A Multivariate Spatiotemporal Model of COVID-19 Epidemic Using Ensemble of ConvLSTM Networks

Swarna Kamal Paul[1] · Saikat Jana[1] · Parama Bhaumik[2]

Abstract The high R-naught factor of SARS-CoV-2 has created a race against time for mankind, and it necessitates rapid containment actions to control the spread. In such scenario short-term accurate spatiotemporal predictions can help understanding the dynamics of the spread in a geographic region and identify hotspots. However, due to the novelty of the disease there is very little disease-specific data generated yet. This poses a difficult problem for machine learning methods to learn a model of the epidemic spread from data. A proposed ensemble of convolutional LSTM-based spatiotemporal model can forecast the spread of the epidemic with high resolution and accuracy in a large geographic region. The feature construction method creates geospatial frames of features with or without temporal component based on latitudes and longitudes thus avoiding the need of location specific adjacency matrix. The model has been trained with available data for USA and Italy. It achieved 5.57% and 0.3% mean absolute percent error for total number of predicted infection cases in a 5-day prediction period for USA and Italy, respectively.

✉ Swarna Kamal Paul
  swarna.kpaul@gmail.com

  Saikat Jana
  saikatjana23091990@gmail.com

  Parama Bhaumik
  bhaumikparama@gmail.com

1 Tata Consultancy Services Kolkata, Kolkata, India

2 IT, Jadavpur University, Kolkata, India

Introduction

Wuhan city in China initially observed an outbreak of Covid-19 disease caused by SARS-CoV-2. Eventually it became a pandemic and more than 200 countries are fighting hard to contain the infection. One of the best ways to contain the infection is rapid identification of positive cases and isolation. Forecasting regional spread can help identify future hotspots and distribution of infection which would eventually help to take containment measures.

A spatiotemporal epidemic spread model can accommodate both spatial and temporal correlations in data. However, most of the models either require disease-specific domain knowledge [1] or are too spatially coarse [2]. Deep learning models can learn the dynamics of epidemic spread with high spatial resolution and high degree of accuracy with minimal initial bias due to its capability of high nonlinear representation. Deep neural network-based spatiotemporal models [3] have already been applied to predict epidemic spread. However, this model is experimented on a small localized region and influence of external factors are ignored. Deep learning models also tend to overfit due to its high representational capability. Due to availability of very limited dataset, the problem of overfitting looms large in this case. Thus, modelling of Covid-19 spread in a wide region with high spatial and temporal resolution is challenging.

To address the problem of spatiotemporal prediction of Covid-19 spread in a large geographical region with high resolution, an ensemble of Convolutional LSTM [4]—based model is proposed which needs to be trained with multilayer temporal geospatial data, transformed as sequence of frames. Each layer of the geospatial data corresponds to a causal factor that might influence the spread of the epidemic. An ensemble of models helps

1138

J. Inst. Eng. India Ser. B (December 2021) 102(6):1137–1142

reduce variance in errors and overfitting on small training dataset. Experimentation is carried out with data of USA and Italy and achieved country-level mean absolute percent error (MAPE) of 5.57% and 0.3%, respectively, on forecasting of total infection cases in 5 days period.

The paper is organized as following. A brief discussion on modelling the epidemic spread and feature engineering is presented in Sect. 2. In Sect. 3, the ensemble of Convolutional LSTM model and performance measurement metrics are presented. Section 4 is about experimental results. The following section concludes the paper.

Modelling Covid-19 Spread

Disease spread is a complex dynamical system, and numerous factors contribute to the dynamics of spread making it non-stationary. Covid-19 is no different. Geographical location, weather conditions [5], human mobility [6], and population statistics might be some of the impacting factors changing the dynamics of the spread. Epidemic spread is correlated in time as well as spatial dimensions. However, it may be spatially autocorrelated in a small localized region but not across wide regions. Thus, a large geographic region is divided into relatively smaller grids and model is trained with samples drawn from local distribution of infection cases. The objective of the model is to forecast new cases of infection on daily basis in different regions across a country which can be added up to calculate total cases of infection.

