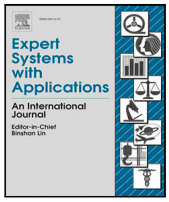
Expert Systems With Applications 258 (2024) 124930

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa



COVID-19 seroprevalence estimation and forecasting in the USA from ensemble machine learning models using a stacking strategy

Gontzal Sagastabeitia [a,*], Josu Doncel [a], José Aguilar [b,c], Antonio Fernández Anta [b], Juan Marcos Ramírez [b]

[a] UPV/EHU, Sarriena Auzoa, z.g., Leioa, 48940, Biscay, Spain
[b] IMDEA Networks, Avenida del Mar Mediterráneo, 22, Leganes, 28918, Madrid, Spain
[c] Universidad de Los Andes, Mérida, Venezuela

ARTICLE INFO

Keywords:
COVID-19
Epidemiology
Stacking ensemble method
Machine learning
Regression modelling
Genetic programming
Neural networks

ABSTRACT

The COVID-19 pandemic exposed the importance of research on the spread of epidemic diseases. In this paper, we apply Artificial Intelligence and statistics techniques to build prediction models to estimate the SARS-CoV-2 seroprevalence in the United States, using multiple estimates of COVID-19 prevalence and other explanatory variables. We propose the use of stacking techniques based on multiple model building techniques (Linear and Beta Regression, Genetic Programming and Neural Networks) to obtain Predictive Ensemble Models. There has been extensive research on this field, but there has not been in-depth research on the application of stacking methods to estimate and forecast seroprevalence in the USA specifically. This paper provides a novel comparison of the behaviour and performance of different building techniques for stacking ensemble models and presents which methods are better for different scenarios. We find that Genetic Programming and Neural Networks are the best models with trained data within single states, and when multiple states are considered Genetic Programming is still better than the Regression models, but Neural Networks fail to estimate the seroprevalence accurately. Another novelty of our work is the use of cross-state validation to evaluate the models with new data, as well as temporal forecasting. Depending on how the data is processed, Linear Regression performs very well with cross-state validation and temporal forecasting, and Genetic Programming is very accurate with the former while Neural Networks work better with the latter.

1. Introduction

The Coronavirus disease 2019 (COVID-19), caused by the severe acute respiratory syndrome Coronavirus 2 (SARS-CoV-2) (Wölfel et al., 2020), has raised public interest in epidemics. During the pandemic, media outlets mainly reported daily updates on the number of COVID-19 infections, hospitalisations, and deaths to provide information about the spread of the disease. COVID-19 estimated number of cases was primarily obtained from large-scale screening using PCR and antigen tests (Cheng et al., 2020). However, this method may not be the most reliable source of information when attempting to understand the full scope of the pandemic and accurately determine the percentage of the population affected, since the accuracy of the information obtained from test screening is affected by various factors such as the limited availability of test kits (Zoabi et al., 2021) (specially at the beginning of the pandemic), the time between infection and the test timing (Akinbami et al., 2021), and the high number of asymptomatic infected individuals, among other reasons (Klompas, 2020).

The traditional approach for estimating the proportion of previously infected individuals within a population relies on the measurement of seroprevalence. Specifically, seroprevalence refers to the proportion of individuals who test positive for a specific antibody in their blood (Farlex Partner Medical Dictionary, 2012). In the case of COVID-19, a seropositive individual is a person who has SARS-CoV-2 antibodies in their blood. The presence of antibodies is considered sufficient evidence to confirm past infection, even without a positive test result. Multiple seroprevalence studies were conducted during the COVID-19 pandemic in different countries, which required blood analyses of thousands of individuals along multiple rounds (Bajema et al., 2021; Pollán et al., 2020). These campaigns required substantial resources for logistical and organisational purposes. Hence, alternative cheaper forms of seroprevalence estimation are desirable (Garcia-Agundez et al., 2021). Therefore, the main objective and motivation of this research has been to study the use of more easily obtained data to estimate seroprevalence

* Corresponding author.
E-mail addresses: gontzal.sagastabeitia@ehu.eus (G. Sagastabeitia), josu.doncel@ehu.eus (J. Doncel), aguilar@ula.ve (J. Aguilar), antonio.fernandez@imdea.org (A. Fernández Anta), juan.ramirez@imdea.org (J.M. Ramírez).

https://doi.org/10.1016/j.eswa.2024.124930
Received 2 March 2024; Received in revised form 27 June 2024; Accepted 26 July 2024
Available online 13 August 2024

in the USA, so that expensive and resource consuming surveys could be substituted. We also wanted to be able to analyse the relationship between the explanatory variables and seroprevalence, hence we focused on more explainable stacking ensemble approaches like Linear Regression and Genetic Programming. Finally, we wanted to improve the accuracy of the stacking ensemble learning methods by combining data with predictive models in their learning process.

In this paper, we explore the use of data obtained from different sources (official number of cases, survey responses, wastewater data) to estimate the seroprevalence evolution in the USA. The models used are built combining different machine learning techniques, and stacking as ensemble method. Although the COVID-19 pandemic is losing its relevance, it has revived the fear of the spread of infectious diseases and made the population aware of the threat that pandemics can pose. Therefore, even though COVID-19 may not be as important anymore, our work can provide a general framework based on data for tracking viral spread. Furthermore, this work is not confined to epidemic tracking or even medical applications. The methods and procedures used for this problem can be applied to any dataset to predict or estimate other aspects of a population. An example of an application in other areas is to use surveys and assemble different models and data to evaluate the voting intention in future elections.

1.1. Contributions

In our work, we use different modelling techniques to estimate the seroprevalence of SARS-CoV-2 in the United States of America (USA) based on data of daily infections obtained from multiple sources. Notice that our approach does not result from mass screenings using PCR or antigen tests. Specifically, we use different artificial intelligence and statistics techniques as stacking machine learning strategies, which learn to combine the estimations of several base models to obtain a final prediction. The artificial intelligence and statistics techniques we used are: *Linear Regression* (LR), *Beta Regression* (BR), *Genetic Programming* (GP), and *Neural Networks* (NN). LR and BR were chosen because they are very simple and widely used models; and GP was chosen because, like the two previous techniques, the models it constructs explicitly show how the different explanatory variables are combined to calculate the seroprevalence rate. NN on the other hand, are a very promising field of study when working with estimations and predictions, and even though their explainability is low, we wanted to see how they perform with this kind of problems.

We consider the Sum of Squared Residuals (SSR) as the performance metric of the models, and we seek predictive models that minimise the SSR. However, we will also use the Mean Absolute Relative Error (MARE) as an additional metric to better understand the accuracy of the models, as MARE is more easy to interpret. We consider two different settings: state by state (statewide models), where data from a single state is considered for training the model; and the whole USA (nationwide models), where the data from all or multiple states is considered for training a single model.

We also present two approaches to dealing with the available data: cumulative and non-cumulative aggregation. Cumulative aggregation uses the first available seropositivity data value as a reference, while non-cumulative aggregation uses the latest available seropositivity data value.

Lastly, we tested how well the models perform with new data. In order to do that, we used a subset of the observed data at our disposal to build the prediction model, and then used it to test the trained model on estimation tasks. We considered two situations that could happen in real live when working with seroprevalence data: cross-state validation and temporal forecasting. The former works with statewide models, while the latter uses nationwide models.

The organisation of this work is as follows. We start by presenting the approach we chose to build the predictive ensemble models in Section 3. In that section, we show the process followed to build said models (Section 3.1), the data used in the models (Section 3.2), and the specifics of the models used (Section 3.4). Afterwards, we present the resulting estimations obtained with both statewide and nationwide models in Section 4, before showing the results of cross-state validation and temporal forecasting in Section 5. Finally, we sum up the results in Section 6, and present our conclusions and future work in Section 7.

2. State of the art

Overall, three research lines were the main focus of this work when approaching the problem at hand: COVID-19 estimation, stacking techniques, and Genetic Programming.

2.1. Works on COVID-19 estimation

Since the start of the COVID-19 pandemic in early 2020, many approaches have been proposed that rely on data analysis and artificial intelligence to estimate the number of daily cases (Astley et al., 2021; Quintero, Ardila, Camargo, Rivas, & Aguilar, 2021; Salomon et al., 2021; Zoabi et al., 2021). These methods exploit the ability of online tools to track health indicators in almost real-time by collecting vast amounts of data from self-reported information. Estimating the seroprevalence from this data type is required to provide healthcare systems with a less expensive method for tracking the spread of diseases. In this context, it is necessary to analyse Ensemble Methods that allow combining the different estimation approaches (regardless of whether they are based on machine learning techniques or not). In particular, we are interested in stacking techniques as an ensemble learning strategy, since it allows learning how to combine the estimates of numerous machine learning models to obtain a final estimate (Gupta, Jain, & Singh, 2022; Zhou & Jiao, 2023).

Regarding the estimation of the COVID-19 spread, there have been extensive studies carried from a statistical point of view, as seen in the thorough reviews (Comito & Pizzuti, 2022; Elsheikh et al., 2021; Jamshidi et al., 2022). Most of these works have focused on forecasting COVID-19 infections, and have extensively taken COVID-19 test positives as reference, although there are some interesting works that have tried to predict other metrics, such as COVID-19 mortality in Cui et al. (2021). Overall not a lot of research has been conducted around SARS-CoV-2 seroprevalence, which is the metric we focus on with our models.

2.2. Works on stacking techniques

Studies on the spread of COVID-19 have mainly focused in implementing machine learning-based models, as seen in Al-Bwana (2021), Lucas, Vahedi, and Karimzadeh (2023) and Vaughan et al. (2023); but stacking ensemble methods have also been widely studied, which is the approach we consider in our work: Cilgin and ÖZDEMİR (2023) for example uses LR as a stacking approach to assemble a time series model to forecast COVID-19 cases; Wang, Harrou, Dairi, and Sun (2024) researches the detection of COVID-19 blood samples via three stacked deep-learning techniques; Sharma, Gupta, and Mishra (2023) also uses deep ensemble learning methods to predict COVID-19 cases; Jin, Dong, Yu, and Luo (2022) builds an ensemble hybrid model by stacking multiple model predictions; and Wang et al. (2020) uses deep-learning stacked with ensemble techniques to build a time series model. In all the cited works except (Wang et al., 2024) they aim to predict COVID-19 positive cases, unlike in our work where we focus on seropositivity. There are also some studies that use similar methods (machine learning and stacking methods) on infectious diseases other than COVID-19, such as Mahajan et al. (2022), Soto-Ferrari, Carrasco-Pena, and Prieto (2023) and Dada, Oyewola, Joseph, Emebo, and Oluwagbemi (2022).

2.3. Works on genetic programming

When it comes to the application of GP, there has also been some work on the topic to estimate COVID-19 prevalence in the United States, like Anđelić et al. (2021); but there is much less research on the application of stacking ensemble approaches using non-machine learning-based models. Other GP application works for the case of COVID-19 are the following. The main focus of the paper of Salgotra et al. is to develop prediction models for COVID-19 confirmed cases and death cases in India, particularly in the states of Maharashtra, Gujarat, and Delhi (Salgotra, Gandomi, & Gandomi, 2020). These models, based on GP, are presented with explicit formulas, to evaluate the significance of prediction variables. Niazkar and Niazkar (2020) introduced multi-gene GP (MGGP) as a novel approach for predicting COVID-19 outbreaks. Despite the challenges posed by the fluctuating daily confirmed cases, MGGP demonstrated promising results, with predicted cases closely aligning with observed values across seven countries studied. Finally, Benolić, Car, and Filipović (2023) explored the application of GP to develop forecasting models for COVID-19 using publicly available datasets from Johns Hopkins University and the GISAID Variant database. The study focuses on Austria and neighbouring countries, creating individual models for each. Short-term models exhibit high R2 scores, while long-term predictions perform slightly lower.

There have been studies on the use of stacking ensemble models for COVID-19 detection (Rahman et al., 2022), but they used biomarkers and did not estimate seroprevalence. There have also been research on the prediction of seroprevalence in the USA, like Larremore et al. (2021), or on the forecasting of COVID-19 incidence, like Lucas et al. (2023) and Larremore et al. (2021), but without the use of stacking methods. As can be seen from the literature review, there have been a lot of research in this field, but there has not been a thorough analysis of the behaviour and performance of different building techniques and approaches for stacking ensemble models. The stacking approaches developed so far for the detection of COVID-19 use biomarkers, the seroprevalence prediction approaches are not stacking-based, and in general, none have used USA data obtained from COVID-19 surveys.

In this work, we present our findings in this direction, analysing the quality of the ensemble models built using four different stacking techniques. Additionally, there has not been much research on the applicability of stacking ensemble models to geographically new data either, so we have tried contributing to this line of research by studying cross-state validation as an evaluation tool for the models built, which is useful to check the performance of the models with untrained data.

On the other hand, ensemble methods usually combine estimation models as input variables to provide predictions, but in our case, we have also considered new explanatory variables (not estimations) alongside the estimation models when stacking the ensemble models, which provides the final model with further data on the pandemic. Finally, we have also evaluated the capabilities of the ensemble models in temporal forecasting tasks.

3. Our approach

There is a lot of data sources that provide useful information with respect to the epidemic, including estimated COVID-19 prevalence rates, via different prediction methods based on COVID-19 prevalence surveys. To reduce the biases each method may have, we propose to use all of them to construct an ensemble model that will use the estimations of all those methods, as well as some extra explanatory variables (mainly, official prevalence data and estimates from wastewater SARS-CoV-2 concentration studies). Our problem consists of finding an appropriate prediction model that combines said estimations and variables to predict the seropositivity of a certain population on a given date.

3.1. Our ensemble method

We require a stacking machine learning strategy, which learns to combine the estimates from several methods with the extra explanatory variables, to obtain a final estimate. The seropositivity values we use as ground truth for our work are the seroprevalence measurements made by the Centers for Disease Control and Prevention (CDC), the national public health agency of the USA (Centers for Disease Control and Prevention, 2023).

We have used only a stacking approach as an ensemble learning method, and have not compared it with other ensemble methods, such as classical bagging and boosting, because our desire was to combine robust base models (strong learners) with other variables that would enrich the estimate. The stacking ensemble method is a learning technique that combines multiple strong base models, often of different types, to improve performance. While bagging and boosting are also ensemble methods, they are based on the premise of combining weak learners, often of the same type, under different creation schemes. Our problem is to find the expression that best combines these strong learners, using stacked learning to capture these more complex relationships from multiple base models. Particularly, in this article we explore several techniques to carry out the stacking learning process.

In Fig. 1, a diagram schematically explains our stacking machine learning strategy to obtain the seroprevalence rate estimations. The modules that compose the diagram can be separated into two types: the circular modules represent variables or data (the independent and dependent variables and the ground truth, see Section 3.2), and the rectangular modules represent processes performed when constructing the ensemble model (data processing and modelling, see Sections 3.4 and 3.2.3). The arrows of the diagram indicate the path that the explanatory variables follow before returning an estimation of the seroprevalence rate, and then, the $\simeq$ symbol indicates that the output should be close to the real seroprevalence (ground truth), which is evaluated using SSR and MARE as metrics.

As the figure shows, we have various input variables from multiple data sources that need to be aggregated before we can use them to build our prediction models. It is important to note that the values of these variables come from estimation/prediction models based on machine learning techniques (Random Forest — RF, Extreme Gradient-Boosting — XGB, etc.), or from estimation models based on wastewater virus concentrations (Wastewater cases — WWC), or are extra explanatory variables (New reported cases — NRC, etc.). The source data used to build these base models are the US COVID-19 Trends and Impact Survey (CTIS) data from the Delphi Group at Carnegie Mellon University and Facebook project (Delphi Group at Carnegie Mellon University, 2022; Salomon et al., 2021), and the wastewater concentration data from Srivastava (2022).

On the other hand, a data aggregation phase is necessary because there is an inconsistency between the number of input data points and the ground truth (seropositivity). The latter comprises at most 30 measurements per USA state, while most input variables have daily values. Therefore, we aggregate the input data into the same number of data points as the ground truth.

After aggregation, the variables are combined into a prediction model based on a stacking ensemble strategy, which outputs estimated seroprevalence rates. We use four different modelling techniques as stacking ensemble methods to build the predictive models: LR, BR, GP and NN. Therefore, we are using regression models, as well as GP and NN-based models as ensemble learning techniques that combine the base models' estimations, as well as new data unrelated to the base models, to obtain the final seroprevalence prediction model. These approaches take the explanatory variables, which, as stated before, some of which are COVID-19 prevalence estimations, and build a prediction model that combines the previous estimations, as well as new data, to produce a SARS-CoV-2 seropositivity estimation. With the exception of NN, the resulting prediction models are provided via

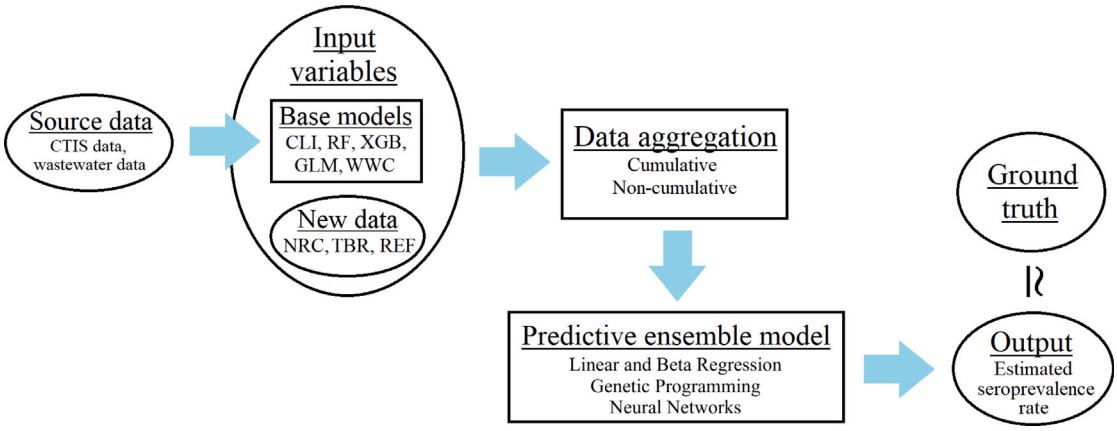


Fig. 1. Diagram showing our stacking machine learning strategy. The modules are separated into two types: the circular modules represent variables or data used, and the rectangular modules represent processes performed on the data to obtain predictions (data processing and modelling). The arrows of the diagram indicate the path that the explanatory variables follow before returning an estimation of the seroprevalence rate, and then, the output is compared to the real seroprevalence using SSR and MARE as metrics.

a mathematical expression that can provide us with information on the relationship of the variables with the seroprevalence rate. Thus, these stacking ensemble approaches use multiple estimations to obtain a more accurate prediction, as they counteract the biases that the individual estimations may have.

