OPEN ACCESS
Fractals, Vol. 30, No. 5 (2022) 2240122 (12 pages)

DOI: 10.1142/S0218348X22401223

EXPLAINABILITY OF NEURAL NETWORK CLUSTERING IN INTERPRETING THE COVID-19 EMERGENCY DATA

ZHENHUA YU,[*,¶] AYESHA SOHAIL,[†,‖] TAHER A. NOFAL[‡]
and JOÃO MANUEL R. S. TAVARES[§]

[*]*Institute of Systems Security and Control*
College of Computer Science and Technology
Xi'an University of Science and Technology
Xi'an 710054, P. R. China

[†]*Department of Mathematics*
Comsats University Islamabad
Lahore Campus, Lahore, Pakistan

[‡]*Department of Mathematics and Statistics*
Faculty of Science, Taif University, Taif, Saudi Arabia

[§]*Instituto de Ciência e Inovação em Engenharia*
Mecânica e Engenharia Industrial
Departamento de Engenharia Mecânica
Faculdade de Engenharia
Universidade do Porto, Porto, Portugal
[¶]*zhenhua_yu@163.com*
[‖]*ayeshasohail81@gmail.com*

[‖]Corresponding author.

Received August 6, 2021
Accepted November 10, 2021
Published May 5, 2022

Abstract

Among other hospitalization causes and cases, the clinical emergency is a critical case and the data of the reporting patients are biased as well as poorly managed due to the chaotic situation. The world has faced chaos over the past year due to the frequent waves of COVID-19 and the resulting emergencies. The data banks, linked with the clinical emergencies require serious quantitative and qualitative analysis to drive interpretable conclusions for necessary future emergency measures and to develop explainable artificial intelligence tools. This important procedure involves the clear understanding of the data patterns and topologies, which is a great challenge for the multidimensional data sets. Mathematically, the topological mapping can resolve this problem by mapping higher-dimensional data to two-dimensional representation, based on the overall association. Proper data mining and pattern recognition can help in improving the rapid patients admission, in providing the medical resources timely and in proper patient administration. In this paper, the importance of self-organizing maps, to interpret the hospital data, particularly for the COVID-19 epidemic is discussed in detail. Important variables are identified with the aid of networks and mappings.

Keywords: Explainable Artificial Intelligence; Self-Organizing Maps; Accuracy and Precision; Time Series Modeling.

1. INTRODUCTION

1.1. Visualizing Fractals in Emergency Data

Fractal analysis is used in the literature to explore the complexity of the research problems.[1,2] In the event of emergency and alarming situations, such as wars, flood, earthquake, disease outbreak, the data handling and management is not only a great challenge but is impossible to maintain sometimes due to the pressure and chaos.[3] An example of such chaotic situation is the current challenge of the COVID-19 frequent outbreaks.

To address such issues, the data management, data cleaning, data retrieval and clear understanding of the data are important.[4–6] To design an explainable machine learning algorithm for such data sets it is important to prepare the data as a preliminary step, since the raw data analysis may lead to false training, during the procedure of the machine learning model development. The raw data cleaning involves removal of incomplete rows or retrieving incomplete/missing data, standardizing data, removing duplication and error fixation.

To deal with the errors, while analyzing the clinical emergency data sets, options of both, qualitative and quantitative approaches are available in the fields of artificial intelligence and data science. The qualitative techniques are based on specific rules, or data relevant constraints, whereas, the quantitative approach is mainly the statistical approach. A script is designed to rectify and eliminate the errors during quantitative approaches.[7,8]

Once the data are ready for the management and policy making, the reinforcement, supervised or unsupervised machine learning tools can be used to interpret the data,[9,10] according to the requirement of the research project.