Feature Engineering

All the observations in the dataset are mapped to a spatial region bounded by defined latitude and longitude. The region is geospatially divided in M × N grids of equal sizes bounded by calculated latitudes and longitudes. Figure 1a illustrates a grid bounded by latitudes and longitudes. The box represented by the dotted line is called as frame. The frames have overlapping areas in all 4 directions. The overlap allows flow of spatial influence from neighbouring grids to another. Each frame is in turn divided into L × L pixels which includes the overlapping area. Each pixel represents a bounded area in geospatial region. The values in each pixel are mapped to certain feature in the bounded geospatial region. Separate frame matrices are constructed for each feature and concatenated through channels. For example, new infection count and population are two features and they represent two separate L × L matrices in a frame concatenated across a third axis. Each pixel in the infection count matrix contains the count of new infections (ΔI) in the pixel area in a day. Infection count is distributed both in spatial and temporal dimensions. To reduce the skewness, the infection count in a pixel is log transformed and normalized in 0-1 scale. Considering R0 factor of Covid-19 between 2 and 3, it is calculated that 60% of the population (P) in an area needs to get infected to attain herd immunity and reduce further spreading [7]. Similar to the SIR model [8], the total population P is compartmentalized into susceptible (S) and infected/recovered/deceased (I) group. Susceptible population at any day is calculated as $0.6P - I$. A pixel value is calculated as $\ln(\Delta I + 1)/\ln(S + 2)$. Total population is distributed spatially in similar fashion, and it is assumed time invariant within a short interval. Pixel value of population matrix is calculated as $\ln(0.6P + 1)/\ln(\max(0.6P))$. Each frame is represented as tensor of dimension $T \times L \times L \times C$, where T is the total time span and C is number of channels or features. As shown in Fig. 1b, each training sample in a frame is
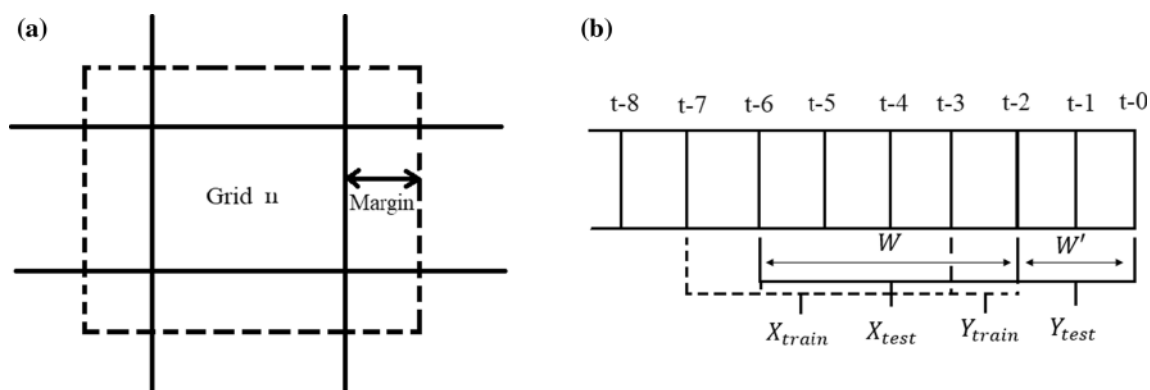

Fig. 1 **a** Illustration of overlapping frames obtained by spatially dividing a geographical region. The bold lines represent latitudes and longitudes which separate the grids. The box with dotted line represents the overlapping frame and it is represented as an image for training the model. Each grid is divided in certain number of pixels. The margin refers to number of pixels in the overlapping region. **b** Illustration of sequence of images in a frame. t-0 is the most recent image. X_{train}, Y_{train} are the training samples and X_{test}, Y_{test} are testing samples

generated by sliding a time window size of W +1 by 1, leaving behind a test case sample of time window size of W' in the most recent period. Number of training samples in a frame can be calculated as $T - W' - W - 1$. Thus, total number of training samples S_{train} for all frames can be calculated as $S_{train} = (T - W' - W - 1) * M * N$.

