



Convolutional neural networks and temporal CNNs for COVID-19 forecasting in France

Lucas Mohimont¹ · Amine Chemchem² · François Alin¹ · Michaël Krajecki¹ · Luiz Angelo Steffene¹

Accepted: 15 March 2021 / Published online: 14 April 2021

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2021

Abstract

This paper focus on multiple CNN-based (Convolutional Neural Network) models for COVID-19 forecast developed by our research team during the first French lockdown. In an effort to understand and predict both the epidemic evolution and the impacts of this disease, we conceived models for multiple indicators: daily or cumulative confirmed cases, hospitalizations, hospitalizations with artificial ventilation, recoveries, and deaths. In spite of the limited data available when the lockdown was declared, we achieved good short-term performances at the national level with a classical CNN for hospitalizations, leading to its integration into a hospitalizations surveillance tool after the lockdown ended. Also, A Temporal Convolutional Network with quantile regression successfully predicted multiple COVID-19 indicators at the national level by using data available at different scales (worldwide, national, regional). The accuracy of the regional predictions was improved by using a hierarchical pre-training scheme, and an efficient parallel implementation allows for quick training of multiple regional models. The resulting set of models represent a powerful tool for short-term COVID-19 forecasting at different geographical scales, complementing the toolboxes used by health organizations in France.

Keywords Deep learning · Convolutional neural networks · Temporal convolutional network · Transfer learning · Quantile regression · COVID-19

1 Introduction

The Severe Acute Respiratory Syndrome Coronavirus 2, or SARS-CoV-2, was initially described in Wuhan, China. Its

spread is responsible for the COVID-19 disease pandemic, with 188 affected countries, 29,897,412 confirmed cases, and 941,363 confirmed deaths (September 17th). This paper focuses on the evolution of the French COVID-19 epidemic, which presented an elevated spreading rate at its beginning. Indeed, the first French cases were confirmed on January 24th and the 100th case was confirmed on February 29th. The transmission of the virus accelerated in March 2020, following an exponential growth of the number of confirmed cases and hospitalizations, and emergency measures had to be taken to avoid hospital services saturation. As a result, a national lockdown was adopted from March 17th to May 11th. Schools, non-essentials businesses, and public parks were closed. Most outdoor activities and long-distant travels were banned, and a signed form was required for every essential trip (to buy food or medicine, to help a vulnerable family member, etc.). The reproductive number (i.e., the average number of people that would get infected by an already infected person) was estimated to be 2.9 before the lockdown and 0.67 during the lockdown [45]. While the lockdown successfully stopped the virus propagation, the death toll was still high, with 26,619 confirmed deaths at the end of lockdown (reaching 31,045 confirmed deaths on September 17th). A second COVID-19 wave hit France

This article belongs to the Topical Collection: *Artificial Intelligence Applications for COVID-19, Detection, Control, Prediction, and Diagnosis*

✉ Lucas Mohimont
lucas.mohimont@univ-reims.fr

Amine Chemchem
Lamine.chemchem@atos.net

François Alin
francois.alin@univ-reims.fr

Michaël Krajecki
michael.krajecki@univ-reims.fr

Luiz Angelo Steffene
angelo.steffene@univ-reims.fr

¹ LICHS Laboratory - LRC CEA DIGIT, Université de Reims Champagne Ardenne, 51097 Reims, France

² ATOS - Pole Intelligence Artificielle, Rue du Mas de Verchant, 34000 Montpellier, France

during Autumn 2020, with confirmed cases rising since the beginning of August. Indeed, the 7-day moving average since August 28th has been higher than the first wave average. Hospitalization rates increased slower than during the first wave, mostly because the affected population now included younger people, but reached critical levels anyway. While mandatory actions such as the use of face masks in closed areas (and often even in the streets), physical distancing, telecommuting work, and massive testing campaigns delayed the rise of new cases, a new lockdown was set in November 2020. This lockdown was less strict than the precedent, mainly because of the negative impacts on the economy. Indeed, the Organization for Economic Cooperation and Development (OECD) estimated a GDP recession of 11.4% for 2020 without a second wave and a 14.1% recession in the case of a second wave by Fall 2020 [38]. Since the end of the lockdown in mid-December 2020, a generalized curfew and several rules to prevent public gatherings have been set to reduce the virus's spread.

All through 2020, the scientific community has been very dynamic in providing useful decision tools for epidemic modeling, not only for contamination or death cases. Providing different projections of the epidemic evolution helps make effective decisions, whose goals are to reduce the number of victims while avoiding a severe economic recession that would increase poverty and political tensions. Multiple French projects have been initiated to model the COVID-19 epidemic and help reduce its harmful effects, often associating French authorities and services at national and regional scales, public research laboratories, and private companies. Most of these projects aim to capitalize on French high-performance computing resources and interdisciplinary skills related to digital technologies.

Our team at the University of Reims Champagne Ardenne (URCA) is currently involved in several projects related to COVID-19 mitigation. At the pharmacology level, the ANR HT-Covid project relies on the ROMEO Supercomputing Center to simulate millions of molecules and protein interactions (molecular docking). The goal is to identify the molecules that can inhibit the SARS CoV-2 virus and lead to a treatment. We may also cite a collaboration with researchers from the French Ministry of Defense, which developed a dashboard to investigate the impact of sanitary and economic restrictions during the lockdown [15]. Contrarily to the models presented in this paper, that work uses multiple epidemic and economic simulators based on traditional parametric models, with an optimization algorithm to select the best resulting scenario. We are also involved in the Grand-Est region project *PrédicEst*,¹ whose objective is to provide a unique

dashboard for COVID-19 monitoring and forecast by combining multiple prediction models and indicators. This project is partially based on the work presented in this paper, and its goal is to aggregate research models into an operational decision tool.

Therefore, this paper aims at presenting our experience using deep neural networks (and especially Deep learning) to model and forecast different COVID-19 indicators and at different geographical scales. Machine learning and deep learning models are powerful modeling tools that revolutionize several domains. Contrarily to classical parametric models, modern neural networks do not depend on the knowledge of a given phenomenon but can use a data-centric approach that can be applied to raw data and still model complex tasks. On several occasions, Deep learning proved to be competitive against well known traditional modeling algorithms, as presented in our previous works [1, 9], and our team specialized in deep learning for agricultural problems, which often rely on small datasets due to the large span of agricultural cycles (seasons, crops, etc.). As a result, our first experiments on COVID-19 were carried with LSTM [24], Prophet [50] and Convolutional Neural Network(CNN) [34], the latter showing the most promising results.

In this study, we demonstrate how data-driven models can produce excellent predictions. Because Deep learning usually depends on massive input data, it was thought that it could not efficiently model the COVID-19 epidemic as the existing data during the epidemic breakout was reduced and fragmented. Nonetheless, our data processing efforts (including the generation of synthetic data for pre-COVID-19 months) proved successful. For instance, our models can perform national forecasts for a 20-30 days window with error rates as low as 1%, in the best cases.

This work was carried on the basis of the official data provided by the French Ministry of Health, and the different models designed by our team were constructed to respond to the current needs of leading authorities. Hence, the first task we focused on was modeling confirmed COVID-19 cases, an important indicator because it represents the transmission of the disease. However, it depends on the number of available tests being conducted and does not relate directly to the actual burden of the epidemic on the health system. Consequently, the second task we focused on was the hospitalization forecast, at both national and regional scales. This is a more meaningful indicator as a sudden increase in hospitalization can lead to hospital services' saturation, which may be forced to move patients to other facilities (often in different regions of the country) with available beds and healthy medical staff. To better anticipate the health structures' impact, we created a deep learning approach to model a comprehensive set of COVID-19 indicators, such as confirmed cases, hospitalizations,

¹<https://predictest.eu/>

hospitalizations requiring artificial ventilation, number of recoveries, and number of deaths.

This paper is organized as follows: Section 2 describes related work on COVID-19 epidemic modeling, Section 3 describes the data sources used in this work, how they were used, as well as the computing environment supporting our experiments. Section 4 describes the proposed models and Section 5 describes the obtained results. Finally, Section 6 discusses the results and their implications, and Section 7 concludes this work.

2 Related works

Epidemic modeling is commonly achieved with compartmental models like *Susceptible - Infectious - Recovered* (SIR) [42]. In the SIR model, the population is divided into three compartments:

- **Susceptible:** the part of the population that can be infected.
- **Infectious:** the part of the population currently infected.
- **Recovered:** the part of the population that recovered from the disease and that is now immune.

Each compartment is associated with a function that represents the evolution of the population. The modeling is performed by solving a system of differential equations. The SIR model can be extended with other compartments like Deceased (SIRD model) or Exposed (SEIR model). Those models need parameters specific to the studied disease like the rate of infection, rate of recovery, and mortality rate. Initial conditions for the compartments population are also needed. In practice, parameters are estimated by fitting the models to the available data, which is achieved using non-linear optimization methods like the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm [36]. Compartmental models can be used for prediction modeling or a-posteriori analysis.

The need for good initial conditions and a good knowledge of the studied disease is one of the main weaknesses of this parametric approach. In the case of COVID-19, a SIRD model predicted the peak of the Italian first wave for March 21th with 26,000 confirmed cases and 18,000 deaths by the end of the epidemic [18]. Those predictions revealed to be highly inaccurate, as the peak was reached on April 19th reaching more than 109,000 cases, and the death toll rose above the predicted 18,000 as earlier as April 9th. Similarly, a SEIR model was proposed to predict the second wave in France and Italy [19]. This model was first used to estimate the fraction of the population infected during the first wave, projecting that 6% of the French population would be infected during the first wave. A similar work was proposed by [45],

using the Diamond Princess outbreak data to estimate the infection fatality ratio in France. A compartmental data was applied to hospital data, estimating that 4.4% of the French population would have been infected by May 11th. Both results are close. However, the uncertainty of the estimate achieved by [19] is high, with estimations ranging from hundreds of thousands to 18 million infections in France. Nonetheless, SIRD models were used to measure the impact of the French epidemic's lockdown, and it was estimated that the reproductive number R_0 was divided by 7. While compartmental models are useful for creating complex scenarios and performing analysis of past epidemics, other methods based on statistical and machine learning techniques seem to be more suitable for real-time forecasting [25].

An alternative to compartmental models is a data-centric approach that does not use predetermined rules about the disease spreading behavior but where the rules are determined from the data. For example, a death prediction model using a mixture of past predictors was proposed by [47]. The main idea is that the death trend of a country can be represented as a mixture of past death trends from other countries. As a result, acceptable accuracy levels were achieved for up to 10 days forecasts. However, this modeling approach is limited to short-term forecasts. The mixture model needs multiple sequences from different countries with higher death rates (countries considered to be ahead in the epidemic trajectory with respect to the studied country). In practice, the authors have shown that accuracy is low beyond ten days forecasts because not enough predictors (countries) are available. Therefore, a SIRD model was used as a predictor for longer forecasts.

Also, Genetic Expression Programming was used by [44] to create formulas for confirmed cases and deaths evolution in 15 countries. This method seems to be more reliable than LSTM with a higher RMSE and higher R^2 . However, this approach is limited as only COVID-19 time series are used in the modeling process. Finally, a hybrid method using ARIMA and Wavelet-based forecasting was proposed for confirmed cases forecasting [8], where Wavelet-based forecasting was used for error remodeling of the ARIMA model. While this approach presented a good accuracy on the training set, the accuracy for real-time forecasts in France is low, with almost twice the RMSE from the training set.

In this work, we chose to use neural networks as modeling tools. Neural networks have the ability to model complex non-linear patterns, and many different architectures are available. One of the most popular architecture for time series analysis is Long Short-Term Memory (LSTM) [24], which is an improvement of the classical neural network developed to solve the vanishing gradient problem. LSTM is suitable for sequence modeling

as it is able to model temporal effects. It was used in multiple research works on COVID-19 forecasting, as in the case of Canada [11] or India [51]. Accuracy levels of 93.4% and 92.67% were found respectively for short and long-term predictions [11], and similar performances were found by [51], with error percentages ranging from 1.64% to 8%.

LSTM was also compared to ARIMA and Nonlinear Auto-Regression Neural Network on confirmed cases predictions of eight European countries [32], and LSTM achieved the best accuracy in both cases. Other works compared multiple neural network architectures to forecast confirmed cases and recovery predictions, including LSTM, Recurrent Neural Network (RNN), Bidirectional LSTM, Gated Recurrent Unit (GRU [12, 13], a variant of LSTM) and Variational Autoencoder (VAE) [53]. The VAE method achieved the best results with a clear margin, probably because VAE needs fewer data than other neural networks. In comparison with VAE, LSTM can achieve a moderate accuracy but with a lower explained variance.

Another popular type of neural network is the Convolutional Neural Network (CNN). It became the new state-of-the-art model for image classification in 2012 when a CNN won the ILSVRC-2012 competition on the ImageNet dataset [31]. For instance, CNNs were mostly applied to COVID-19 detection from x-ray images [3, 40, 49]. Although CNNs were first successful on image processing tasks, they can also be applied on 1D data like COVID-19 time series, and CNNs were used for confirmed cases forecasting in China [26]. We observe that CNNs are not as popular as LSTM, despite their potentially better accuracy levels and more efficient training with GPU acceleration [4]. Hence, [4] compared CNNs to LSTM, GRU, and the Multi-layer Perceptron, and CNN achieved better accuracy levels. In the present work, we chose to explore the use of CNNs instead of LSTM, as it presented better results during our preliminary experiments with confirmed cases forecasting.

We may also cite a hybrid of LSTM and CNN, the ConvLSTM model, an adaptation of CNN for image sequence modeling. It was used by [41] for COVID-19 spatiotemporal modeling of confirmed cases. In that work, pixel maps for Italy and the USA were created from the available datasets, and an ensemble of ConvLSTM was trained. Indeed, the ConvLSTM uses both convolutional layers to process image inputs and LSTM layers for sequence modeling.