Once we have the stacking ensemble models, we compare the output of the models with the seroprevalence ground truth to evaluate the accuracy of the constructed predictive ensemble model.

3.2. Data processing

3.2.1. Explanatory variables

Our main source of data has been the US COVID-19 Trends and Impact Survey (CTIS). This project, operated by the Delphi Group at Carnegie Mellon University in collaboration with Facebook, has continuously operated surveys between the 6th of April 2020 and the 25th of June 2022, and has collected over 20 million responses (Delphi Group at Carnegie Mellon University, 2022; Salomon et al., 2021). Every day for the duration of the project, a random sample of Facebook users were invited to complete a questionnaire about the COVID-19 pandemic: symptoms, COVID testing, social distancing, vaccination, mental health and economic security.

In this work, we have not used directly the raw data obtained by the CTIS. Instead, we have used daily prevalence estimates obtained from the responses to the CTIS using various machine learning or statistical methods, as described in Rufino, Baquero et al. (2023), Rufino et al. (2024a) and Rufino, Ramirez et al. (2023). The resultant dataset has four prevalence estimates per record, and a total of 4 1369 records. Each state (and the District of Columbia) has 811 or 812 entries, one per surveyed day.

Moreover, in addition to the estimates obtained from the CTIS, we have also chosen four other input variables for our models, from three different sources: official daily reported new COVID cases, wastewater SARS-CoV-2 concentration (Srivastava, 2022), previous seroprevalence measurement, and normalised time since the previous seroprevalence measurement. The dataset of the official reported cases has 57 000 daily records for all USA states and the District of Columbia, each region having 950 records, between the 22nd of January 2020 and the 28th of August 2022. The wastewater concentration from Srivastava (2022) has 6273 entries, and 123 entry per region, from the 25th of January 2020 to the 28th of May 2022.

In summary, our models consist of eight explanatory variables, of which three are predicted by machine learning or statistical models. Those eight variables are the following:

- **COVID-like illness** (*CLI*). Daily rates for reported COVID compatible symptoms (CLI is defined as reporting a fever of at least 37.8 °C, along with cough, shortness of breath, or difficulty breathing), from the CTIS (National Notifiable Diseases Surveillance System (NNDSS), 2020; World Health Organization, 2020).
- **Random Forest** (*RF*). Daily estimated prevalence rate via an RF model from CTIS data.
- **Extreme Gradient-Boosting** (*XGB*). Daily estimated prevalence rate via an XGB model from CTIS data.
- **Generalised Linear Model** (*GLM*). Daily estimated prevalence rate via a Generalised Linear Model from CTIS data.
- **New reported cases** (*NRC*). Official total number of daily reported SARS-CoV-2 test positives. The data has been divided by its maximum value so that the scale lines up with the other variables.
- **Wastewater cases** (*WWC*). Daily estimated total active COVID-19 cases via wastewater virus concentrations. The data is divided by its maximum value so that the scale lines up with the other variables.
- **Time between rounds** (*TBR*). Number of days of the time interval we are aggregating (days from the end of the reference value's round). The number is divided by the maximum value of the variable so that the scale lines up with the other variables.
- **Reference value** (*REF*). The official rate of seropositivity in the round from which we are aggregating the daily data (see Section 3.2.3).

3.2.2. Ground truth

We have chosen to work with seroprevalence data from the USA because data is available for each one of its states. The US CDC has collected extensive data with respect to SARS-CoV-2 seroprevalence in their Nationwide Commercial Laboratory Seroprevalence Survey, which can be found in Centers for Disease Control and Prevention (2023). For that survey, the CDC conducted 30 rounds of seroprevalence testing among the population of most states within the USA, between July 2020 and February 2022, for a dataset of 1535 records in total.

Unfortunately, there are 13 states with less than 30 rounds: Arizona, Indiana, Maryland, Montana, Nevada, New Hampshire, New Jersey, Utah and Virginia have 29 rounds; Hawaii has 27 rounds; Wyoming has 26 rounds; South Dakota has 21 rounds; and North Dakota has 4 rounds. Note that the 10 most populous states all have had 30 rounds conducted in them. In order to identify the states, we have used the abbreviations defined by the American National Standards Institute (ANSI), as specified on Table 1.

It is important to highlight that all the input data presented in sections 3.2.1 and 3.2.2 were pre-processed in the works (Rufino,

Table 1
Table showing the ANSI state codes. Note that DC is not a state, but a federal district.

Code	State or territory
AL	Alabama
AK	Alaska
AZ	Arizona
AR	Arkansas
CA	California
CO	Colorado
CT	Connecticut
DE	Delaware
DC	District of Columbia
FL	Florida
GA	Georgia
HI	Hawaii
ID	Idaho
IL	Illinois
IN	Indiana
IA	Iowa
KS	Kansas
KY	Kentucky
LA	Louisiana
ME	Maine
MD	Maryland
MA	Massachusetts
MI	Michigan
MN	Minnesota
MS	Mississippi
MO	Missouri
MT	Montana
NE	Nebraska
NV	Nevada
NH	New hampshire
NJ	New jersey
NM	New mexico
NY	New york
NC	North carolina
ND	North dakota
OH	Ohio
OK	Oklahoma
OR	Oregon
PA	Pennsylvania
RI	Rhode island
SC	South carolina
SD	South dakota
TN	Tennessee
TX	Texas
UT	Utah
VT	Vermont
VA	Virginia
WA	Washington
WV	West virginia
WI	Wisconsin
WY	Wyoming

Baquero et al., 2023; Rufino et al., 2024a, 2024b), just like the selection of characteristics considered in this work was that carried out in these works.

3.2.3. Data aggregation

The data can be classified into two groups based on its frequency: sporadic and daily data. The ground truth has one measurement per seropositivity survey round for a total of at most 30 data points per state (sporadic data), while every explanatory variable we are going to use has daily values (daily data). Note that each seropositivity survey round spans several days of data collection. Therefore, we need to establish some criteria for how we are going to unify these two types of data. We have to choose how to aggregate daily data so that it aligns with the sporadic ground truth.

We have defined two different approaches to this aggregation problem, namely, "cumulative" and "non-cumulative" aggregation, both of which add the daily values of the explanatory variables. A graphical representation of how the daily data is aggregated with each approach can be seen in Fig. 2. The cumulative aggregation approach (Fig. 2(a)) adds up the daily data into Aggregate Round n, for the nth survey round, starting at the end-date of the first round, up to the end of the current nth round, so that the data aggregated for each round is a subset of the data aggregated for the next round. On the other hand, the non-cumulative approach (Fig. 2(b)) adds up the daily data into Aggregate Round n from the end-date of the round $n-1$ up to the end-date of round n, so that the aggregates of each round are disjoint.

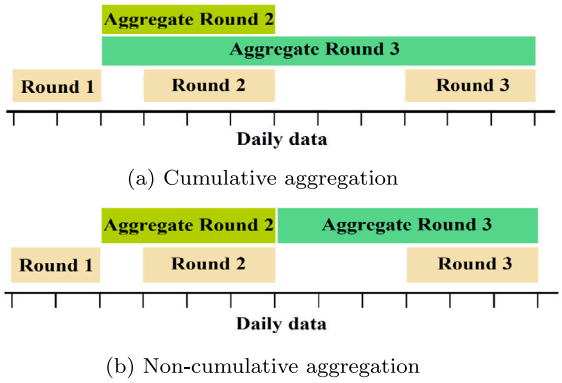


Fig. 2. Diagrams showing how daily data points are aggregated to transform daily data into sporadic data.

3.3. Performance metrics

Therefore, the CDC seroprevalence measurements are the ground truth of this problem, and our models' output will be predictions of these values. When building the prediction models, we are going to try to minimise the SSR, which reduces the variance of the resultant residuals. Its formula is shown in (1), where y_i is the real rate for the ith observation and $\hat{y}_i$ is its predicted value.

$$\mathrm{SSR}(\hat{\mathbf{y}}) = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2. \tag{1}$$

However, as the SSR is a relatively abstract measurement of error, when evaluating the accuracy of the models by comparing it to the ground truth, we also use the MARE alongside the SSR. The MARE represents the relative deviation from the observed data on average. It is more explicit and easily interpretable than the SSR. The formula of the MARE for a dataset of n observations is presented in Eq. (2), where y_i is the real rate for the ith observation and $\hat{y}_i$ is its predicted value.

$$\mathrm{MARE}(\hat{\mathbf{y}}) = \frac{1}{n} \sum_{i=1}^{n} \frac{|y_i - \hat{y}_i|}{y_i}. \tag{2}$$

3.4. Artificial intelligence and statistics techniques

3.4.1. Linear regression

Linear Regression (LR) models are one of the simplest and most thoroughly studied predictive models in Statistics. These models describe the relationship between one or various independent variables and another scalar variable, a relationship that is assumed to be linear. The scalar variable is usually referred to as the response variable, and the independent variables as explanatory variables. We are working with multiple explanatory variables, so the linear model we are going to build is a multiple LR model.

Let $\{x_i^1, \ldots, x_i^p, y_i\}_{i=1}^n$ be a set of $n \in \mathbb{N}$ observations, of which $\{x_i^j\}_{j=1}^p$ is the set of $p \in \mathbb{N}$ explanatory variables, and y_i is the response variable.

Then, an LR model defines the relationship between the response and explanatory variables using the following characterisation in Eq. (3)

$$y_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_i^j + \epsilon_i\,, \quad \forall i \in \{1, \dots, n\} \tag{3}$$

where ϵ_i is the error variable (unpredictable noise).

That LR model is usually denoted using matrix notation as in Eq. (4)

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \tag{4}$$

where $\mathbf{y}^T = (y_1 \cdots y_n)$ is an n-dimensional vector containing all observations of the response variable, $\boldsymbol{\beta}^T = (\beta_0 \cdots \beta_p)$ is the $p+1$-dimensional vector of weights or regression parameters, $\boldsymbol{\epsilon}^T = (\epsilon_1 \cdots \epsilon_n)$ is the error term of the model, and $\mathbf{X}$ is a $n \times (p+1)$-dimensional matrix containing the observations of all explanatory variables, plus a column of ones for the intercept of the model (β_0), as shown in Eq. (5).

$$\mathbf{X} = \begin{pmatrix} 1 & x_1^1 & \cdots & x_1^p \\ 1 & x_2^1 & \cdots & x_2^p \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n^1 & \cdots & x_n^p \end{pmatrix} \tag{5}$$

The aim of LR is to estimate the weights $\boldsymbol{\beta}$ so that the resulting predictive model fits the observed data. When we search for the best LR model we are going to minimise its SSR. The estimation method for LR that is used for that task is called least-squares regression, as seen in Eq. (6), where $\hat{\boldsymbol{\beta}}$ are the estimated weights.

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_i^j \right)^2. \tag{6}$$

And we can rewrite the problem in its matrix form, so that the minimisation problem can be reduced to Eq. (7).

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \|\epsilon\|^2 = \arg\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2. \tag{7}$$

Using its matrix form, we can expand the SSR formula and find its global optimum, which is its minimum because the function is convex. That way, we can obtain the simple matrix formula to compute the least-squares parameters shown in Eq. (8).

$$\hat{\boldsymbol{\beta}} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y}. \tag{8}$$

3.4.2. Beta regression

When working with rates or percentages, it is known that the values must be in the interval $[0, 1]$. This can be a problem for LR models because those models have no restriction that prevents the estimated response variable to fall outside of said interval. That is why, when the response variable is a rate, Beta Regression (BR) models are usually considered to be a better pick.

BR assumes the response variable to follow some beta distribution $B(\mu, \phi)$, where μ is the mean and ϕ the precision. We use a link function g to map from the bounded space $[0, 1]$ to the real numbers $\mathbb{R}$. The link function we are using is the logit function in Eq. (9).

$$g(x) = \text{logit}(x) = \ln\left(\frac{x}{1-x}\right). \tag{9}$$

Once in $\mathbb{R}$, we perform a regression assuming the data follows a beta distribution by maximising the likelihood. Then, we map the data back to $[0, 1]$ using the g^{-1} function. The inverse of the logit function is the expit function (10).

$$g^{-1}(x) = \text{expit}(x) = \frac{1}{1+e^{-x}} \tag{10}$$

Let $\{x_i^1, \dots, x_i^p, y_i\}_{i=1}^n$ be a set of $n \in \mathbb{N}$ observations, of which $\{x_i^j\}_{j=1}^p$ is the set of $p \in \mathbb{N}$ explanatory variables, and y_i is the response variable. Then, a BR model assumes that the target variable y_i is represented by the mean of a beta distribution μ, and models the relationship between the response and explanatory variables using the characterisation (11).

$$g(y_i) = \text{logit}(y_i) = \sum_{j=1}^{p} x_i^j \beta_j, \tag{11}$$

where β_j are the regression parameters. To obtain the parameters β_j we maximise the likelihood of the data under this model, which is defined as L in Eq. (12).

$$L = \prod_{i=1}^{n} f(y_i; \mu_i, \phi), \tag{12}$$

where f is the probability density function of the beta distribution, defined as (13), where Γ is the Gamma function.

$$f(y_i; \mu_i, \phi) = \frac{\Gamma(\phi)}{\Gamma(\mu_i\phi)\Gamma((1-\mu_i)\phi)} y_i^{\mu_i\phi-1}(1-y_i)^{(1-\mu_i)\phi-1} \tag{13}$$

When finding the optimal β_i parameters, it is easier to maximise the logarithm of the likelihood (called log-likelihood) rather than the likelihood itself, so we will work with the former: $l = \log(L)$.

3.4.3. Genetic programming

Genetic Programming (GP) is a method inspired by natural genetic processes that tries to find the best solution to a problem by evolving a set of equations. In our specific case, we are working with a population formed by mathematical expressions, i.e., equations that combine our explanatory variables. Each combination of the explanatory variables will be called a model in the following. The aim is to minimise the error between the values provided by these equations considering the available dataset and the ground truth. Said error in our case is the SSR.

GP uses operations based on natural genetic evolution to evolve and update a set of given equations (prediction models) so that they get better over time. GP works with tree structures to define the equations (individuals). This tree structure allows the algorithm to change and swap parts of an equation by manipulating nodes or subtrees in a given tree.

For our work, we use the following operators to define each individual (tree structure): addition, subtraction, multiplication, division, negative sign, exponential, and natural logarithm. We have also added a set of constants to the explanatory variables to make easier to get constant terms and factors in the equations. The set of chosen constants is the following set of powers of ten: $\{10^{-2}, 10^{-1}, 10^0, 10^1, 10^2\}$.

The GP algorithm has three phases: initialisation, selection, and reproduction. The algorithm starts by generating a random initial population of trees of a predetermined size using the available operators and variables. The randomness of the initial population allows the algorithm to start with a wide range of possibilities to cover enough of the search space. The search space of GP is the set of all possible combinations of operators and variables at its disposal.

Once the population is initialised, the algorithm evaluates the individuals to determine their quality, and uses a selection method to pick several individuals. In our case, the evaluation criteria for selection is SSR. Therefore, our algorithm uses SSR to select a population subset, using tournament selection of size three, which consist on randomly taking three individuals to then pick the best individual among those three, and repeating until the desired number of individuals are selected. Then, the algorithm manipulates the selected individuals ("parents") to create a new generation ("children"). For the reproduction of the parents, our algorithm uses three operations:

- Crossover, which picks pairs of parents and uses one-point crossover to generate one or two children.
- Mutation, which picks a single child and uses subtree replacement mutation to randomly change it.
- Replication, when the child is just a copy of its parent.

Crossover is usually applied before mutation. In our algorithm, both operations have assigned probabilities and are applied to the parents based on those probabilities: p_c, p_m.

After the children have been created, a new selection takes place to form a new generation of the same size. In our case, this selection picks the best individuals among a set formed by the previous generation and the newly created children. With the new generation, the algorithm repeats the process of selection and reproduction, until the stopping criteria is met. As stopping criteria, we have set a maximum number of generations, but if the fitness (SSR) of the best individual in each generation does not improve beyond a specified threshold δ for a total of m_s generations, the algorithm is stopped. δ and m_s are predefined hyper-parameters. Another hyper-parameter of the algorithm is the maximum depth of the GP-based models. This hyper-parameter prevents the model's complexity to grow too much. We do not need an excessive complexity to obtain good prediction models, as we see below.

In summary, the pseudo-code of the constructed GP algorithm for our problem is presented below on Algorithm 1. The variables introduced to the algorithm are as follows:

- P is the initial population, a set of models.
- $p_c \in [0, 1]$ is the probability of crossover.
- $p_m \in [0, 1]$ is the probability of mutation.
- $\delta \in \mathbb{R}^+$ and $m_s \in \mathbb{N}$ are the previously mentioned hyper-parameters for the stopping criteria.
- $g_{max} \in \mathbb{N}$ is the maximum number generations of the algorithm.
- $d_{max} \in \mathbb{N}$ is the maximum allowed depth of the model.

Algorithm 1: The GP algorithm. (Individuals cannot be deeper than d_{max})

Require: P, $p_c \in [0,1]$, $p_m \in [0,1]$, $\delta \in \mathbb{R}^+$, $m_s \in \mathbb{N}$, $g_{max} \in \mathbb{N}$
  $best \leftarrow \text{argmin}_{f \in P}\{\text{Eval}(f)\}$
  $locked\_eval \leftarrow \text{Eval}(best)$ ▷ *Eval* returns the SSR of the individuals
  $m \leftarrow 0$
  $p_m^* \leftarrow \min\{1, 2p_m\}$
  for $g = 1, \ldots, g_{max}$ **do**
    $C \leftarrow \text{Select}(P, |P|)$ ▷ Select $|P|$ random individuals from P with repetition
    if $g = 100$ **then**
      $p_m^* \leftarrow p_m$ ▷ After 100 generations p_m is restored
    end if
    $\text{Crossover}(C, p_c)$ ▷ Each pair of individuals in C is crossed with probability p_c
    $\text{Mutate}(C, p_m^*)$ ▷ Each individual in C is mutated with probability p_m^*
    $P \leftarrow \text{BestOf}(P \cup C, |P|)$ ▷ Picks the $|P|$ best individuals (smallest SSR) from $P \cup C$
    $best \leftarrow \text{argmin}_{f \in P}\{\text{Eval}(f)\}$
    if $locked\_eval - \text{Eval}(best) \leq \delta$ **then**
      $m \leftarrow m + 1$
      if $m = m_s$ **then**
        break for
      end if
    else
      $locked\_eval \leftarrow \text{Eval}(best)$
      $m \leftarrow 0$
    end if
  end for
  return $(best, g)$

After a hyper-parameter optimisation process, the final values for the hyper-parameters used were: a population size of $|P| = 300$, crossover and mutation probabilities of $p_c = 0.8$ and $p_m = 0.3$, the stopping parameters $\delta = 0.005$ and $m_s = 100$, a maximum number of generations of $g_{max} = 1000$, and maximum depths $d_{max} \in \{4, 6, 8, 10\}$.