The forecasting problem is framed as supervised learning problem. Given a sequence of observed matrices of spatial data as images $X_1, X_2...X_t$, the final objective of the model is to predict the next image X_{t+1}. The training samples are divided into input sequences and output sequences each of length W. The input sequence is time lagged images of output sequence. The model forecasts the normalized log transformed new infection count in each pixel in a frame for each timestep. Thus, the output frame consists of only 1 channel. The input training dataset (X_{train}) can be represented as a tensor of size $S_{train} \times W \times L \times L \times C$ and the output dataset (Y_{train}) as $S_{train} \times W \times L \times L \times 1$. For training, the input sequences are selected from all frames having nonzero total infection count. Figure 1b illustrates the sequence of images in a frame. The image t-7 to t-3 represents an input training sequence (X_{train}) of length W. The output image sequence (Y_{train}) for this training sample is t-6 to t-2. Other training samples are generated by sliding the window W backwards in time by 1. The most recent images t-0 and t-1 represent the test output images (Y_{test}), and immediate sequence of images t-6 to t-2 is the test input sample (X_{test}). The test set X_{test} is represented by a tensor of size $(M * N) \times W \times L \times L \times C$ and Y_{test} by $(M*N) \times W' \times L \times L \times 1$.

Ensemble of Convolutional LSTM Models

The model comprises of a Convolutional LSTM network [4] configured to take multichannel input such that distinct features can be passed through different channels. Multiple convolutional LSTM layers are stacked sequentially to form a network with high representational capability. The network terminates with a 3D convolutional layer having one filter which constructs a single channel output as the next frame prediction.

A single model may be prone to overfitting on training dataset and loose stability in terms of prediction made. Creating an ensemble of diverse models intended to solve the same task, and combining the predictions made by them typically improves test accuracy and stability [9]. We used bootstrap aggregation [10] to create an ensemble of models. 60% random samples are drawn with replacement from the original training dataset, and an ensemble of five models is trained individually. During prediction the output of each of the models is weighted as per following equation.

$$o = \sum_{\forall \xi E} \sigma \left(\ln \left(\frac{\sum I_{train}^{\xi}}{MSE_{\xi}} + 1 \right) \right)_{\xi} * o_{\xi}, \text{ where } \sigma(z)_i$$
$$= \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{1}$$

o is output of the ensemble, o_{ξ} is output of the model ξ, E is set of all models in the ensemble, I_{train}^{ξ} is number of infected patients in the training samples of model ξ, MSE_{ξ} is mean squared error of model ξ on validation dataset, and σ is softmax function. The weights are proportional to the amount of positive cases used for training the model and inversely proportional to the validation error.

During testing the model is given a sequence of most recent frames as input, and the next frame is predicted. The predicted frame is temporally appended to the input sequence of frames and fed to the model again to obtain the next predicted frame. This continues until required number of future frames are predicted. The accuracy of a model is tested with the metric "mean absolute percent error" (MAPE) and Kullback–Liebler (KL) divergence [11]. The pixel values are transformed to ΔI and summed up cumulatively to calculate total infection cases I, up till a specific day. MAPE is calculated at pixel level for total infection cases at the end of prediction period and averaged. The pixels with 0 susceptible population count are ignored while calculating MAPE. Pixel MAPE is calculated as per Eq. 2, where G_p is set of all unique pixels in all grids such that the frame for each corresponding grid has nonzero total infection count, W' is prediction time period, $W'' = T - W'$ is total time period in training set, p is a pixel from a set of unique pixels in the total region having nonzero actual susceptible population, p_i is predicted pixel value on i th day, S_p^i is susceptible population at pixel p on i th day calculated from p_i and S_p^{i-1}, ΔI_p^i is actual new infection count at pixel p on i th day, and N_p is total number of pixels p in the region. $\hat{I}_p$ and I_p are total predicted and actual infection cases, respectively.