CNNs have also been adapted to generic sequence modeling tasks by using causal and dilated convolutional layers. This family of models is called Temporal Convolutional Network (TCN). TCN models are fully convolutional networks that can be applied to sequences of any length. TCNs have been applied to different application such as stock price prediction, energy consumption forecasting, and automatic sepsis prediction in hospitals [30, 33, 37]. Both

classical CNN and TCN are applied to multiple COVID-19 modeling tasks. To the best of our knowledge, this is the first time TCNs are used for epidemic modeling. Another difference from our work to related work is the development of a multi-level forecast model. Indeed, most works are focused only on national level modeling, but the epidemic crisis management is often operated at regional levels. It is therefore essential to provide projections at a local scale. In this work, we preferred to keep the data at a meaningful regional level (France's official regional organization) during both training and prediction.

The next Section will detail the main data sources and data curation procedures used in our work.

3 Datasets

Our work started in mid-March, at the moment the first lockdown was declared in France. Several indicators, such as French hospitalization data, were not yet available. Therefore, a substantial effort was made on data curation.

3.1 Data sources

In this work, we rely on four datasets composed of different data sources. They are summarized in Table 1 and were used to train the neural networks. COVID-19 data come from different sources like testing laboratories and hospitals. The centralization of French COVID-19 data is a difficult but necessary task that involves hundreds of establishments. Hospitals and testing data are collected and verified by *Santé Publique France*, a French public organization for public health surveillance. Those data are openly available, and they can be visualized on the official dashboard [20]. Data reliability is important for modeling, but some mistakes are expected when many different establishments are involved. As a consequence, data anomalies are checked and corrected regularly.

The first dataset (#1) was built from the COVID-19 data collected by the Center for Systems Science Engineering at John Hopkins University. It is used for worldwide visualization of the epidemic on an online dashboard and it is available on their Github repository [17]. This COVID-19 time-series dataset starts on January 22nd and is updated every day. The variables include confirmed COVID-19 cases, confirmed recoveries, and confirmed deaths. Each sample corresponds to one geographic area and one specific date (one sample per day). Data for 188 countries are available. Demographic, economic, and health indicators coming from the United Nations were added to the dataset.

The second dataset (#2) was built from the official French dataset regarding COVID-19 hospitalizations [21]. It includes time-series of current COVID-19 hospitalizations,

Table 1 Summary of the training datasets. The starting dates of the time series are shown in the Dataset column

| | Dataset | Target | Granularity | Features |
|----|---|--|---------------------------------|---|
| #1 | Worldwide dataset (Jan 24) | Confirmed cases | Daily / country or region | Demographic, health, international mobility and economic features |
| #2 | French dataset (Mar 18) | Hospitalizations | Daily / departments and regions | Population by age, population density. Static mobility data (changes before and after lockdown) extracted from Orange operator. |
| #3 | French dataset with mobility data (Mar 18) | Hospitalizations, Artificial ventilation, Recoveries, Deaths | Daily / departments and regions | Population, population density, Google mobility data |
| #4 | Worldwide dataset with mobility data (Feb 15) | Confirmed cases | Daily / country or region | Google mobility data, demographic, health, international mobility and economic features |

current hospitalizations with artificial ventilation, cumulative recoveries, and cumulative deaths, grouped at national, regional, and departmental levels. These time series started on March 18th and is updated every day. Furthermore, demographical indicators from the National Institute of Statistics and Economic Studies (INSEE) and mobility data extracted by the French mobile operator Orange were integrated into this dataset.

The third dataset (#3) is similar to the second one, only with Google mobility data [27] instead of Orange mobility data. This dataset includes six indicators of mobility changes from a baseline period in different sectors: grocery and pharmacy, parks, public transports, retail and recreation services, residences, and workplaces. Finally, the fourth dataset (#4) is built by merging the dataset #1 and the Google Mobility data.

As indicated before, most of the development time was dedicated to data curation (data collection, cleaning, and preparation) at the beginning of the project. One problem was to find interesting dynamical data for our models. Google created mobility reports at different geographical scales (e.g., national, regional, or city), including the time series of six mobility indicators described previously. Such variables are useful because population mobility has an impact on the epidemic spreading. Finding relevant dynamical data is difficult because systematic data collection is sometimes limited by concerns such as ethics, security, or processing capability.

3.2 Data extraction

Data curation was partially performed with Excel, then with the Pandas Python library. Data processing was performed in Python with Pandas and Numpy. Deep learning models were created and trained with Tensorflow and Keras. To adapt the data sources to our needs, we developed some assumptions. Our primary assumption for this work is that

convolutional neural networks (CNN) are robust enough to learn valuable features from data without much pre-processing. Another assumption is that CNN can benefit from data built from countries with different COVID-19 epidemic evolution.

Also, the datasets contain multivariate time-series of different durations, depending on the country or the data source. As CNNs require input samples to have the same size, time-series were converted into input and target sequences of constant size for supervised learning. One problem was the size of the datasets at the beginning of the project. For French hospitals' data, only 12 days of data were available for each department. As data before March 18th was unavailable, the input sequences length was initially limited to 5 days, a value we conserved afterward to simplify comparisons.

Because available data was limited at the beginning of the work, a data augmentation step was performed using the classical CNN proposed in Section 4 and by reversing the order of the value in sequences. This data augmentation technique allowed the addition of ten days of sequences prior to March 18th. While the datasets were updated every day with new sequences, there were only a few hundreds of entries at the end of March 2020, when we started our project. In comparison, datasets available at the end of 2020 count with thousands of sequences available for training the neural networks. Even though, due to the entries' nature, the datasets are relatively small and can be processed quickly on consumer-grade hardware.

Features are normalized to zero mean and unit variance. The Orange mobility data was originally in the form of daily time-series, but it had to be simplified as constant features. Indeed, the Orange mobility features represent the change in mobility before and after the beginning of lockdown, but it was challenging to obtain frequent data updates, which motivated the simplification stated above. Similarly, A 14-days sliding-window average was used to

smooth Google's mobility time series and remove noise and weekly seasonality.

In the next section, we will describe the different models we experimented.

4 Developing models for COVID-19 forecast

As presented in Section 1, we developed several models to cover different indicators for the COVID-19 epidemic. This section presents these models and how they were applied to the datasets to produce forecasts.

4.1 Convolutional neural network for time series

Convolutional Neural Networks (CNNs) were designed in the late 1980s to solve image classification tasks, and they were successfully applied to computer vision tasks like handwritten digits recognition [14]. CNNs are not a new idea, but many limitations have prevented them from widespread success in the 1990s. Indeed, CNN training requires powerful computers and a vast amount of data, which were not available at the time. Multi-layer neural networks training was also known to be difficult, and other techniques like the Support Vector Machine (SVMs) were successful alternatives. CNN only achieved widespread recognition when a CNN architecture, Alexnet [31], won the ILSVRC-2012 challenge. This challenge's goal was to achieve the best accuracy on the Imagenet dataset (1000 class tasks), and AlexNet got a 10% improvement over the second-best entry.

CNNs are Multilayer Perceptron (MLP) adapted to image processing. A typical MLP is made of fully connected layers only, which is impractical for image inputs as the number of parameters would be too high. For example, an MLP first layer of 32 neurons and 28x28 pixels grayscale image input would have 25,088 parameters (and over 66 million parameters for a full HD image), and fully connected layers do not account for the pixel's neighborhood, which limits the ability to learn complex image pattern. The CNN solves this issue using convolutional layers, where 2D convolutional filters replace the fully connected neurons. These filters can be seen as shareable neurons as they are applied in a sliding window manner on the whole image, acting as feature extractors that are reused on every part of the image. The motivation for this is that shapes, textures, or objects can be anywhere in the image. Convolutions are followed by a non-linear activation to produce the final output. Each filter produces a feature map, and the feature maps are stacked together to be used as input in the next layer. A sub-sampling (or pooling) layer can be used after a convolutional layer to reduce the feature maps' size. A typical CNN uses a

succession of convolutional/pooling layers to produce a robust feature extractor. Fully connected layers are then used for classification with a soft-max function. The filters' parameters and the fully connected parameters are both calculated by optimizing a loss function, which is generally performed by stochastic gradient descent and with the back-propagation algorithm [43]. As a result, a trained CNN can be described as a hierarchical feature extractor: the first layers can be used to extract low-level features like edges or lines, while the next layers can be used for more complex shapes, textures, or object parts [52].

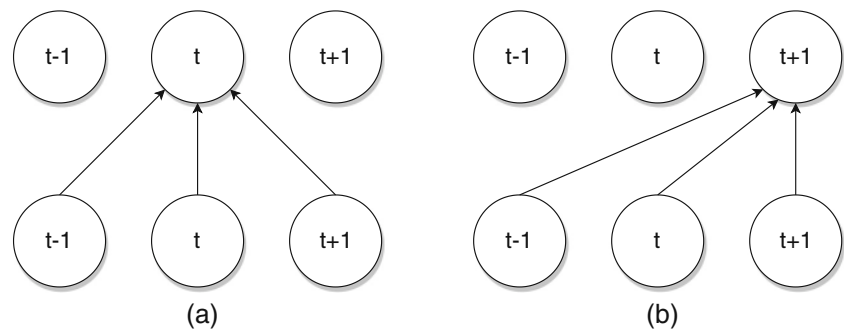
The idea of using CNN for time series processing is not new and was proposed in the 1990s by the original inventor of CNN, Yann LeCun [35]. In this case, a CNN for time series processing uses 1D filters instead of 2D filters. Time series can be represented as 1D arrays in the same way images are represented as 2D arrays. Few changes can also be made to solve regression tasks, like selecting the appropriate output activation and loss function, which can be, for example, ReLU (Rectified Linear Unit) and the mean-squared error.

4.2 Temporal convolutional neural networks

In addition to the CNN-based architecture for time series presented in the previous section, the literature contains other time-series processing architectures that worth being studied. One of them is the Temporal Convolutional Network (TCN) architecture, a technique that was first used in Wavenet [39]. This model was initially designed for sound-related predictions, such as music or speech synthesis, using raw data, and setting a new state-of-the-art in Text-to-Speech systems. TCNs use causal dilated convolutional layers, in which the convolutions preserve the time causality. Causal convolutions do not use future values of the input sequence to calculate their activation. The difference between a classical and a causal convolution is illustrated in Fig. 1. Causal convolutions are used in TCNs with an increasing dilation rate. The dilation rate is a parameter that can be used to expand the input window of the convolution while keeping the same kernel size (some values of the input are ignored). Therefore, a higher dilation rate corresponds to a higher receptive field, and many causal and dilate convolutional layers are stacked to process long sequences. This is illustrated in Figs. 2 and 3.

One inconvenience of most TCNs is that they do not have pooling layers to avoid losing information. This can be circumvented by using a residual skip connection, first introduced in the ResNet models [23]. Skip connections are inserted between the input and the convolutional layers' output to overcome the vanishing (or exploding) gradient problem.

Fig. 1 **a** Classical convolution with a size 3 kernel that does not respect time causality. **b** Causal convolution with a size 3 kernel [39]



4.3 Adapting CNN models for the COVID-19 epidemic

4.3.1 Description of the proposed CNN

Our first attempt at COVID-19 modeling, at the beginning of the first French lockdown in mid-March 2020, relies on a classical CNN for time series regression. Because the available dataset was small, most of the development time was focused on data curation, and we chose to apply a CNN as a one-step-ahead regression model. In this first attempt, the model receives a five-day input sequence and produces a forecast for the next day. This limited input size was selected because there were only 11 days of hospitalizations data available during the beginning of the project, and a five days input size was chosen to perform a training/validation split. Longer input sizes were tested when more data were available, but their performances were similar or slightly worse in most cases. Deeper CNN models also shown similar performances, which advantaged the relatively lightweight model presented here.

To perform long-term forecasts, multi-step predictions shift the input and use the one-day forecast as the input sequence's last value. The CNN uses a single 1D convolutional layer with 2×1 filters followed by a max-pooling layer, and two fully connected layers are then used to produce the final output. The model architecture is described in Fig. 4. Another version of this CNN was also

developed, using two separates inputs: a convolutional layer for the COVID-19 sequences and a dense layer for the static data. The output of both layers is then concatenated to be used in the networks' final dense layers. The summary of this CNN architecture can be seen in Fig. 5. In this version, dropout with a probability of 0.5 was used on the dense input and on the dense hidden layer to reduce the effect of potential over-fitting [48]. Dropout was also used to produce confidence intervals [22]. Both CNN versions were trained with the Adam optimizer with a learning rate of 0.0001 and a mini-batch size of 64 [28].

4.3.2 Model for confirmed cases prediction

The first version of the CNN-based model was applied to cumulative confirmed cases forecasting. Three scenarios, corresponding to three independent training, were built by filtering the first dataset (Dataset #1), based on the data available on March 29th. Hence, the first scenario (optimistic) uses data from countries where the epidemic was stopped (this corresponds to China and South Korea data, only). The second scenario (compromise) uses data from every country with at least 1000 confirmed cases (including China and South Korea). Finally, the third scenario is a pessimistic approach to the second scenario, where China and South Korea were removed from the training set.