We want to analyse the performance of GP-based models with this particular problem. The GP-based models are known to provide a more general framework than LR and BR. Therefore, one of the objectives of this work is to investigate the performance improvement of GP-based models with respect to the LR models.

The GP algorithm is stochastic: it has an element of randomness that can cause the results of each iteration to be different from each other. Therefore, one execution is not enough to see how good the GP algorithm is at finding accurate prediction models. For that reason, we test the algorithm by executing it 20 times for each combination of state, aggregation, and maximum depth. After those 20 executions, we average the SSR (and MARE) of all the resultant prediction models, in order to have a better approximation of the accuracy of our GP algorithm.

3.4.4. Neural networks

We decided to also use Neural Networks (NN) to estimate the seroprevalence rate, as a fourth modelling method along LR, BR and GP. NN are machine learning models, loosely based on the behaviour of biological neurons in animal brains. NN are made up of a set of connected nodes, called neurons, which can transmit signals (real numbers) between each other. When a neuron receives the signals of other neurons, it then returns an output by applying a function (activation function) to the linear combination of these inputs. The connections between neurons are called edges, and they usually have assigned weights that are adjusted along the training of the model. The neurons are usually organised in layers, and the signals travel through the different layers, from an input layer all the way to an output layer. The layers that are between the input and output layers are called hidden layers (Hardesty, 2017).

NN are iteratively trained with a training dataset. At the start, the weights of the edges are picked randomly, and then, for each iteration, they are updated based on the performance of the network on the training dataset. There are multiple methods to update the weights, but we have decided to use gradient descent, the most common way to optimise NN. Gradient descent consists on minimising an objective function (the SSR in our case) by updating the parameters of the function (the weights of the NN) in the opposite direction to the gradient of said objective function with respect to the parameters. When updating the parameters, the size of the steps taken is determined by the learning rate, a predetermined hyper-parameter (Ruder, 2016).

The NN we are working with have multiple hyper-parameters that have to be set beforehand. These parameters are:

- *Batch size*: The size of the sub-samples that are used to train the model. In each iteration, the model is trained with each of the batches.
- *Number of epochs*: The epochs are the iterations of the training loop for the NN. The number of iterations must be set.
- *Learning rate*: The size of the steps taken along the gradient when updating the weights of the NN if using gradient descent.

The values we have chosen for the hyper-parameters are different for each studied case (state-by-state estimations, nationwide models, forecasting), except for the stopping criteria of the gradient descent algorithm. We have applied a similar stopping criteria to the GP algorithm: the gradient descent will stop when the mean training loss (SSR) does not improve more than 0.0001 for 5000 epochs.

For the problem at hand, we tried building different types of NN. One of the simple and most widely used activation functions is the RELU function, which can be defined as (14).

$$\text{RELU}(x) = \begin{cases} x & x > 0 \\ 0 & x \leq 0 \end{cases} \tag{14}$$

We have used a NN with 7 hidden layers of 9 neurons each, with RELU as the activation function of all neurons, except for the output neuron, which has the identity function. We will call this the RELU NN (RNN). A graphical representation of a simplified version of the RNN, with two layers of three neurons each, can be seen in Fig. 3. Even though this

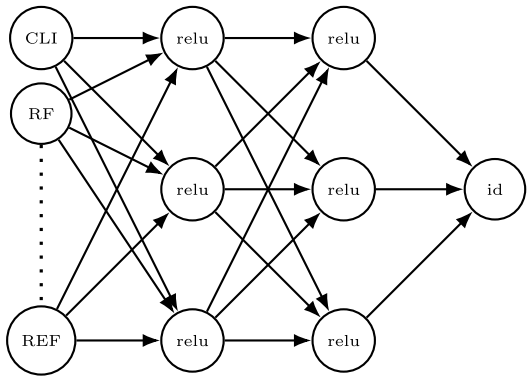


Fig. 3. A simplified version of the RNN, with two hidden RELU layers, with three neurons each. The names of the input neurons represent the explanatory variables, and the names of all other neurons are their activation functions.

is a fairly simple NN, the seven intermediate layers, with a total of 63 neurons, give the model a large number of degrees of freedom, resulting in a quite accurate prediction model, as we will see below.

NN are stochastic, like GP, and hence we execute the gradient descent algorithm 20 times for each combination of state and aggregation, and average the SSR (and MARE) of all the resultant NN models, in order to obtain the mean errors of the NN approach.

4. Model evaluation

We start by applying the previously mentioned approaches (LR, BR, GP, and NN) to the data we have at hand, to see how well they can approximate the observed rates of seroprevalence. Note that both LR and BR are deterministic methods, but GP and NN are stochastic methods, so when analysing and comparing the results of those two latter methods, we are going to check the mean and median errors of multiple executions, as well as the distribution of the resulting errors.

There can be two prediction models built with the data, depending on the observations used. On the one hand, we have separate data for all USA states (and the District of Columbia), so we build a statewide prediction model that only uses said state's data. On the other hand, we take data from multiple US states to build a bigger prediction model that uses observations from various states. That model provides us with more accurate information on the pandemic, even if the state level estimations may get less accurate.

4.1. Results by state

First, we build a model for each state and aggregation method (cumulative and non-cumulative), which we will refer to as statewide models, using all four modelling methods.

4.1.1. Linear regression

When obtaining the regression parameters of the explanatory variables, we will find that very often some of the coefficients are not significant, i.e., if the model had zero as a coefficient for that variable, then the distribution of the estimations would be virtually the same. Therefore, when building our LR models, we are going to purge the non-significant variables, as long as removing them results in a lower SSR.

As an example, we are going to take cumulative aggregation and the biggest state in terms of population, California, and determine the best LR model for its data. If we include all the variables to the model, then we get the regression coefficients of Table 2, and an SSR of 0.00732731189382156. However, if we check the p-values of each regression coefficient, then there are multiple p-values over the usual alpha-level of 0.05. When a p-value is over that value, then the null hypothesis is not rejected (in this case, the null hypothesis is that a null coefficient would result in a notable difference in estimations), and the respective variable is non-significant.

Table 2
Regression parameters and p-values of said parameters for the cumulative CA linear model with all variables.

	CLI	RF	XGB	GLM	NRC	WWC	TBR	REF
Coeff.	−0.204	0.638	−0.380	−0.154	0.005	−0.011	0.320	0.835
p-value	0.122	0.001	0.037	0.123	0.806	0.161	0.703	0.008

Table 3
Regression parameters and p-values of said parameters for the cumulative CA linear model without the variable NRC.

	CLI	RF	XGB	GLM	WWC	TBR	REF
Coeff.	−0.180	0.642	−0.355	−0.178	−0.011	0.137	0.866
p-value	0.039	<0.001	0.015	<0.001	0.157	0.714	0.002

Table 4
Regression parameters and p-values of said parameters for the cumulative CA linear model without the variables NRC, TBR and WWC.

	CLI	RF	XGB	GLM	REF
Coeff.	−0.159	0.638	−0.371	−0.173	1.003
p-value	<0.001	<0.001	0.008	<0.001	<0.001

However, we cannot remove all non-significant variables because the removal of one may affect the significance of the others. Therefore, we are going to remove them one by one, checking the significance of the variables at each step. The first variable we remove is that with the highest p-value: NRC. The new model, without the NRC variable, has the coefficients and p-values of Table 3, and an SSR of 0.00732731189382156. So, with the removal of a non-significant variable we have obtained virtually the same SSR.

We repeat this procedure until all variables left are significant, and then take the set of variables that result in the lowest SSR to build the definitive statewide model. In the next step, we would remove TBR. After this procedure, we find that even though all combinations for CA with cumulative aggregation result in very similar SSR, the one with all the variables, shown in Table 2 is marginally better, so we take that model. We have replicated the procedure outlined for the cumulative CA model with all states and both aggregation methods, and we have saved the best model for each (see Table 4). We have then looked into the resultant SSR and MARE of each model, and the resulting values are displayed in Tables 5 and 6 respectively.

As highlighted in the table, some of the states have had less than 30 survey rounds conducted on them. There are nine states with 29 rounds, and another four have been surveyed on even fewer rounds, North Dakota (ND) being an outlier with just four rounds. Clearly, the extremely low number of data-points available for ND results in a very accurate LR model, because there is a lot of data to estimate just three points: the predictions do not deviate at all from the ground truth with both non-cumulative and cumulative aggregation.

On the other hand, if we compute the mean of MARE values (more interpretable than SSR for the analysis) of each group of states, we can see how their mean more or less stays in the range $[0.10, 0.25]$, as can be seen in Table 7. So, the LR models generally work well with the data, specially for the states with the most data available. However, the perfect predictions of the models in ND are not replicated in any other state, which reinforces the view that ND is an outlier due to the lack of data, so the models for that state must not be taken at face value.

The mean MARE of all states is relatively low at 0.153 for non-cumulative aggregation and 0.130 for cumulative aggregation, and if we dismissed ND as an outlier then the mean MARE would only increase by 0.003 for both aggregations, so the perfect predictions for the ND models do not overly distort the accuracy of the LR models. The states that have less than 29 rounds of surveys besides ND have

Table 5
Table with the SSR for each LR state estimation. Some states are highlighted by the number of survey rounds: those with 29 rounds in grey, those with between 27 and 21 rounds in pink, and ND with its 4 rounds in red.

State	By aggregation	
	Non-cum.	Cum.
AL	0.01142	0.00683
AK	0.01919	0.01673
AZ	0.01290	0.00664
AR	0.01298	0.00706
CA	0.01003	0.00683
CO	0.01291	0.01761
CT	0.00372	0.00346
DE	0.00879	0.00619
DC	0.00808	0.00406
FL	0.01046	0.00753
GA	0.01381	0.02561
HI	0.00462	0.00328
ID	0.02383	0.01073
IL	0.01902	0.00871
IN	0.01797	0.01130
IA	0.02018	0.02522
KS	0.01359	0.02608
KY	0.00564	0.00441
LA	0.00522	0.00664
ME	0.00284	0.00192
MD	0.02820	0.01411
MA	0.00733	0.00613
MI	0.00589	0.00242
MN	0.01086	0.02025
MS	0.01439	0.00486
MO	0.01174	0.00981
MT	0.01739	0.03085
NE	0.00800	0.02946
NV	0.00936	0.00434
NH	0.00408	0.00536
NJ	0.00976	0.00580
NM	0.01496	0.03249
NY	0.01850	0.02121
NC	0.00783	0.00876
ND	0.0	0.0
OH	0.00845	0.00902
OK	0.01204	0.03083
OR	0.00335	0.00842
PA	0.01055	0.00548
RI	0.00965	0.01308
SC	0.00684	0.00447
SD	0.04251	0.03419
TN	0.00928	0.00510
TX	0.03755	0.01956
UT	0.02252	0.01146
VT	0.00349	0.00252
VA	0.00553	0.00462
WA	0.00611	0.00441
WV	0.00536	0.00614
WI	0.01328	0.00766
WY	0.03555	0.02155

Table 6
Table with the MARE for each LR state estimation. Some states are highlighted by the number of survey rounds: those with 29 rounds in grey, those with between 27 and 21 rounds in pink, and ND with its 4 rounds in red.

State	By aggregation	
	Non-cum.	Cum.
AL	0.09753	0.06600
AK	0.44970	0.15167
AZ	0.12103	0.07093
AR	0.16116	0.12336
CA	0.12131	0.07883
CO	0.18267	0.20950
CT	0.14229	0.11444
DE	0.11827	0.09041
DC	0.08934	0.06188
FL	0.09791	0.07715
GA	0.10605	0.13506
HI	0.29858	0.22759
ID	0.17954	0.11771
IL	0.10986	0.07052
IN	0.17170	0.11586
IA	0.13385	0.13889
KS	0.16808	0.18424
KY	0.11432	0.14272
LA	0.07878	0.08793
ME	0.35681	0.25162
MD	0.13730	0.09537
MA	0.14759	0.16455
MI	0.08613	0.06643
MN	0.13212	0.18550
MS	0.11921	0.06040
MO	0.14210	0.10816
MT	0.20557	0.23445
NE	0.10078	0.14802
NV	0.07642	0.04836
NH	0.19181	0.19958
NJ	0.07117	0.05934
NM	0.16004	0.17442
NY	0.12704	0.11876
NC	0.12217	0.12125
ND	0.0	0.0
OH	0.13703	0.13971
OK	0.10654	0.21261
OR	0.15845	0.17095
PA	0.09449	0.08304
RI	0.17801	0.19413
SC	0.08183	0.05624
SD	0.38136	0.30147
TN	0.08563	0.05586
TX	0.15352	0.08973
UT	0.14575	0.08063
VT	0.46729	0.33242
VA	0.12723	0.09436
WA	0.21834	0.18763
WV	0.16107	0.17861
WI	0.10165	0.07748
WY	0.24293	0.12947

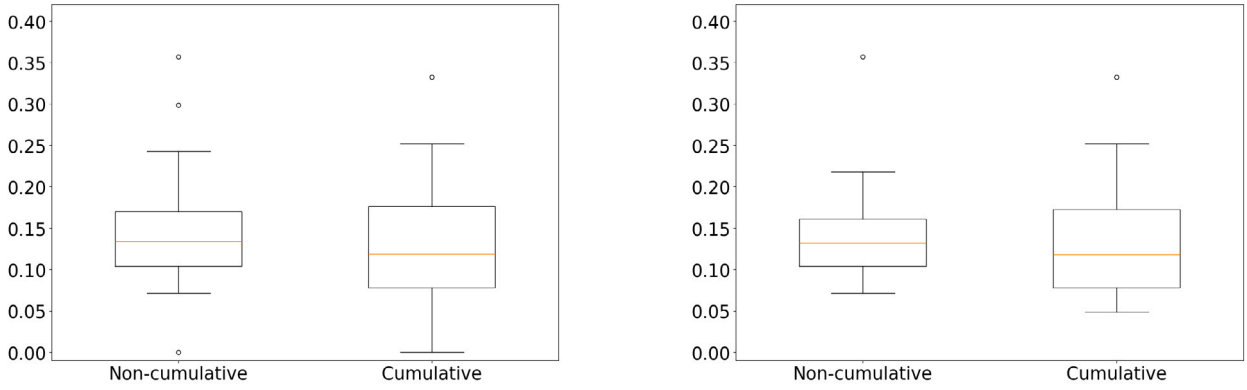
a higher than average MARE (Hawaii, South Dakota and Wyoming), with a mean of 0.251 and 0.200 among the three for non-cumulative and cumulative aggregation, respectively. This higher MARE could be the result of poorly executed survey rounds, but we do not have the information to properly conclude that.

In order to better represent the behaviour of the MARE for each of the aggregation methods, Fig. 4 provides box-plots of the error in two cases: with All states, and only with the states with 29 or 30 rounds. The exclusion of the four states with less than 29 rounds results in little change for both aggregation methods. ND's value is an outlier in non-cumulative aggregation, and only lowers the whisker of the cumulative box-plot by about 0.05. Comparing both plots one can also see that another one of the other three removed states is an outlier for non-cumulative aggregation too, and the variation on the median and quartiles is not big enough to worry about it.

Table 7
Arithmetic means of the MARE of different combinations of states depending on their number of rounds, using LR.

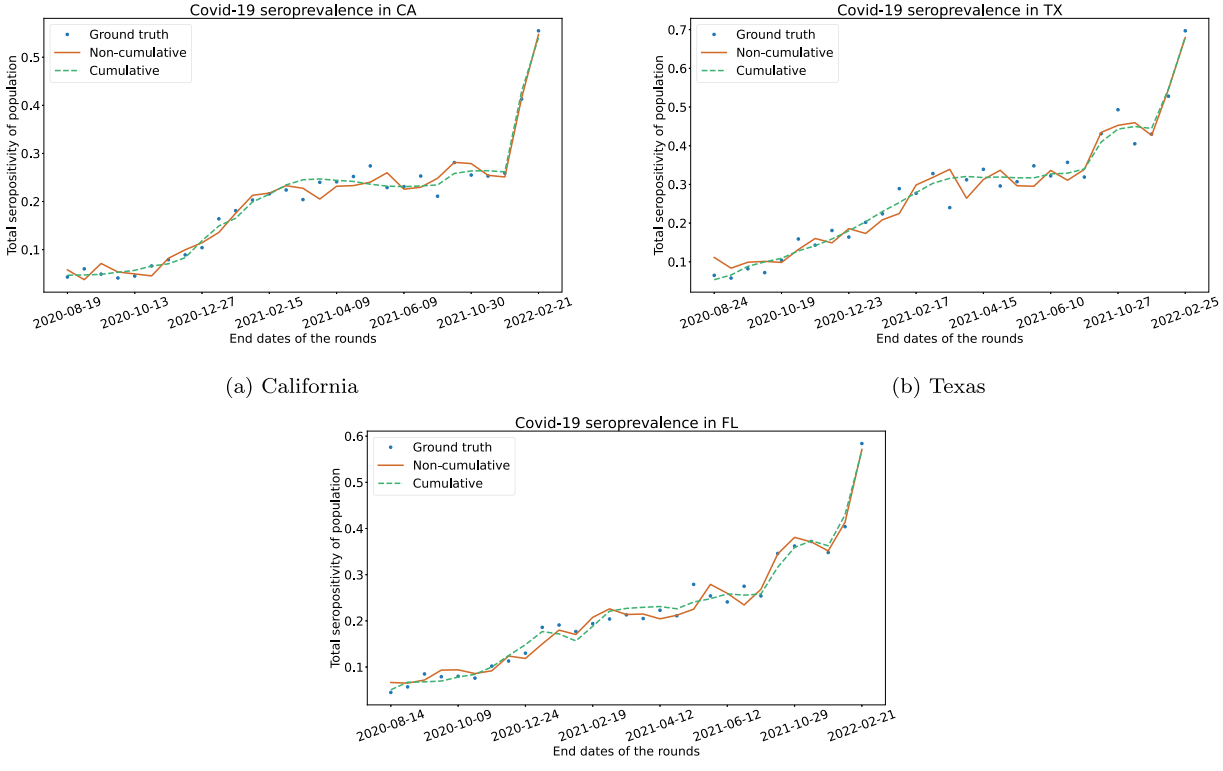
	MARE		SSR	
	Non-cum.	Cum.	Non-cum.	Cum.
Mean of 30 round states	0.15233	0.13231	0.01124	0.01152
Mean of 29 round states	0.13866	0.11099	0.01419	0.0105
Mean of HI, SD and WY	0.25094	0.20032	0.02756	0.01967
ND	0.0	0.0	0.0	0.0
Mean of all states	0.15273	0.12995	0.0125	0.01159
Mean of all states but ND	0.15579	0.13255	0.01275	0.01182
Mean of 29–30 round states	0.14971	0.12823	0.01181	0.01132

In order to show some examples of the estimation capacity of our LR models, we have taken the three most populous states of the USA and plotted the ground truth we have for them (the CDC seroprevalence



(a) All states (b) Without HI, ND, SD and WY

Fig. 4. MARE box-plots for all states, with and without the states with less than 29 rounds (HI, ND, SD and WY), using LR.



