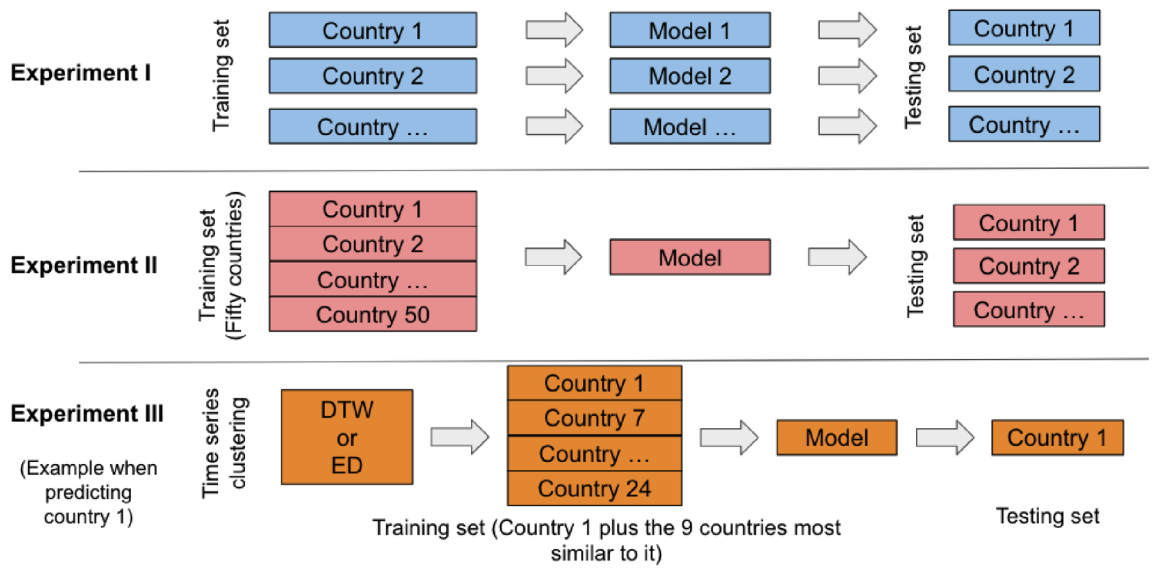

FIGURE 1 Experiment I only uses historical cases of the country it is predicting, Experiment II uses historical cases from all the countries to predict every single one, and Experiment III, here it is only shown for one country, only uses data from the nine most similar countries based on time similarity measures such as DTW and ED.

2 | STATE OF THE ART

This section describes some important research work on modelling the spread of the COVID-19 virus on a population. It also describes the main ML methods applied, emphasizing online incremental ML algorithms and their main differences compared to static ML methods in terms of model evaluation.

2.1 | Modelling the evolution of COVID-19

The impact of the pandemic on society has pushed researchers to find ways to combat the virus from their respective disciplines. Due to the coronavirus' impact on society, the volume of publications related to COVID-19 since it was detected has become enormous. To give an example, just the number of publications related to COVID-19 in the database of the Web of Science until 14 October 2020, only 10 months after the pandemic started, is 12,021. This database gathers data from the bioRxiv platform (2040), the medRxiv platform (7555), the Preprints platform (1046) and the SSRN platform (2028) (Wang & Tian, 2021).

Our investigation falls into the category of modelling the evolution of the virus in the population. On this topic, there is a high number of publications covering a broad spectrum that goes from those applying the simple method ARIMA (Benvenuto et al., 2020) to those implementing the latest deep learning techniques (Zeroual et al., 2020). Forecasting the evolution of positive cases with precision is the backbone for planning the optimal governmental actions to avoid damaging the economy and protect public health (Miralles-Pechuán, Ponce, & Martínez-Villaseñor, 2020). Creating inaccurate models can mislead governments into taking the wrong actions, which could have terrible effects on the health and economy of the population (Chin et al., 2020).

The epidemiological models are the ones used to estimate the evolution of viruses. Broadly speaking, there are two categories: the mechanistic models and the statistical models (Adiga et al., 2020). The mechanistic models are based on principles and equations that explain how infectious diseases spread. Those models simulate the dynamics of the virus and are quite similar to the typical models in physics. Mechanistic models range from simple ones that only consider the incubation period to complex ones that consider multiple groups based on variables such as age, location and vaccination rate. On the other hand, the statistical models are data-driven and use historical data of the viruses' evolution, e.g. the registered positive cases per day (Maleki et al., 2020). They use equations from statistics, although, in recent years, they are using AI and ML techniques more frequently (Zeroual et al., 2020). Statistical methods learn from the experience using the data with which they are fed. There are also models that combine both mechanistic and statistical approaches and are becoming popular as well (Adiga et al., 2020).

Inside the mechanistic models, there is a class called mass compartmental models. Compartmental models represent the virus evolution using differential equations that divide the population into dynamic groups (or compartments) (Brauer, 2008). The SIR model is a typical example of compartmental model. SIR models divide the population into three groups: Susceptible (S), Infected (I) and Removed (R). Infected individuals can transmit the virus to the Susceptible ones who have not yet suffered it. The recovered individuals are the ones that have recovered or died. In its simplest version, the virus' evolution can be modelled using three parameters: β (contact rate), σ (incubation period) and γ (recovery rate or period in which individuals are infectious) (Yang et al., 2020).

An extended version of the SIR model called SEIR includes a new group called Exposed (E). Exposed individuals have contracted the virus, but they are not infectious yet (Brauer, 2008). Since these models are a bit simplistic and, in general, populations are composed of different groups, for example, by age, location, or other similar patterns, there is an extension of the SIR models called Structured Metapopulation Models (Gyllenberg et al., 1997).

Metapopulation models consider the heterogeneity of the groups within a population by considering multiple factors such as age, lockdowns, two-meter distance separation and restrictions on mask use. They are more complex, but they are better at adapting to the behaviour of a population. Metapopulation models have the overhead of fine-tuning a large number of parameters to represent the changing environment accurately, and a mismatch in the value of the parameters can render the model flawed. This creates a need to adjust the values of the model continuously, for example the virus can mutate, becoming more harmful or more contagious (Toyoshima et al., 2020). On top of that, the more complex the models are, the more difficult to understand they become. Metapopulation models require a deep understanding and study of the virus. Their main advantages are that they enable the simulation of different scenarios (pessimistic, realistic, optimistic) and that once the values are set, they are also very fast to run.

Agent-based network models are an extension of metapopulation models. They use computer simulations that approximate the real-world scenario by representing the population's individuals using agents inside a virtual environment. They apply a 'bottom-up' approach, whereas compartmental models apply a 'top-down' one. The individuals obey some coded rules and help to understand how the virus can propagate across the population under different scenarios such as lockdowns and self-isolation. For example, at the University of Australia, Chang et al. (2020) implemented a fine-grained simulation and calibrated the model to match the COVID-19 transmission rate in Australia.

On the other hand, statistical models try to find patterns in the data. They are also known as time-series or curve-fitting models. They are based on sequential historical data points to predict future records. These models are less useful for simulating or understanding the virus but are

quicker to develop. There are a lot of variables that influence the behaviour of the virus, such as the mental health of the population; the influence of social networks; the correlation between the transmissibility of the virus and the temperatures; the number of lighting hours; the accumulative fatigue of a lockdown; the transmissibility rates of the new strains, or the unemployment rate of the country.

It is extremely challenging to estimate their actual impact. On top of that, those variables are constantly changing over time. The collected data from the virus intrinsically considers a high number of variables that are very difficult to understand and measure, even for the experts. Time-series models have been proven to be effective, and they have an overall advantage over the compartmental models; the simplicity and quickness to be modelled (Harvey & Kattuman, 2020). Those two advantages are very important when the time for taking the right measures is critical.

The approach of our paper, based on curve-fitting models, was also used by the Institute for Health Metrics and Evaluation (IHME) at the University of Washington. The institute predicted quite accurately when the United States was going to have a peak on the curve of the number of cases by creating models that fitted the curve of positive cases of the virus in other countries such as Italy, Spain, or the UK (COVID, IHME, 2020). The work of Zeroual et al. (2020) presents a general overview of some of the most important studies on modelling the virus and compares the performance of five different deep learning methods for modelling time-series data such as simple RNN, LSTM, Bidirectional LSTM (BiLSTM), Gated Recurrent Units (GRUs) and Variational AutoEncoder (VAE). One of the downsides of these models is that they do not have parameters to consider different scenarios of the evolution of the virus.