$$MAPE_{pixel} = \frac{1}{N_p} \sum_{\forall p} \frac{\left| \hat{I}_p - I_p \right|}{I_p} | p \in G_p, \text{ where} \tag{2}$$

$$\hat{I}_p = \sum_{i \in W'} e^{p_i * \ln \left(S_p^{i-1} + 2 \right)} - 1 + \sum_{i \in W''} \Delta I_p^i$$

$$I_p = \sum_{\forall i \in W'} \Delta I_p^i + \sum_{i \in W''} \Delta I_p^i$$

KL divergence at pixel level is calculated for total infection cases at the end of prediction period to measure the dissimilarity of distribution of predicted infection cases with respect to actual. σ is softmax function applied after scaling a series in 0 to 1 scale, and $P(X)$ is probability distribution of X. Softmax is applied to convert total infection cases as probability distribution across pixels.

$$D_{KL}^{pixel} = \sum_p P\left(\sigma(\hat{I})_p\right) \log\left(\frac{P\left(\sigma(\hat{I})_p\right)}{P\left(\sigma(I)_p\right)}\right) \tag{3}$$

MAPE is also calculated at country level with respect to total predicted infection cases across the region during the prediction period.

$$MAPE_{country} = \frac{1}{N_{W'}} \sum_{\forall k \in W'} \left(\frac{\sum_p \hat{I}_p^k - \sum_p I_p^k}{\sum_p I_p^k + \sum_p \sum_{i \in W''} \Delta I_p^i}\right), \text{ where} \tag{4}$$

$$\hat{I}_p^k = \sum_{i \in W' \Delta i \leq k} e^{p_i * \ln\left(S_p^{i-1} + 2\right)} - 1$$

$$I_p^k = \sum_{\forall i \in W'} \Delta I_p^i$$

Experimental Results

Experiments have been carried out to predict the future new infection cases in Italy for a period of 5 days and 10 days and in USA for a period of 5 days and 8 days. Data have been collected from Harvard Dataverse [12–14]. For USA the data collection period is '2020-03-09' to '2020-04-08' and for Italy it is '2020-02-05' to '2020-04-10'. Test data period for Italy data is '2020-04-01' to '2020-04-10' and for USA it is '2020-04-01' to '2020-04-08'. Figure 2a, b shows the region of USA and Italy which has been divided into grids. The length W of each training input sequence is taken as 10 days. For both the countries, the frames containing at least a single Covid-19 infection

case are considered for training and testing the model. Each frame in turn is divided in 16 × 16 pixels with an overlap *Margin* of 4 pixel.

The model consists of ensemble of 5 Convolutional LSTM networks. For Italy each network contains 4 hidden layers with sigmoid activation. The output layer is a Convolutional 3D layer with exponential linear unit as activation. The models are trained for 30 epochs with mean squared error as loss function. The input and hidden layers have 32 filters. The input layer is configured to take images of size 16 × 16 × 2. The second channel is fed with normalized population data. The region in USA is approximately 13 times than Italy, and the distribution of Covid-19 cases in USA is geospatially highly skewed. Thus, grids are divided in four equal sections by latitude and longitude with each section containing 9 × 15 grids. A set of 4 heterogenous ensembles are trained for each of the 4 sections. The configuration of networks is same as that of Italy except they contain 2 hidden layers and each network is trained for 20 epochs.

Table 1 shows the performance of the models in terms of KL divergence and MAPE. For both USA and Italy, low KL divergences state that the predicted geospatial probability distribution of total infection cases nearly matches with the actual probability distribution. The pixel level MAPE for Italy stays below 30%. For USA in 8-day forecasting period MAPE is 44% as there are many pixels in USA with low total patient count. A slight deviation in the prediction for these pixels shoots up the MAPE. Country level MAPE is low for both Italy and USA.