The next step is the clustering of the data based on the topological measures. This important step is executed with the aid of the networks called the self-organizing maps (SOMs). With the aid of this clustering, it is helpful to identify the useful attributes or variables that have influence on the clinical emergency. This important screening can be managed with the aid of the "SOMs" algorithm analysis, that allows the pattern recognition.[11] In the literature, three most frequently used software are "Matlab", "R" and "Python", where the clustering tool is well designed with the necessary utilities. For Matlab, the "neural network clustering" is used, whereas, for Python, "sklearn" (Ref. 12) is used.

The phase of clustering via the networks is of great significance since it improves the "human interpretation" of the "presented data".[13–15] Basically, most of the fascinating achievements of artificial intelligence still require the "explainability" and termed as the "Explainable Artificial Intelligence" (XAI), an emerging field of data science and artificial intelligence.[16–18] This step makes the human analysis (not black box algorithm-based) and interpretation more easier. The final results may be improved in three steps, by improving the data, by improving the efficiency of the procedure and by shortlisting the key variables and attributes with maximum impact.

SOMs or Kohonen's map is a data visualizing technique, introduced by Kohonen in the early 1980s. It is a form of artificial neural network that is based on unsupervised learning. It is a powerful software tool for the visualization of high-dimensional input data. It converts complex, nonlinear statistical relationships between high-dimensional data into simple geometric relationships on a low-dimensional display (typically two-dimensional).[11,19]

The main purpose of SOMs is to cluster the high-dimensional data into easy-to-visualize outputs. Human brain is an example of the SOM. The brain maps the external multidimensional representation of different objects it senses into one- or two-dimensional internal representation, it then maps them onto "topologically distinct" areas. In scientific language, the brain processes external signals in a topology-preserving manner.

An SOM is different from typical "Artificial Neural Networks (ANNs)" in its architecture and algorithmic properties. The organizing maps use competitive learning whereas the ANNs use error correction learning such as back-propagation with gradient descent. The self-organizing neighborhood function preserves the topological properties of the input space. The SOM has the ability to learn and organize data without the prior knowledge of the desired outputs for the given input vectors.

In general, the SOM defines a mapping from the higher dimension of input data space onto a regular two-dimensional mapping array. The Kohonen neural network (SOM) has two layers: input layer and output layer (Kohonen layer). SOM does not use activation function like ANN to defines the output. These two layers are demonstrated with the aid of schematic Fig. 1. The input layer is fully connected with the competitive layer of processing

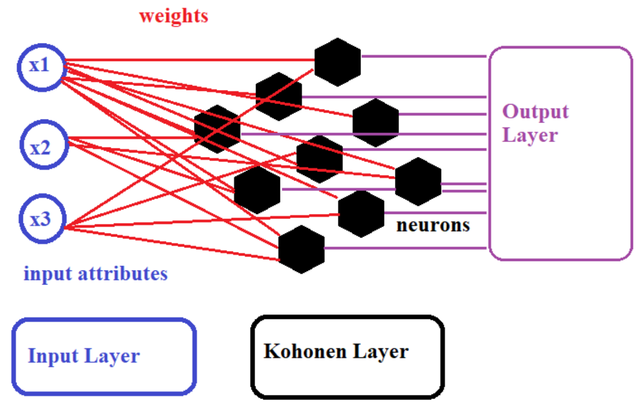

Fig. 1 Kohonen neural network (SOM) with two layers.

neurons. During this research, the COVID-19 emergency data, specifically for Italy, are analyzed with the aid of this clustering approach.

1.2. COVID-19 Emergency Data Fractal Analysis

The initial COVID-19 outbreak was sudden and alarming as the health care system across the globe was not efficient to deal with such a deadly pandemic. The countries where coronavirus continued to be severe had to struggle hard to control the outbreak. These included seven European countries and among them, Italy was at the top of the list of COVID-19 cases,[20–23] according to the statistics updated by world health organization. The first patient of COVID-19 was admitted in Lodi and since then, Italy is still struggling to fight the war against COVID-19 at the cost of public health. The intensity of this deadly viral infection was too high in Lombardy, Emilia-Romanga and other cities such as Veneto, during the first quartile of 2020 and led to higher number of admissions to the intensive care unit (ICU).