2.2 | Machine learning algorithms

Pandemic curves are non-stationary (constantly changing) by nature because, depending on the period, they can show clear trends, cycles and seasons where the random component is more prevalent. Moreover, the virus spread in each region varies differently (Li et al., 2020; Wang et al., 2020). Under these circumstances, incremental and online ML techniques (Ditzler et al., 2015) that adapt to the evolution of the trend and its changes on the fly are gaining traction in different domains (Gama, Sebastião, et al., 2013; Suárez-Cetrulo et al., 2019). One of the drawbacks of static ML methods is their inability to deal with data updates efficiently (Suárez-Cetrulo & Cervantes, 2017). This paper refers to state-of-the-art ML algorithms that need to be trained from scratch to add new knowledge as static algorithms.

ILMs and the notion of concept drift (when the predicted target changes) have not gained enough attention in the coronavirus prediction domain. However, incremental ML algorithms can deal either actively or passively in a better way (Elwell & Polikar, 2011) with the non-stationary nature of data streams such as the COVID-19 curve evolution (Tsymbal, 2004). This is by adapting (passively) to the non-stationary nature of the data or by using drift detectors (actively). These methods can find an equilibrium between prioritizing new knowledge, adapting to changes, and retaining previous relevant information through different forgetting mechanisms. This balance is known as the stability-plasticity dilemma, and it is very suitable for dynamics scenarios like the COVID-19 spreads (Kukar, 2003; Singh et al., 2020).

One of the main differences when comparing static with ILMs for data streams is the convention used to measure the model performance. Different model evaluation metrics create a challenge when comparing static with ILMs, as the evaluation for both methods needs to be consistent and fair. Hold-out is a typical evaluation technique in ML where the original set is separated into two independent sets for training and testing. The idea is that instances from the training set differ from those in the testing set to evaluate how accurately the model can predict. Using 80% for training and 20% for testing is a very common split. Hold-out is the evaluation scheme performed in traditional ML approaches (Gama, Sebastiao, & Rodrigues, 2013).

A prequential evaluation (or Interleaved Test-Then-Train) is a conventional setting in ILMs for continuous streams, where data is evaluated as soon as it is available (Cerqueira et al., 2020). In this evaluation technique, every data example (instance) is used initially for making a prediction. Then, when its target label is available, the instance is first used to compute the prediction error and later update the algorithm. This differs from hold-out evaluation, where testing splits are not used for training. The prequential evaluation makes more efficient use of the data (Cerqueira et al., 2020), and it is more suitable for ILMs, which can be updated and adapted to new instances and, unlike static methods, do not need to train the whole model again from scratch (Žliobaite et al., 2015).

2.2.1 | Static machine learning algorithms

We selected some of the most relevant regression ML models to compare them with ILMs. The description of the chosen algorithms goes as follows: Linear regression (LR) minimizes the residual sum of squares between the prediction and the target feature and fits a line with coefficients (Montgomery et al., 2012). Ridge regression, as an improvement of LR, uses regularization, minimizes β coefficients and adds a λ scalar to the learning process (Bishop, 2006). Bayesian Ridge Regression (MacKay, 1992; Tipping, 2001) is the Bayesian interpretation of a Ridge Regression Estimator. Thus, it performs linear regressions through probability distributors rather than point estimates. Support Vector Machines (SVM) construct a hyperplane through Support Vectors to use it as a discriminator for classification tasks. Support Vector Regression (SVR) is the regression

version of the SVM algorithm (Chang & Lin, 2011). Random Forest (RF) is a popular ensemble method that constructs a set of decision trees through bagging and feature bagging (Liaw et al., 2002). The final prediction depends on voting or an aggregation mechanism. The regression version aggregates the votes by averaging their predictions. Gradient Boosting trains its base learners gradually and sequentially (Friedman, 2001; Hastie et al., 2009). It uses gradients in the base learners' loss function to measure the outcome of each observation and improves weak learners iteratively. LSTM neural networks (Hochreiter & Schmidhuber, 1997) is an architecture with dense layers. LSTMs are used to keep adjacent temporal information while remembering information for a long time in their memory blocks. In feed-forward neural networks, learning occurs by changing these connection weights, often through a gradient descent-based approach like the back-propagation algorithm, to minimize the obtained error (Murtagh, 1991). ARIMA from AutoRegressive Integrated Moving Average is one of the most popular regression methods, and it has also been used for predicting coronavirus cases (Benvenuto et al., 2020). It is a stochastic statistical method mostly used for non-stationary series. It is based on moving averages to predict future values, which has proved inefficient when sudden changes happen.

2.2.2 | Incremental machine learning algorithms

We also selected a set of four incremental regression ML models to compare them with the state-of-the-art static methods aforementioned. Our choice covers incremental models that are notoriously used for regression problems in the literature. A description of the selected methods goes as follows: A Hoeffding tree (HT) is an incremental algorithm that assumes that the data distribution is constant over time. This tree relies mathematically on Hoeffding bounds, which supports that a small sample may suffice to choose an optimal splitting attribute. Hoeffding Tree for regression calculates the decrease of the variance of the target to decide the splits. Its leaf nodes fit linear perceptron models by default (Domingos & Hulten, 2000). The Hoeffding Adaptive Tree (HAT) is an adaptive version of the Hoeffding Tree. It replaces old branches with new ones if the error of the old ones increases over time and new branches perform better. To monitor the evolution of the errors, it uses the Adaptive Windowing (ADWIN) algorithm Bifet and Gavalda (2007); Bifet and Gavaldà (2009). HAT also proposes bootstrap sampling as an improvement over Hoeffding Trees. Adaptive Random Forest (ARF) (Gomes et al., 2017) is an adaptive version of the Random Forest ensemble for Data Streams. It manages a pool of trees that are replaced with new ones when a concept drift is detected. As an improvement to RF, each adaptive tree is trained with different samples and feature sets as part of the bagging process. The Passive-Aggressive algorithm (PA) is an online learning algorithm that updates the model depending on the obtained error (Crammer et al., 2006).

2.3 | Time series similarity measures for COVID-19 forecasting

A Time Series Similarity Measure (TSSM), such as DTW or ED, quantifies the degree of similarity between two sets of values using a real value. We can use TSSM to see how similar the time series of COVID-19 cases are between two countries. Some publications have applied TSSM like DTW (Müller, 2007) to compare the COVID-19 curve among countries to predict cases in the future. Other authors (Rojas et al., 2020) used hierarchical clustering to determine the most similar countries to the eastern and western zone of the United States and to create models such as the Logistic, Gompertz and SIR models to estimate the future cases of the virus. The DTW is used very often because it has the great advantage of being able to find similarities even when there is a shift in time between the compared time series.

The work of Stübinger and Schneider (2020) is a bit similar to ours. They used DTW to analyse lead–lag effects (one variable is correlated with another, but with time-lags) between different countries. They used information from a related country where the virus had already been spread to predict countries in which the virus is emerging. However, they used a statistical approach to predict new cases rather than applying ML methods. Additionally, they only used one country to train the models and only predicted over the nine countries with more cases. The publication of Landmesser (2020) is related to our work since the author applies DTW to perform hierarchical grouping for countries but uses ARIMA, which is not as effective as ML methods in forecasting new cases.

The work of Zeroual et al. (2020) conducts a comparative analysis over nine countries for short-term cases forecast using six promising deep learning methods: simple RNN, LSTM, Bidirectional LSTM (BiLSTM), Gated recurrent units (GRUs) and Variational AutoEncoder (VAE). They do not use similarity measures, which is a step that could potentially increase the performance of the methods. Many other publications related to the virus and time similarity measures focus on one particular country or consider a different sector.

3 | METHODOLOGY

The experiments were conducted to answer the following question: What is the best approach for training incremental and static methods to predict the future number of coronavirus cases for a given country? To answer this question, we performed three experiments. In Experiment I, we trained the static and incremental models with only one country, and we predicted future cases for that same country. In Experiment II, we

predicted over one country, but this time, we trained the models with the 50 selected countries. Lastly, in Experiment III, we first selected the nine most similar countries to the ones predicted using TSSM, such as ED and DTW. Then, we trained the model with those nine countries and the selected country.

These methods were evaluated at eight points in time to calculate the average performance of the different approaches, which we called milestones. Each milestone represents a date on which we predicted future cases, considering only previous information to that point. Figure 2 illustrates how we evaluated the applied ML methods. Each milestone test set covers a month interval after its respective training set. Using the subset of dates given by each monthly milestone, we created samples that contained training and testing data for the subsequent experiments.

To make ILMs comparable, we trained and tested all the algorithms using the same training and testing sets. However, in the literature, ILMs for data streams are designed to be trained continuously and static training and testing splits are not generally applied to ILMs. Thus, to follow the convention with this type of learner, we performed an additional experiment for the second approach (multi-country) to measure the effect of using prequential evaluation (Cerqueira et al., 2020; Gama, 2010) for the incremental learners rather than using defined training and test splits (hold-out). The difference between the prequential and the hold-out evaluation is that the prequential retrains the model after each prediction during the 30 days that are predicted in each milestone, whereas the hold-out scheme does not.