Figures 3a and 4a shows predicted versus actual total Covid-19 cases for a period of 8 and 10 days in USA and
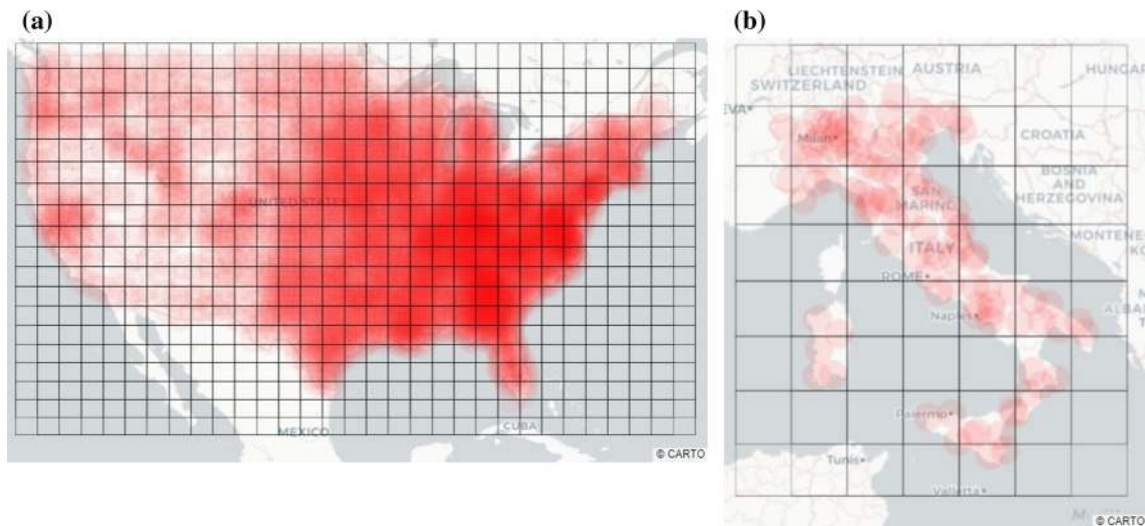

Fig. 2 **a** A region of USA divided in 18 × 30 grids. The red bubbles denote cumulative number of Covid-19 cases. **b** A region of Italy divided in 8 × 7 grids with cumulative Covid-19 cases denoted by red bubbles

Table 1 KL divergence at pixel level and mean absolute percent error at pixel and country level for Italy and USA

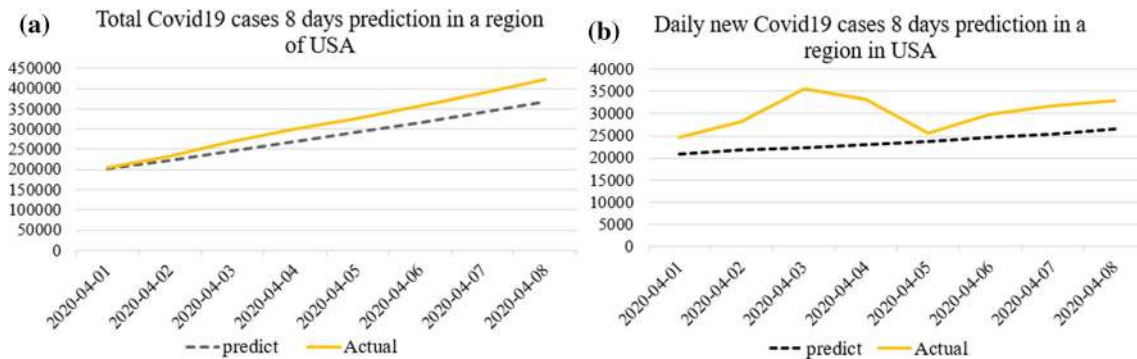| Country	Pixel KL divergence	Pixel MAPE total cases	Country MAPE total cases	Predicted total cases	Test span in days
Italy	5.72×10^{-5}	11.51%	0.3%	129,665	2020-04-01–2020-04-05
Italy	0.00010	28.64%	1.4%	151,803	2020-04-01–2020-04-10
USA	3.06×10^{-6}	30.4%	5.57%	292,032	2020-04-01–2020-04-05
USA	0.0014	44%	9%	368,660	2020-04-01–2020-04-08