(a) California (b) Texas

(c) Florida

Fig. 5. LR estimations of seropositivity rates for the three most populous states.

surveys) along with the LR estimations for both aggregation methods: cumulative and non-cumulative. The result can be seen in Fig. 5.

The plots clearly show that the constructed LR models closely fit the observed data, and follow the rate of seropositivity throughout time quite well. However, a difference between the estimations of each aggregation method can be perceived at first glance: the cumulative approach results in a smoother estimated line, as the noise of the seropositivity rate is damped by using data from a longer period of time per round; while the non-cumulative approach seems to result on a delayed estimated rate with respect to the ground truth, at least up to the abrupt rise at the end, which both approaches predict very well.

4.1.2. Beta regression

Overall, the results for LR were accurate, but when working with rates and percentages, it is not the best approach to choose. BR is a regression model that is usually more appropriate for rates, so we are going to check how it performs. When building the BR models, we proceed just as we did with LR, removing the non-significant variables as long as the SSR of the model is reduced. The resultant SSR and MARE values of each model are displayed in Tables 8 and 9, respectively.

Then, we can check the mean of MARE values, just like we did with LR, and see how the means of both regression models differ. The results are shown in Table 10. The non-cumulative results are less accurate with BR, specially for 29 round states, but BR cumulative MAREs are fairly similar to LR, and sometimes even better. So we could say that depending on the aggregation approach LR can be notably better than BR (non-cumulative), or BR marginally better than LR. Overall the results with BR are also good. The mean MARE of all states is 0.03 higher for non-cumulative BR, at 0.153, and virtually the same for cumulative BR, at 0.131. The states that have less than 29 rounds of

Table 8
Table with the SSR for each BR state estimation. Some states are highlighted by the number of survey rounds: those with 29 rounds in grey, those with between 27 and 21 rounds in pink, and ND with its 4 rounds in red.

State	By aggregation	
	Non-cum.	Cum.
AL	0.01437	0.00634
AK	0.01526	0.02073
AZ	0.02022	0.008
AR	0.01617	0.00459
CA	0.00969	0.00733
CO	0.01702	0.01605
CT	0.00787	0.00458
DE	0.01077	0.00561
DC	0.01204	0.00216
FL	0.01733	0.00713
GA	0.01136	0.02207
HI	0.00412	0.00372
ID	0.02441	0.01419
IL	0.03351	0.01791
IN	0.03078	0.01103
IA	0.01596	0.02038
KS	0.01317	0.02467
KY	0.01632	0.00424
LA	0.00389	0.00543
ME	0.00666	0.00114
MD	0.03608	0.01443
MA	0.00545	0.00669
MI	0.01155	0.0045
MN	0.00709	0.01604
MS	0.01335	0.00587
MO	0.0219	0.01105
MT	0.02442	0.03096
NE	0.01528	0.02666
NV	0.01659	0.00413
NH	0.00683	0.00557
NJ	0.01088	0.00495
NM	0.02026	0.03559
NY	0.01495	0.02576
NC	0.01262	0.0075
ND	0.0	0.0
OH	0.0181	0.02409
OK	0.01453	0.02991
OR	0.00421	0.00813
PA	0.01238	0.00561
RI	0.0116	0.01126
SC	0.00931	0.00622
SD	0.0581	0.0494
TN	0.01479	0.0053
TX	0.05215	0.02191
UT	0.0336	0.01383
VT	0.00279	0.00202
VA	0.00993	0.00494
WA	0.00417	0.00287
WV	0.01855	0.00345
WI	0.02369	0.01341
WY	0.06077	0.02656

Table 9
Table with the MARE for each BR state estimation. Some states are highlighted by the number of survey rounds: those with 29 rounds in grey, those with between 27 and 21 rounds in pink, and ND with its 4 rounds in red.

State	By aggregation	
	Non-cum.	Cum.
AL	0.10034	0.06354
AK	0.60248	0.43711
AZ	0.13438	0.08283
AR	0.1589	0.08439
CA	0.12426	0.07924
CO	0.22012	0.16465
CT	0.20386	0.10883
DE	0.1213	0.08879
DC	0.10827	0.05474
FL	0.11556	0.07639
GA	0.10587	0.12634
HI	0.26002	0.24945
ID	0.21814	0.12406
IL	0.1455	0.1135
IN	0.22337	0.09965
IA	0.11859	0.11954
KS	0.18792	0.16101
KY	0.19709	0.10096
LA	0.07088	0.07612
ME	0.51159	0.17916
MD	0.15336	0.09807
MA	0.15117	0.1379
MI	0.11108	0.09022
MN	0.12272	0.15238
MS	0.1116	0.06764
MO	0.18244	0.10671
MT	0.42478	0.19123
NE	0.13375	0.14519
NV	0.09507	0.04247
NH	0.39216	0.22044
NJ	0.07345	0.0528
NM	0.19983	0.19419
NY	0.11261	0.12655
NC	0.1161	0.11312
ND	0.0	0.0
OH	0.16372	0.2299
OK	0.14951	0.1531
OR	0.2011	0.17351
PA	0.10268	0.08266
RI	0.22131	0.16445
SC	0.08477	0.06129
SD	0.2523	0.27991
TN	0.11202	0.05877
TX	0.1739	0.09074
UT	0.17053	0.08836
VT	0.5569	0.31411
VA	0.15223	0.09407
WA	0.17227	0.14096
WV	0.30915	0.13082
WI	0.13116	0.10077
WY	0.37047	0.16419

surveys (except ND) have a higher than average MARE, just like we saw with LR, with a mean of 0.294 and 0.231 for non-cumulative and cumulative aggregation, respectively.

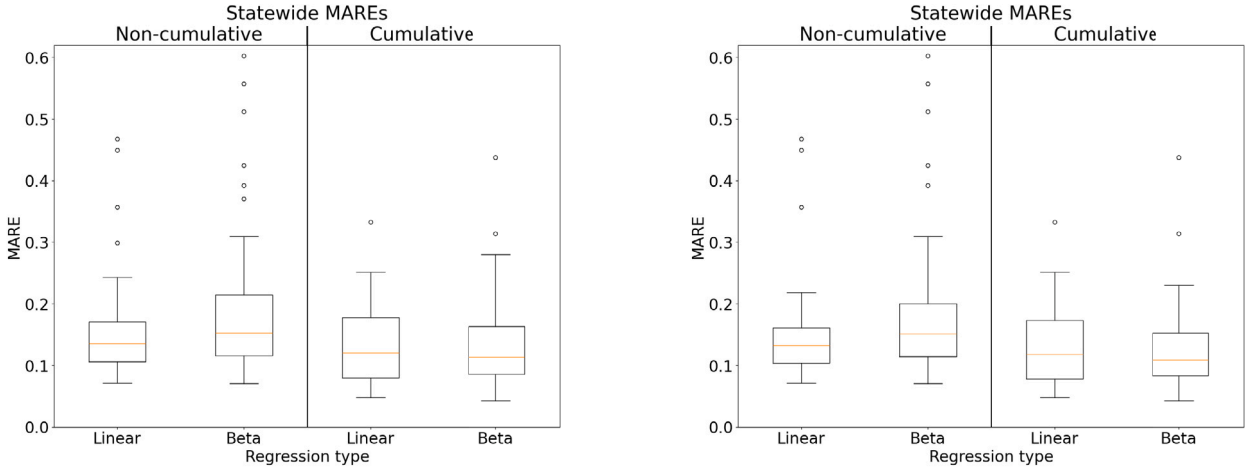
In order to compare the MAREs of LR and BR more easily, we represent both methods' results next to each other in the box-plots of Fig. 6. We observe that the removal of states with less than 29 rounds reduces the upper quartiles and whiskers of the box-plots, and without said states, the BR MARE values are very similar but slightly lower than the LR MARE values when cumulative aggregation is used, which coincides with our observations of the mean MAREs. The BR MAREs with non-cumulative aggregation, on the other hand, are quite higher than with LR. Lastly, we show some examples of the BR estimations for the three most populous states, just like with LR. The result can be seen on Fig. 7. It can be seen that BR also obtains good fits for the data, cumulative aggregation being smoother.

Table 10
Arithmetic means of the MARE of different combinations of states depending on their number of rounds, using BR.

	MARE by aggregation	
	Non-cum.	Cum.
Mean of 30 round states	0.18238	0.13140
Mean of 29 round states	0.20215	0.10777
Mean of HI, SD and WY	0.29426	0.23119
ND	0.0	0.0
Mean of all states	0.18887	0.13053
Mean of all states but ND	0.19265	0.13314
Mean of 29–30 round states	0.18617	0.12688

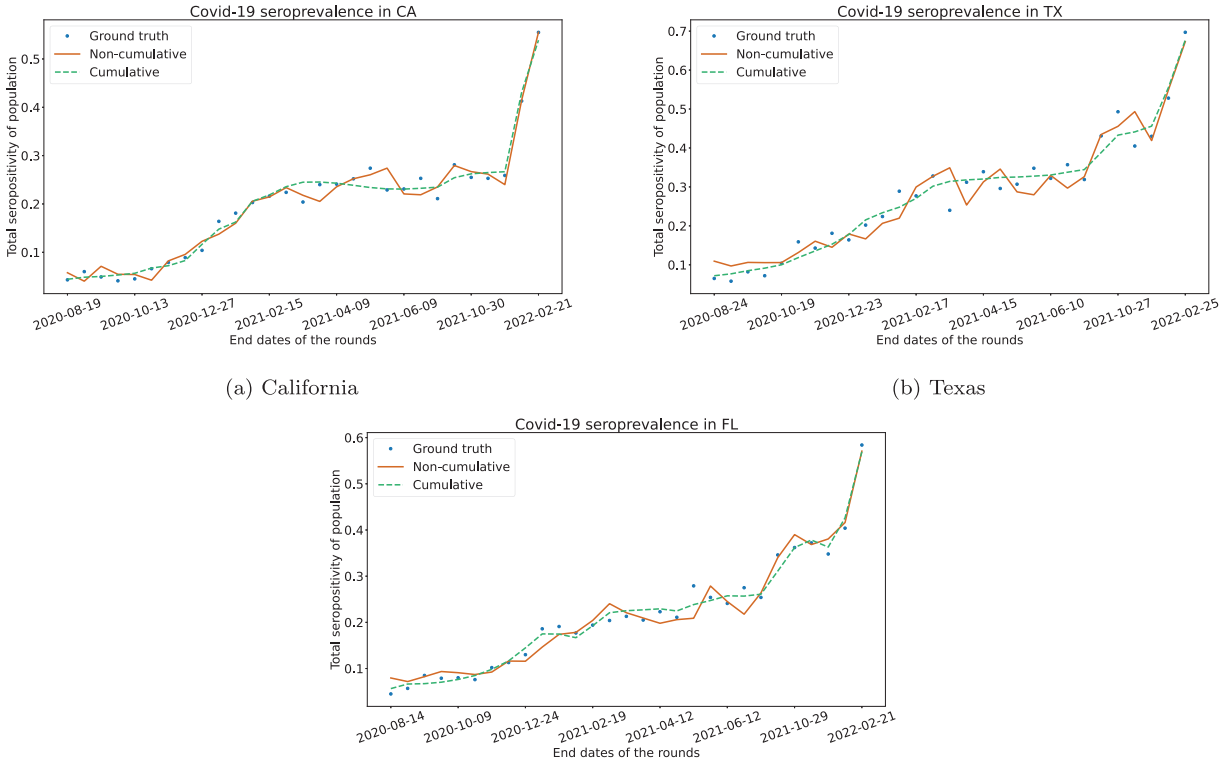
4.1.3. Genetic programming

We move on to GP modelling. Remember that a specially important hyper-parameter has to be set, the maximum depth. We use four



(a) All states but ND (b) Without HI, ND, SD and WY

Fig. 6. MARE box-plots for all states except ND, with and without the states with less than 29 rounds (HI, SD and WY).



(a) California (b) Texas

(c) Florida

Fig. 7. BR estimations of seropositivity rates for the three most populous states.

maximum depths: 4, 6, 8 and 10. We observe that a tree with less than 4 levels is too simple to represent the observed data accurately, and that 10 levels are enough to get a relatively low MARE (higher values may result in over-fitting the data). In Table 11, we display the mean MARE of three example states per maximum depth for both aggregation methods. These three are the most populous states and they are representative of most statewide models. On the other hand, the box plots of the MARE per maximum depth for each state aggregation are also displayed in Fig. 8. As we can see in the box plots, the larger depth they are allowed to have, the more precise GP-based models get (Texas, Florida, and cumulative California), even though there are a few cases where more depth beyond a certain point is shown to produce higher MARE (non-cumulative California).

When observing the behaviour of the MARE, the cumulative approach results in lower MARE than the non-cumulative on average for the three examples, as we saw in LR and BR. If we compare the GP-based models' MARE to that of the LR and BR models, we see that for all three examples (and all states studied beyond these examples), GP achieves a lower MARE than both LR and BR, specially with non-cumulative aggregation, where even the 4-level model is below the linear MARE for all executions of the three examples (this is replicated in most but not all states). The cumulative aggregation model usually

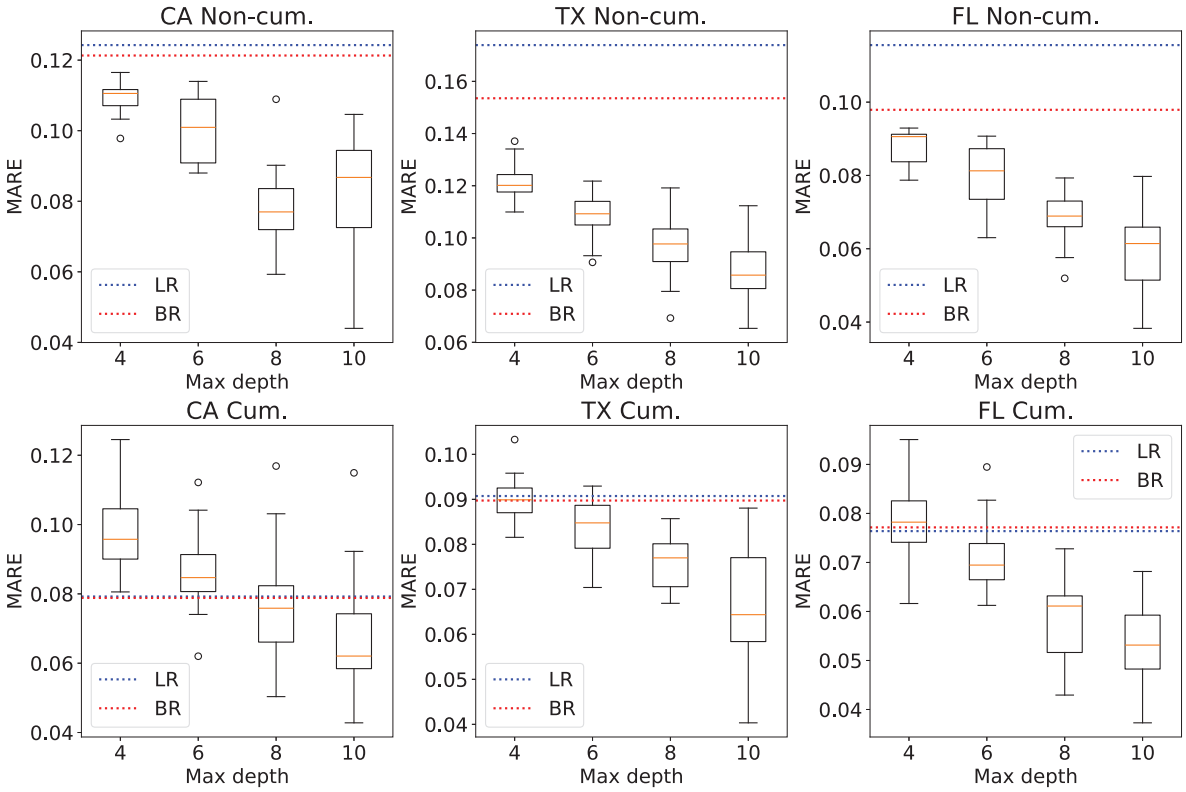


Fig. 8. Box-plots of the MARE of 20 executions of the GP algorithm with different maximum depths for California (CA), Texas (TX) and Florida (FL). LR an BR MARE are shown as horizontal lines for reference.

Table 11
Table with the mean MARE of 20 executions of the GP algorithm for different maximum tree depths.

		By maximum depth			
		4	6	8	10
California	Non-cum.	0.109	0.100	0.078	0.083
	Cum.	0.097	0.086	0.077	0.067
Texas	Non-cum.	0.122	0.108	0.097	0.087
	Cum.	0.090	0.083	0.076	0.067
Florida	Non-cum.	0.088	0.080	0.069	0.059
	Cum.	0.079	0.071	0.058	0.053

needs more depth than the non-cumulative to achieve lower errors than its linear and beta counterparts, and in California, there are some executions with depth 10 where the GP-based model was worse, but it is better than LR and BR on average.

With these statewide GP-based models, we are achieving very low mean MAREs, below a 10% deviation from the observed data on average. This low MARE looks like the models are working extremely well, and could lead us to think that allowing even more depth would be desirable, as we may be able to reduce the MARE even more. However, there are two main reasons why that may not be a good idea. On the one hand, the more levels the model has, the more complex and confusing it becomes. Hence, if we want to understand the internal workings of the model, more complex trees could be a problem. Besides, a small reduction of the MARE may not be worth the great growth in complexity. On the other hand, when building a prediction model, reducing the error of the training data to a minimum (the observed data per state in our case) runs the risk of over-fitting the model to said training data, including the noise of the observations into the model, which gravely reduces the usefulness of the model outside the small dataset used. Therefore, we decided that the small MARE obtained with maximum depths of up to 10 levels is a good enough result and that it is unnecessary to try to lower it even more by raising the maximum depth further.