Since the total number of models was 400 (eight milestones $\times$ 50 models), all algorithms were implemented using their default vanilla configuration in each library used, covered at the end of this section. The autoregressive order (p), the degree of differencing (d) and the moving average order (q) were optimized for ARIMA in the range [0–10], selecting the ones with the lowest Akaike Information Criteria during the first milestone. Then the resulting values for these parameters, which are necessary in ARIMA in order to fit a model, are used as input in the rest of the milestones. The seasonal parameter in ARIMA was set to false as this obtained better results in an exploratory phase using the first milestone. For the LSTM, we used the architecture from Figure 3, a validation set for back-propagation was created using the last 10 days of the training set at each

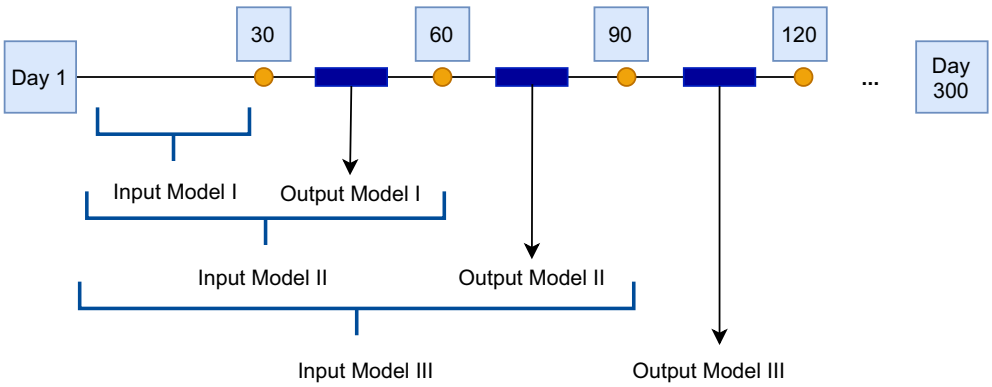

FIGURE 2 For evaluating the models, we defined eight milestones (represented with orange circles). Each milestone performs a prediction for a month ahead of a certain point in time. In milestone one, there is only 1 month of training data (30 training examples). In the second, third and fourth milestones, there are 2, 3 and 4 months of training data, respectively and so on. The test set used for each milestone is the month after the last month used for training in such milestone. Thus, in milestones one and two, the model will be tested in months two and three, respectively. The performance of the methods was computed by averaging the model errors obtained across the eight milestones.

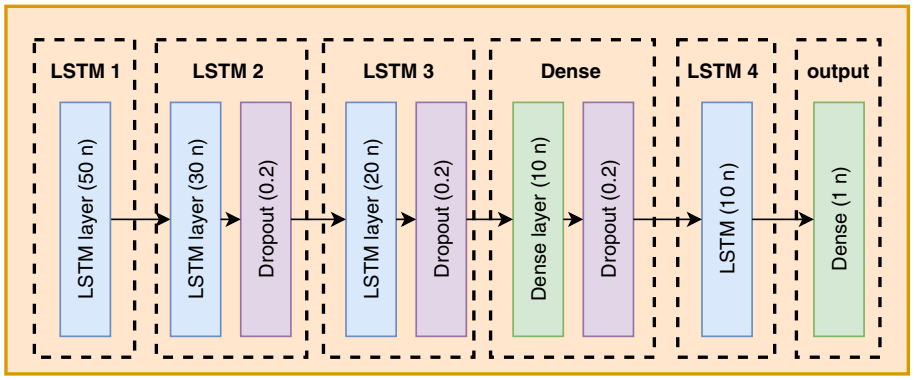

FIGURE 3 LSTM architecture used. The input layer receives the feature set and consists of 50 LSTM cells. The network has a total of four intermediate layers, three of them applying dropout at 0.2, and a final output dense layer.

respective milestone. The LSTM was trained for 500 epochs in all the experiments. However, the batch size and *patience* were different in the SC and the MC approaches.

The *batch size* refers to the number of training examples used in one iteration when training the LSTM sequentially. *Patience* represents the number of epochs to wait before early stopping training the algorithm if the model does not lower its error. For training the LSTM method, in Experiment I, we defined a batch size of 10 examples (one example per day) and a *patience* of 20 examples. In Experiment II, we defined a batch size of 500 examples (10 days multiplied by 50 countries) and a *patience* of 1000 examples (20 days multiplied by 50 countries). And in Experiment III, we defined a batch size of 100 examples (10 days multiplied by 10 countries) and a *patience* of 200 examples (20 days multiplied by 10 countries).

This methodology of dividing the dataset into milestones and calculating the error as the average of the milestones was used throughout all the experiments. The performance of the models was measured using the root mean squared error (RMSE) and the mean absolute error (MAE), which are common metrics for regression in the literature (Botchkarev, 2018). The results are calculated by comparing the model's predictions to the target feature in the test set (or test-then-train set in incremental learners). Rows in these datasets are sorted first by date and then by country for the different training and testing splits and for the batches already mentioned. Figure 4 shows the end-to-end process followed for conducting the experiments.

For the implementation of the LSTM, we have used the Python library *Keras*. For the static methods and the Passive-Aggressive Regressor, we used the provided versions of the Python library Scikit-Learn Pedregosa et al. (2011). For the rest of the incremental approaches, we used the implementation provided in another Python library called Scikit-multiflow. We have used the Python package RustDTW (version: 0.1.14) for calculating the DTW distances due to its quick processing speed. Finally, for the baseline, we used a non-seasonal auto ARIMA (p,d,q) using a Python library called 'pmdarima.arima: ARIMA estimator & differencing tests'[2], which handles automatically the optimisation of the autoregressive order (*p*), the degree of differencing (*d*) and the moving average order (*q*).

4 | EXPERIMENTS AND RESULTS

In this section, we conducted some experiments to compare the performance of three different approaches for training models. We did so using ILMs and state-of-the-art static methods, including the popular deep learning method called LSTM, to see which methods are more suitable under these approaches.

4.1 | Dataset description

For this work, we used the dataset 'COVID-19 Coronavirus data - daily (up to 14 December 2020)' available in the European Open Data Portal [3] and provided by the European Centre for Disease Prevention and Control. The original dataset contains twelve columns with daily information about the disease in 213 countries during 2020. To create the dataset for the experiments, we first selected the countries with eight or more months of data by 30 November 2020 (66 in total); then, we selected the 50 countries with the highest number of accumulated cases of COVID-19 to conduct the experiments. We considered that countries with few cases are not very helpful for making predictions.

The dataset is structured as follows: one column represents the number of positive cases; another one the number of deaths; four columns are related to the current date; four other columns are related to country-specific information; one column refers to the continent; and lastly, one column represents the cumulative number of the COVID-19 cases for 14 days per 100,000 inhabitants.

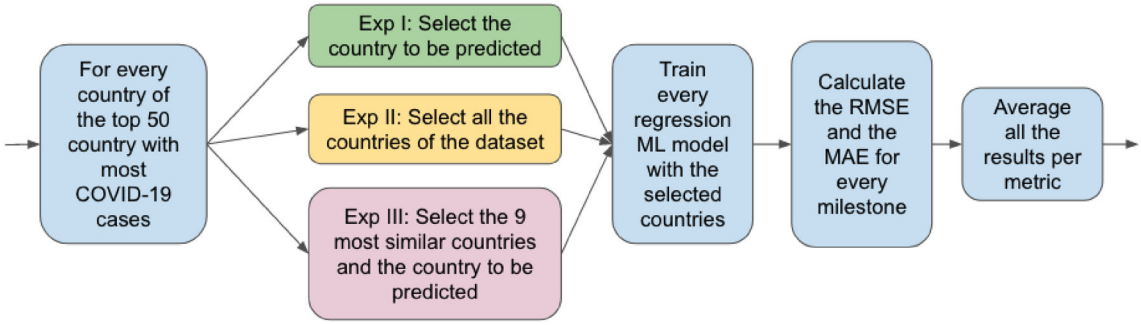

FIGURE 4 Methodology for implementing the training and testing of the three conducted experiments

Regarding the preprocessing steps for creating the final dataset for the experiments, columns related to dates and countries were used to split the original dataset into training and testing sets. The number of new cases is the only column used to generate the feature set (input of the model) and the target (output of the model). The rest of the columns were removed. Each data example used as the input of the models corresponds to a moving time window of 50 consecutive days. And the target/output of the model is the average of ten consecutive days, where the first of those 10 days is 30 days ahead of the last day of the input. The main reason for using the average of 10 days is to cushion some spikes that arise due to potential delays when reporting the test results. We also wanted to predict 30 days because it gives governments some margin of time to lift or apply new restrictions on the population.