Fig. 2 **a** A causal convolution with a dilation rate of 1. **b** A causal convolution with a dilation rate of size 2. The receptive field is bigger while having the same number of parameters [39]

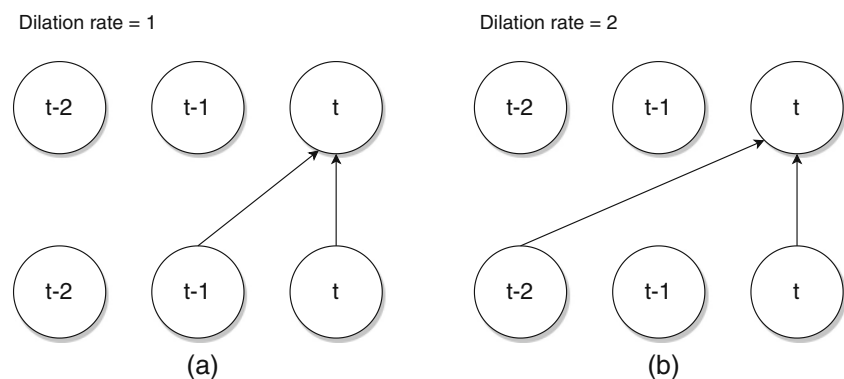
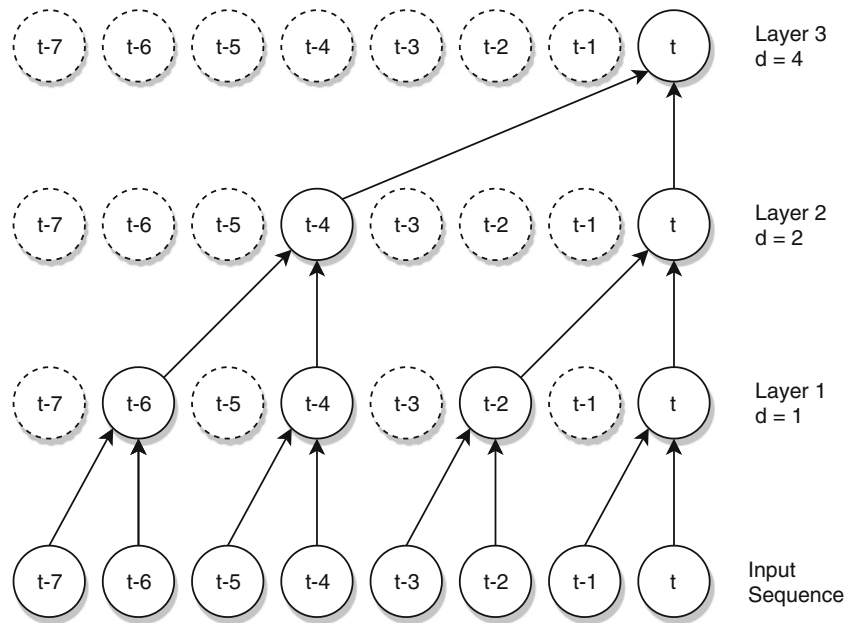


Fig. 3 Stacked causal and dilated convolutional layers. The increasing dilatation rate is used to artificially increase the receptive field while keeping small kernel of size 2 [39]



The second version of the CNN was applied to daily confirmed cases forecasting. The goal of this model was to create an optimistic scenario of the daily increase of confirmed cases. The training was achieved using decreasing sequences only (corresponding to the countries where the epidemic was receding). Data available on May 8th were used during the training.

4.3.3 Model for hospitalizations prediction

The second version of the CNN was applied to French COVID-19 hospitalizations modeling. Like the precedent model, this modeling experiment started in late March when only 12 days of historical data were available (as data collection from French hospitals started on March 18th). In the following weeks, the training was re-executed every week to include the new data, and projections at both regional and national levels were systematically sent to the French Ministry of Health. Indeed, we were committed to creating a model that could be used to monitor the epidemic after the end of lockdown on May 11th. Therefore, a baseline model was trained for

this purpose, using the data available on May 10th for training, and only decreasing sequences were kept in the training set. The model was then used for short-term projections, corresponding to a fictional extended lockdown where COVID-19 hospitalizations decrease. Projections and observations were therefore compared at both regional and national levels to detect a slow-down of recoveries or an increase in hospitalizations compared to that ideal scenario.

4.4 Adapting TCN models for the COVID-19 epidemic

4.4.1 Description of the proposed TCN

The CNN-based models proposed in the previous sections have good short-term accuracy but has several limitations. For example, it only used small sequences of five days as input when it was designed, mostly because the datasets were too small. With time, new data becomes available and more complex models could be used for longer sequence processing. Therefore, this work also proposes the use of a TCN model. The training of a more complex model was possible because COVID-19 data are updated daily,

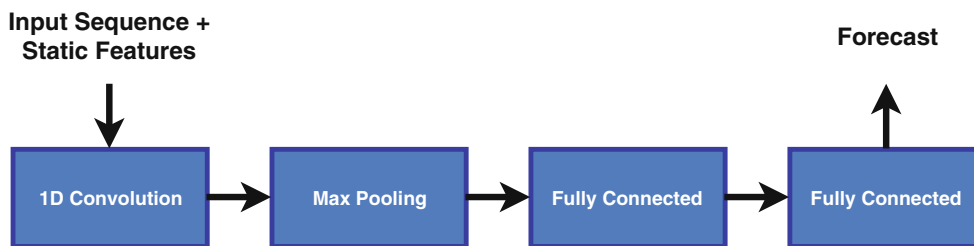


Fig. 4 The proposed CNN for confirmed cases modelling scenarios

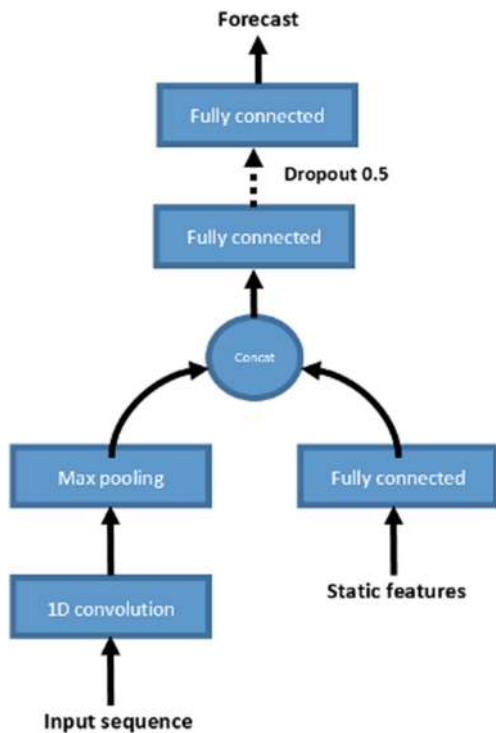


Fig. 5 Multi-input CNN with dropout

and our datasets were large enough when the development of the TCN model began (mid-May). The TCN proposed here is an adaptation of the conditional TCN introduced by [6], applied initially to financial data forecasting with time series conditioning. This model is built with three blocks. The input block has two separate inputs, one for the main sequences and one for conditional sequences. The two separate paths contain a 1D causal convolutional layer as explained in Fig. 6. Residual skip connections are used for both inputs. 1x1 convolutional layers can be used in the skip connections to change the number of feature maps (to have the same sequence sizes for the addition operation of

Fig. 6 Input block with 2 separate paths to process the target sequences and the conditional sequences. Residual skip connections are also used with a 1x1 convolution layer to have the same number of features maps before the final concatenation [6]

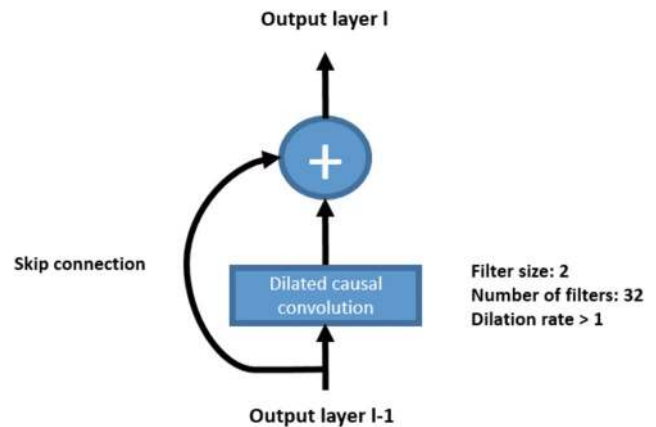
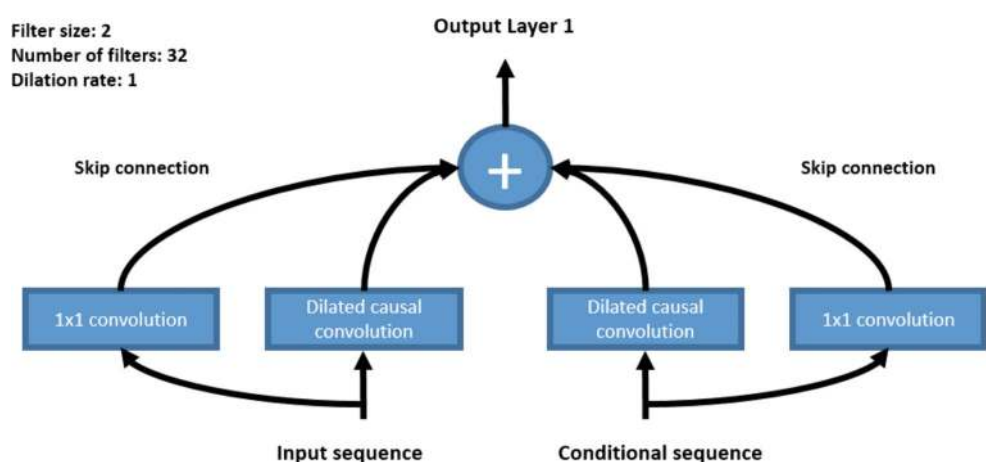


Fig. 7 A central block made of one dilated causal convolutional layer with a residual skip connection. The dilation rate is increased when multiple central blocks are stacked together to process longer sequences [6]

the skip connection). The central blocks are residual blocks with causal convolutions and an increasing dilation rate (Fig. 7).

Multiple central blocks are stacked with an increasing dilation rate. The last central block will have a receptive field that covers the complete sequence length. From the original publication, the output block is a single 1x1 convolutional layer with one filter per output and followed by a global pooling layer (Fig. 8). The TCN proposed in this work uses the same input and central blocks, where each layer has 32 1D filters of size 2 with ReLU activation. However, the output block was adapted to produce confidence intervals with quantile regression as proposed in [10, 29]. This is achieved with three independent output blocks with 5%, 50% and 95% quantile losses as explained in Fig. 9. The following quantile loss equations were used to train the TCN (1 and 2):

$$l(y, \hat{y}, q) = q \times \max(0, y - \hat{y}) + (1 - q) \times \max(0, \hat{y} - y) \quad (1)$$

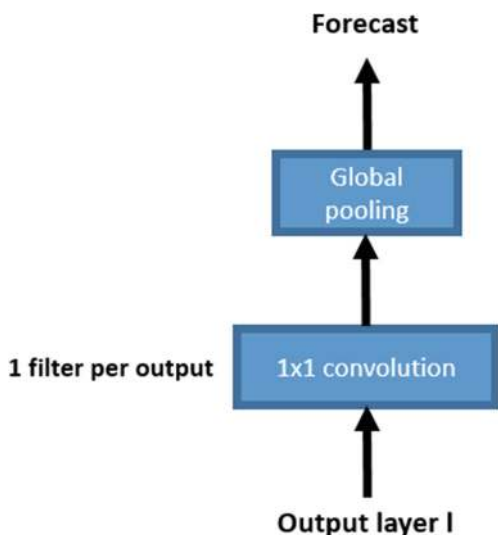


Fig. 8 The original output block with one 1x1 convolution layer and a final global pooling layer to produce a one-step ahead prediction. The 1x1 layer can be adapted for multi-task forecasting by using a filter for each prediction target. Adapted from the figure shown in [6]

$$LOSS = \sum_{(i=1)}^k l(y, \hat{y}, q_i) \tag{2}$$

Each output block contains a convolutional layer with 1x1 filters with ReLU activation and dropout regularization. This layer acts as a dense layer applied in a sliding window manner on the feature maps produced by the last central block. A final convolutional layer then follows it, with a single 1x1 filter with ReLU activation and global pooling, and a multi-task version of the TCN was studied in this work. It is a multi-task model with hard-parameters sharing because the whole network is used to produce the outputs. It was found that the 1x1 layers with dropout in the output blocks were necessary for the multi-task model to achieve good accuracy (without this layer, the model performs terribly). The multi-task model has three output blocks, one for each quantile, and each block produces a single day prediction for each target. Another possible way was to add one output block for each target and each quantile. This approach is closer to the one used by other multi-task CNN [2, 54], but it was not able to converge appropriately during training.

The final objective is to produce multiple outputs from different COVID-19 time series, as several COVID-19 variables like hospitalizations and deaths are highly correlated. The multi-task output is achieved by using one filter for each target in the network’s last convolutional layer.

4.4.2 Transfer learning and multi-task TCN for regional COVID-19 prediction

The TCN with quantile regression was applied to the four COVID-19 time series of the French dataset #3: current

hospitalizations, current hospitalizations with artificial ventilation, cumulative recoveries, and cumulative deaths.

The training was first performed independently for each COVID-19 indicators (one-target models): the main input included one of the four COVID-19 time series and the daily variation. The Google mobility data, total population, and population density were used in the conditional input. Subsequently, a multi-task TCN was trained with the four COVID-19 time series altogether. This training was performed with the Adam optimizer with a 0.0001 learning rate and a 256 batch size. Early stopping was used for regularization purposes.

We observed that performances at regional and departmental levels were generally lower compared to the national level. Hence, a hierarchical transfer learning scheme was applied to train regional models with improved accuracy, as illustrated in Fig. 10. There, a global (or initial) model is trained with every available data from Dataset #3. Regional models are then trained by using the weights of the initial model instead of random weight initialization. The training at the regional level only uses data from the concerned region.

Departmental models can also be trained by following the same technique (the concerned regional model replaces the global model for the pre-training step). In our case, the focus was set onto regional levels as the French epidemic is managed at the regional level.

Training multiple models, one for each region, can be highly demanding in computation power. For this step, we could rely on resources from the ROMEO Supercomputing Center.² ROMEO resources include a mixed CPU-GPU cluster dedicated to HPC and Artificial Intelligence (ranked 249th in the TOP500 list, June 2018), as well as a Nvidia DGX-1 server (8x Nvidia V100 GPUs with 16 GB RAM each and NVLink), specially dedicated to AI.

The training of the initial model was performed on a single Nvidia V100 GPU. Regional models were trained with a multi-thread CPU implementation. Each training was performed on a single thread on an Intel Xeon 40 cores CPU.