In order to see a couple of examples from all the models generated, we are going to show some estimations for both the minimum depth allowed (4) and the maximum depth (10). We have picked the models that are closest to the mean MARE of all 20 executions as examples, for the most populous state (CA). The model of depth 4 that the algorithm returned for CA with non-cumulative aggregation, is Eq. (15).

$$\left(REF + 0.01RF + \frac{0.01}{TBR}(RF - 0.1)\right)e^{REF(WWC-TBR)} \tag{15}$$

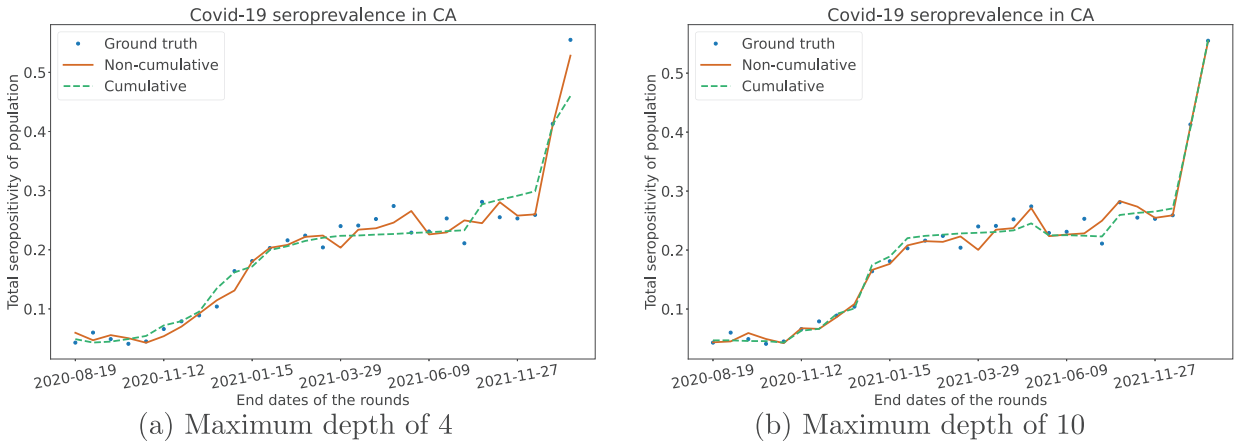
And with cumulative aggregation, the result is Eq. (16).

$$(NRC - TBR - \ln(CLI))\frac{CLI + 0.1}{100RF} + \frac{RF}{(REF + 10)e^{TBR}} \tag{16}$$

It becomes evident at first glance that these models are more complex than a simple LR model, even if these are the GP-based models with the smallest depth. They are also clearly non-linear. These models have the SSR, MARE and R^2 values shown in Table 12. The resultant estimated seropositivity rates can be seen in Fig. 9a. We have done the same with the maximum depths of ten. The MARE and R^2 values can be seen on Table 12, and the estimations on Fig. 9b.

4.1.4. Neural networks

Lastly, we have also used NN to model the states' seropositivity. When working state by state we have built an NN with 7 hidden layers and 9 neurons per layer, for a total of 63 neurons, all of them, with RELU as their activation function; and we picked a learning rate of 0.05 and a batch size of 5. We have chosen the same three example states as in GP (California, Texas and Florida) and applied the gradient descent algorithm to obtain a NN that fits their seroprevalence rates. The algorithm has a stochastic element (the initialisation), so we have executed it 20 times, like the GP algorithm. The resulting mean SSR and MARE values are shown in Table 13. We also plotted the MARE values alongside the MAREs of the other models in Fig. 10.



(a) Maximum depth of 4 (b) Maximum depth of 10

Fig. 9. GP estimations of seropositivity rates for California, with a maximum depth of 4 and 10.

Table 12
Table with the SSR, MARE and R^2 of the California GP-based models with a maximum tree depth of 4 and 10.

		By depth	
		4	10
SSR	Non-cum.	0.0592	0.0313
	Cum.	0.0713	0.0246
MARE	Non-cum.	0.1053	0.0799
	Cum.	0.0980	0.0637
R^2	Non-cum.	0.9708	0.9854
	Cum.	0.9483	0.9872

Table 13
Table with the mean SSR and MARE of 20 executions of NN gradient descent.

		SSR	MARE
California	Non-cum.	0.00513	0.07217
	Cum.	0.00390	0.06008
Texas	Non-cum.	0.00706	0.04612
	Cum.	0.01547	0.07083
Florida	Non-cum.	0.00860	0.07347
	Cum.	0.00440	0.06009

As we can see, the constructed NN obtains fairly good results with the example states, with MARE values below 0.1. I.e., the NN deviates from the observed data less than 10% on average. The results of the NN are close in accuracy to the GP model with a maximum depth of 10, over-performing their MARE in some cases. In most cases, the MAREs and SSRs of the NN models present slight or no improvements with respect to their GP equivalents, but it has to be noted that there is barely any room for notable improvements, given that the MARE values of the GP10 models get as low as 0.04 in many cases. If we compare the MARE and SSR values of the NN to the LR and BR models there are almost no executions of gradient descent that obtain higher errors, showing that NN is a much better approach than said regressions.

4.2. Nationwide results

We have also used different modelling methods to obtain nationwide prediction models, aggregating all available data from all the USA. As previously mentioned, there are some states that have had less than 30 rounds of the CDC survey conducted on them, which may indicate that the accuracy of those measurements is lower. In order to see whether the accuracy of the nationwide models works better with some states, we have built three nationwide models using three sets of states: all states, only the states with 29 or 30 rounds surveyed (all but HI, ND, SD and WY), and the top 10 most populous states (CA, TX, FL, GA, NY, PA, IL, OH, MI and NC).

Table 14
SSR, MARE and R^2 of the LR nationwide model for both aggregation methods, for different sets of states.

		SSR	MARE	R^2
All states	Non-cum.	1.39802	0.21116	0.94901
	Cum.	5.69245	0.55992	0.79236
29–30 round states	Non-cum.	1.21596	0.18663	0.95242
	Cum.	5.20820	0.51069	0.79622
Top 10	Non-cum.	0.25986	0.14470	0.94980
	Cum.	0.80138	0.25460	0.84520

4.2.1. Linear regression

We start building the LR nationwide models. We have estimated the regression coefficients by least-squares regression for both aggregation methods, as we did with the statewide models. We also tried removing the non-significant variables and taking the model with the variables that resulted in the lowest SSR. We have worked with the three sets of states stated above, and the resulting SSR and MARE values, as well as the R^2, of the chosen nationwide models, can be seen in Table 14.

The SSR and MARE of the models, when constructed by including or not the data from the states with less than 29 survey rounds, provide us with an interesting insight of the LR model. Let us start with the nationwide model that combines all 50 states and the District of Columbia, whose SSR and MARE are on Table 14. Even if the separate statewide models worked better with cumulative aggregation, the nationwide model seems to be much more accurate with non-cumulative aggregation: both errors of non-cumulative aggregation are less than half the errors of cumulative aggregation. Besides, the accuracy of the models is greatly reduced when compared to statewide LR models. This is specially the case for cumulative aggregation, which gets to a MARE of 0.56, up from 0.130, and an SSR of 5.69, up from 0.012 (see Table 7). And even though non-cumulative aggregation fairs better, it still gets a MARE of 0.21 (up from 0.153) and an SSR of 1.4 (up from 0.013).

However, the big drop in the accuracy of the model may be the result of either inaccurate surveyed measurements or different epidemic behaviour in the states with fewer rounds or smaller states. This possibility comes to mind when noticing that when removing the four states with less than 29 survey rounds the SSR of the models drops to 1.216 and 5.208 for non-cumulative and cumulative aggregation respectively, as displayed on Table 14. The MARE also decreases slightly for both non-cumulative (0.187) and cumulative aggregation (0.511).

Furthermore, if we just consider the ten most populous states (California, Texas, Florida, New York, Pennsylvania, Illinois, Ohio, Georgia, North Carolina and Michigan), all of which have had 30 rounds conducted on them, we find that the SSR and MARE get even lower. As shown in Table 14, with non-cumulative aggregation, we are left with

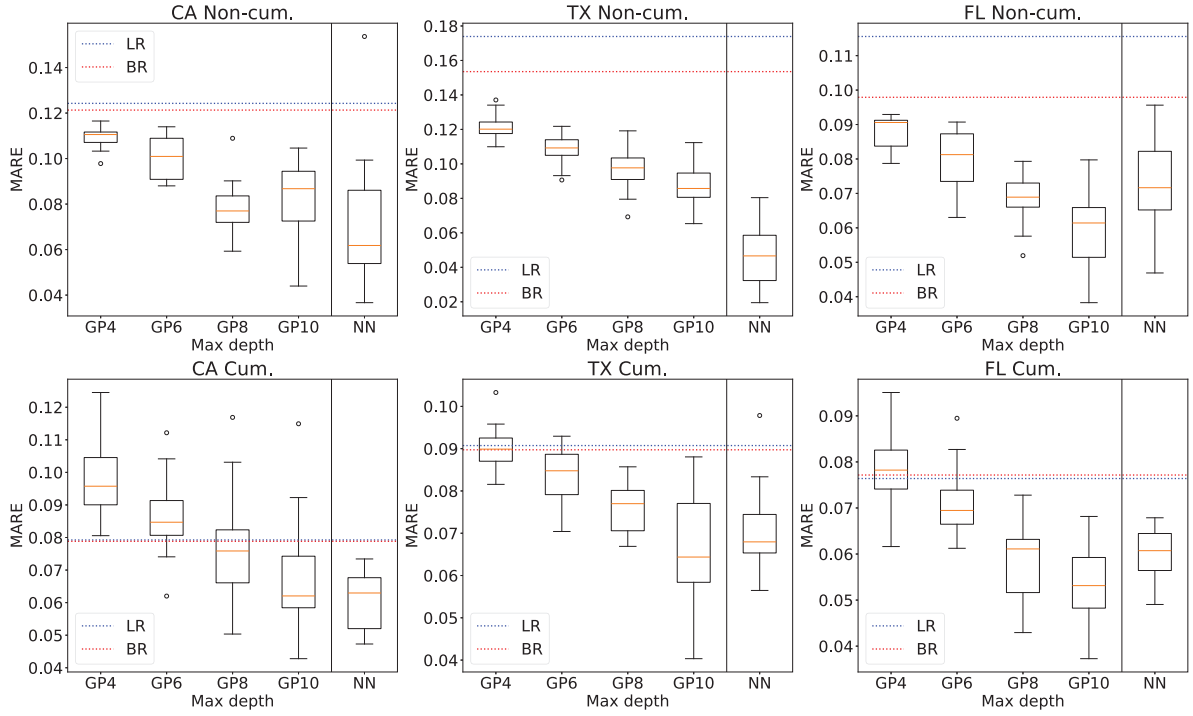


Fig. 10. Box-plots of the MARE of 20 executions of the NN gradient descent algorithm for California (CA), Texas (TX) and Florida (FL). For reference, LR and BR MARE are shown as horizontal lines, and GP are shown as box-plots.

Table 15
SSR, MARE and R^2 of the BR nationwide model for both aggregation methods, for different sets of states.

		SSR	MARE	R^2
All states	Non-cum.	2.60927	0.46267	0.90550
	Cum.	6.70242	0.73248	0.69044
29–30 round states	Non-cum.	2.26936	0.41811	0.90772
	Cum.	6.05642	0.67217	0.71076
Top 10	Non-cum.	0.41682	0.21699	0.82130
	Cum.	0.91739	0.30174	0.70164

Table 16
Table with the mean MARE of 20 executions of the GP nationwide algorithm with all states, states with 29–30 rounds and the top 10 states; for different maximum tree depths.

		By maximum depth			
		4	6	8	10
All	Non-cum.	0.176	0.171	0.167	0.168
	Cum.	0.362	0.351	0.311	0.325
29–30	Non-cum.	0.162	0.157	0.155	0.154
	Cum.	0.313	0.302	0.291	0.291
Top 10	Non-cum.	0.127	0.122	0.120	0.118
	Cum.	0.211	0.193	0.191	0.183

an SSR of 0.26 and MARE of 0.145. With cumulative aggregation, the SSR drops by more than 4.0 to a total of 0.801, and the MARE gets to 0.255.

So, we see that the accuracy of the LR model works better when we ignore the smaller states, and those without all 30 rounds of the survey. This behaviour may be due to bias or inaccuracies in the CDC seroprevalence surveys for those problematic states. So, maybe the real seropositivity of some states did not evolve as suggested by the CDC survey, making the ground truth we use for LR inaccurate, which would result in very big residuals for those specific states, driving the SSR and MARE of the model up.

4.2.2. Beta regression

If we repeat the process with BR to build the equivalent models with the same three sets of states, we obtain the errors displayed in Table 15. We can see that the slight (if any) worsening on average in accuracy seen with statewide models is not replicated with these three nationwide models. With individual states, there was not much change between LR and BR, and in the three most populous states, the BR errors were even lower than with LR; but when all states are considered the SSR increases significantly from 1.39 to 2.61 with non-cumulative aggregation, and from 5.69 to 6.70 with cumulative. The MARE also goes up notably to 0.46 and 0.73 for non-cumulative and cumulative aggregations respectively.

The same behaviour is observed when fewer states are used for the model, and both the 29–30 round states and the top 10 states have a notably higher SSR and MARE than their linear counterparts. Overall, we can conclude that the BR nationwide models we built are worse than the LR nationwide models, so the latter is clearly a preferable approach.

4.2.3. Genetic programming

After running the GP algorithm for those three sets of states 20 times, we computed the mean SSR and MARE for each maximum depth, just like we did with the statewide models. The resultant mean errors are on Table 16; and the MAREs of all 20 executions with box-plots on Fig. 11. We have also computed the R^2 of the GP nationwide models, as displayed in Table 17.

Looking at the box-plots, we can see that the MARE of the nationwide GP-based models is higher on average than the MARE of the statewide models. However, the GP-based models greatly over-perform the LR and BR nationwide models, specially with non-cumulative aggregation. Furthermore, the GP nationwide models seem to suggest that the nationwide model performs poorly with smaller states, driving the mean MARE up, because the GP-based models without the states with less than 29 rounds show better results, and we get even better MARE if we only account for the ten most populous states. This behaviour is also observed with LR and BR.

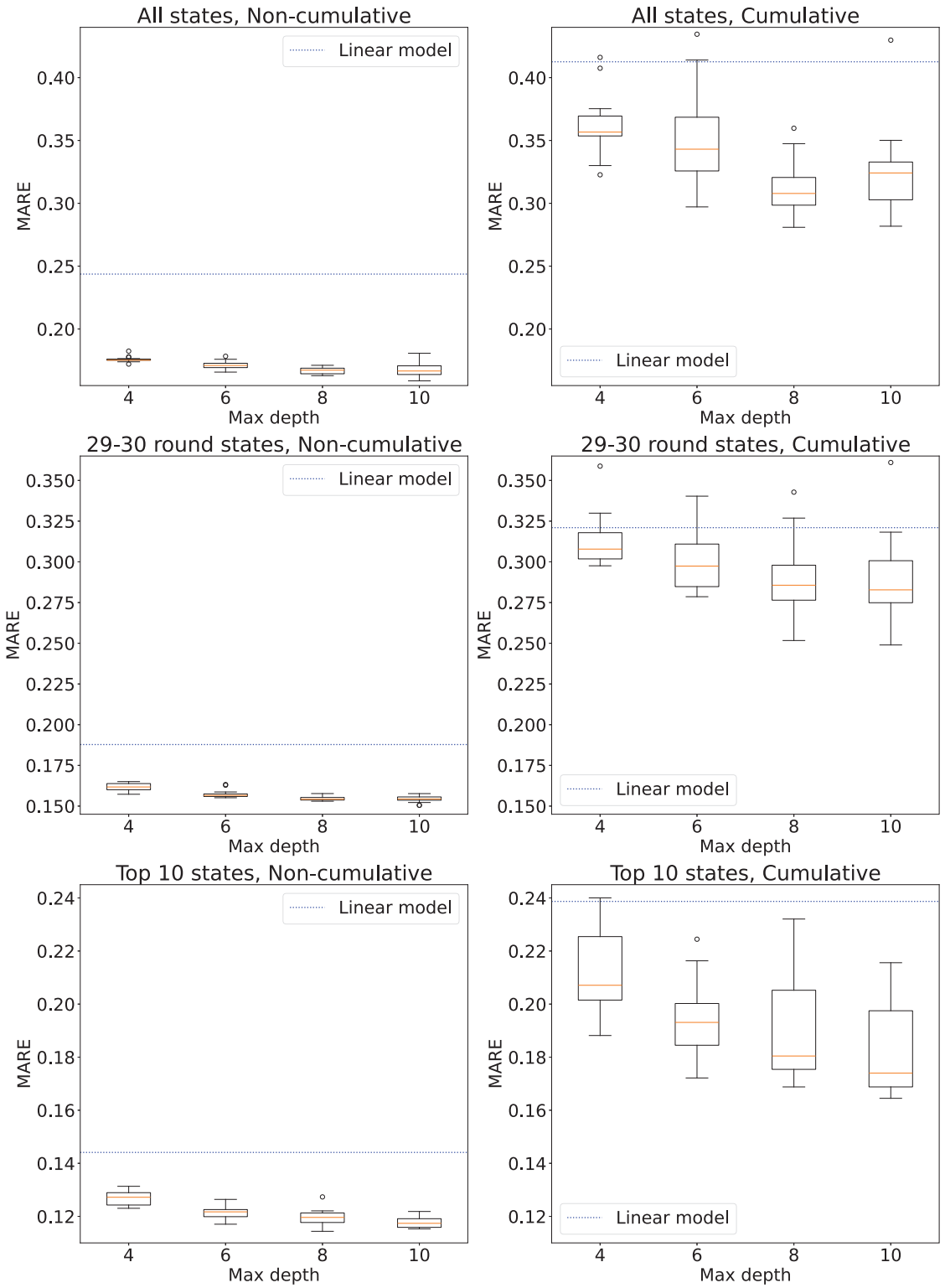


Fig. 11. Box-plots of the MARE of 20 executions of the nationwide GP algorithm with different maximum depths for five sets of states: all states, 29–30 round states, and the top 10 most populous.

Besides, just like the statewide GP-based models, the larger the maximum depth of the models, the more accurate they get. However, there is barely any improvement from maximum depth 8 to 10 for non-cumulative aggregation when states with 29–30 rounds are considered, and for both aggregations with all states. That seems to indicate that a maximal depth beyond 8 levels does not result in a big improvement in accuracy, which leads us to think that it is not worth sacrificing simplicity for the relatively minuscule improvements beyond depth 8.

It is also worth noting that when multiple states are considered, the aggregation that results in the best MARE is the non-cumulative, opposite to what was observed with the statewide models. Furthermore, the accuracy of the GP-based models found with cumulative aggregation varies much more than those found using non-cumulative aggregation.

Table 17
Table with the mean R^2 of 20 executions of the GP nationwide algorithm with all states, states with 29–30 rounds and the top 10 states; for different maximum tree depths.