During this research, we have used the SOMs to cluster the clinical emergency data. The data were extracted with permission from the data repository (https://ourworldindata.org/covid-hospitalizations). To explain the approach in detail, both SOMs as the neural network clustering tool and the sklearn packages are used. The key features and attributes were identified with the help of clustering behavior throughout the period of 54 weeks, starting from February 2020 to February 2021.

Although the machine learning tools are ruling the scientific world over the past few decades, the current challenge is to design tools that are more interpretable and equipped with algorithms, that are more explainable. The important data analysis steps, followed during this research, can be used to make future interventions, in case of calamities and chaos.

The rest of the paper is organized as follows: the SOMs and the mathematical motivation behind the algorithm are discussed in detail in the following section, the algorithms, from two frequently used and easy to use programming frameworks of Matlab™ and Python, are then applied on the pre-processed COVID-19 emergency data of Italy. These results showed chaos and fractals, and their detailed analysis is presented during this research. Useful conclusions are drawn at the end of this paper.

2. MATERIALS AND METHODS

In this section, we will discuss the application of neural network clustering mechanism and its mathematical background. The entire process is divided into training, testing and validation steps.

During the training process,[24] input data is given to the network through the processing units (neurons) in the input layer. For a given network, input vector has fixed dimension n. The n-components of input vector x $(x_1, x_2, \ldots, x_n)$ are connected with each neuron in the array (the output layer of $M = m$ by m processing neurons) value associated with ith component of input vector to jth neurons.

Weights in SOMs connect the input data to the output. These are similar to slope in linear regression, where a weight is multiplied to the input, to add up to form the output. As an input is processed by the neuron, it gets multiplied by a weight value and the resulting output is obtained. The procedure of updating is described in the algorithm below. The input layer is connected to the Kohonen layer by weight:

$$w_j = (w_{j1}, w_{j2}, \ldots, w_{jn}), \tag{1}$$

where w_{ij} is the weight.

2.1. Training of the Neural Network Classifier

The training algorithm has two phases:

- The similarity matching phase.
- The weight modification phase.

During the first step, the input values are given to the model, initially the weights that SOM model sets are of small random values. During the similarity matching phase, the "Euclidean distances" between the inputs and weights, associated with the output neurons are computed. The neurons in the output layer compete with each other and the winning output unit is the one whose distance is minimum among the M output neurons. The winning neuron is called the "Best Matching Unit (BMU)". The BMU is more precise when the size of the two-dimensional map is enlarged, i.e. from Fig. 2, we can see that when SOM was applied on an example data, where the cluster is obvious in the bottom right corner, as the two-dimensional map is enlarged, from 5 to 10 and then to 15, with the blue dots (the training vectors), the black dots (the neurons), the convergence improved.

During the weight modification phase, a topological neighborhood, say K_t of neurons in a circle form vicinity to the winning neuron t, is created and the weights of the winning neuron and the neurons that are in the circle are updated.