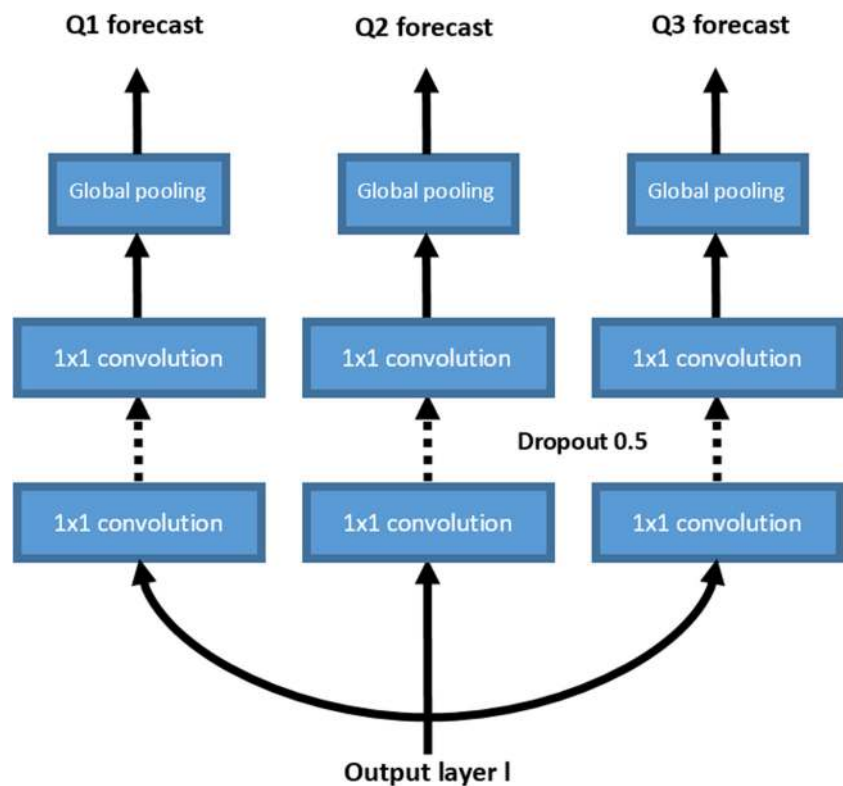
4.4.3 TCN for confirmed cases forecasting

The proposed TCN with quantile regression was also applied on French confirmed cases forecasting (Dataset #4) in three different ways:

1. the TCN was trained on French data only for cumulated confirmed cases forecasting,
2. the TCN was trained on Worldwide data for cumulated confirmed cases forecasting

²<https://romeo.univ-reims.fr>

Fig. 9 The quantile regression output block. Each output is similar to the original output block. They used the same feature maps as input and can be adapted for multi-task forecasting by adding more filters to the 1x1 convolutional layers



- the TCN was trained on Worldwide data for daily confirmed cases forecasting

The training used the Adam optimizer with, respectively, 32, 1024, and 1024 samples per mini-batches and with a learning rate of 0.0001. Early stopping was used as a regularization technique. Negative values (data error) were removed as confirmed cases sequences should only contain positive values.

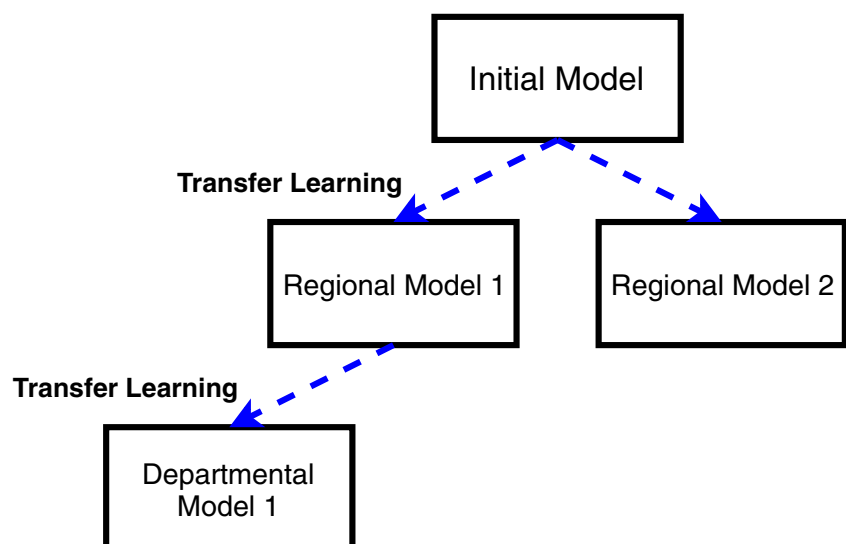
The three training were performed twice, with and without data augmentation. The augmentation technique

used is random masking noise (random values of the sequences are set to 0). Training with augmentation was performed by creating nine new sequences with between 6 and 20 random corrupted values.

4.5 Evaluation strategies

Classical CNNs showed in Figs. 4 and 5 were respectively applied to cumulative confirmed cases predictions scenarios and daily confirmed cases predictions. The second classical CNN was applied to hospitalizations forecasting (Dataset

Fig. 10 Example of a hierarchical transfer learning for regional and departmental training



#2). The TCN model was first applied to French COVID-19 hospital data (hospitalizations, artificial ventilation, recoveries, and deaths) with the Google Mobility reports (Dataset #3). The classical CNN performed poorly when Google mobility data were used as inputs, as predictions were too sensitive to mobility changes. Finally, the TCN was applied to both cumulative and daily confirmed cases forecasting with worldwide COVID-19 and Google mobility data (Dataset #4). The different experiments are summarized in Table 2. Performances were measured with the normalized root mean-squared-error (NRMSE) and the adjusted R-squared as explained in (3), (4) and (5):

$$MSE(\hat{y}, y) = \sum_{i=1}^n \frac{(\hat{y}_i - y_i)^2}{n} \tag{3}$$

$$RMSE(\hat{y}, y) = \sqrt{MSE(\hat{y}, y)} \tag{4}$$

$$NRMSE(\hat{y}, y) = \frac{RMSE(\hat{y}, y)}{Mean(y)} \tag{5}$$

The TCN transfer learning scheme was evaluated by comparing the sum of regional predictions (which accounts for the whole country) to country observations. The average regional NRMSE and the weighted average regional NRMSE were calculated to evaluate the improvement of regional forecast performances. Section 5 presents a dedicated analysis for each model described in Table 2.

5 Results and analysis

This section describes the main results of this paper. A summary of the results is first presented, and a detailed description for each indicator is presented later.

As presented before, our first proposed model for COVID-19 was applied to confirmed cases forecasting.

Table 3 Performances for the confirmed cases modelling scenarios

| Evaluation | Optimistic | Compromise | Pessimistic |
|------------------|------------|------------|-------------|
| First prediction | 45,873 | 45,570 | 44,882 |
| Last prediction | 98,830 | 110,703 | 290,864 |
| NRMSE | 0.0986 | 0.1257 | 0.99 |
| R ² | 0.85 | 0.88 | 0.93 |

This first model is a classical CNN designed to have a low number of parameters to avoid over-fitting. It was trained on worldwide data (Dataset #1) for cumulative confirmed cases modeling with optimistic, compromise, and pessimistic scenarios. Results show that confirmed cases evolution followed a different trend compared to projection. Observed data were below the lower predictions from May 8th to June 10th and followed a linear increase. A limitation of this approach is that data filtering must be updated before each training to produce coherent scenarios.

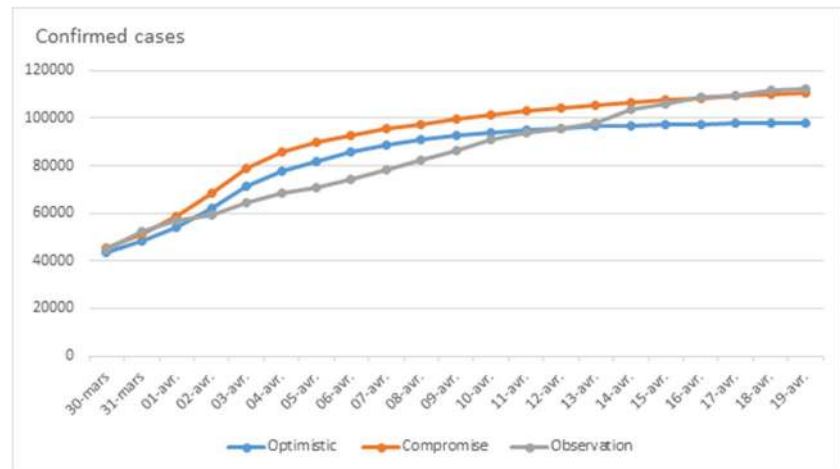
The same model was retrained later when more data were available, providing daily confirmed cases forecasts. Only decreasing sequences were kept during training to produce an optimistic scenario corresponding to an extended lockdown. Results show that the model was too sensitive to extreme values, which produce high daily increase predictions. The model was indeed too optimistic, predicting a stop of the daily increase by July 2020. Instead, observed data from the end of the lockdown to early July show a linear increase in the cumulative number of confirmed cases.

The number of confirmed cases is an indicator that must be carefully evaluated because, while it reflects the spread of the virus in the population, it does not reflect the actual burden caused by the virus on hospitals. Indeed, a massive testing campaign can increase the number of confirmed

Table 2 Summary of the 6 modelling experiments

| Model | Target | Dataset | Samples | Training period | Forecast period |
|--|---|---------|-------------------|--|---|
| Optimistic, compromise and pessimistic scenarios | Total Confirmed cases | #1 | 366 / 2745 / 2379 | January 22 nd to March 29 th | March 30 th to April 19 th |
| CNN with dropout | Daily Confirmed cases | #1 | 332 | January 22 th to May 7 th | May 8 th to July 6 th |
| CNN with dropout | Hospitalizations | #2 | 1676 | March 18 th to May 10 th | May 11 th to May 24 th |
| TCNs with quantile regression | One model per target (Hospitalizations, artificial ventilation, recoveries, deaths) One multi-task model with the 4 targets | #3 | 5170 | March 18 th to June 5 th | June 6 th to June 30 th |
| Regional models | Same as TCNs | #3 | 5170 | March 18 th to June 5 th | June 6 th to June 30 th |
| Worldwide models | Daily or cumulative confirmed cases | #4 | 18426 | February 15 th August 6 th | August 7 th to September 9 th |

Fig. 11 Optimistic and compromise predictions from March 30th to April 19th compared to observed data



cases by detecting asymptomatic infected people who do not overload the healthcare system. Instead, COVID-19 hospitalizations is a better indicator because it describes the actual impact of the virus. This indicator is crucial to manage the crisis, as its forecast is directly related to decisions on the number of available beds and medical staff workers. It is also important to decide when a patient must be transferred from a heavily impacted region to a less impacted region with available beds.

The classical CNN for confirmed cases forecasting was then applied to hospitalizations forecasting. The training was performed using French data at departmental, regional and national levels (Dataset #2). Subsequent training campaigns were performed every week, and projections were communicated to the French Ministry of Health. A baseline monitoring model was created using data available up-to May 11th as training data. Again, we adopted an optimistic scenario corresponding to an extended lockdown, and projections were compared to observed data every week. This way, we could observe that the decrease in hospitalizations was lower than expected from mid-May to mid-July, which can be explained by the slow-down of

hospitals' patient recoveries (fewer people are getting out of the hospitals). These results have shown that modeling can still be performed despite the small size of the datasets.

Despite these results, the classical CNN model suffers from many limitations. Performances at the regional and departmental levels were not satisfying, and the model was too limited to exploit dynamical data such as Google Mobility Reports. Therefore this work was extended using a modified Temporal Convolutional Network (TCN) with mobility data (Dataset #3).

The TCN model was successfully applied to five COVID-19 indicators: confirmed cases, hospitalizations, artificial ventilation, recoveries, and deaths. Besides, the TCN was successful at different scales: regional predictions could be made from national data, and national predictions were performed from worldwide data.

TCN was first applied to hospitalizations predictions, achieving good performances at the national level, although regional level performances were insufficient. Regional forecasts were later improved with a hierarchical pre-training scheme, as presented in Section 4. Finally, the TCN was used to confirmed cases forecasting, successfully

Fig. 12 Predictions from May 8th to July 6th compared to observed data

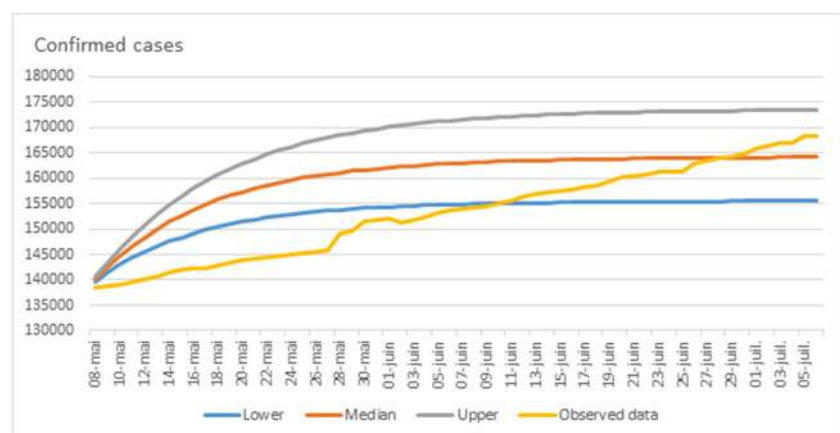


Table 4 Performances for confirmed cases forecasting

| Evaluation | Lower | Median | Higher |
|------------------|---------|---------|---------|
| First prediction | 139,642 | 140,155 | 140,579 |
| Last prediction | 155,496 | 161,134 | 173,438 |
| NRMSE | 0.041 | 0.057 | 0.099 |
| R^2 | 0.49 | 0.53 | 0.58 |

Table 5 Performances for hospitalizations forecasting

| Evaluation | Lower | Median | Higher |
|------------------|--------|--------|--------|
| First prediction | 20,967 | 21,554 | 22,115 |
| Last prediction | 7,149 | 18,597 | 20,292 |
| NRMSE | 0.032 | 0.06 | 0.12 |
| R^2 | 0.99 | 0.99 | 0.96 |

modeling both cumulative and daily confirmed cases (Dataset #4).

The next sections present the detailed analysis for each modeling approach.

5.1 Results for CNN-based COVID-19 modeling

5.1.1 Scenarios for cumulative confirmed cases forecasting

The results presented in this section were obtained with the first classical CNN proposed earlier. The main objective of these scenarios is to model the evolution of the epidemic with the reduced worldwide data available at the time. From March 30th to April 19th the number of confirmed cases grew from 44,550 to 112,606 with an average of 3,403 daily increase. The daily increase peak was achieved on March 31st with 7,578 new cases. Table 3 contains the performance values for the three scenarios. The optimistic scenario predicted an average of 2,708 daily increase for the same period with a peak on April 3rd with 8,889 new cases. The predicted total cases grew slowly to 98,000 on April 19th with less than 200 daily new cases. The prediction error rate is equal to 9.86% NRMSE for the entire period and 12.94% for the last day of the period. Predictions were optimistic as intended, with a final prediction lesser

than 14,000 than the observed total cases. The model was optimistic by predicting a slow down to 218 daily increase for the last five days of the periods against 1,600 observed daily increase. The model also predicted a higher peak with a delay of two days.