To illustrate how the dataset was transformed, we provide the following example. Imagine we had 200 hundred days of positive COVID-19 new cases for a given country. Then, the first row of the dataset has the input from day 1 to day 50, and as the output, the average from day 80 to 90; the input for the second row goes from day 2 to 51, and the output is the average from 81 to 91; and so on. As a result, the generated dataset has 50 columns representing the number of new cases in the previous days and a single value representing the output of the model. The number of rows for each trained model varies according to the experiment, as explained in more detail in Section 4.2. The feature set (inputs of the model) was created according to the scheme shown in Figure 5.

4.2 | Experimentation

This subsection shows the results in different tables and plots the results of the three approaches.

4.2.1 | Experiment I: single-country training

This experiment trains the supervised ML models with data from a single country and predicts the cases for that same country. Results are obtained by averaging the eight points called milestones for which predictions are made. The mean performance of the incremental and static methods for the single country (SC) approach is shown in Table 1. We added the statistical model ARIMA to our experiments to use it as a baseline for our experiments.

Table 1 shows the average RMSE and MAE metric and time in seconds for the 50 countries with the most significant number of cases on 30 November 2020, when running over a single country. It can be seen how the algorithms Gradient Boosting, Decision Trees and Random Forests algorithm outperform the rest of the algorithms in terms of RMSE and MAE. The LSTM algorithm obtains a comparable predictive accuracy, but it requires more time and its training is more complex. If we had a dataset with more samples, it would probably perform better since deep learning is known for requiring a lot of data to perform well. We can also see that the order of the countries by RMSE and by MAE is the same.

Figure 6 shows a boxplot comparing all the algorithms averaging the results for the eight milestones and the 50 countries. It can be seen how static methods outperform the rest of the methods in terms of RMSE. ARIMA obtains comparative results in terms of mean accuracy to static Machine Learning algorithms in this experiment. These results are also visible in Table 1, and a similar trend can be observed there in terms of MAE. Gradient Boosting outperforms the rest of the methods in the single-country experiment and also outperforms methods like ARIMA and the LSTM model in terms of computational time. Albeit the top performers can change in different countries, it is visible that static methods

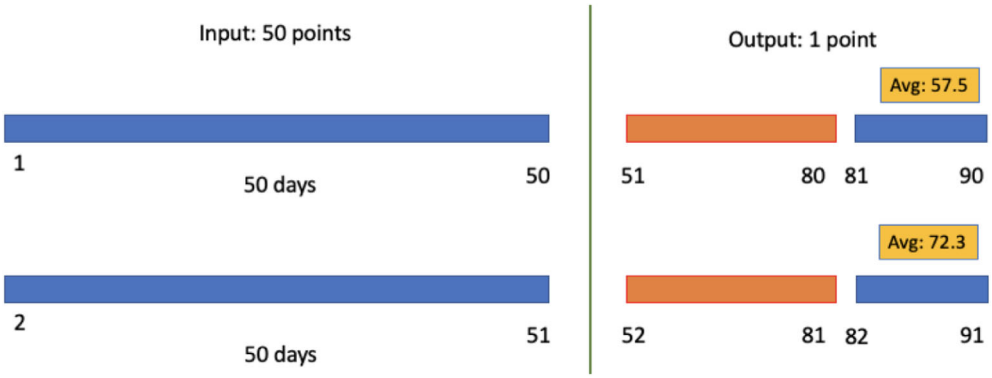

FIGURE 5 Feature set used in the experiments over the training and test sets explained in Figure 2. The number of new cases in the previous 50 days is passed individually and in order (day by day) to the model as an input feature vector. The prediction target is the average of new cases per day in the 10 consecutive days after a month ahead; the 30 days ahead are applied because the test set is always 1 month after the training set.

TABLE 1 Models trained with the single-country (SC) approach

Metric	RMSE	MAE	Time (s)
Gradient Boosting	1821	1635	0.147
Decision Tree	1976	1713	0.005
Random Forest	2187	2031	0.197
ARIMA	2483	2180	2.082
LSTM	3311	3053	20.525
Bayesian Ridge	7165	5934	0.011
Hoeffding Adaptive Trees*	15,653	11,502	0.146
Hoeffding Trees*	19,412	13,548	0.115
Adaptive Random Forest*	23,923	17,336	2.909
Linear SVR	37,518	31,370	0.02
Passive Aggressive Regressor*	111,570	91,809	0.002

Note: Results show the mean average RMSE and MAE for the prediction of the 50 countries. The symbol (*) indicates that it is an incremental method. We used ARIMA as a baseline.

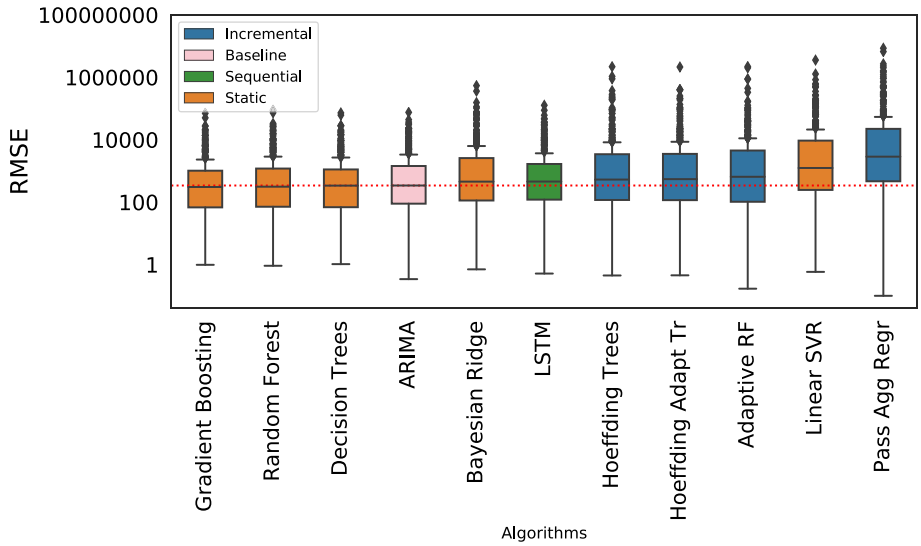

FIGURE 6 Boxplot of the RMSE for each algorithm for the eight milestones of each of the 50 predicted countries applying the single-country approach (SC).* ARIMA was used as a baseline across experiments and its median is marked as a red dotted line.

outperform ILMs in predictive accuracy. Most of the implemented algorithms had an average run-time per milestone and a country lower than 1 s for predicting each country. Thus, none of these algorithms presents issues for working in real-time with the current training set sizes.

4.2.2 | Experiment II: multiple-countries training

In the second experiment, we predicted over the same 50 countries at the same eight points as in Experiment I. Still, this time we used a single model trained with 50 countries rather than training 50 models using a single country as we did in Experiment I. Table 2 shows that Gradient Boosting is still the best-performing method for predicting COVID-19 cases. The results show some discrepancies between the performance of the methods when they are ranked using the RMSE and the MAE. However, the ranking of the methods between the two metrics is still consistent, and the differences are just one position up or down in the ranking. There are very few differences in predictive performance between the static methods. Conversely, there are clear differences between the incremental (marked with an asterisk) and the static methods. In this case, ARIMA cannot be used as a baseline due to its constraints to be trained with multiple countries at once, as it is a univariate technique. ARIMA cannot be trained with multiple time-series data at the same time. Therefore, we could not implement ARIMA in multi-country approaches. Figure 7 shows that the SC experiment obtains lower errors than the MC experiment. We believe that this behaviour is a consequence of grouping the number of

TABLE 2 Average of the RMSE and MAE for 50 predictions results using multiple-countries (MC) to train each model ordered by RMSE

Metric	RMSE	MAE	Time (s)
Gradient Boosting	7500	3,052	6.86
Bayesian Ridge	7671	2943	0.04
Random Forest	7739	3197	3.11
LSTM	8272	3089	138.57
Decision Tree	8764	3663	0.46
Linear SVR	12,824	4577	1.91
Hoeffding Adaptive Trees*	13,946	4803	21.23
Adaptive Random Forest*	14,151	4599	163.51
Hoeffding Trees*	17,381	5496	7.26
Passive Aggressive Regressor*	130,604	40,041	0.005

Note: The symbol (*) indicates it is an incremental method.