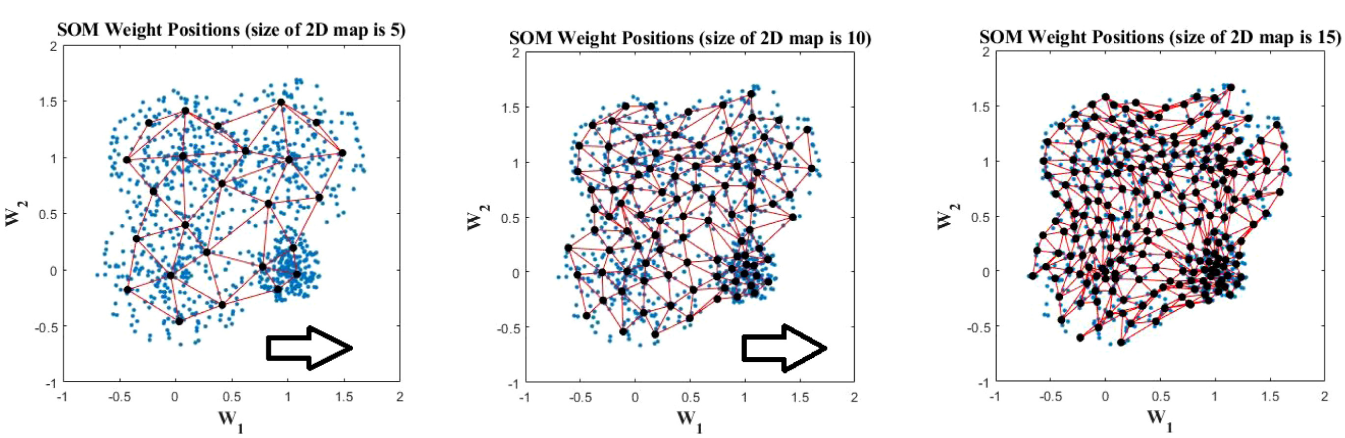

Fig. 2 BMU and neighboring neurons are updating towards the training vector which are shown with blue dots (nods). The black hexagons are the neurons. Left to right, improved mapping with increased number of maps.

Closest the neuron is from the winning neuron, the associated weight updates itself in a better way. In simple words, the SOMs work like a net which is spread onto the input data (see Fig. 2). SOM training algorithm updates weight vectors so they span the input space and the map organizes itself accordingly.

2.2. Neural Network Clustering Algorithm

The algorithm of SOMs for the neural network clustering follows a detailed procedure.[25] The algorithm is divided into five steps:

(1) Initialization: To initialize all the weight w_{ij} with small random values, select a value for the initial radius of the topological neighborhood K_0 and set the initial learning rate α_0.

(2) Competition: An input vector is chosen randomly from the input training space. Every neuron j is examined, to calculate, whose weight w_{ij} is closer to input vector x and the winning neuron is called the "BMU". The winning neuron has the smallest distance from the input vector and weight of neuron. Distance is calculated by "Euclidean distance" formula that is widely used to find distance which is given by

$$D_{\min}(t) = \min\{D_i(t)\}$$

$$= \min_i \sum_i (x_j(t) - w_{ij}(t)). \qquad (2)$$

(3) Cooperation: In the third step, topological neighborhood for the neurons in the SOM is created. This step is achieved by kernel function such as Gaussian function h_{ij}, this function depends on two factors:

(a) Time (time incremented each new input data).

(b) Distance between the winner neuron and the other neuron.

$$h_{i,j}(t) = \exp\left(\frac{d_{i,j}^2}{2\sigma(t)^2}\right) \qquad (3)$$

$\sigma(t)$ is a feature of SOM, with width of the neighborhood linked with the Gaussian function:

$$\sigma(t) = \sigma_o \exp\left(\frac{-t}{\tau_o}\right), \qquad (4)$$

where $d_{i,j}$ is the lateral distance between winning neuron (j) and neighboring neuron (i) and is defined as

$$d_{i,j} = \|w_j - w_i\|. \qquad (5)$$

(4) Adaptation is the next step, where the winning node is adjusted to be more similar to the input x. Weight of winning neuron j and its neighborhood neurons are updated by W, which is mathematically written as

$$W_i(t+1) = W_i(t) + \Delta W_i,$$

$$W_i(t+1) = W_i(t) + \alpha(t)h_{i,j}(t)(x - W_i(t)),$$

$$\alpha(t) = \alpha_o \exp\left(\frac{-t}{\tau_1}\right),$$

where α is the learning rate function. Both (α and σ) of these parameters decrease monotonically over time (t).

(5) Repeat steps 2–4 for N iterations or until the desired output is achieved. A schematic depiction of the flow chart is presented in Fig. 3.