The compromise scenario is similar to the optimistic model, with a predicted peak of 9,467 new cases on April 3rd but with a slighter decrease of new daily cases with an average of 811 for the five last days of the period. Predictions grew from 45,570 predicted cases on March 30th and 110,703 on April 19th. The prediction error rate is higher than the optimistic scenario with 12.57% NRMSE but was low for the last day of the period with only a 1.7% error rate. This scenario predicted an average daily increase of 3,257 new cases for the whole period, close to the observed daily increased (3,403 cases). Predictions for optimistic and compromise scenarios are compared to the real observations in Fig. 11 (the pessimistic scenario was omitted for visibility purpose). The pessimistic scenario was different, with a continuous predicted increase in new daily cases. Predictions grew from 44,882 predicted cases on March 30th to 290,864 on April 19th.

Overall, the three scenarios successfully described the optimistic, compromise, and pessimistic evolution of the epidemic for the considered period. Optimistic and compromise scenarios were close to observations, with the

Fig. 13 Predictions from May 11th to May 24th compared to observed data

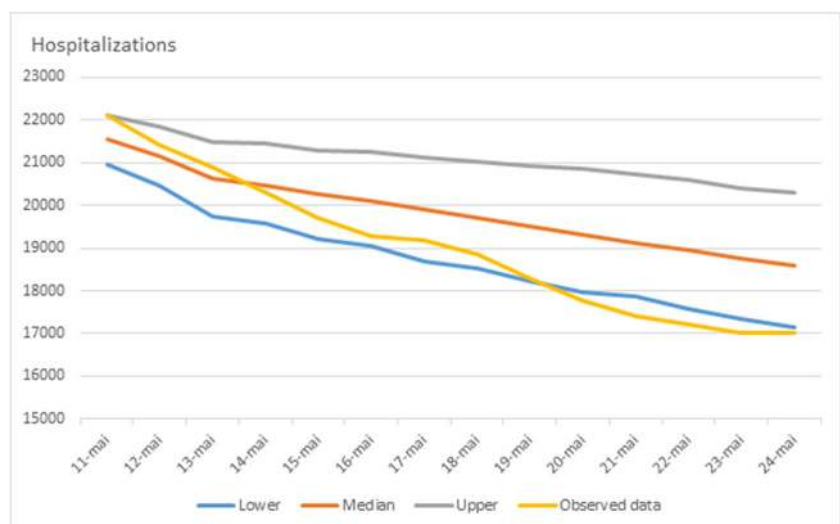
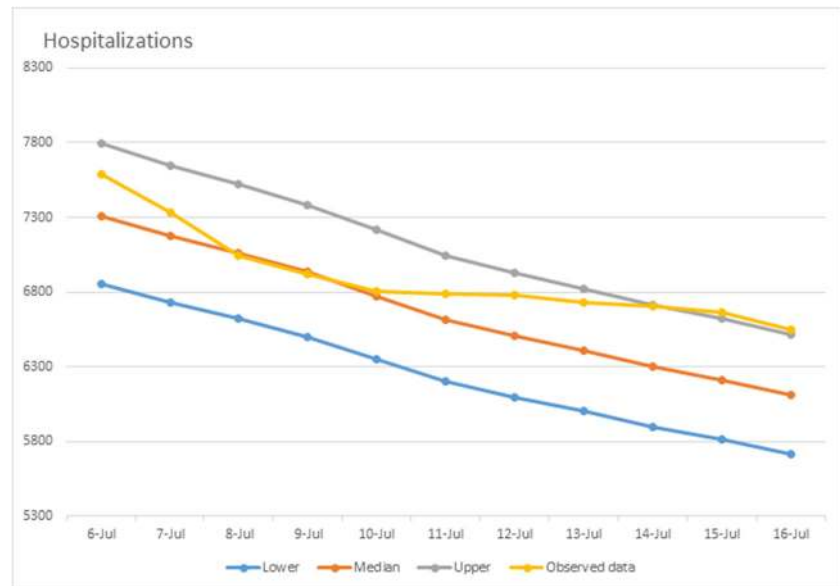


Fig. 14 Predictions from July 6th to July 16th compared to observed data



main differences being a higher predicted peak and a higher decrease in new daily cases. This can be seen in Fig. 11 with observed data below both scenarios predictions until April 12th.

5.1.2 CNN for daily confirmed cases forecasting

Another approach to confirmed daily cases modeling was carried out with the second version of the CNN described earlier. This approach was conducted when more data were available, and data from every country were used. Predictions and observations are shown in Fig. 12 and performances are shown in Table 4. Dropout was activated at inference time to create 95% confidence intervals. The predicted total confirmed cases grew from about 140,000 cases on May 8th to 155,496 for lower predictions, 164,134 for median predictions, and 173,438 for higher predictions on July 6th. In the same period, the observed total number of confirmed cases grew from 138,421 to 168,335. However, observations and predictions have different slopes. The model predicted a daily increase much higher, over 1,800 new cases, that would decrease to a few dozen cases, while observation shows a stagnation with 509 daily confirmed cases on average. This much higher predicted daily increase can be explained by the peak in confirmed cases observed

on May 6th as raw data were used. Overall, this model was too sensitive to extreme values, and it was overly optimistic by predicting a stop of the virus spread by July 2020.

5.1.3 CNN for hospitalizations forecasting

The second version of the proposed CNN was also trained to predict the number of hospitalization cases in France at the country and regional levels. Predictions and observations are shown in Fig. 13 and Table 5 contains the performances of the predictions. From May 11th to May 24th, the number of observed hospitalizations in France decreased from 22,115 to 17,021, with an average of 140 daily decreases in the number of hospitalizations. Lower predictions decrease from 20,966 to 17,149, median predictions from 21,554 to 18,587 and higher predictions decrease from 22,115 to 20,292. Observations were inside the confidence intervals for 12 of the 13 regions. Observations for the whole country were close to the lower predictions with 3.2% NRMSE. This model was applied every two weeks on the new data to compare predictions and observations and detect a potential new wave. Predictions were very close to observations at the country level but not for each region.

In most cases, predictions tend to be overly pessimistic, with a predicted stagnation or increase for some regions. As

Table 6 Performances for hospitalizations predictions with TCNs

| Hospitalization models | Training RMSE | Valid. RMSE | Valid. R^2 |
|-----------------------------|---------------|-------------|--------------|
| One-target - country | 0.02 | 0.01 | 0.99 |
| One-target - sum of regions | 0.03 | 0.07 | 0.95 |
| Multi-task - country | 0.02 | 0.04 | 0.98 |
| Multi-task - sum of regions | 0.02 | 0.04 | 0.88 |

Fig. 15 Hospitalizations predictions from June 6th to June 30th compared to observed data

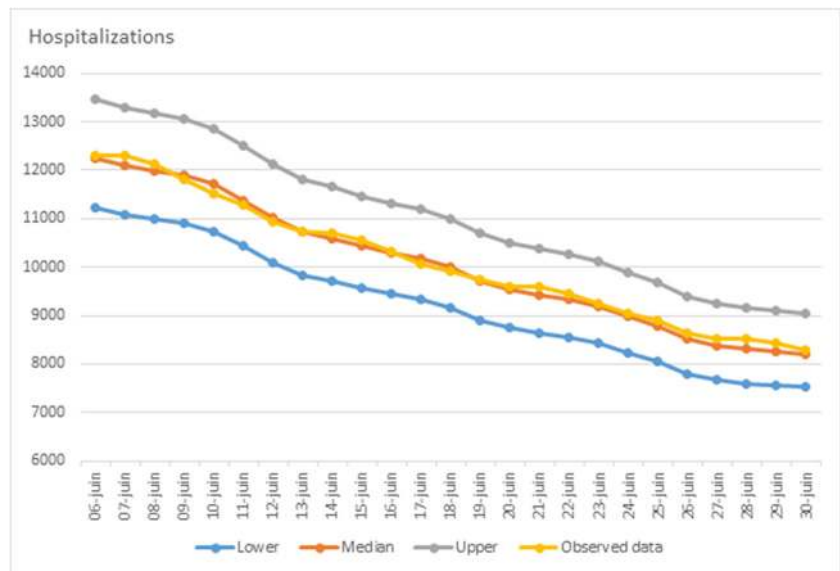


Fig. 16 Artificial Ventilations predictions from June 6th to June 30th compared to observed data

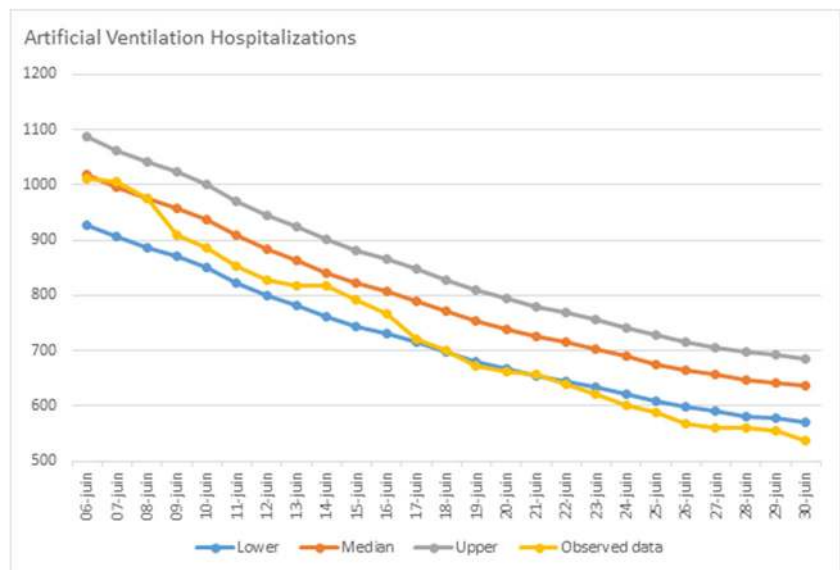


Table 7 Performances for Artificial Ventilations hospitalizations predictions with TCNs

| Artificial vent. models | Training RMSE | Valid. RMSE | Valid. R^2 |
|-----------------------------|---------------|-------------|--------------|
| One-target - country | 0.03 | 0.09 | 0.98 |
| One-target - sum of regions | 0.03 | 0.05 | 0.94 |
| Multi-task - country | 0.06 | 0.41 | 0.96 |
| Multi-task - sum of regions | 0.05 | 0.4 | 0.94 |

Table 8 Performances for recovery predictions with TCNs

| Recovery models | Training RMSE | Valid. RMSE | Valid. R^2 |
|-----------------------------|---------------|-------------|--------------|
| One-target - country | 0.01 | 0.03 | 0.98 |
| One-target - sum of regions | 0.01 | 0.02 | 0.98 |
| Multi-task - country | 0.01 | 0.01 | 0.95 |
| Multi-task - sum of regions | 0.01 | 0.01 | 0.99 |

Table 9 Performances for deaths predictions with TCNs

| Death models | Training RMSE | Valid. RMSE | Valid. R^2 |
|-----------------------------|---------------|-------------|--------------|
| One-target - country | 0.01 | 0.05 | 0.95 |
| One-target - sum of regions | 0.03 | 0.04 | 0.94 |
| Multi-task - country | 0.01 | 0.01 | 0.95 |
| Multi-task - sum of regions | 0.01 | 0.01 | 0.50 |

we can see in Fig. 13, observed hospitalizations from May 11th to May 24th were close to the lower predictions because the virus spreading was stopped and new hospitalizations were low. These observations were closer to the upper predictions from July 6th to July 16th as shown in Fig. 14. A slow-down in the number of recoveries (fewer people are leaving the hospitals), while new hospitalizations were still observed, can explain such a situation.

5.2 Results for TCN-based COVID-19 modeling

5.2.1 TCNs for COVID-19 hospital indicators forecasting

The TCN with quantile regression described in the previous section is implemented and trained to predict the number of hospitalizations, hospitalizations with artificial ventilation, recoveries, and deaths. These models are compared to the multi-task model that predicts the four targets, and the NRMSE is calculated for the validation set between median predictions and observations. Country-level performances were compared to regional model performances by calculating the sum of regional predictions that should be close to the national level. With TCNs, the Hospitalization model achieved good accuracy at the country level with 1.2% NRMSE and 0.99 R^2 . The sum of

regional prediction accurate well with 6.8% NRMSE and 0.98 R^2 , and observed data were above median predictions and below higher predictions. For both regional and country predictions, the multi-task model creates wider intervals with an upper limit than stagnate (64 daily decreases against 177 for the higher predictions at the country level). In both cases, the observed data are close to the median predictions with 4% RMSE but with lower R^2 . Performances are shown in Table 6, and country-level French predictions are plotted in Fig. 15.

Similar behavior can be observed with the Artificial Ventilation model, where observed data are close to the lower predictions with 5.6% NRMSE and 0.98 R^2 at the country level. Furthermore, observed data even went below the lower limit from June 19th to June 30th. As shown in Fig. 16 (country-level French predictions), observed data are close to the median predictions with 5.2% NRMSE and 0.96 R^2 at the regional level. The multi-task model also produced wider intervals that were expected because artificial ventilation hospitalization is part of the total number of hospitalizations. The multi-task model higher limit has a 2-daily decrease average compared to 16-daily decreases at the country level. The performances of the different models are detailed in Table 7.