Fig. 3 **a** Predicted versus Actual total Covid-19 cases for a period of 8 days in a region of USA. **b** Predicted versus Actual daily new Covid-19 cases for a period of 8 days in a region of USA
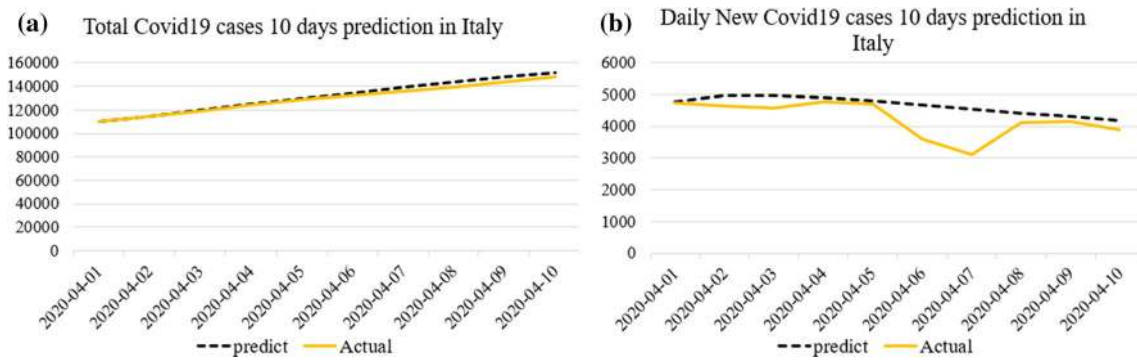

Fig. 4 **a** Predicted versus Actual total Covid-19 cases for a period of 10 days in a region of Italy. **b** Predicted versus Actual daily new Covid-19 cases for a period of 10 days in a region of Italy

Italy, respectively. For Italy the prediction follows closely with the actual, whereas for USA it is little underestimated. Figures 3b and 4b shows predicted versus actual daily new Covid-19 cases in USA and Italy, respectively. Figures 5a, b shows the distribution of total predicted versus actual infection cases in each pixel after 10 day and 8 day in Italy and USA, respectively. The predicted distribution closely follows with actual with residuals distributed both on negative and positive sides.

Conclusion

An ensemble of Convolutional LSTM-based spatiotemporal epidemic spread model has been proposed for short-term forecasting of Covid-19 spread. Experiments done on data obtained for USA and Italy reveal acceptable prediction accuracy with high resolution. Since the model has option to fed in any number of external features, we are experimenting with multiple external features that might influence the spread. This might help to find important
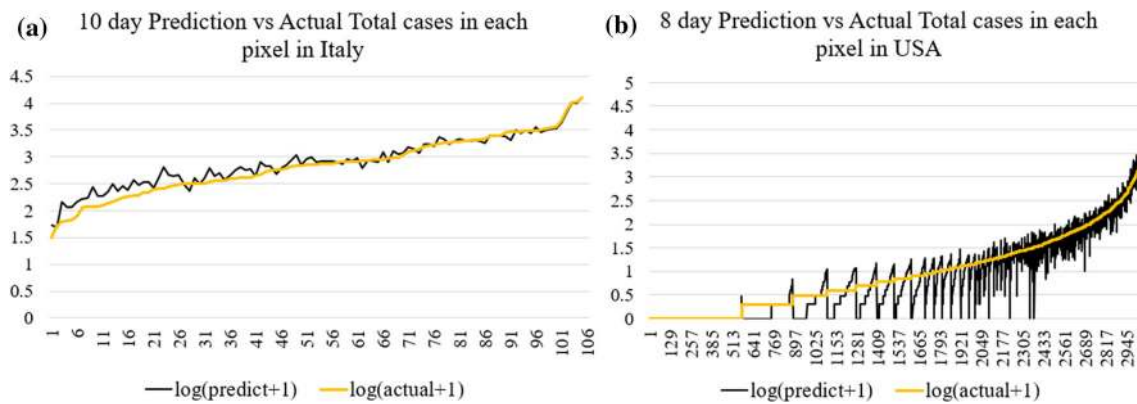
Fig. 5 a Log transformed Predicted versus Actual total Covid-19 cases after a period of 10 days in each pixel in Italy. **b** Log transformed Predicted versus Actual total Covid-19 cases after a period of 8 days in each pixel in USA