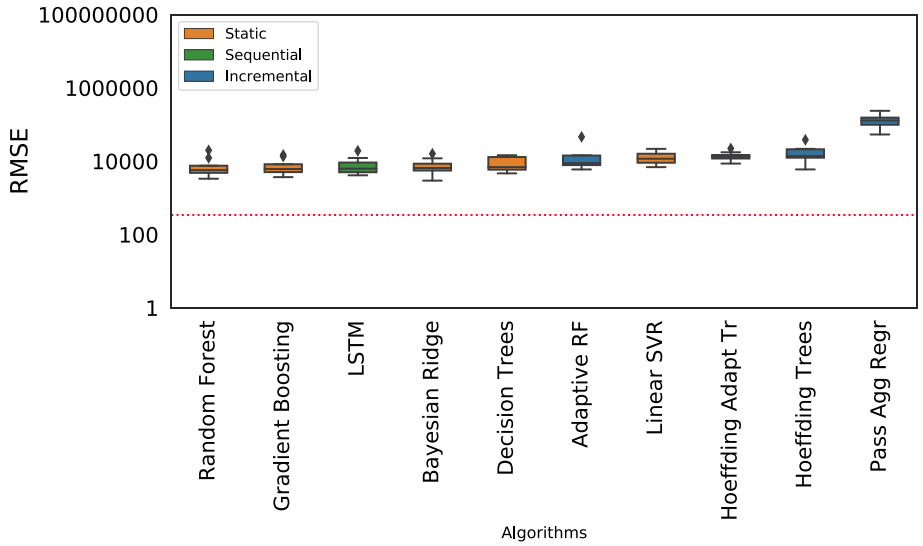

FIGURE 7 RMSE per algorithm for the six time-points (monthly) for the multi-country experiment (MC) covering the 50 countries with the most cases. The performance of ARIMA for the SC experiment (Figure 6) is shown as a baseline using a red dotted line.

cases across countries regardless of the state of the epidemiological curve. This will be addressed in the next section, grouping using similarity metrics.

Figures 6 and 7 are ordered by the median of the RMSE values in eight-time points. Static ensembles outperform ILMs in the single-country and multi-country experiments in our study. As aforementioned, the common framework to compare incremental to traditional ML methods used in this paper could be a constraint jeopardizing the performance of the incremental algorithms. Thus, we performed an experiment to compare the predictive accuracy of the incremental learners using hold-out with a test-then-train evaluation (prequential). Prequential evaluation is used as a convention in the incremental learning literature.

Table 3 shows that all but one of the ILMs improved their performance in terms of MAE by using a prequential evaluation. However, their RMSE is higher for all the algorithms than the Passive-Aggressive Regressor, which benefited from a prequential evaluation. These results do not suggest a significant improvement in predictive accuracy for any incremental learners based on decision trees. These algorithms also perform low when compared to static learners, and the reason for this behaviour could be beyond the simple selection of the training mechanism. We believe that this is because the selected feature sets act as a limiting factor for the batch size and the frequency of model updates. The adaptive approaches may need shorter sliding windows to adapt on time to any changes in the COVID-19 curve. Hence, compared to ILMs, the static learners show the best predictive performance and run times across all the algorithms in the current experimental setup. Figure 8 shows that as months go by, the RMSE error tends to increase, although the models have more data. However, we need to bear in mind that this is probably because there are more cases, and therefore the absolute errors in the predictions are higher. The option of using percentage error metrics has

TABLE 3 Average RMSE results for the eight milestones of multi-country experiments comparing the hold-out versus the prequential evaluation

Metric	RMSE	MAE
Adaptive Random Forest (prequential)	16,398	4258
Adaptive Random Forest (hold-out)	14,151	4599
Hoeffding Adaptive Trees (prequential)	18,074	4891
Hoeffding Adaptive Trees (hold-out)	13,946	4803
Hoeffding Trees (prequential)	18,075	4892
Hoeffding Trees (hold-out)	17,381	5496
Passive Aggressive Regressor (prequential)	83,661	22,229
Passive Aggressive Regressor (hold-out)	130,604	40,041

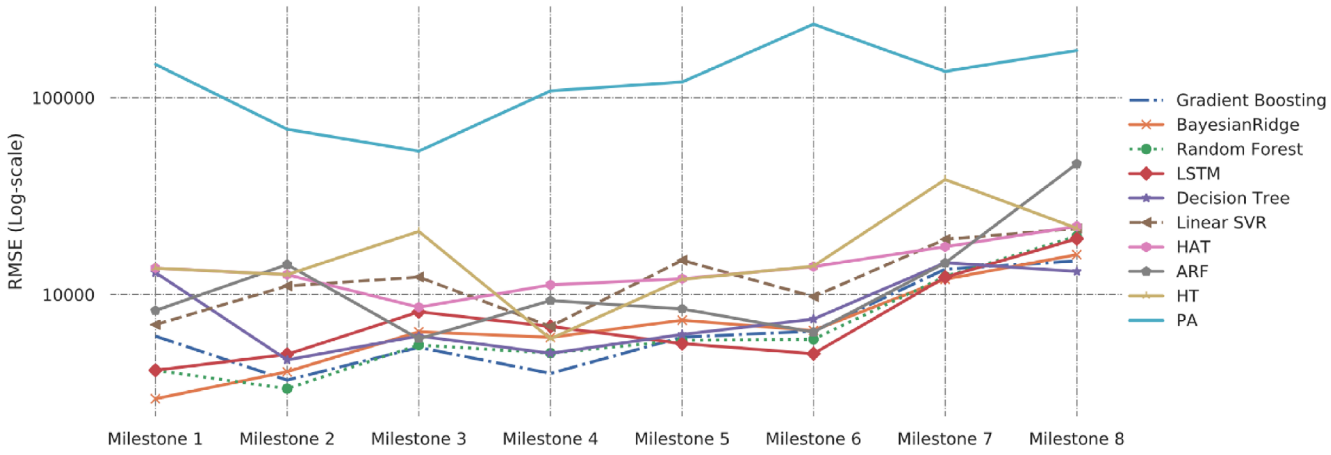

FIGURE 8 Evolution of the RSME across milestones. Average RMSE across countries per algorithm in the MC experiment

the opposite issue; it penalizes when low cases are predicted, especially when there are zero cases. That is why most papers use RMSE and MAE as metrics.

4.2.3 | Experiment III: multi-countries training by similarity

In the third experiment, we applied an approach that has hardly been applied in the literature. First, we used the time similarity measures (ED and DTW) to calculate the nine more similar countries to the predicted country based on the COVID-19 time series. The similarity measures compute the time series of COVID-19 cases (curves) of two countries and return a value representing the distances between the points of the two curves. And then, we create a dataset to train the models with the data from those nine countries and the predicted one. We did this at each milestone, as in Experiments I and II. The obtained results are displayed in Table 4. The ML algorithms are represented as rows and are sorted by the average RMSE obtained across experiments. Experiments are represented as columns to make them easier to compare.

By computing the mean of each of the experiments (columns), we could observe that the average RMSE was 22,885 for MC and 22,454 for SC. The ED reduced by 74.21% and 74.70% the RMSE obtained in the single country and multiple country approach, respectively. Likewise, the DTW reduced by 74.89% and 75.36%, a little more than the previous performance. For some algorithms, such as Linear SVR, the obtained error was 8.5 times lower when grouping by DTW than in the SC approach. In Table 4, DTW is shown as the best similarity metric since eight of the 10 algorithms trained after grouping with this metric obtain a lower RSME than the ones trained after grouping by ED. The results show that applying clustering based on time similarity measures is a very efficient and promising approach.

Table 4 compares the mean results of all algorithms across experiments in terms of MAE. There is a high-performance gain in terms of MAE in the multi-country experiment that does not occur in terms of the RSME. This means that there are more significant errors in the multi-country approach at specific points in time. Bear in mind that RSME penalizes large errors as under or over-estimations due to squaring the differences. In contrast, these large errors at certain points in time can be reduced in terms of MAE by periods of very accurate forecasts. The multi-country experiment obtains a lower cumulative mean absolute error overall. This does not occur for the static algorithms, which are overall the best

TABLE 4 Aggregate results from Experiments I, II and III (DTW and ED)

N	Algorithm	RMSE					MAE				
		SC	MC	ED	DTW	Mean	SC	MC	ED	DTW	Mean
1	Gradient Boosting	1822	7500	1908	1896	3282	1636	3052	1425	1419	1883
2	Random Forest	2188	7739	1811	1800	3385	2032	3197	1389	1382	2000
3	Decision Tree	1977	8764	2171	2155	3767	1713	3663	1555	1547	2120
4	LSTM	3311	8272	2277	2253	4028	3054	3089	1770	1748	2415
5	Bayesian Ridge	7166	7671	1839	1845	4630	5935	2943	1422	1429	2932
6	Hoeffding Adaptive Trees*	15,654	13,946	2594	2827	8755	11,502	4803	2012	2229	5137
7	Hoeffding Trees*	19,412	17,381	2424	2894	10,528	13,548	5496	1881	2273	5800
8	Adaptive Random Forest*	23,923	14,151	3120	3102	11,074	17,336	4599	2308	2313	6639
9	Linear SVR	37,518	12,824	4448	4405	14,799	31,371	4577	3366	3321	10,659
10	Passive Aggressive Regressor*	111,570	130,604	35,319	33,204	77,674	91,809	40,041	23,550	21,724	44,281

Note: The ED reduced by 74.21% and 74.70%, and the DTW reduced by 74.89% and 75.36% the RMSE obtained in the single country and multiple country approach, respectively. And the ED reduces by 77.39% and 46.09%, and the DTW reduces by 78.11% and 47.80% the MAE. Results are sorted by the mean RMSE across the experiments.