Fig. 17 Recovery predictions from June 6th to June 30th compared to observed data

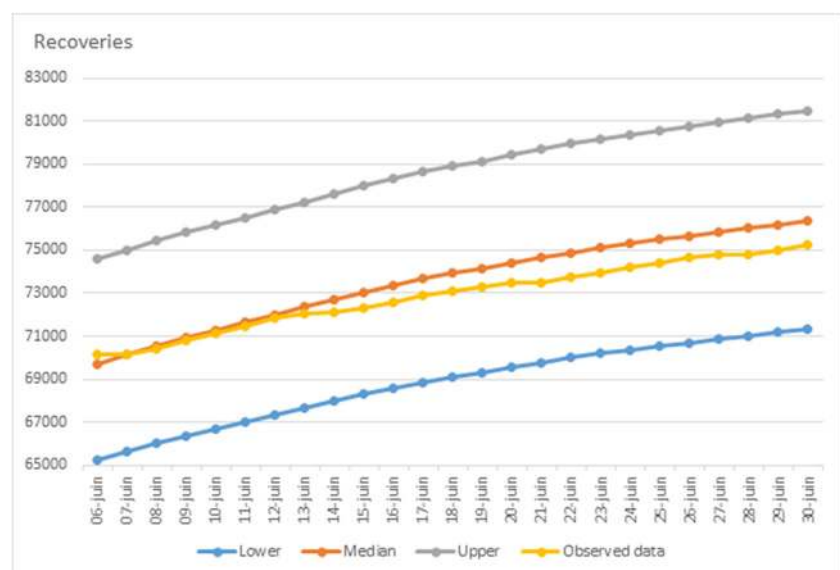
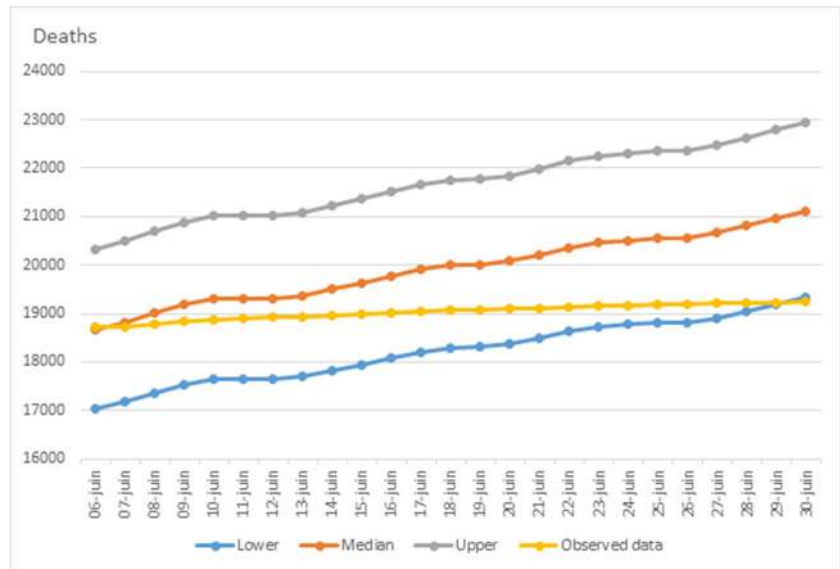


Fig. 18 Deaths predictions from June 6th to June 30th compared to observed data



The Recovery model also achieved good accuracy at both country and regional level with, respectively, 3.5% NRMSE / 0.98 R^2 and 2.3% NRMSE / 0.95 R^2 . The multi-task model improved the performances on both levels with respectively 1.2% NRMSE / 0.98 R^2 and 0.7% NRMSE / 0.99 R^2 . The intervals produced by the multi-task model were narrower with a 277 daily increase against 410 daily increase for higher predictions at the country level. It is coherent with hospitalization behavior because total recoveries are directly linked to the decrease in hospitalization. Performances are shown in Table 8, and country-level French predictions are plotted in Fig. 17.

In the case of the number of deaths, we observe that the error rate is low with, respectively, 5.5% and 4.3% NRMSE for country and regional levels. However, the increasing trend is overestimated with a predicted daily increase of 98 compared to the 21 daily increase observed during the validation period. The predicted increase is lower for

the multi-task models, but they produced anomalies with a small decreasing cycle. The prediction trends are still increasing, but there are decreasing values that seem to correspond to weekends. This is related to the way data are reported, as fewer deaths are reported on weekends, and they are added to the following weekdays. Performances are shown in Table 9, and country-level French predictions are plotted in Fig. 18.

Finally, we can say that good performance can be achieved at the country level on the four targets with individual models. The multi-task model can also achieve good performances, with broader or narrower intervals coherent with the task's high uncertainty. However, performances on regional data are not homogeneous among the regions. High accuracy can be achieved in the most impacted regions such as Ile-de-France (Paris region) or Grand-Est (north-east of France), while lower accuracy is still observed in smaller regions.

Fig. 19 Performances of the 4 models on hospitalizations predictions

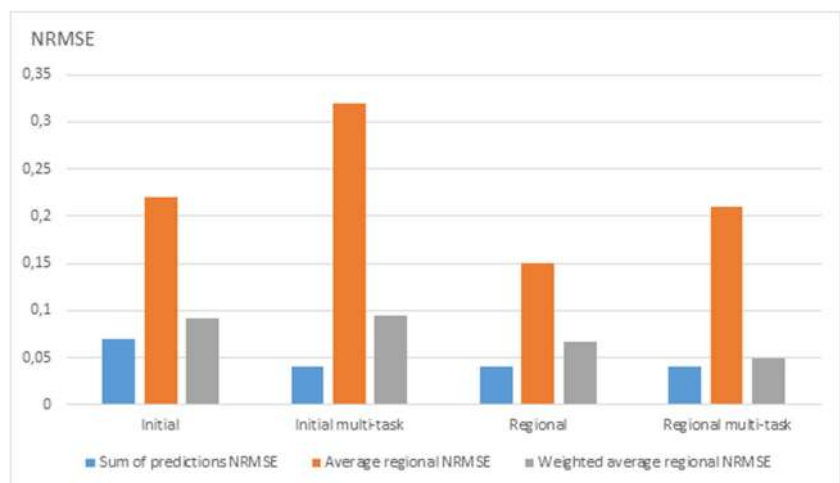


Fig. 20 Performances of the 4 models on artificial ventilation hospitalizations predictions

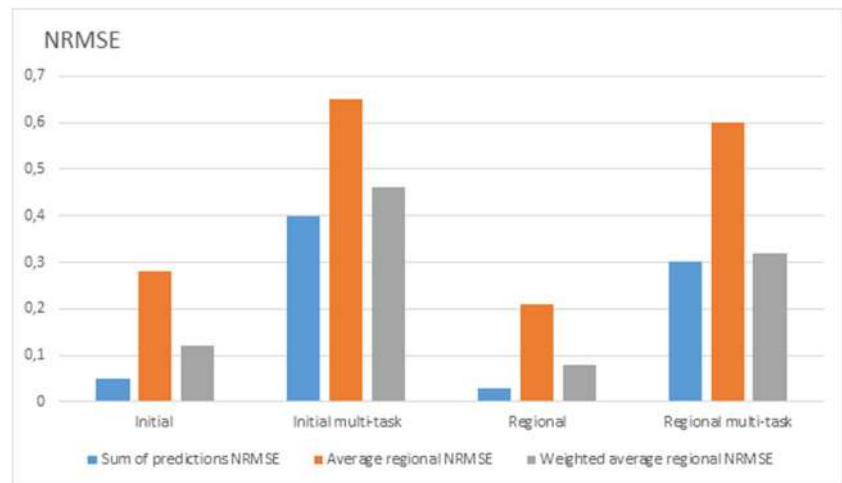


Fig. 21 Performances of the four models on recovery predictions

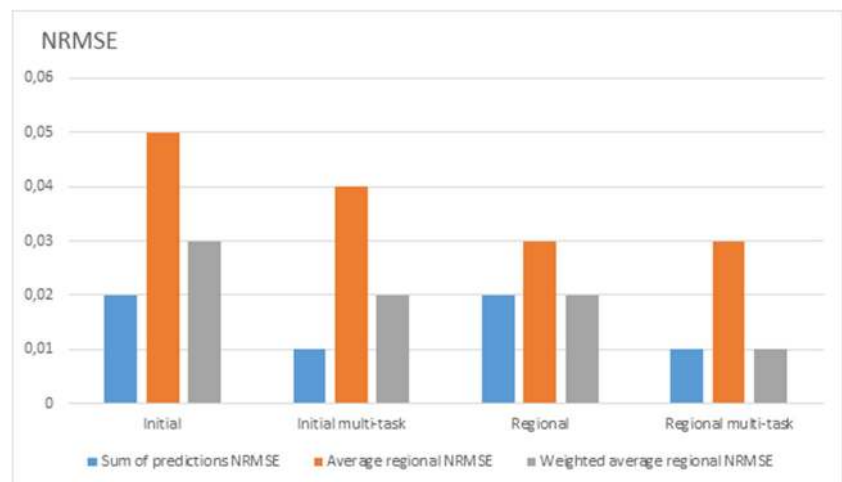


Fig. 22 Performances of the four models on deaths predictions

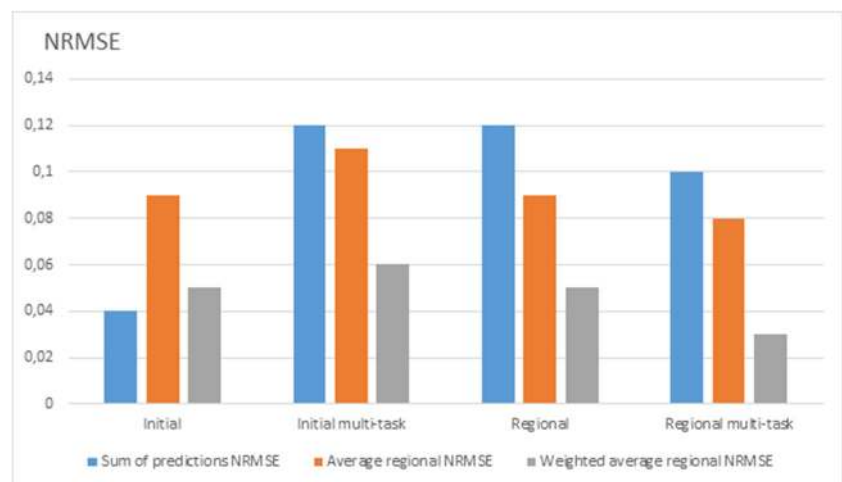


Table 10 Performances for the TCN trained on French data only

| Data | Training RMSE | Validation RMSE |
|-------------------------------|---------------|-----------------|
| France | 4.4% | 14.5% |
| France with data augmentation | 8.2% | 12.6% |

5.2.2 TCNs with transfer learning for regional forecasting

Regarding the hospitalization predictions, better performances were found with regional training with transfer learning with a weighted average of 6.7% NRMSE, an average of 15% NRMSE, and a 4% sum of predictions NRMSE against 9%, 22%, and 7% with the initial model, respectively. Performances are better for eight of the 13 regions and slightly worse for five regions. The initial multi-task model achieved worse performances with 9.5% weighted average NRMSE, 32% average NRMSE, but presented a better sum of prediction performances, reaching 4%. The regional multi-tasks models achieved the best weighted average performances with 4.7% but with an average RMSE similar to the initial model with 21.5%. The regional multi-task models improved performances in the region that contributed the most to total hospitalization accounts, while performances were worse in other regions. All these results are shown in Fig. 19.

Similar results with Artificial Ventilation predictions were found. Regional models achieved 8.2% weighted average NRMSE, 20.5% average NRMSE, and 3% sum of predictions NRMSE against 12.3% weighted average NRMSE, 28.5% average NRMSE, and 5% sum of predictions NRMSE. Performances were better on nine of the 13 regions, while four regions presented slightly worse results. Performances were worse on each region with both initial multi-task and regional multi-task models, as summarized in Fig 20.

Recoveries NRMSE were lower than 10% for 11 of the 13 regions on both initials and regional models. Performances were improved slightly by the multi-task regional models with 1.3% weighted average NRMSE, 2.7% average NRMSE, and 1% sum of predictions NRMSE against respectively 3.1%, 5.4%, and 2% for the initial model as shown in Fig. 21.

Table 11 Performances for the TCN trained on worldwide data for cumulative confirmed cases

| Data | Training RMSE | Validation RMSE |
|----------------------------|---------------|-----------------|
| Whole dataset (Dataset #4) | 4.1% | 2.7% |
| France | 1% | 1% |

The weighted average NRMSE is slightly lower for death predictions with multi-task regional models. However, decreasing predictions can be found with up to 5 values out of the 25 of the validation periods despite having no decreasing trend on the training set. The one-target regional models' performances are similar to the initial models, while the initial multi-task model shows slightly worse performances. While one-target initial and regional models show a high correlation with 0.94 R^2 , the predicted daily increase is off with more than 94 predicted daily increase compared to the observed 21 daily increase. All these results are condensed in Fig. 22.

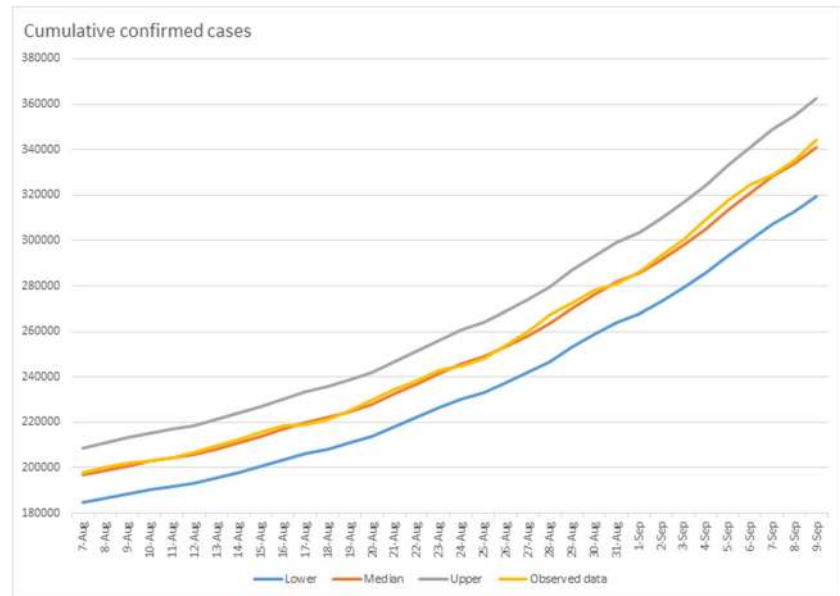
5.2.3 TCNs for confirmed cases forecasting

The TCN trained only on French data shows overfitting signs with 4.4% NRMSE on the training set and 14.5% NRMSE on the validation set. Also, this model's projections were incoherent because they were below the input sequences' values (confirmed cases can only increase or stay constant). The TCN trained with data augmentation had similar performances with 8.2% NRMSE on the training set and a slightly better validation NRMSE of 12.6%, but the projections remain incoherent. Performances for this model are shown in Table 10.