		By maximum depth			
		4	6	8	10
All	Non-cum.	0.94007	0.94502	0.94762	0.94770
	Cum.	0.74739	0.77148	0.78977	0.77822
29–30	Non-cum.	0.94596	0.95312	0.95738	0.95737
	Cum.	0.77908	0.80614	0.81293	0.81315
Top 10	Non-cum.	0.94449	0.95001	0.95587	0.95825
	Cum.	0.83916	0.87322	0.87418	0.89001

Table 18
Table with the mean SSR, MARE and R^2 of 20 executions of the NN nationwide algorithm with all states, states with 29–30 rounds and the top 10 states.

SSR			MARE		
	Non-cum.	Cum.		Non-cum.	Cum.
All	3.65925	4.19375	All	0.57238	0.62955
29–30	2.01734	3.62840	29–30	0.32490	0.53027
Top 10	2.80141	3.10637	Top 10	0.31806	0.46093

R^2		
	Non-cum.	Cum.
All	0.94292	0.78004
29–30	0.94947	0.79108
Top 10	0.94279	0.81750

The box-plots of the cumulative models show that the MARE values are more spread out. This suggests that the non-cumulative approach results in more deterministic or predictable behaviour for the GP algorithm, while cumulative aggregation is more random and variable.

4.2.4. Neural networks

NN were very accurate with statewide models, so one could expect that they would also be the best with nationwide models. However, if we execute gradient descent 10 times per set of states, it is enough to see that the NN we are using fail to accurately estimate the seroprevalence of multiple states at once. We have used a NN with 7 hidden layers, 6 neurons per layer, a batch size of 10, and a learning rate of 0.0005, as these hyper-parameters were much more optimal for nationwide results. The resulting mean SSR and MARE values and R^2 are displayed in Table 18.

By looking into individual executions of the gradient descent, we see that there are multiple executions that get stuck in local minima with around 4 SSR and 0.8 MARE, which drives up the mean errors. The executions that do not get stuck do not obtain very accurate results either, staying around 1 SSR and 0.3 MARE. Even with these bad results, non-cumulative aggregation is slightly better than cumulative, just like we saw in the other three models.

5. Model validation

Finally, we test the predictive models with new data. For that, we build the predictive model using a subset of the observed data at our disposal, and test said model on a new data subset. We have separated two validation approaches: one based on spacial validation (cross-state validation), which trains the model with a set of data from a geographical area and applies it to a new area; and another one based on temporal validation (temporal forecasting), which takes a time period to train the model and tests it with future seroprevalence data.

5.1. Cross-state validation

This first testing method consists of taking a set of states (training states) to build a nationwide model using the data of those states, and then validating the model with the data of a new state (test state), not included in the set of training states. For example, in Fig. 12, we take California, Texas, Florida and New York as training states, and evaluate the resulting model with Kentucky. This procedure allows us to see how well our models can be adapted to different geographical regions, as long as the data available is equivalent.

Table 19
SSR and MARE of the LR cross-state validation for the ten most populous states.

SSR			MARE		
State	By aggregation		State	By aggregation	
	Non-cum.	Cum.		Non-cum.	Cum.
CA	0.01885	0.10528	CA	0.1421	0.3904
TX	0.04779	0.21749	TX	0.14477	0.22631
FL	0.02429	0.01466	FL	0.10838	0.10443
NY	0.04153	0.18646	NY	0.16009	0.31258
IL	0.02623	0.2684	IL	0.11162	0.31025
PA	0.01966	0.03034	PA	0.13586	0.15547
OH	0.02539	0.08005	OH	0.22749	0.4797
GA	0.0483	0.44332	GA	0.12732	0.58368
NC	0.01985	0.05169	NC	0.17197	0.3103
MI	0.01577	0.01921	MI	0.14363	0.24509
Mean	0.02877	0.14169	Mean	0.14732	0.31182

5.1.1. Linear regression

As an example of the result of cross-state validation, we have taken all the top 10 most populous states except California to build an LR model, and then applied it to the data from California. The results for both aggregations are plotted in Fig. 13. The results look promising, as we can clearly see that the estimations are close to the ground-truth, and overall they follow the trend of the seroprevalence, specially with non-cumulative aggregation.

We did the same with the rest of the top ten states, and computed the SSR and MARE values of each cross-state validation. Said values are displayed in Table 19. We can see that with non-cumulative aggregation the SSR is not too high and the MARE is quite small on average. The cumulative models deviate more from the ground-truth, 31% on average, so the pattern of less accurate cumulative nationwide models with respect to non-cumulative nationwide models is replicated with cross-state validation. Overall, if we use non-cumulative aggregation, it looks like the LR models built can be used for data from new regions quite accurately.

5.1.2. Beta regression

We repeat the cross-validation of the ten biggest states for BR. First of all, we see in the example of California in Fig. 14 that, again, the cumulative model more closely follows the ground truth, but both are fairly close to the Californian seroprevalence. Then, if we apply cross-state validation to the top ten states, we see that the results are, on average, less accurate than LR (see Table 20). The mean SSR and MARE values of the non-cumulative models' cross-state validation are notably higher than their LR counterparts, and the cumulative models are closer to the LR results, but still worse than the linear cross-state validation on average. Overall, the models are not inaccurate, but they are a less appropriate fit than with LR.

5.1.3. Genetic programming

We also perform cross-state validation with GP. We take the ten most populous states as training states and use the model to test the top five states. As the GP algorithm is stochastic, we apply cross-state validation 20 times per state and aggregation, with a maximum depth of 8. Fig. 15 shows the result of one iteration for California. At first glance the cumulative model's prediction looks worse than with LR or BR, as it is displaced downwards in the middle section of the plot; but the non-cumulative prediction seems very accurate.

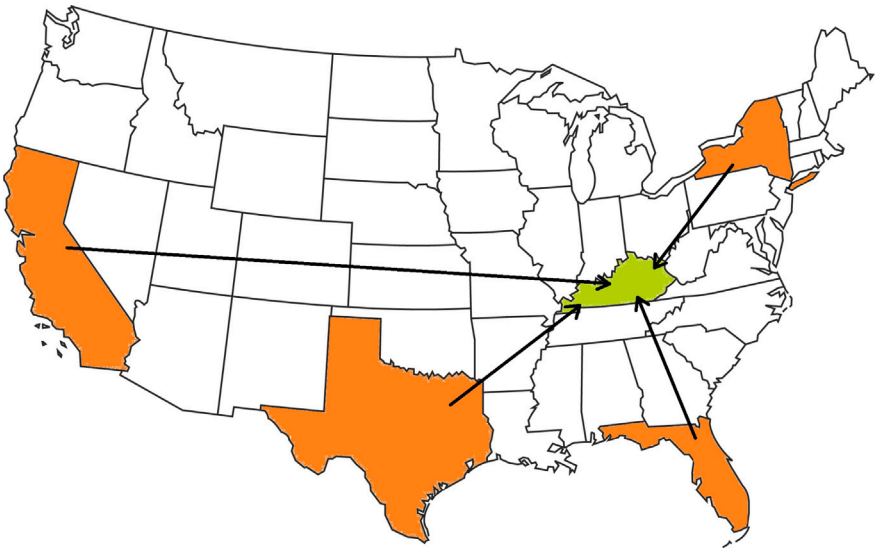


Fig. 12. Graphical representation of the procedure of cross-state validation.

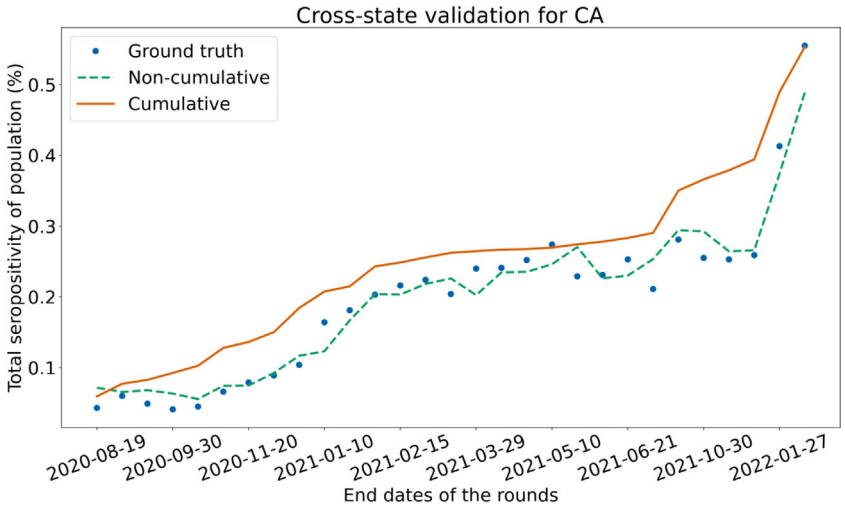


Fig. 13. LR cross-state validation of California (CA) using the other ten most populous states.

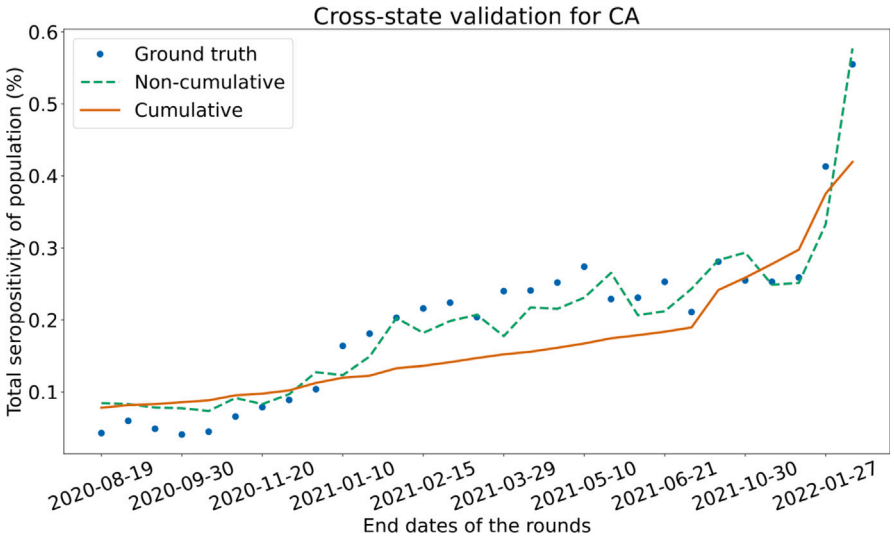


Fig. 14. BR cross-state validation of California (CA) using the other ten most populous states.

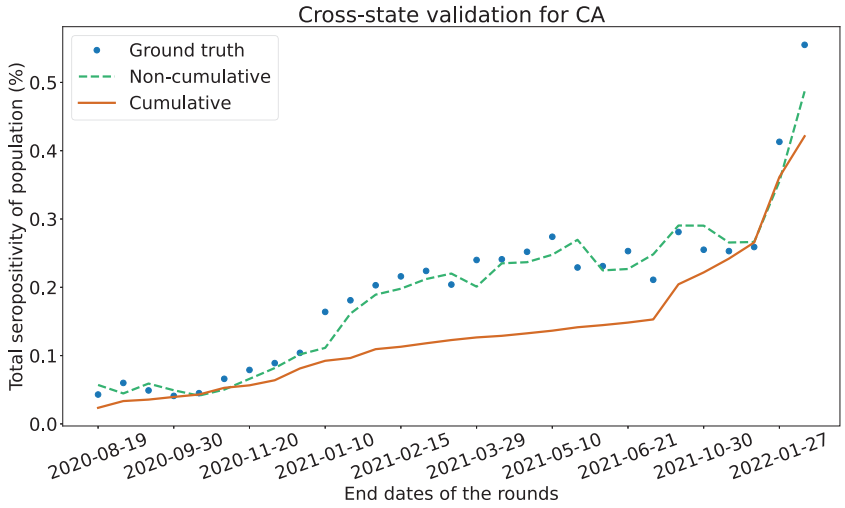


Fig. 15. GP cross-state validation of California (CA) using the other ten most populous states.

Table 20
Results of SSR and MARE of the BR cross-state validation for the ten most populous states.

SSR			MARE		
State	By aggregation		State	By aggregation	
	Non-cum.	Cum.		Non-cum.	Cum.
CA	0.03184	0.10386	CA	0.22891	0.34003
TX	0.10904	0.25427	TX	0.17601	0.28975
FL	0.03978	0.07574	FL	0.18359	0.22073
NY	0.04508	0.12885	NY	0.13923	0.28308
IL	0.07787	0.49094	IL	0.19995	0.46107
PA	0.05707	0.03265	PA	0.15447	0.16355
OH	0.07108	0.38902	OH	0.39208	0.68404
GA	0.07675	0.08541	GA	0.1664	0.25201
NC	0.02691	0.05412	NC	0.26297	0.29929
MI	0.05503	0.11743	MI	0.24701	0.39352
Mean	0.05905	0.17323	Mean	0.21506	0.33871

Table 21
Table with the SSR and MARE of the GP cross-state validation for the five most populous states.

SSR			MARE		
State	By aggregation		State	By aggregation	
	Non-cum.	Cum.		Non-cum.	Cum.
CA	0.02255	0.55530	CA	0.12553	0.55358
TX	0.05165	0.30743	TX	0.13885	0.33710
FL	0.02468	0.24262	FL	0.09825	0.41445
NY	0.04462	0.51597	NY	0.14841	0.49240
PA	0.04233	0.10989	PA	0.13019	0.30062
Mean	0.03717	0.34624	Mean	0.12825	0.41963

In Table 21, we see the mean SSR and MARE of all 20 executions of the GP cross-state validations. We clearly see that just like in the example in Fig. 15, non-cumulative cross-state validation is very accurate on average, without any mean SSR surpassing 0.06, and all mean MARE values below 0.15. The same cannot be said about cumulative aggregation, as the mean SSR is relatively high for most cases, as well as the mean MARE. Overall, the non-cumulative MARE is better than with LR and BR (not so with the SSR), and non-cumulative is still the best aggregation approach.

5.1.4. Neural networks

Lastly, we have tried cross-state validation for NN. Following with the previously shown examples, we have plotted the results of the predictions of NN cross-state validation using the top ten states (except California) as training states to predict the seroprevalence rate of California, shown in Fig. 16. The optimal hyper-parameters chosen were a batch size of 5 and a learning rate of 0.0005 for the RNN of 6 neurons per layer. In the chosen example, the predictions for non-cumulative aggregation are very close to the ground-truth, and the cumulative predictions follow the trend of the seroprevalence fairly well, although it overestimates the rate at the start. However, this is not the norm for NN cross-state validation, and many executions result in fairly bad results.

Table 22
Table with the SSR and MARE of the NN cross-state validation for the five most populous states.

SSR			MARE		
State	By aggregation		State	By aggregation	
	Non-cum.	Cum.		Non-cum.	Cum.
CA	0.234	0.28302	CA	0.55282	0.80358
TX	0.5915	0.48555	TX	0.53336	0.45121
FL	0.35608	0.27787	FL	0.62877	0.55312
NY	0.30288	0.3322	NY	0.4813	0.50359
PA	0.28961	0.24774	PA	0.44584	0.46343
Mean	0.35482	0.32528	Mean	0.52842	0.55498

Just like GP, we have to execute the gradient descent for NN multiple times to get an idea of the average performance of NN with cross-state validation. We also executed it 20 times per state and aggregation, and the resulting means are in Table 22. By looking at the means, we clearly see that the inaccuracy of the nationwide NN models is carried on to the cross-state validation models. The mean SSR and MARE values are very high for both non-cumulative and cumulative aggregations, and NN is clearly the worst model when applying cross-state validation, with a mean MARE of more than 0.5 for both aggregations.

5.2. Temporal forecasting

Lastly, we have used what we call temporal forecasting in order to evaluate the accuracy of statewide models when presented with new data. Temporal forecasting consists of training the statewide prediction models with all available rounds except the last four rounds, and then estimating the immediate next round with said model. This is repeated iteratively, estimating the next round each iteration, until all rounds

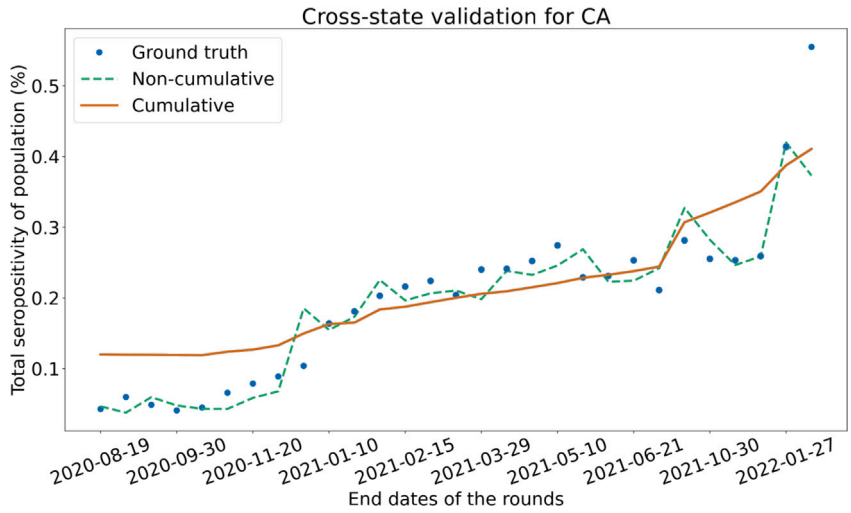


Fig. 16. NN cross-state validation of California (CA) using the other ten most populous states.

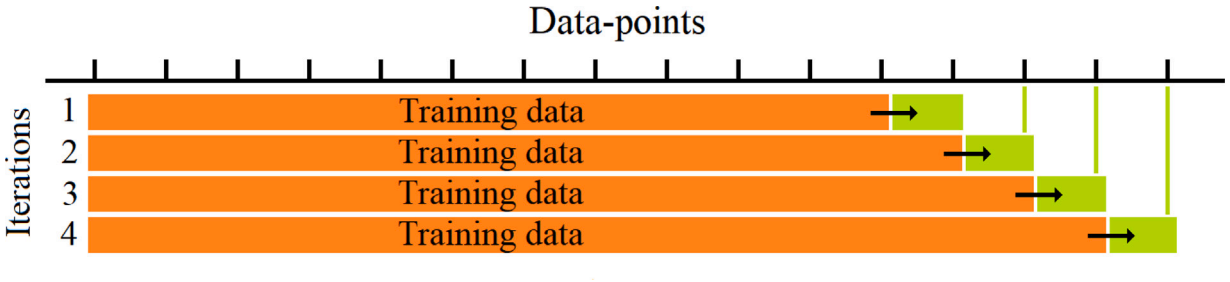


Fig. 17. Graphical representation of the procedure of temporal forecasting.

have been estimated. We then evaluate the SSR and MARE of the estimated rounds. This way, we can see how accurately the models predict the immediately future round. This procedure is shown in Fig. 17. Even though estimating the last four rounds does not sound like much, take into account that those four rounds cover four months in real-time.