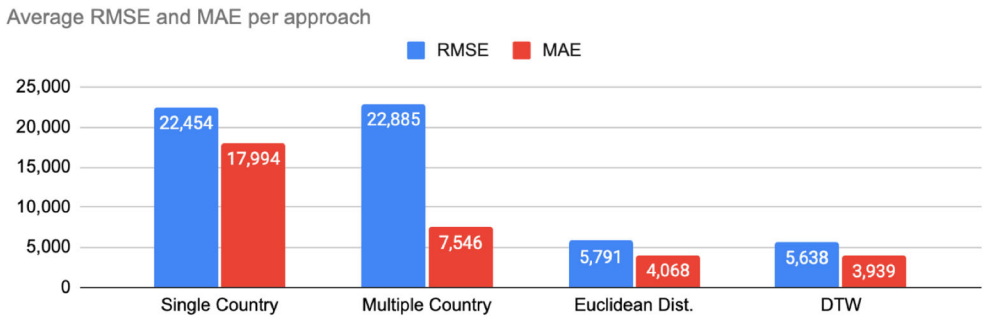

FIGURE 9 Average performance in terms of RMSE and MAE for the 10 MI methods (incremental plus static) used in the experiments.

performers in terms of MAE and RSME across experiments. Gradient Boosting and Random Forest have been proven to be the best-performing models for MAE and RSME, obtaining lower predictive error under both metrics. A summary of the performance for all the approaches can be seen in Figure 9, which represents the average performance of the 10 MI methods for each approach. It can be seen that for both RMSE and MAE, the use of time series similarity measures improves the predictive performance significantly.

In terms of the SC and MC approaches in Tables 4, it is visible how these experiments outperform each other for different algorithms. In terms of MAE, the tree-based static models, which are some of the top performer methods, obtained significantly lower MAE in the SC experiment, negatively affected by the grouping of 50 countries. Conversely, Bayesian Ridge, Linear SVR and the ILMs are positively affected in terms of MAE in the MC experiment. In terms of RSME, a similar trend can be observed; the LSTM and the Passive-Aggressive Regressor are also negatively impacted by the MC experiment. Differences between MAE and RSME can occur due to the nature of both metrics computing the error since RSME penalizes large errors. Hence, the MC approach makes the LSTM and the Passive-Aggressive Regressor less reliable methods (greater deviations) in error over time. Similar differences between MAE and RSME between any of these approaches in our experiments can be attributed to the same effect. When looking at these experiments at algorithm level in Table 4, training models with similar countries obtains the best average results overall. Gradient Boosting goes down to an MAE of 1419 when using Dynamic Time Warping, compared to ARIMA, which has an MAE of 2180 (Table 1 since it only applies to SC) as it can only be trained with one country at a time due to its univariate nature.

4.3 | Discussion

It is very positive to see how the use of time similarity measures can help reduce by a factor of four the RMSE of most methods, either using the DTW or the ED. We think that the DTW performs slightly better because it automatically aligns the time series curves even when they are shifted on time. Although in the experiments, we used a shorter version of DTW to accelerate the experiments that only aligned within a threshold. In

this regard, our selection of ML methods is crucial to allow a multivariate representation of the daily cases; this enables learning patterns from multiple countries concurrently, something not achievable using univariate methods such as ARIMA.

Experiments I, II and III compare traditional ML regression algorithms to Incremental Learners for 50 countries, eight-time points, and a common hold-out evaluation scheme. These results show how traditional static ML techniques perform better than the ILMs in the three experiments. ILMs are oriented to online scenarios and continuous adaptation. Models like HT and ARF are designed for data streaming scenarios and to handle large amounts of data, like in the stock market. The time series of coronavirus cases are not big enough for the standards of ILMs. The same applies to LSTM, which performs very well in other domains but not in our experiments.

While one may think that the MC approach should give enough information to most models to improve their performance, the results show the opposite. This is probably because many of these countries have very different behaviours in the evolution of COVID-19 cases and can mislead rather than help the model predict a particular country. That is to say, at the same time point, different countries may be in states of an outbreak different from each other, such as at the start or the end of a different COVID-19 wave. The presence of various disease states across countries adds extra complexity to the MC approach compared to the SC approach.

Gradient Boosting and Random Forests are the best performers in terms of predictive accuracy. On top of that, they have shown low execution times for our experiments. The linear SVR is the less accurate static learner, but apart from this method, static learners have been the best performing methods in our study.

The Passive–Aggressive Regressor shows the worst predictive accuracy among all the methods. This can be explained by the fact that this technique is targeted at purely incremental scenarios, such as other techniques from the ML literature, like the Stochastic Gradient Descent regressor. This technique is highly dependent on continuous model updates to produce accurate predictions, which is not the case in our study due to the long sliding windows used to represent each data example and for each of the milestones (see Figures 2 and 5).

The performance of the incremental learners can be justified in a similar manner. Many of these approaches are aimed at continuous data streams. Hoeffding Adaptive Tree and Adaptive Random Forest have active drift detectors designed to align to the speed of changes. In the approach followed, a batch of many weeks may hold abrupt changes in the COVID-19 spread curve. Under this design, the adaptive learners are unable to adapt until the end of the batch. These would benefit from a different feature set and an experimental methodology oriented to streaming settings. But this is out of scope in the current work since it would penalize the static learners that can clearly handle this scenario well with the current data.

Finally, in terms of time execution, the LSTM method requires a long time to be trained, about 40 times more than that of the first three methods. This is due to the iterative design of training a neural network. Surprisingly, the incremental learners also consume more execution time, which could be justified by the incremental updates every batch, and by the abrupt drifts that would trigger internal actions in the algorithms (removal of base classifiers and pruning of branches) at every drift detected.

5 | CONCLUSION

This work explores the suitability of using time series similarity measures like DTW in combination with ML algorithms trained to predict the time series of coronavirus cases. It is crucial to do more research to find the best strategies and methodologies to tackle outbreaks of future viruses so that their effects on public health can be addressed in a more effective way.

Our research utilized the daily information on COVID-19 cases during 2020 for 50 different countries. We recreated the situation in which countries had to flatten the curve of the number of cases while protecting the economy of the country at the same time. At that moment, there was little information about the spread of the virus and the new outbreaks. Our research is valuable because of the insights we got from the experiments. We compared the performance of training models under three approaches: using a single country, using multiple countries and using only those countries similar to the one that was predicted. Additionally, we compared the performance of state-of-the-art static versus online ILMs for predicting the number of new cases. We used ARIMA as a baseline, and many of the methods proposed were notably more accurate than this baseline.

We highlight that using time similarity measures such as ED and DTW worked out very well, drastically increasing the performance of all methods. The methodologies based on compartmental models that try to learn the behaviour of a virus on the population based on epidemiological models are much more complex since they need to consider many parameters involved in a pandemic, and these parameters need to be properly calibrated. This is very difficult because a lot of different variables (sociological, medical, political, economic, cultural, etc.) affect the virus's evolution, and their values are constantly changing. On the other hand, our methodology is straightforward and quick to implement, and we have proved its effectiveness.

We have implemented online incremental ML models which had not been applied for forecasting coronavirus cases. ILMs, a priory, represented a relevant alternative due to their ability to adapt to non-stationary behaviours, which is a characteristic of epidemic curves. However, possibly because the feature set selected with long sliding windows adds constraints to continuous updates, the ILMs did not perform well compared to the static ones even when applying a more suitable training scheme for them called prequential evaluation, in which the model is updated continuously in batches.