On the contrary, the TCN trained on worldwide data for cumulative confirmed cases forecasting achieved 4.1% NRMSE on the training set and 2.7% on the validation set. The performances were of 1% NRMSE for the training set and also 1% NRMSE for the validation in France. The TCN trained for daily confirmed cases forecasting achieved 69.5% NRMSE on the training set and 83.4% NRMSE on the validation set on daily sequences. Performances are shown in Table 11 and predicted values for the validation period are compared to observed data in Fig. 23. Performances for France were 65% on the training set and 29% on the validation set for daily sequences and 5% on both training and validation sets for cumulative sequences. Performances are shown in Table 12.

The high NRMSE on daily sequences for this last model was expected because the daily increase of confirmed cases is subject to noise and cyclical patterns related to the COVID-19 test centers' organization. However, the actual error on cumulative cases remains low, with 5% in France. Data augmentation did not improve the performances of those two models. The performances achieved by both TCN trained with worldwide data show that the proposed TCN can be applied successfully to different COVID-19 related modeling tasks like confirmed cases, hospitalizations, artificial ventilation, recoveries, and deaths forecasting. The TCN achieved good performances on both national modeling (confirmed cases forecasting with worldwide data) and regional modeling (hospitalizations forecasting

Fig. 23 Predictions and observed data from August 6th to September 9th for confirmed cases forecasting



with French data). Finally, the proposed TCN overcame the limitations of the classical CNN proposed in Section 4.1.

6 Discussions and future works

The results presented in the previous section show that CNN and TCN can produce accurate projections of multiple COVID-19 indicators. Best performances for confirmed cases and hospitalizations of the proposed CNN and TCN are compared in Fig. 24. TCN seems to achieve good accuracy, but this comparison is limited because the error rates were calculated for different periods (TCN also benefited from more data). Overall, TCN can achieve a 1% error rate or even less in the best case and continued to present good results. For instance, confirmed cases from September 11th to September 24th have been computed, as seen in Fig. 25, demonstrate high accuracy, with only 0.5% NRSME.

If these results comfort our choices, the algorithm choices must also be observed in the context of the COVID-19 outbreak. Since March 2020, multiple data-driven methods have been applied, with more or less success. In the specific case of French COVID-19 data, five main methods

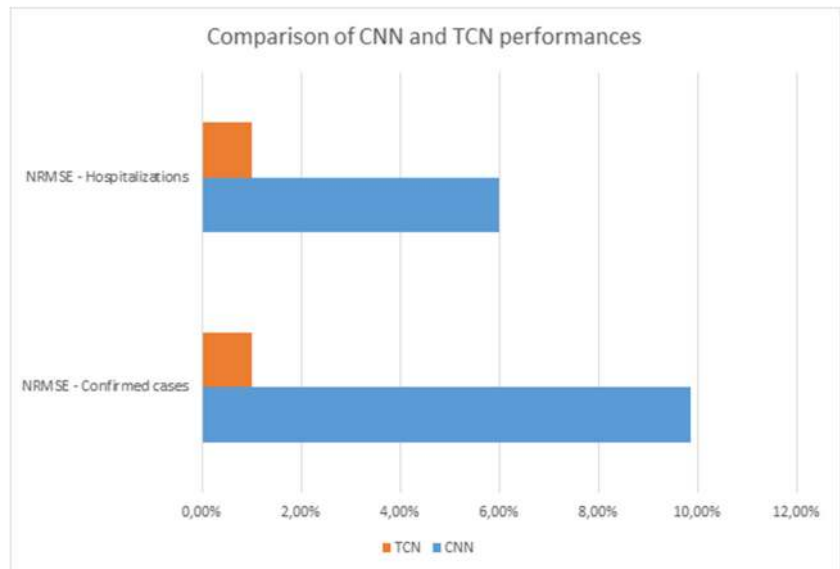
were explored to forecast confirmed cases during the first epidemic wave. Table 13 illustrates the principal results from these works (to perform a fair comparison, reported metrics such as RMSE were normalized). One of the main methods used by the authors is ARIMA, a popular time-series analysis algorithm. It was applied to French data by several authors such as [8, 32, 46]. Table 13 indicates that these works present an NRMSE of 23.5%, 7.1%, and 2.9%, respectively, which can be compared to the NRMSE of 9.9% obtained with the optimistic CNN method we employed. This difference in performances between authors can be explained by the duration of the validation period. [32] achieved 2.9% with only seven days of validations, while [46] achieved 23.5% over 22 days. Also, only the training RMSE was reported by [8]. The performances achieved by the optimistic CNN were obtained with 20 days of validation, with less than half of the NRMSE reported by [46].

Another popular method is Least-square Support Vector Machine (LS-SVM), which was compared to ARIMA by [46]. Training NRMSE of less than 1% and validation NRMSE of 10% were reported, showing signs of overfitting. It is very close to the optimistic CNN. Nonlinear Autoregression Neural Network (NARNN) and LSTM were also compared to ARIMA by [32]. They achieved 1% and 0.3% NRMSE on a 7-days validation period, respectively. While both of these methods gave better results than the CNN method used in our paper, they are similar to the results we obtained with the TCN method, which gives about 1% and 0.35% NRMSE for 30 and 14 days forecasts. The longer validation duration shows that the proposed TCN is competitive compared to other existing methods. It is also worth noting that the proposed TCN achieved

Table 12 Performances for the TCN trained on worldwide data for daily confirmed cases

| Data | Training RMSE | Validation RMSE |
|----------------------------|---------------|-----------------|
| Whole dataset (Dataset #4) | 70% | 83% |
| France (daily) | 65% | 29% |
| France (cumulative) | 5% | 5% |

Fig. 24 Comparison of CNN and TCN error rates on confirmed cases and hospitalizations



good performances on hospitalization predictions. Indeed, 1.2% NRMSE was reported from June 6th to June 30th, during the drop in cumulative hospitalizations after the first lockdown. Later, the TCN was retrained with data available up to November 8th. An NRMSE of 2.49% was reported for a 28 days forecast (from November 9th to December 6th). Predictions and observed data are shown in Fig. 26. This result shows that the proposed TCN can provide good forecasts with different dynamics.

Another modeling method based on Genetic Expression Programming (GEP) was proposed by [44]. It has the

advantage of giving an interpretable formula for different COVID-19 prediction tasks (confirmed cases and deaths, for example). This is important because confirmed case modeling is not easy to model, as confirmed cases are too dependent on the testing strategy. Hence, only a few thousand tests per day were performed during the first wave (from March to May) in France. As part of the new public health strategies after the first wave, this number has grown to at least 200,000 tests since July 2020.

Consequently, the formula calculated by [44] is no longer valid as the testing strategy has changed. In comparison,

Fig. 25 Comparison of predicted confirmed cases and observed data from September 11th to September 24th in France

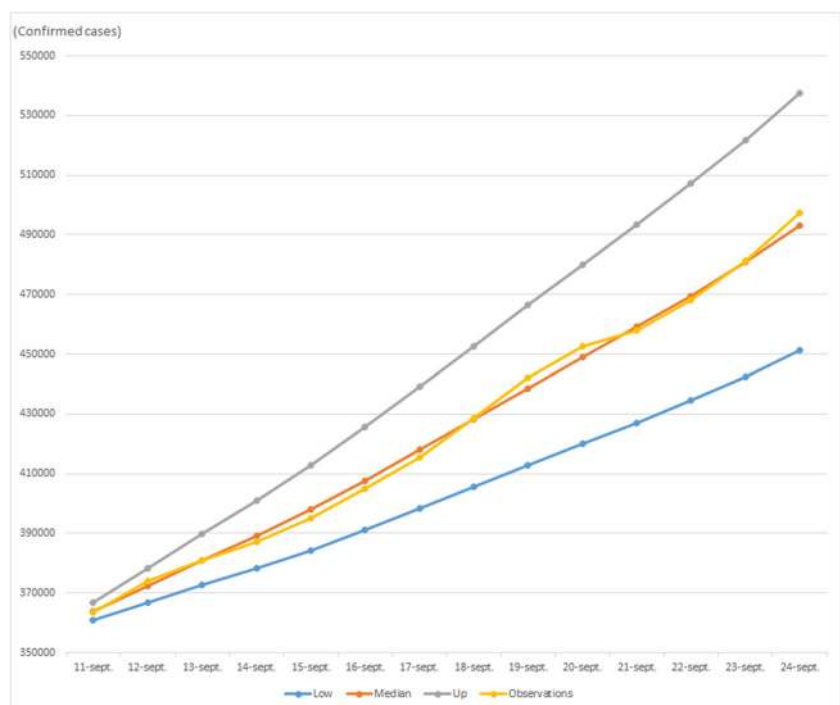


Table 13 Performances of the discussed methods(confirmed cases)

| Authors | Methods | Validation period | NRMSE |
|---------------------------|-----------------|--|-------|
| Singh et al [46] | ARIMA | 22 days(April/May) | 24% |
| Singh et al [46] | LS-SVM | 22 days(April/May) | 10% |
| Chakraborty and Ghosh [8] | ARIMA + Wavelet | Training accuracy only(January to April) | 7.1% |
| Salgotra et al. [44] | GEP | 7 days(June) | 0.6% |
| Salgotra et al. [44] | GEP | 30 days(June) | 0.6% |
| Salgotra et al. [44] | GEP | 32 days(August to September) | 24% |
| Kırbaş et al [32] | ARIMA | 7 days(April/May) | 3% |
| Kırbaş et al [32] | NARNN | 7 days(April/May) | 1% |
| Kırbaş et al [32] | LSTM | 7 days(April/May) | 0.35% |

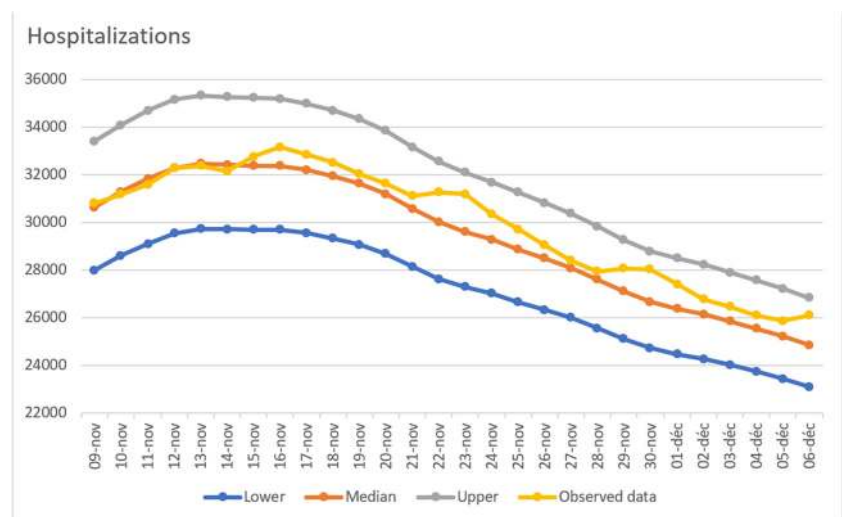
the proposed TCN uses worldwide data to overcome this limitation. The published formula was able to predict confirmed cases, starting from May 29th, with 0.6% error rate at both 7 days and 30 days forecast. However, the same formula applied from August 6th to September 9th achieved 24% error rate (compared to 1% with TCN).

Besides the modeling method, another legitimate question often overlooked is the training time/cost threshold. Compared to other artificial intelligence applications, the datasets are relatively small (even after one year). Both CNN and TCN models are sufficiently simple to allow fast training in consumer-grade hardware. For instance, if the TCN model's training can be achieved in about 20 minutes with the help of a high-end Nvidia Tesla V100 GPU, it can also be performed in a reasonable time with a mid-range GPU, like the Nvidia GTX 1050. The training cost problem has a more significant impact on the proposed transfer learning scheme for regional modeling. Each region needs to be trained on different data, and a single region dataset is too small to benefit from GPU acceleration. A simple technique

to reduce the training time is multi-threaded CPU parallelization. Sequential training takes about 6 hours, while parallelized training requires about 45 minutes on two Intel Xeon E5-2698 v4 (20 cores each). This is an expensive solution if we consider the material cost (about 6500\$ for the two CPUs), but the use of non-dedicated hardware such as a cloud computing instance may considerably reduce this cost.

As future work, we aim at overcoming some limitations of the current models. For example, the current TCN model uses constant mobility data for conditional forecasting. Ideally, mobility data would be selected to create optimistic and pessimistic scenarios by setting mobility constraints in the model's conditional input. One problem is the availability of dynamic data. Many factors have an impact on the evolution of the epidemic and cannot be monitored in real-time. Those factors include the effectiveness of sanitary measures. People's behavior has changed since the first outbreak in March-April. Masks are now mandatory in closed spaces, telecommuting is recommended, bars

Fig. 26 Predictions and observed data from November 9th to December 6th for hospitalizations forecasting



and restaurants have been kept closed since November 2020, etc. Another necessary dynamical data is regional mobility. It could be used to create projection maps of the virus spreading. An approach using ConvLSTM to generate disease spreading maps have been proposed by [41], but they did not use mobility data.

The modeling tool GleanViz uses mobility data like commuting networks and air travel, but they use compartmental models only [7]. This tool is also designed for a-posteriori analysis or scenario explorations. We believe that a hybrid solution can be designed where, for example, compartmental models could be used to generate artificial datasets. This synthetic data would include many scenarios based on different sanitary constraints (an idea proposed by [5]). Another hybrid solution would be to use deep learning to estimate a compartmental model's parameters, as proposed by [16] to model the US epidemic. This approach reached good results, predicting that the number of confirmed cases would reach 5 million on August 7th, a number that was finally attained on August 5th.

Another development front is related to the multi-task model, which did not improve performances and show signs of over-fitting on two targets (Artificial ventilation hospitalizations and deaths). We believe that this naive multi-task model with hard-shared parameters can be improved with soft-shared parameters and regularization techniques. Constraints would also be necessary to ensure that the predicted values are coherent (e.g., the number of death cannot decrease, or the artificial ventilation hospitalizations should be less than the total hospitalizations). Finally, it is important to ensure that sufficient data is available to feed data-driven models. This implies that data-driven models are less adapted to predict an epidemic's first wave unless automatic health data collection systems are implemented at the regional or the establishment level. Such observation should be debated as automatic health data collection poses many security and ethical concerns, a subject under strict regulation in France.