causal features that are impacting the spread across multiple locations.

References

1. R.M. Neilan, S. Lenhart, Optimal vaccine distribution in a spatiotemporal epidemic model with an application to rabies and raccoons. J. Math. Anal. Appl. **378**(2), 603–619 (2011)
2. V.E. Pitzer, C. Viboud, L. Simonsen, C. Steiner, C.A. Panozzo, W.J. Alonso, B.T. Grenfell, Demographic variability, vaccination, and the spatiotemporal dynamics of rotavirus epidemics. Science **325**(5938), 290–294 (2009)
3. G. Xi, L. Yin, Y. Li, S. Mei, A deep residual network integrating spatial-temporal properties to predict influenza trends at an intra-urban scale, in *Proceedings of the 2nd ACM SIGSPATIAL International Workshop on AI for Geographic Knowledge Discovery* (2018), pp. 19–28
4. S.H.I. Xingjian, Z. Chen, H. Wang, D.Y. Yeung, W.K. Wong, W.C. Woo, Convolutional LSTM network: a machine learning approach for precipitation nowcasting, in *Advances in Neural Information Processing Systems* (2015), pp. 802–810
5. M.M. Sajadi, P. Habibzadeh, A. Vintzileos, S. Shokouhi, F. Miralles-Wilhelm, A. Amoroso, *Temperature and Latitude Analysis to Predict Potential Spread and Seasonality for Covid-19* (2020). Available at SSRN 3550308
6. C. Zhan, C. Tse, Y. Fu, Z. Lai, H. Zhang, *Modelling and Prediction of the 2019 Coronavirus Disease Spreading in China Incorporating Human Migration Data* (2020). Available at SSRN 3546051
7. P. Fine, K. Eames, D.L. Heymann, Herd immunity: a rough guide. Clin. Infect. Dis. **52**(7), 911–916 (2011)
8. H.W. Hethcote, Qualitative analyses of communicable disease models. Math. Biosci. **28**(3–4), 335–356 (1976)
9. S. Lee, S. Purushwalkam, M. Cogswell, D. Crandall, D. Batra, *Why M Heads are Better Than One: Training a Diverse Ensemble of Deep Networks* (2015). arXiv:1511.06314
10. L. Breiman, Bagging predictors. Mach. Learn. **24**(2), 123–140 (1996)
11. S. Kullback, R.A. Leibler, On information and sufficiency. Ann. Math. Stat. **22**(1), 79–86 (1951)
12. https://www.citypopulation.de/en/italy/admin/
13. China Data Lab, *US COVID-19 Daily Cases with Basemap*, Harvard Dataverse (2020). https://doi.org/10.7910/DVN/HIDLTK
14. Dipartimento della Protezione Civile, *Coronavirus Disease 2019 (COVID-19) in Italy*, Harvard Dataverse (2020). https://doi.org/10.7910/DVN/KDFYZW