5.2.1. Linear regression

We start by applying the forecasting to LR. We picked the three most populous states as examples once again. If we follow the procedure presented above, we obtain the seroprevalence estimations plotted in Fig. 18. Just by looking at the plots we see that, overall, the temporal forecasting does not work very well with these three states. If we check the SSR and MARE values of all the 10 most populous states, shown in Table 23, we see that the results are not too bad, with the notable exception of New York (NY). The mean SSR of the ten states is 0.06 for non-cumulative and 0.104 for cumulative, and the mean MARE values are 0.18967 and 0.21221 for non-cumulative and cumulative. Overall, the error is not very high. Note that the relationship between the aggregations is reversed with temporal forecasting: now non-cumulative aggregation is better.

5.2.2. Beta regression

We also apply BR temporal forecasting to the three most populous states, and plot them in Fig. 19. We can see at first glance that the results are worse than LR, but we need to check the SSR and MARE values. In Table 24 we clearly see that both the SSR and the MARE are worse than for LR throughout the ten biggest states. We obtain a mean SSR of 0.183 and 0.167 for non-cumulative and cumulative aggregations, and a mean MARE of 0.322 and 0.307. So, the models deviate by almost a third of the real values. Such poor results clearly signal that the LR models we built work better when predicting future seroprevalence rates.

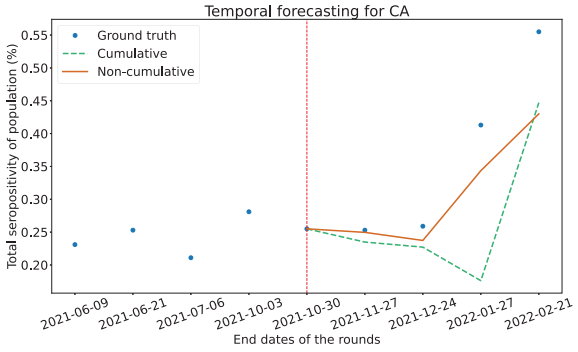
Table 23
Results of SSR and MARE of the LR temporal forecasting for the ten most populous states.

SSR			MARE		
State	By aggregation		State	By aggregation	
	Non-cum.	Cum.		Non-cum.	Cum.
CA	0.02092	0.06911	CA	0.12258	0.24065
TX	0.15389	0.06515	TX	0.30174	0.17305
FL	0.06717	0.06476	FL	0.20583	0.17241
NY	0.07419	0.46524	NY	0.24542	0.57502
IL	0.08248	0.02562	IL	0.20153	0.15114
PA	0.01055	0.00728	PA	0.13756	0.11016
OH	0.03824	0.11331	OH	0.16177	0.24337
GA	0.08645	0.16903	GA	0.20953	0.20766
NC	0.05794	0.05624	NC	0.20461	0.20188
MI	0.01153	0.002	MI	0.10613	0.04672
Mean	0.06034	0.10377	Mean	0.18967	0.21221

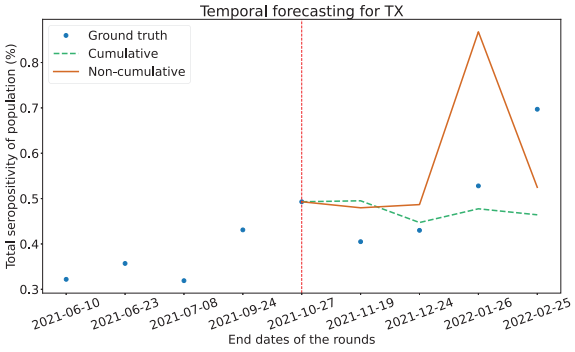
5.2.3. Genetic programming

When applying temporal forecasting to GP, we have to execute the algorithm multiple times to see how it performs, just like we did with normal estimations. We have taken the five most populous states and executed GP temporal forecasting on them 20 times. An example of these executions for the top three states is shown in Fig. 20. These examples seem to indicate that GP is still not very accurate when it comes to predicting future data but does not look much worse than the LR and BR temporal forecasting.

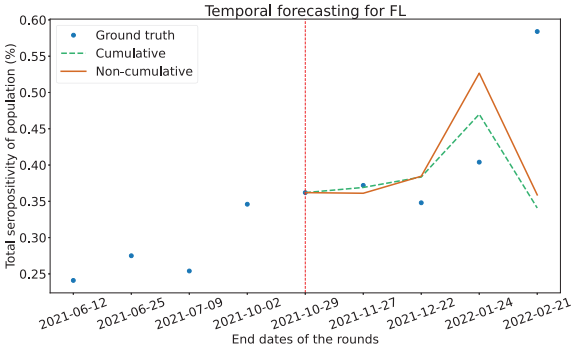
However, if we look into the mean SSRs and MAREs of all the executions for the five biggest states, shown in Table 25, there are clearly some big problems with GP temporal forecasting: some state-aggregation combinations result in exorbitant mean errors (both SSR and MARE). If we look into the individual executions, we notice that



(a) California

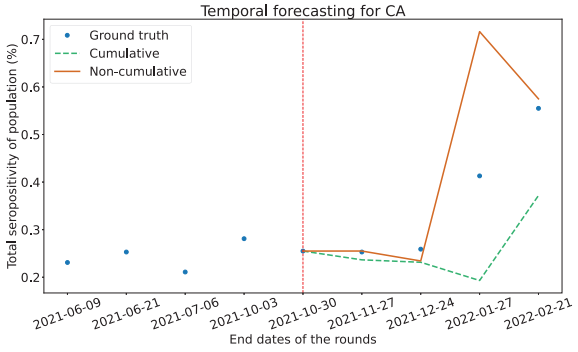


(b) Texas

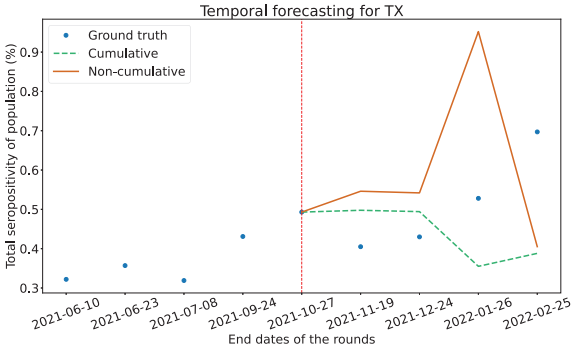


(c) Florida

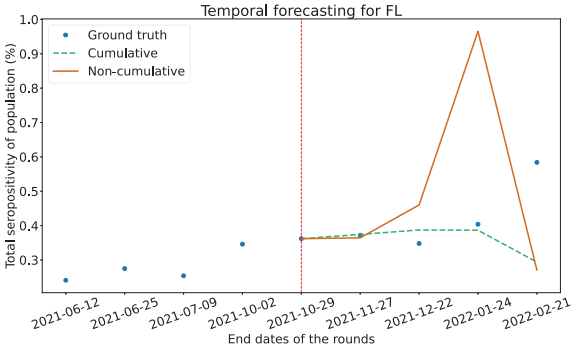
Fig. 18. LR temporal forecasting of the last four rounds for the three most populous states.



(a) California

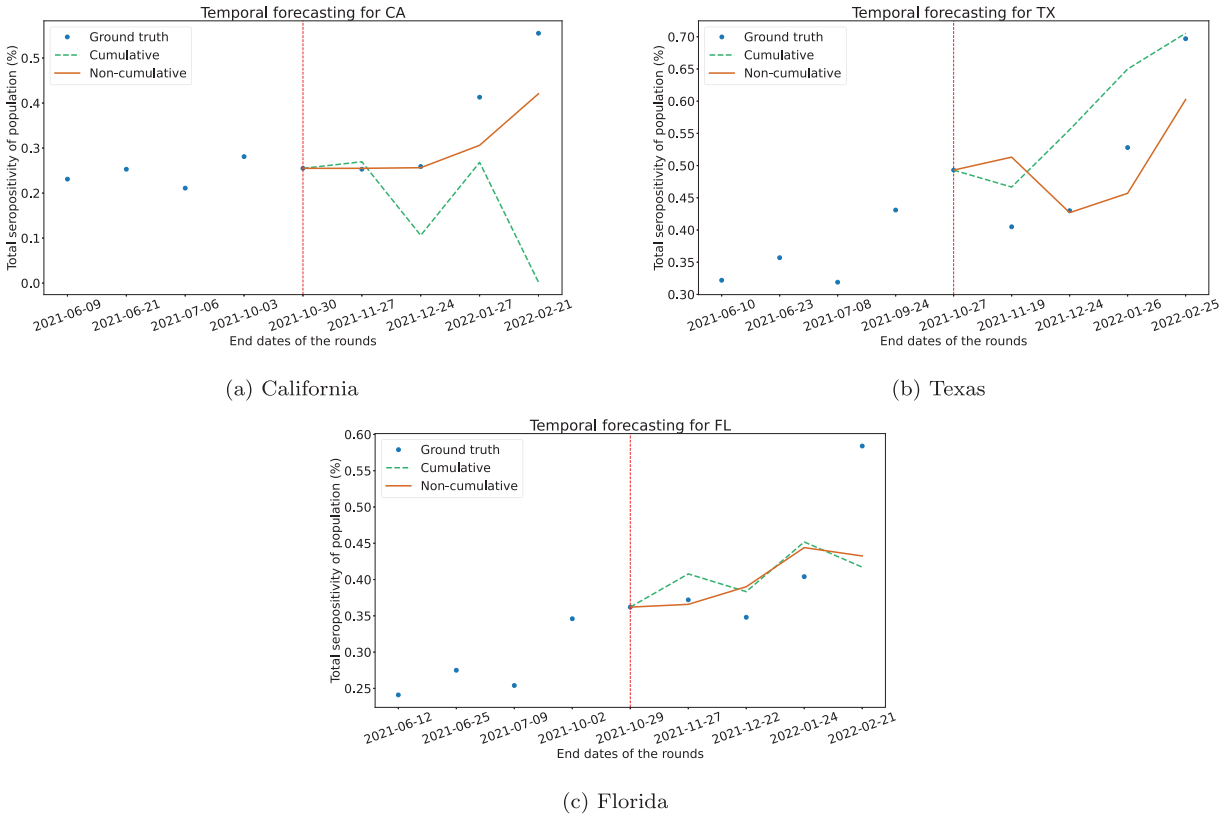


(b) Texas



(c) Florida

Fig. 19. BR temporal forecasting of the last four rounds for the three most populous states.



(a) California (b) Texas

(c) Florida

Fig. 20. GP temporal forecasting of the last four rounds for the three most populous states.

Table 24
Table with the SSR and MARE of the BR temporal forecasting for the ten most populous states.

SSR			MARE		
State	By aggregation		State	By aggregation	
	Non-cum.	Cum.		Non-cum.	Cum.
CA	0.09299	0.08296	CA	0.21854	0.25847
TX	0.29739	0.13807	TX	0.45756	0.28722
FL	0.42564	0.08603	FL	0.56694	0.16481
NY	0.10899	0.33781	NY	0.28967	0.56531
IL	0.22894	0.28911	IL	0.38833	0.36873
PA	0.01258	0.01713	PA	0.14574	0.16058
OH	0.15163	0.23927	OH	0.2643	0.38807
GA	0.20537	0.28326	GA	0.33587	0.32268
NC	0.29298	0.13301	NC	0.43862	0.29888
MI	0.01587	0.06015	MI	0.11093	0.25134
Mean	0.18324	0.16668	Mean	0.32165	0.30661

these spikes in the errors of some states is due to a handful of executions that resulted in huge SSR and MARE values. Therefore, even though most times the GP algorithm results in fairly low errors, there are a few times when the SSR gets to the hundreds, or even thousands. This is probably because the new data that is introduced to the resulting models may take some unexpected values (with respect to the training data), which may result in, for example, a division by a minuscule number.

So, even though GP can provide very accurate estimations with the training data and can usually predict future seroprevalence rates fairly well, its probabilistic nature makes it unpredictable when working with new data.

5.2.4. Neural networks

Lastly, we apply temporal forecasting to NN. Remember that these were the most accurate statewide models. As an example, we have picked California and applied temporal forecasting to it with an NN with the following hyper-parameters: seven hidden layers with nine neurons each, batch size of 5, and learning-rate of 0.1. The result is plotted in Fig. 21. This example looks good overall, and even though there is a dip in the second to last round, the estimations somehow follow the trend of the seroprevalence rate, specially the cumulative model.

Table 25
Table with the mean SSR and MARE of the GP temporal forecasting for the five most populous states.

SSR			MARE		
State	By aggregation		State	By aggregation	
	Non-cum.	Cum.		Non-cum.	Cum.
CA	1633.25	0.15671	CA	19.0925	0.32762
TX	0.07744	0.13307	TX	0.18128	0.26670
FL	0.02500	0.25912	FL	0.11644	0.26916
NY	149.169	936 262	NY	3.65417	281.106
PA	0.04242	0.25610	PA	0.19211	0.37915

We have also executed temporal forecasting on the top five most populous states 20 times with NN. To do so, we picked a batch size of 5 and a learning rate of 0.1. The mean SSR and MARE values of each state and aggregation, as well as the mean of all executions, are shown in Table 26. The results are clearly better than GP, as there have been no executions with exorbitant errors. The non-cumulative mean SSR and MARE, at 0.098 and 0.260, respectively, are better than that of BR, but still notably higher than LR. With cumulative aggregation, on the other hand, the NN SSR and MARE values are the best out of the four models, with a mean SSR of 0.034 and a mean MARE of 0.149.

6. Summary of results

Let us remember the experiments that we have conducted with the ensemble models built. First of all, we have simply evaluated how well

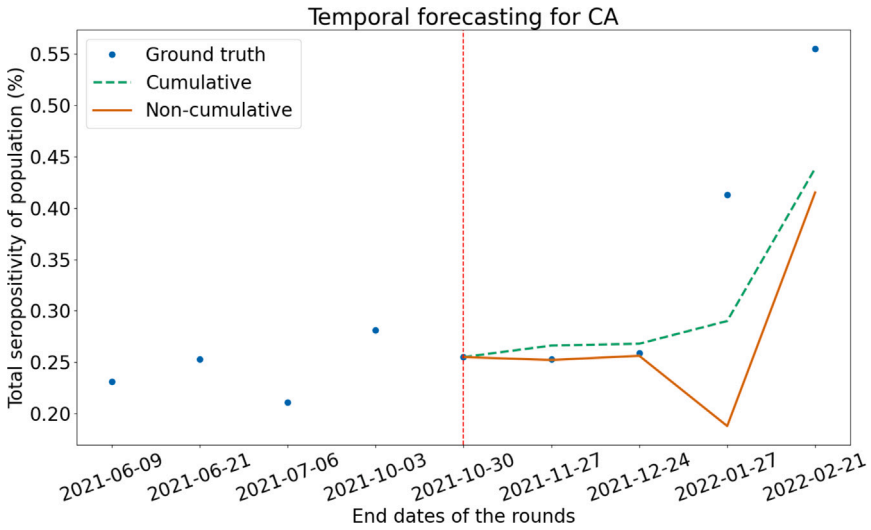


Fig. 21. NN temporal forecasting of the last four rounds for California.

Table 26
Table with the SSR and MARE of the NN temporal forecasting for the ten most populous states.

SSR			MARE		
State	By aggregation		State	By aggregation	
	Non-cum.	Cum.		Non-cum.	Cum.
CA	0.07406	0.03232	CA	0.24074	0.15345
TX	0.12257	0.04916	TX	0.28449	0.17512
FL	0.09577	0.03264	FL	0.23121	0.12526
NY	0.11502	0.04296	NY	0.28193	0.1644
PA	0.08304	0.01432	PA	0.26266	0.12438
Mean	0.09809	0.03428	Mean	0.26021	0.14852

the models could fit the data available, by training the model and testing its accuracy with the same set of data. We have done this for both data-aggregation approaches (cumulative and non-cumulative), and for two types of datasets: statewide data (all the data from single states) and nationwide data (all the data from a set of multiple states put together). Afterwards, we have checked the performance of the models with new data via two validation approaches: cross-state validation, which uses nationwide models to predict the seropositivity of untrained states; and temporal forecasting, which uses the data of a state up to a set date to build a model and predicts the future seropositivity of the state.

Now let us present a summary of the results obtained with each model, aggregation, and scenario. In Table 27, we have displayed which models are the best and worst to choose (with respect to their errors) in each of the studied cases, for each aggregation approach. First of all, we conclude that cumulative aggregation is better when working with individual states within the USA, but non-cumulative aggregation more accurately fits the data when multiple states are considered together. We find that GP obtains much more accurate prediction estimations than those yielded by LR and BR on average, and that, as expected, the complexity of the models obtained using GP is usually inversely correlated with its SSR and MARE. This finding suggests that the non-linearity and complexity of GP models is an advantage with respect to LR and BR, and improves the performance of the models, as could be expected. We also find that the NN built are the most accurate models for statewide models, but notably under-perform the other models for nationwide models.

Furthermore, when it comes to the introduction of new untrained data via cross-state validation and temporal forecasting, we find that the models are better at predicting the data from new states rather than future seroprevalence data. Besides, some models have some clear problems when presented with new data, like NN in cross-state validation and GP in temporal forecasting, which indicates that not every model is appropriate for new untrained data.

The LR and BR models also have a notorious advantage with respect to GP and NN: the execution time. The regression models can build the prediction models much faster than GP and NN, which makes them easier to work with. Indeed, LR and BR can build the models in seconds, while GP and NN usually need minutes or even hours (for nationwide models).

When it comes to the theoretical time complexity of the different approaches, LR and BR have a complexity of $O(np^2+p^3)$, where n is the number of observations and p is the number of explanatory variables (8 in this case). On the other hand, GP has a theoretical complexity of $O\left(P\cdot G\cdot T(\text{fitness})\left(P_c\cdot T(\text{crossover})+P_m\cdot T(\text{mutation})\right)\right)$; where P, P_c and P_m are the total population, the crossed population, and the mutated population, respectively; G is the number of generations; and $T(\text{fitness})$, $T(\text{crossover})$ and $T(\text{mutation})$ are the execution times of the fitness function (SSR in our case) and the crossover and mutation operations, respectively (Lissovoi & Oliveto, 2020). In our case, the crossed and mutated populations are equal to P, so we have a complexity of (17).

$$O\left(P^2\cdot G\cdot T(\text{fitness})\left(T(\text{crossover})+T(\text{mutation})\right)\right) \tag{17}$$

Note that the execution time of the fitness function is dependent on the number of observations and variables (n and p).

For NN, the gradient descent we used is called Mini-Batch Gradient Descent (MBGD), which shuffles and divides the training dataset into batches of size k in each iteration. Suppose the algorithm needs I iterations to stop. In that case, the complexity is given by $O\left(\frac{dnI}{k}\right)k = O(dnI)$, where n is the number of training samples and d the number of parameters being optimised (the weights of the neurons). The complexity is multiplied by the batch size k because MBGD needs to traverse the k data-points of the batches for each update, which increases complexity (Jagadeesha & Bhandari, 2021).