7 Conclusion

In this paper, multiple data-driven models were proposed for COVID-19 forecasting in France. The proposed TCN can achieve a 1% error rate at the national level for confirmed cases and hospitalizations predictions (compared to 9 and 5% for the proposed CNN). Competitive performances were also achieved on artificial ventilation hospitalizations, recoveries, and deaths predictions with respectively 9%, 3.5%, and 4.5% error rates. The proposed transfer-learning scheme was able to improve accuracy in 8 regions. Several challenges had to be faced when developing these models. The first challenge was the lack of data at the

COVID-19 outbreak. Hence, one of our contributions was designing a model based on Convolutional Neural Networks and capable of providing short-term confirmed cases and hospitalizations forecasts, even when little data is available.

Another challenge was the need for more representative metrics on the advance of the COVID-19 epidemic. Hence, we trained a Temporal Convolutional Network and achieved good accuracy forecasts for indicators such as confirmed cases, hospitalization, artificial ventilation hospitalization, and recoveries.

We also had to deal with different granularity levels for our forecasts, as we had to provide both national scale forecasts and regional/departmental forecasts. Indeed, the French management of COVID-19 is mainly performed at a regional scale, with national-level coordination for resource deployment and patient transfers when needed. We achieved good national and regional accuracy with improved performances by using a hierarchical transfer learning scheme presented in this work.

Our models are now integrated into a COVID-19 monitoring and forecast dashboard developed for the Grand-Est region within the PrédicEst project. Upon request, the model can be accessed by other researchers through a REST API.

Acknowledgements This work was partially supported by the AI4DI H2020 project³. The authors would like to thank Cyril Voisard for his help during the data curation process and his feedback on the models' performances. The authors would also like to thank Frédéric Mauguère and Arnaud Renard for the development of the REST API used to query our models.

We also would like to thank the ROMEO Computing Center of the University of Reims Champagne Ardenne, where part of the computation (dataset creation and model training) was performed. The Nvidia DGX-1 server allowed us to accelerate the training steps and compare several model approaches.

References

1. Alin F, Chemchem A, Nolot F, Flauzac O, Krajecki M (2020) Towards a hierarchical deep learning approach for intrusion detection. In: Boumerdassi S., Renault É., Mühlethaler P. (eds) Machine Learning for Networking. Springer International Publishing, Cham, pp 15–27
2. Amyar A, Modzelewski R, Li H, Ruan S (2020) Multi-task deep learning based ct imaging analysis for covid-19 pneumonia: Classification and segmentation. *Comput Biol Med* 126:104037. <https://doi.org/10.1016/j.combiomed.2020.104037>, <http://www.sciencedirect.com/science/article/pii/S0010482520303681>
3. Apostolopoulos ID, Mpesiana TA (2020) Covid-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks. *Phys Eng Sci Med* 43:635–640. <https://doi.org/10.1007/s13246-020-00865-4>
4. Bai S, Kolter JZ, Koltun V. (2018) An empirical evaluation of generic convolutional and recurrent networks for sequence modeling

³<https://ai4di.automotive.oth-aw.de/>

5. Behrens T (2018) Deep learning and diseases. <https://ai-guru.de/deep-learning-and-diseases/>
6. Borovykh A, Bohte S, Oosterlee CW (2018) Conditional time series forecasting with convolutional neural networks. arXiv: [1703.04691](https://arxiv.org/abs/1703.04691) [stat]
7. Broeck WVd, Gioannini C, Gonçalves B, Quaggiotto M, Colizza V, Vespignani A (2011) The gleamviz computational tool, a publicly available software to explore realistic epidemic spreading scenarios at the global scale. *BMC Infect Dis* 11:37. <https://doi.org/10.1186/1471-2334-11-37>
8. Chakraborty T, Ghosh I (2020) Real-time forecasts and risk assessment of novel coronavirus (covid-19) cases: A data-driven analysis. *Chaos Solitons Fractals* 109850:135. <https://doi.org/10.1016/j.chaos.2020.109850>
9. Chemchem A, Alin F, Krajecki M (2019) Combining smote sampling and machine learning for forecasting wheat yields in france. In: 2019 IEEE second international conference on artificial intelligence and knowledge engineering (AIKE). pp 9–14
10. Chen Y, Kang Y, Chen Y, Wang Z (2020) Probabilistic forecasting with temporal convolutional neural network. arXiv: [1906.04397](https://arxiv.org/abs/1906.04397) [cs stat]
11. Chimmula VKR, Zhang L (2020) Time series forecasting of covid-19 transmission in Canada using lstm networks. *Chaos Solitons Fractals* 135:109864. <https://doi.org/10.1016/j.chaos.2020.109864>
12. Cho K, van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y (2014) Learning phrase representations using rnn encoder–decoder for statistical machine translation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). Association for Computational Linguistics, pp 1724–1734. <https://doi.org/10.3115/v1/D14-1179>
13. Chung J, Gulcehre C, Cho K, Bengio Y (2014) Empirical evaluation of gated recurrent neural networks on sequence modeling. In: NIPS 2014 workshop on deep learning, December 2014. <https://nyuscholars.nyu.edu/en/publications/empirical-evaluation-of-gated-recurrent-neural-networks-on-sequen>
14. Cun YL, Boser B, Denker JS, Howard RE, Hubbard W, Jackel LD, Henderson D (1990) Handwritten digit recognition with a back-propagation network. Morgan Kaufmann Publishers Inc., San Francisco, pp 396–404
15. DECOV: Decov gitlab repository. https://gitlab.com/covid_dia/deconf
16. Deng Q (2020) Dynamics and development of the covid-19 epidemic in the united states: a compartmental model enhanced with deep learning techniques. *J Med Internet Res* 22(8). <https://doi.org/10.2196/21173>
17. Dong E, Du H, Gardner L (2020) An interactive web-based dashboard to track covid-19 in real time. *Lancet Infect Dis* 20(5):533–534. [https://doi.org/10.1016/S1473-3099\(20\)30120-1](https://doi.org/10.1016/S1473-3099(20)30120-1)
18. Fanelli D, Piazza F (2020) Analysis and forecast of COVID-19 spreading in China, Italy and France. *Chaos Solitons Fractals* 109761:134. <https://doi.org/10.1016/j.chaos.2020.109761>
19. Faranda D, Alberti T (2020) Modelling the second wave of covid-19 infections in France and Italy via a stochastic SEIR model
20. France, SP. Covid-19 - dashboard - suivi de l'épidémie de covid-19 en france. <https://dashboard.covid19.data.gouv.fr>
21. France, SP. Données hospitalières relatives à l'épidémie de covid-19. <https://www.data.gouv.fr/fr/datasets/donnees-hospitalieres-relatives-a-lepidemie-de-covid-19/>
22. Gal Y, Ghahramani Z (2016) Dropout as a bayesian approximation: Representing model uncertainty in deep learning. arXiv: [1506.02142](https://arxiv.org/abs/1506.02142) [cs stat]
23. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: 2016 IEEE conference on computer vision and pattern recognition (CVPR). pp 770–778
24. Hochreiter S, Schmidhuber J (1997) Long short-term memory. *Neural Comput* 9(8):1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>
25. Holmdahl I, Buckee C (2020) Wrong but useful — what covid-19 epidemiologic models can and cannot tell us. *N Engl J Med* 383(4):303–305. <https://doi.org/10.1056/NEJMp2016822>
26. Huang CJ, Chen YH, Ma Y, Kuo PH (2020) Multiple-input deep convolutional neural network model for covid-19 forecasting in china. medRxiv p. 2020.03.23.20041608. <https://doi.org/10.1101/2020.03.23.20041608>
27. Inc., G. Covid-19 community mobility report. <https://www.google.com/covid19/mobility>
28. Kingma DP, Ba J (2017) Adam: A method for stochastic optimization arXiv: [1412.6980](https://arxiv.org/abs/1412.6980) [cs]
29. Koenker R, Bassett G (1978) Regression quantiles. *Econometrica* 46(1):33–50. <https://doi.org/10.2307/1913643>
30. Kok C, Jahmunah V, Oh SL, Zhou X, Gururajan R, Tao X, Cheong KH, Gururajan R, Molinari F, Acharya U (2020) Automated prediction of sepsis using temporal convolutional network. *Comput Biol Med* :103957. <https://doi.org/10.1016/j.compbiomed.2020.103957>
31. Krizhevsky A, Sutskever I, Hinton G (2012) Imagenet classification with deep convolutional neural networks. In: Proceedings of the 25th international conference on neural information processing systems - Volume 1, NIPS'12. Curran Associates Inc., Red Hook, pp 1097–1105
32. Kirbaş İ, Sözen A, Tuncer AD, Şinasi Kazancıoğlu F (2020) Comparative analysis and forecasting of covid-19 cases in various european countries with arima, narnn and lstm approaches. *Chaos Solitons Fractals* 110015:138. <https://doi.org/10.1016/j.chaos.2020.110015>
33. Lara-Benítez P, Carranza-García M, Luna-Romera JM, Riquelme JC (2020) Temporal convolutional networks applied to energy-related time series forecasting. *Appl Sci* 10(77):2322. <https://doi.org/10.3390/app10072322>
34. LeCun Y, Bengio Y (1998) Convolutional networks for images, speech, and time series. MIT Press, Cambridge, pp 255–258
35. LeCun Y, Bengio Y (1998) Convolutional networks for images, speech, and time series. MIT Press, Cambridge, pp 255–258
36. Liu DC, Nocedal J (1989) On the limited memory bfgs method for large scale optimization. *Math Program* 45(1):503–528. <https://doi.org/10.1007/BF01589116>
37. Liu Y, Dong H, Wang X, Han S (2019) Time series prediction based on temporal convolutional network. In: 2019 IEEE/ACIS 18th international conference on computer and information science (ICIS). pp. 300–305
38. OECD (2020) OECD economic outlook, Volume 2020 Issue 1: Preliminary version. OECD Economic Outlook. OECD. <https://doi.org/10.1787/0d1d1e2e-en>
39. Oord Avd, Dieleman S, Zen H, Simonyan K, Vinyals O, Graves A, Kalchbrenner N, Senior A, Kavukcuoglu K (2016) Wavenet: A generative model for raw audio. arXiv: [1609.03499](https://arxiv.org/abs/1609.03499) [cs]
40. Ozturk T, Talo M, Yildirim EA, Baloglu UB, Yildirim O, Rajendra Acharya U (2020) Automated detection of covid-19 cases using deep neural networks with x-ray images. *Comput Biol Med* 103792:121. <https://doi.org/10.1016/j.compbiomed.2020.103792>
41. Paul SK, Jana S, Bhaumik P (2020) A multivariate spatiotemporal spread model of covid-19 using ensemble of convlstm networks. medRxiv p. 2020.04.17.20069898. <https://doi.org/10.1101/2020.04.17.20069898>

42. Ross R (1916) An application of the theory of probabilities to the study of a priori Pathometry. Part I. <https://doi.org/10.1098/rspa.1916.0007>
43. Rumelhart DE, Hinton G, Williams RJ (1986) Learning representations by back-propagating errors. *Nature* 323(60886088):533–536. <https://doi.org/10.1038/323533a0>
44. Salgotra R, Gandomi M, Gandomi AH (2020) Evolutionary modelling of the covid-19 pandemic in fifteen most affected countries. *Chaos Solitons Fractals* 110118:140. <https://doi.org/10.1016/j.chaos.2020.110118>
45. Salje H, Kiem CT, Lefrancq N, Courtejoie N, Bosetti P, Paireau J, Andronico A, Hozé N, Richet J, Dubost CL, et al. (2020) Estimating the burden of SARS-CoV-2 in France. *Science* 369(6500):208–211. <https://doi.org/10.1126/science.abc3517>
46. Singh S, Parmar KS, Makkhan SJS, Kaur J, Peshoria S, Kumar J (2020) Study of arima and least square support vector machine (ls-svm) models for the prediction of sars-cov-2 confirmed cases in the most affected countries. *Chaos Solitons Fractals* 139:110086. <https://doi.org/10.1016/j.chaos.2020.110086>. <http://www.sciencedirect.com/science/article/pii/S0960077920304835>
47. Soubeyrand S, Ribaud M, Baudrot V, Allard D, Pommeret D, Roques L (2020) Covid-19 mortality dynamics: The future modelled as a (mixture of) past(s). *PLOS ONE* 15(9):1–9. <https://doi.org/10.1371/journal.pone.0238410>
48. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R (2014) Dropout: A simple way to prevent neural networks from overfitting. *J Mach Learn Res* 15:1929–1958
49. Swapnarekha H, Behera HS, Nayak J, Naik B (2020) Role of intelligent computing in covid-19 prognosis: A state-of-the-art review. *Chaos Solitons Fractals* 138:109947. <https://doi.org/10.1016/j.chaos.2020.109947>
50. Taylor SJ, Letham B (2018) Forecasting at scale. *Am Stat* 72(1):37–45. <https://doi.org/10.1080/00031305.2017.1380080>
51. Tomar A, Gupta N (2020) Prediction for the spread of covid-19 in India and effectiveness of preventive measures. *Sci Total Environ* 138762:728. <https://doi.org/10.1016/j.scitotenv.2020.138762>
52. Zeiler MD, Fergus R (2014) Visualizing and understanding convolutional networks. In: Fleet D., Pajdla T., Schiele B., Tuytelaars T. (eds) *Computer vision – ECCV 2014*. Springer International Publishing, Cham, pp 818–833
53. Zeroual A, Harrou F, Dairi A, Sun Y (2020) Deep learning methods for forecasting covid-19 time-series data: A comparative study. *Chaos Solitons Fractals* 110121:140. <https://doi.org/10.1016/j.chaos.2020.110121>
54. Zhang YF, Thorburn PJ, Fitch P (2019) Multi-task temporal convolutional network for predicting water quality sensor data. In: Gedeon T, Wong KW, Lee M (eds) *Neural Information Processing*. Springer International Publishing, Cham, pp 122–130

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.