The most interesting points we consider in our paper are:

- The RMSE of implementing ARIMA is 2483 and the RMSE when using TSSM with Gradient Boosting is reduced to 1822, which is a clear improvement. When compared to univariate methods like ARIMA, selecting multivariate Machine Learning methods brings the possibility of learning patterns from multiple countries concurrently (MC, ED and DTW experiments). In some of these experiments, training models with similar countries gives the best mean results overall. For instance, Gradient Boosting goes down to an MAE of 1419 cases when using Dynamic Time Warping, compared to ARIMA, which had the best result with an MAE of 2180. In this domain, a difference of this sort is significant since the number of cases can grow exponentially in a matter of days. Thus, ARIMA obtained competitive results in the SC experiment but its performance is not so good when compared to multivariate algorithms trained with the similar-countries approach.
- The version of ARIMA used displayed higher runtimes, not being competitive in this aspect with other Machine Learning algorithms like Gradient Boosting.
- The idea of using time series similarity measures increases the accuracy of the methods and reduces the RMSE by a factor of four, and by a greater factor in terms of MAE. We think this is an exciting path to explore in the future.
- Training models with all the countries give slightly worse performance than doing so training with only one country. This is probably because COVID-19 behaves differently across countries at different points in time, and some countries may mislead the model rather than help it.
- ILMs obtained lower performance than that static methods. We believe this is due to abrupt changes in the COVID-19 spread curve and constraints for the models to adapt in time due to the selected feature set and experimental methodology. These changes are well captured by the feature set selected and handled by the static learners.
- Gradient Boosting, Random Forest, and Decision Trees are very accurate and robust methods for the three experiments. On the other hand, the performance of the Passive-Aggressive Regressor is very poor in all the experiments, demonstrating that it is not a suitable algorithm for the current experimental design. Also, Linear SVR does not perform very well either.
- The LSTM model did not perform as well as expected even though it is considered the best method for time series in many domains. It is quite likely that if we run some experiments to optimize the values for its parameters (hyperparameter tuning), we can get better performance from this method. Still, this step also adds more complexity to the experiments, which we are trying to keep as simple as possible.

For future work, we would like to explore further the idea of applying time series to select similar countries. Some aspects of this approach can be explored, such as the optimal threshold for a given measure, the most suitable measures for each method, or the methods that work best for this approach. We should also consider time as a relevant parameter when selecting time similarity measures. There are many time similarity measures whose performance in the context of coronavirus forecasting could be explored. Lastly, we would like to deepen on how changing the size of the time window prediction may affect the performance of the methods.

ACKNOWLEDGMENT

Open access funding provided by IReL.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available in European Centre for Disease Prevention and Control at https://data.europa.eu/data/datasets. These data were derived from the following resources available in the public domain: - COVID-19 Coronavirus data - daily (up to 14 December 2020), https://data.europa.eu/data/datasets/covid-19-coronavirus-data-daily-up-to-14-december-2020?locale=en

ORCID

Luis Miralles-Pechuán https://orcid.org/0000-0002-7565-6894
Andrés L. Suárez-Cetrulo https://orcid.org/0000-0001-5266-5053

ENDNOTES

[1] http://www.github.com/ankitk2109/Covid_Evolution_Using_Incremental_ML

[2] https://alkaline-ml.com/pmdarima/modules/classes.html

[3] https://data.europa.eu/euodp/en/data/dataset/covid-19-coronavirus-data-daily-up-to-14-december-2020

REFERENCES

Adiga, A., Dubhashi, D., Lewis, B., Marathe, M., Venkatramanan, S., & Vullikanti, A. (2020). Mathematical models for covid-19 pandemic: A comparative analysis. *Journal of the Indian Institute of Science*, *100*(4), 793–807.

Barnard, R. C., Davies, N. G., Pearson, C. A., Jit, M., & Edmunds, W. J. (2021). Projected epidemiological consequences of the omicron sars-cov-2 variant in England, december 2021 to april 2022. *medRxiv*, 2021.12.15.21267858. https://doi.org/10.1101/2021.12.15.21267858

Benvenuto, D., Giovanetti, M., Vassallo, L., Angeletti, S., & Ciccozzi, M. (2020). Application of the Arima model on the covid-2019 epidemic dataset. *Data in Brief*, *29*, 105340.

Bifet, A., & Gavalda, R. (2007). Learning from time-changing data with adaptive windowing. Proceedings of the 2007 siam international conference on data mining (pp. 443–448).

Bifet, A., & Gavaldà, R. (2009). Adaptive learning from evolving data streams. International symposium on intelligent data analysis (pp. 249–260).

Bishop, C. M. (2006). *Pattern recognition and machine learning*. springer.

Bonaccorsi, G., Pierri, F., Cinelli, M., Flori, A., Galeazzi, A., Porcelli, F., Schmidt, A. L., Valensise, C. M., Scala, A., Quattrociocchi, W., & Pammolli, F. (2020). Economic and social consequences of human mobility restrictions under covid-19. *Proceedings of the National Academy of Sciences*, *117*(27), 15530–15535.

Botchkarev, A. (2018). Performance metrics (error measures) in machine learning regression, forecasting and prognostics: Properties and typology. *arXiv Preprint*, arXiv:1809.03006.

Brauer, F. (2008). Compartmental models in epidemiology. In *Mathematical epidemiology* (pp. 19–79). Springer.

Cerqueira, V., Torgo, L., & Mozetic, I. (2020). Evaluating time series forecasting models: An empirical study on performance estimation methods. *Machine Learning*, *109*(11), 1997–2028. https://doi.org/10.1007/s10994-020-05910-7

Chang, C.-C., & Lin, C.-J. (2011). Libsvm: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, *2*(3), 1–27.

Chang, S. L., Harding, N., Zachreson, C., Cliff, O. M., & Prokopenko, M. (2020). Modelling transmission and control of the covid-19 pandemic in Australia. *Nature communications*, *11*, 5710. https://doi.org/10.1038/s41467-020-19393-6

Chin, V., Samia, N. I., Marchant, R., Rosen, O., Ioannidis, J. P., Tanner, M. A., & Cripps, S. (2020). A case study in model failure? Covid-19 daily deaths and icu bed utilisation predictions in New York state. *European Journal of Epidemiology*, *35*(8), 733–742.

Crammer, K., Dekel, O., Keshet, J., Shalev-Shwartz, S., & Singer, Y. (2006). Online passive-aggressive algorithms. *Journal of Machine Learning Research*, *7*(19), 551–585. http://jmlr.org/papers/v7/crammer06a.html

Ditzler, G., Roveri, M., Alippi, C., & Polikar, R. (2015). Learning in nonstationary environments: A survey. *IEEE Computational Intelligence Magazine*, *10*(4), 12–25. https://doi.org/10.1109/MCI.2015.2471196

Domingos, P., & Hulten, G. (2000). Mining high-speed data streams. Proceedings of the sixth acm sigkdd international conference on knowledge discovery and data mining (pp. 71–80).

Elwell, R., & Polikar, R. (2011, October). Incremental learning of concept drift in nonstationary environments. *IEEE Transactions on Neural Networks*, *22*(10), 1517–1531.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, *29*, 1189–1232.

Gama, J. (2010). *Knowledge discovery from data streams* (1st ed.). Chapman & Hall/CRC.

Gama, J., Sebastião, R., & Rodrigues, P. P. (2013). On evaluating stream learning algorithms. *Machine Learning*, *90*, 317–346. https://doi.org/10.1007/s10994-012-5320-9

Gama, J. A., Žliobaite, I., Bifet, A., Pechenizkiy, M., & Bouchachia, A. (2013, mar). A survey on concept drift adaptation. *ACM Computing Surveys*, *1*(35), 1–37. https://doi.org/10.1145/0000000.0000000

Gomes, H. M., Bifet, A., Read, J., Barddal, J. P., Enembreck, F., Pfharinger, B., Holmes, G., & Abdessalem, T. (2017). Adaptive random forests for evolving data stream regression. *Machine Learning*, *106*(9–10), 1469–1495. https://doi.org/10.1007/s10994-017-5642-8

Gyllenberg, M., Hastings, A., & Hanski, I. (1997). Structured metapopulation models. In *Metapopulation biology* (pp. 93–122). Elsevier.

Harvey, A., & Kattuman, P. (2020). *Time series models based on growth curves with applications to forecasting coronavirus*. Harvard Data Science Review.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: Data mining, inference, and prediction*. Springer Science & Business Media.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780.

Hsu, M. W., Lessmann, S., Sung, M. C., Ma, T., & Johnson, J. E. (2016). Bridging the divide in financial market forecasting: Machine learners vs. financial economists. *Expert Systems with Applications*, *61*, 215–234. https://doi.org/10.1016/j.eswa.2016.05.033

IHME COVID-19 health service utilization forecasting team, & Murray, C. J. L. (2020). *Forecasting covid-19 impact on hospital bed-days, icu-days, ventilator-days and deaths by us state in the next 4 months*. MedRxiv.

Kukar, M. (2003). Drifting concepts as hidden factors in clinical studies. In M. Dojat, E. T. Keravnou, & P. Barahona (Eds.), *Artificial intelligence in medicine* (pp. 355–364). Springer Berlin Heidelberg.

Lalmuanawma, S., Hussain, J., & Chhakchhuak, L. (2020). *Applications of machine learning and artificial intelligence for covid-19 (sars-cov-2) pandemic: A review* (p. 110059). Chaos.