It is easy to see that the theoretical time complexity of GP and NN are bigger than that of LR and BR. With regards to spatial complexity, it is negligible for LR and BR, and it is not a problem for GP and NN either because both approaches use a constant population/iteration throughout their algorithms.

Overall, clearly the simplest models are the LR models, followed by BR models. These models provide a simple combination of the explanatory variables which can help us understand the relationships some variables have with the seropositivity. This lack of complexity may

Table 27
Summary of the best and worst models for each scenario considered.

Best models			Worst models		
	Non-cum.	Cum.		Non-cum.	Cum.
Statewide models	NN	NN	Statewide models	BR	BR
Nationwide models	GP	GP	Nationwide models	NN	BR
Cross-state validation	GP	LR	Cross-state validation	NN	NN
Temporal forecasting	LR	NN	Temporal forecasting	GP	GP

result in less accurate predictions with respect to GP and NN, but the simplicity can be an advantage to be taken into account when choosing the model to use. On the other hand, the GP models, even though more complex, the depths of the GP trees that we have worked with were not big enough to cause a problem due to excessive complexity. As explained before, more depth results in more accurate models, but we considered the improvement beyond 10 to be too small with respect to the trade-off of complexity.

The deterministic nature of LR and BR is also an advantage, because they can only have one outcome, and they are not dependent on the individual execution like GP and NN are. Finally, one notable problem with NN is its lack of explainability. While the other three models provide an analytic formula, which can be analysed to get information about the relationship between explanatory variables and seroprevalence rates, NN is a black box whose inner workings are very hard to understand.

7. Conclusions and future work

Throughout this work, we have presented four different stacking ensemble approaches for the problem of estimating the seroprevalence rate in the USA: LR, BR, GP and NN. These estimation methods have given us insights into how indirect surveys can be used in combination with other data sources to estimate a hidden population, without surveying for said population. Specifically applied to the SARS-CoV-2 pandemic in the USA, we have looked into how these models can be used to get very accurate estimations of seroprevalence at both state and nation levels.

GP and NN are always stochastic, so each execution of the algorithm may result in different models, with varying levels of accuracy; but after building models using the four stacking approaches, we have seen that, when it comes to working with a single state, the NN models are on average the best fit for the observed data, followed by GP and LR models, with respect to both SSR and MARE. Sometimes, the GP and NN algorithms may result in models with higher SSR and MARE than the LR and BR models, specially when not enough complexity is allowed, but those cases are the exception to the norm. With nationwide models, on the other hand, GP models are the most accurate, followed by LR and BR, and our results with NN were poor.

However, when working with prediction models, the error should not be minimised in excess, as that may result in over-fitting the training data. However, the MARE is not too small to take over-fitting as a considerable threat to the accuracy of the model (it is around 0.07 for the best obtained GP models). With statewide NN models, the error has been reduced further, so over-fitting should be taken into account.

We have also seen that the nationwide models have a poorer performance when compared to the statewide models. This drop in accuracy may be the result of poor measurement quality among small states, and states with few survey rounds conducted on them, because the exclusion of these states results in lower SSR and MARE values on average, for every model tried. The drop in accuracy when multiple states are considered may also be due to inconsistencies among states' data, as the data collected in some states might not be as accurate as in others, or the correlation between explanatory variables and seroprevalence may not be identical throughout all states. There may also be more important explanatory variables such as population size and density, climate, social interactions, vaccination or COVID-19 denialism, that were not considered in our work but could have a great impact on the spread of an infectious disease in different regions. In all nationwide models, the increase in error is much bigger for cumulative aggregation with respect to non-cumulative aggregation. Therefore, cumulative was the best aggregation approach with statewide models but is outperformed by non-cumulative aggregation in nationwide models. So, when we are working with the data from a single state, it is better to use cumulative aggregation if we want to minimise the SSR or MARE of the model, regardless of the model we are using (LR, BR, GP or NN); but if we are going to account for multiple states, the non-cumulative approach is a better choice. NN fare specially bad with nationwide models, which may indicate that further research on the application of NN to this kind of problem is needed. After all, we have used a very simple NN.

When it comes to GP nationwide models, the cumulative approach results in a relatively big variance in MARE, which may result in worse MARE than LR for some executions of the GP algorithm. For non-cumulative nationwide models, on the other hand, the MARE's variation is much smaller, which means that the behaviour of the GP algorithm is much more predictable.

We have also used cross-state validation and temporal forecasting to test how well the models perform with new data, and the results offer us insights into the applicability of these models. First of all, cross-state validation resulted, overall, in very accurate results for non-cumulative aggregation, the best being the GP models, closely followed by LR. BR was notably worse with this aggregation, and NN performed badly, as expected from the nationwide models. Cumulative aggregation was not as accurate with cross-state validation, which is in line with the results of the nationwide models. The best cumulative model for cross-state validation was again LR, and in this case, BR over-performed GP. NN were once again notably less accurate.

Secondly, temporal forecasting shows us that even though GP models are very accurate at estimating known data, they perform poorly when presented with new data to predict. GP was indeed the worst model with respect to temporal forecasting, followed by BR. The best model for predicting future seroprevalence was LR when using non-cumulative aggregation, followed by NN; and NN with cumulative aggregation. Therefore, the simplest models were actually very accurate with new data, obtaining better results than the much more complex NN models in some cases. This reminds us of the effectiveness of these widely used models, and that there are some cases where too much complexity may be detrimental, like we saw with GP.

It is important to note that this work has some limitations. Mainly, the use of GP and NN requires a large computing power and a long time to execute the algorithms multiple times. We did not have access to the computational capabilities required for some tests or research lines we would have liked to dive into, so there are some aspects that are left as future work.

Possible future work includes the following research lines:

- Studying the application of GP more in-depth by, for example: trying to implement GP with more operators beyond the ones used in this paper, such as trigonometric functions; allowing the GP algorithm to run for more time before stopping it or using a bigger population size, in order to give the algorithm more space to find a more optimal solution; or trying to minimise the MARE instead of the SSR.
- Focusing on the explanatory variables used, and checking whether adding new variables such as the number of deaths by COVID-19 or vaccination rates can notably improve the models. In turn, exploring the indirect effect of possible new variables (for example, population density) on the quality of the models, without impacting their complexity.
- Further research on the application of NN: adding more hidden layers and neurons, trying different and more complex activation functions.

- Considering new data aggregation approaches other than cumulative and non-cumulative, or studying the application of statistical or deep learning models directly to the non-aggregated data.
- The modelling approaches presented in this paper are not limited to the study of SARS-CoV-2 spread, and could be applied elsewhere. The application of the stacking ensemble models presented here could be done in other areas such as electoral research or sociological studies.

CRediT authorship contribution statement

Gontzal Sagastabeitia: Methodology, Software, Formal analysis, Investigation, Writing – original draft, Visualisation. **Josu Doncel:** Methodology, Resources, Writing – review & editing, Supervision. **José Aguilar:** Conceptualisation, Methodology, Resources, Writing – review & editing, Supervision. **Antonio Fernández Anta:** Conceptualisation, Methodology, Resources, Writing – review & editing, Supervision, Project administration. **Juan Marcos Ramírez:** Conceptualization, Methodology, Resources, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The authors do not have permission to share data.

Acknowledgements

The authors would like to thank Alexander Brodbelt and Mohamed Kacem for their contribution to pre-processing the data used in this work. The work has been partially supported by grant TED2021-131264B-I00 (SocialProbing), funded by MCIN/AEI/10.13039/501100011033, the European Union "NextGenerationEU"/PRTR, by the Department of Education of the Basque Government, Spain, through the Consolidated Research Group MATHMODE (IT1456-22) and by the Marie Sklodowska-Curie grant agreement N. 777778.

References

Akinbami, L. J., et al. (2021). Coronavirus disease 2019 symptoms and severe acute respiratory syndrome coronavirus 2 antibody positivity in a large survey of first responders and healthcare personnel, may-july 2020. *Clinical Infectious Diseases: An Official Publication of the Infectious Diseases Society of America*, *73*, e822–e825.

Al-Bwana, E. (2021). *Coronavirus (COVID-19) detection using ensemble learning* (Ph.D. thesis), Zarqa University.

Anđelić, N., Šegota, S. B., Lorencin, I., Jurilj, Z., Šušteršič, T., Blagojević, A., et al. (2021). Estimation of covid-19 epidemiology curve of the united states using genetic programming algorithm. *International Journal of Environmental Research and Public Health*, *18*(3), 959.

Astley, C. M., et al. (2021). Global monitoring of the impact of the COVID-19 pandemic through online surveys sampled from the facebook user base. *Proceedings of the National Academy of Sciences*, *118*(51), Article e2111455118.

Bajema, K., et al. (2021). Estimated SARS-CoV-2 seroprevalence in the US as of september 2020. *JAMA Internal Medicine*, *181*(4), 450–460.

Benolić, L., Car, Z., & Filipović, N. (2023). Mathematical modeling of COVID-19 spread using genetic programming algorithm. In N. Filipovic (Ed.), *Applied artificial intelligence: medicine, biology, chemistry, financial, games, engineering* (pp. 320–331). Cham: Springer International Publishing.

Centers for Disease Control and Prevention (2023). Nationwide commercial laboratory seroprevalence survey. https://data.cdc.gov/Laboratory-Surveillance.

Cheng, M. P., et al. (2020). Diagnostic testing for severe acute respiratory syndrome–related coronavirus 2: a narrative review. *Annals of Internal Medicine*, *172*, 726–734.

Cilgin, C., & ÖZDEMİR, M. O. (2023). *Vol. 26, Time series forecasting of covid-19 confirmed cases in Turkey with stacking ensemble models* (pp. 504–520). Bingöl Üniversitesi Sosyal Bilimler Enstitüsü Dergisi.

Comito, C., & Pizzuti, C. (2022). Artificial intelligence for forecasting and diagnosing COVID-19 pandemic: A focused review. *Artificial Intelligence in Medicine*, *128*, Article 102286.

Cui, S., Wang, Y., Wang, D., Sai, Q., Huang, Z., & Cheng, T. (2021). A two-layer nested heterogeneous ensemble learning predictive method for COVID-19 mortality. *Applied Soft Computing*, *113*, Article 107946.

Dada, E. G., Oyewola, D. O., Joseph, S. B., Emebo, O., & Oluwagbemi, O. O. (2022). Ensemble machine learning for monkeypox transmission time series forecasting. *Applied Sciences*, *12*(23), 12128.

Delphi Group at Carnegie Mellon University (2022). Delphi's COVID-19 trends and impact surveys (CTIS). https://delphi.cmu.edu/covid19/ctis/.

Elsheikh, A. H., Saba, A. I., Panchal, H., Shanmugan, S., Alsaleh, N. A., & Ahmadein, M. (2021). Artificial intelligence for forecasting the prevalence of COVID-19 pandemic: An overview. *Healthcare*, *9*(12), 1614.

Farlex Partner Medical Dictionary (2012). Seroprevalence. https://medical-dictionary.thefreedictionary.com/seroprevalence.

Garcia-Agundez, A., Ojo, O., Hernández-Roig, H. A., Baquero, C., Frey, D., Georgiou, C., et al. (2021). Estimating the COVID-19 prevalence in Spain with indirect reporting via open surveys. *Front. Public Health*, *9*, Article 658544.

Gupta, A., Jain, V., & Singh, A. (2022). Stacking ensemble-based intelligent machine learning model for predicting post-COVID-19 complications. *New Generation Computing*, *40*, 987—-1007.

Hardesty, L. (2017). Explained: Neural networks. https://news.mit.edu/2017/explained-neural-networks-deep-learning-0414.

Jagadeesha, N., & Bhandari, N. (2021). Computational complexity of gradient descent algorithm. http://dx.doi.org/10.36227/techrxiv.14544000.

Jamshidi, M., Roshani, S., Daneshfar, F., Lalbakhsh, A., Roshani, S., Parandin, F., et al. (2022). Hybrid deep learning techniques for predicting complex phenomena: A review on COVID-19. *AI*, *3*(2), 416–433.

Jin, W., Dong, S., Yu, C., & Luo, Q. (2022). A data-driven hybrid ensemble AI model for COVID-19 infection forecast using multiple neural networks and reinforced learning. *Computers in Biology and Medicine*, *146*, Article 105560.

Klompas, M. (2020). Coronavirus disease 2019 (COVID-19): protecting hospitals from the invisible. *Annals of Internal Medicine*, *172*(9), 619–620.

Larremore, D. B., Fosdick, B. K., Bubar, K. M., Zhang, S., Kissler, S. M., Metcalf, C. J. E., et al. (2021). Estimating SARS-CoV-2 seroprevalence and epidemiological parameters with uncertainty from serological surveys. *eLife*, *10*, Article e64206.

Lissovoi, A., & Oliveto, P. S. (2020). Computational complexity analysis of genetic programming. In *Theory of evolutionary computation: recent developments in discrete optimization* (pp. 475–518). Springer.

Lucas, B., Vahedi, B., & Karimzadeh, M. (2023). A spatiotemporal machine learning approach to forecasting COVID-19 incidence at the county level in the USA. *International Journal of Data Science and Analytics*, *15*(3), 247–266.

Mahajan, A., Sharma, N., Aparicio-Obregon, S., Alyami, H., Alharbi, A., Anand, D., et al. (2022). A novel stacking-based deterministic ensemble model for infectious disease prediction. *Mathematics*, *10*(10), 1714.

National Notifiable Diseases Surveillance System (NNDSS) (2020). Coronavirus disease 2019 (COVID-19) 2020 interim case definition, approved april 5, 2020.

Niazkar, M., & Niazkar, H. (2020). COVID-19 outbreak: Application of multi-gene genetic programming to country-based prediction models. *Electronic Journal of General Medicine*, *17*.

Pollán, M., et al. (2020). Prevalence of SARS-CoV-2 in Spain (ENE-COVID): a nationwide, population-based seroepidemiological study. *The Lancet*, *396*(10250), 535–544.

Quintero, Y., Ardila, D., Camargo, E., Rivas, F., & Aguilar, J. (2021). Machine learning models for the prediction of the SEIRD variables for the COVID-19 pandemic based on a deep dependence analysis of variables. *Computers in Biology and Medicine*, *134*, Article 104500, URL https://www.sciencedirect.com/science/article/pii/S0010482521002948.

Rahman, T., Khandakar, A., Abir, F. F., Faisal, M. A. A., Hossain, M. S., Podder, K. K., et al. (2022). QCovSML: A reliable COVID-19 detection system using CBC biomarkers by a stacking machine learning model. *Computers in Biology and Medicine*, *143*, Article 105284.

Ruder, S. (2016). An overview of gradient descent optimization algorithms. CoRR, abs/1609.04747. arXiv:1609.04747. URL http://arxiv.org/abs/1609.04747.

Rufino, J., Baquero, C., Frey, D., Glorioso, C. A., Ortega, A., Reščič, N., et al. (2023). Using survey data to estimate the impact of the omicron variant on vaccine efficacy against COVID-19 infection. *Scientific Reports*, *13*(1), 900.

Rufino, J., Ramirez, J. M., Aguilar, J., Baquero, C., Champati, J. P., Frey, D., et al. (2023). Consistent comparison of symptom-based methods for COVID-19 infection detection. *International Journal of Medical Informatics*, *177*, Article 105133.

Rufino, J., Ramírez, J. M., Aguilar, J., Baquero, C., Champati, J., Frey, D., et al. (2024a). Performance and explainability of feature selection-boosted tree-based classifiers for COVID-19 detection. *Heliyon*, *10*(1).

Rufino, J., Ramírez, J. M., Aguilar, J., Baquero, C., Champati, J., Frey, D., et al. (2024b). Performance and explainability of feature selection-boosted tree-based classifiers for COVID-19 detection. *Heliyon*, *10*(1), Article e23219, URL https://www.sciencedirect.com/science/article/pii/S2405844023104270.

Salgotra, R., Gandomi, M., & Gandomi, A. H. (2020). Time series analysis and forecast of the COVID-19 pandemic in India using genetic programming. *Chaos, Solitons & Fractals*, *138*, Article 109945, URL https://www.sciencedirect.com/science/article/pii/S0960077920303441.

Salomon, J. A., et al. (2021). The US COVID-19 trends and impact survey: Continuous real-time measurement of COVID-19 symptoms, risks, protective behaviors, testing, and vaccination. *Proceedings of the National Academy of Sciences*, *118*(51), Article e2111454118.

Sharma, S., Gupta, Y. K., & Mishra, A. K. (2023). Analysis and prediction of COVID-19 multivariate data using deep ensemble learning methods. *International Journal of Environmental Research and Public Health*, *20*(11), 5943.

Soto-Ferrari, M., Carrasco-Pena, A., & Prieto, D. (2023). Deep learning architectures framework for emerging outbreak forecasting of mpox: A bagged ensemble scheme to model accurate prediction intervals.

Srivastava, A. (2022). The variations of SIkJalpha model for COVID-19 forecasting and scenario projections. http://dx.doi.org/10.48550/arXiv.2207.02919, arXiv preprint arXiv:2207.02919.

Vaughan, L., Zhang, M., Gu, H., Rose, J. B., Naughton, C. C., Medema, G., et al. (2023). An exploration of challenges associated with machine learning for time series forecasting of COVID-19 community spread using wastewater-based epidemiological data. *Science of the Total Environment*, *858*, Article 159748.

Wang, L., Adiga, A., Venkatramanan, S., Chen, J., Lewis, B., & Marathe, M. (2020). Examining deep learning models with multiple data sources for COVID-19 forecasting. In *2020 IEEE international conference on big data* big data, (pp. 3846–3855). IEEE.

Wang, W., Harrou, F., Dairi, A., & Sun, Y. (2024). Stacked deep learning approach for efficient SARS-CoV-2 detection in blood samples. *Artificial Intelligence in Medicine*, *148*, Article 102767.

Wölfel, R., et al. (2020). Virological assessment of hospitalized patients with COVID-2019. *Nature*, *581*, 465–469.

World Health Organization (2020). Coronavirus disease (COVID-19) Q&A. https://www.who.int/news-room/q-a-detail/coronavirus-disease-covid-19.

Zhou, T., & Jiao, H. (2023). Exploration of the stacking ensemble machine learning algorithm for cheating detection in large-scale assessment. *Educational and Psychological Measurement*, *83*(4), 831–854, arXiv:https://doi.org/10.1177/00131644221117193.

Zoabi, Y., et al. (2021). Machine learning-based prediction of COVID-19 diagnosis based on symptoms. *NPJ Digital Medicine*, *4*, 1–5.