Landmesser, J. (2020). Analysis of covid-19 dynamics in eu countries using the dynamic time warping method and arima models. In *Conference of the section on classification and data analysis of the polish statistical association* (pp. 337–352).

Li, G., Li, W., He, X., & Cao, Y. (2020, April). Asymptomatic and Presymptomatic infectors: Hidden sources of coronavirus disease 2019 (COVID-19). *Clinical Infectious Diseases*, *71*(8), 2018. https://doi.org/10.1093/cid/ciaa418

Liaw, A., & Wiener, M. (2002). Classification and regression by randomforest. *R News*, *2*(3), 18–22.

MacKay, D. J. (1992). Bayesian interpolation. *Neural Computation*, *4*(3), 415–447.

Maleki, M., Mahmoudi, M. R., Wraith, D., & Pho, K.-H. (2020). Time series modelling to forecast the confirmed and recovered cases of covid-19. *Travel Medicine and Infectious Disease*, *101742*, 101742.

Miralles-Pechuán, L., Jiménez, F., Ponce, H., & Martínez-Villaseñor, L. (2020). A methodology based on deep q-learning/genetic algorithms for optimizing covid-19 pandemic government actions. Proceedings of the 29 th acm international conference on information & knowledge management (pp. 1135–1144).

Miralles-Pechuán, L., Ponce, H., & Martínez-Villaseñor, L. (2020). Optimization of the containment levels for the reopening of Mexico City due to covid-19. *IEEE Latin America Transactions*, *19*(6), 1065–1073.

Montgomery, D. C., Peck, E. A., & Vining, G. G. (2012). *Introduction to linear regression analysis* (Vol. 821). John Wiley & Sons.

Müller, M. (2007). *Information retrieval for music and motion* (Vol. 2, p. 59). Springer.

Murtagh, F. (1991). Multilayer perceptrons for classification and regression. *Neurocomputing*, *2*(5–6), 183–197.

Nicola, M., Alsafi, Z., Sohrabi, C., Kerwan, A., Al-Jabir, A., Iosifidis, C., Agha, M., & Agha, R. (2020). The socio-economic implications of the coronavirus pandemic (covid-19): A review. *International Journal of Surgery (London, England)*, *78*, 185–193.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn: Machine learning in python. *The Journal of Machine Learning Research*, *12*, 2825–2830.

Rojas-Valenzuela, I., Valenzuela, O., Delgado-Marquez, E., & Rojas, F. (2021). Estimation of covid-19 dynamics in the different states of the United States during the first months of the pandemic. *Engineering Proceedings*, *5*(1), 53. https://doi.org/10.3390/engproc2021005053

Serra, J., & Arcos, J. L. (2014). An empirical evaluation of similarity measures for time series classification. *Knowledge-Based Systems*, *67*, 305–314.

Shereen, M. A., Khan, S., Kazmi, A., Bashir, N., & Siddique, R. (2020). Covid-19 infection: Origin, transmission, and characteristics of human coronaviruses. *Journal of Advanced Research*, *24*, 91–98.

Singh, B., Sun, Q., Koh, Y. S., Lee, J., & Zhang, E. (2020). Detecting protected health information with an incremental learning ensemble: A case study on new zealand clinical text. In *2020 ieee 7th international conference on data science and advanced analytics* (*dsaa*) (pp. 719–728). https://doi.org/10.1109/DSAA49011.2020.00082

Stübinger, J., & Schneider, L. (2020). Epidemiology of coronavirus covid-19: Forecasting the future incidence in different countries. In *Healthcare* (Vol. 8, p. 99). MDPI.

Suárez-Cetrulo, A. L., & Cervantes, A. (2017). An online classification algorithm for large scale data streams: iGNGSVM. *Neurocomputing*, *262*, 67–76. https://doi.org/10.1016/j.neucom.2016.12.093

Suárez-Cetrulo, A. L., Cervantes, A., & Quintana, D. (2019). Incremental market behavior classification in presence of recurring concepts. *Entropy*, *21*(1), 25. https://doi.org/10.3390/e21010025

Tipping, M. E. (2001). Sparse bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, *1*(Jun), 211–244.

Toyoshima, Y., Nemoto, K., Matsumoto, S., Nakamura, Y., & Kiyotani, K. (2020). Sars-cov-2 genomic variations associated with mortality rate of covid-19. *Journal of Human Genetics*, *65*, 1–8.

Tsymbal, A. (2004). The problem of concept drift: Definitions and related work. Technical report: TCD-CS-2004-15, Department of Computer Science Trinity College, Dublin.

Vashi, A. P., & Coiado, O. C. (2021). The future of covid-19: A vaccine review. *Journal of Infection and Public Health*, *14*(10), 1461–1465.

Wang, L., Didelot, X., Yang, J., Wong, G., Shi, Y., Liu, W., Gao, G. F., & Bi, Y. (2020). Inference of person-to-person transmission of covid-19 reveals hidden super-spreading events during the early outbreak phase. *Nature Communications*, *11*(1), 1–6.

Wang, P., & Tian, D. (2021). Bibliometric analysis of global scientific research on covid-19. *Journal of Biosafety and Biosecurity*, *3*(1), 4–9.

Wynants, L., Van Calster, B., Bonten, M. M., Collins, G. S., Debray, T. P., De Vos, M., Dhiman, P., Haller, M. C., Harhay, M. O., Henckaerts, L., Heus, P., Kammer, M., Kreuzberger, N., Lohmann, A., Luijken, K., Ma, J., Martin, G. P., McLernon, D. J., Andaur Navarro, C. L., … van Smeden, M. (2020). Prediction models for diagnosis and prognosis of covid-19 infection: Systematic review and critical appraisal. *BMJ*, *369*, m1328.

Yang, Z., Zeng, Z., Wang, K., Wong, S.-S., Liang, W., Zanin, M., Liu, P., Cao, X., Gao, Z., Mai, Z., Liang, J., Liu, X., Li, S., Li, Y., Ye, F., Guan, W., Yang, Y., Li, F., Luo, S., … He, J. (2020). Modified SEIR and AI prediction of the epidemics trend of covid-19 in China under public health interventions. *Journal of Thoracic Disease*, *12*(3), 165–174.

Zeroual, A., Harrou, F., Dairi, A., & Sun, Y. (2020). Deep learning methods for forecasting covid-19 time-series data: A comparative study. *Chaos, Solitons & Fractals*, *140*, 110121.

Žliobaite, I., Budka, M., & Stahl, F. (2015). Towards cost-sensitive adaptation: When is it worth updating your predictive model? *Neurocomputing*, *150*(Part A), 240–249. https://doi.org/10.1016/j.neucom.2014.05.084

Zu, Z. Y., Jiang, M. D., Xu, P. P., Chen, W., Ni, Q. Q., Lu, G. M., & Zhang, L. J. (2020). Coronavirus disease 2019 (covid-19): A perspective from China. *Radiology*, *200490*, E15–E25.

AUTHOR BIOGRAPHIES

Luis Miralles-Pechuán is currently an Assistant Lecturer at Technological University Dublin. He obtained his PhD and Bachelor in Computer Science at the University of Murcia (Spain). He worked as a full-time researcher/lecturer at University Panamericana in Mexico for more than three years. He decided to start a PhD in 2012 on creating new approaches within the Online Advertising world. During his PhD, he got familiar with ML and he published a good number of papers on topics related to how to apply ML to online advertising. After finishing his PhD, he worked in postdoc levels I and II in CeADAR, University College Dublin.

Ankit Kumar is a skilled computer science professional with a strong background in software engineering and data science. He earned his Bachelor's degree in Computer Science from the University of Pune and went on to work as a software engineer at Tibco for 2 years. Seeking to expand his knowledge and skills in the field, Ankit then pursued a Master's degree in Computer Science with a specialization in Data Science from University College Dublin (UCD). He is currently working as a Senior Data Scientist at Liberty IT, where he utilizes his expertise to drive business success through the analysis and interpretation of complex data sets.

Andrés L. Suárez-Cetrulo received his BSc and MSc in Computer Science at Carlos III of Madrid (Spain)in 2013 and 2014, respectively. He received his PhD in Computer Science at Carlos III of Madrid in 2022. He is currently a Data Science Architect at Ireland's National Centre for Applied AI, based at University College Dublin. His current interests focus on online machine learning for data streams, regime changes in financial markets, deep learning,transformers and generative models.

How to cite this article: Miralles-Pechuán, L., Kumar, A., & Suárez-Cetrulo, A. L. (2023). Forecasting COVID-19 cases using dynamic time warping and incremental machine learning methods. *Expert Systems*, *40*(6), e13237. https://doi.org/10.1111/exsy.13237