2.3. Supervised Neural Network Clustering Algorithm

Supervised SOM is the extended SOM. It is structured in a way so that it can solve complex problems by an adaptable combination of the SOM approach with supervised learning. Using this method the learning process of SOM and the clustering can be improved.[26]

Supervised SOM can be used for various purposes, most common of which are regression and classification problems. Supervised and unsupervised SOMs follow the basic structure of SOM but they are different in their dimension of the weights and also in their training process.

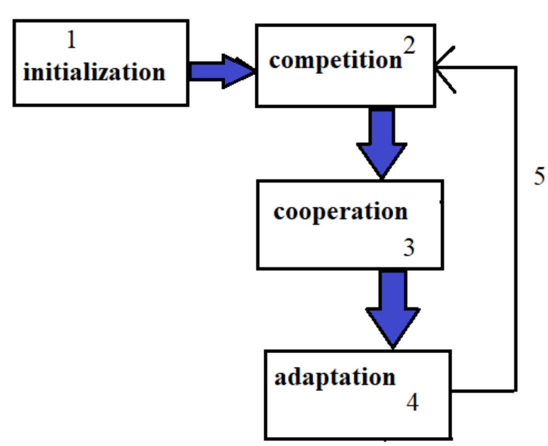

Fig. 3 Flowchart for the SOM.

Unsupervised SOM have weights of same dimension as the input data, contrary to that, supervised SOM have weights of same dimension as the output variables. Thus, adapting these weights often changes the BMU for each input data point. Supervised SOMs in the case of regression have weights which have continuous number and while classifying, the weights contains a class.

The key steps are as follows:

(1) In the initialization step every neuron is assigned to the class representing the most of data points.
(2) Randomly an input is chosen with classes (labels).
(3) BMU is calculated in a similar manner as in the case of unsupervised SOM.
(4) Neighborhood is created (neighborhood function and neuron weight matrix are calculated) similar to corporation step.
(5) To update the discrete weights, probability is calculated which is defined as

$$P_{c,i}(t) = w_{y(t)} * \alpha(t) * h_{c,t}(t). \tag{6}$$

(6) A random number $ru_i(t)$ is generated between 0 and 1, then the weights are updated based on $P_{c,i}(t)$ as follows:

$$w_i(t+1) = \begin{cases} y(t) & \text{if } u_i(t) < P_{ci}(t), \\ w_i(t) & \text{otherwise.} \end{cases}$$

(7) Repeat for desired number of times.
Finally every neuron of the SOM map is connected to a class of the data set. The details of this algorithm can be obtained from recent research approach.[9,27] To understand these steps, a schematic is presented in Fig. 4.

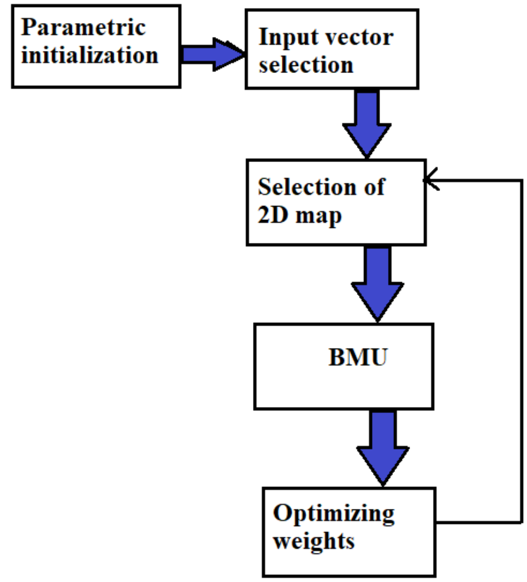

Fig. 4 Flowchart for the SOM with data optimization.

2.4. Application of LIME

The word LIME stands for "Local Interpretable Model-Agnostic Explanations". The word LIME is also used as a Python interpretation tool/library. It is basically used to explain in detail the predictions obtained from the AI model. The correct choice of predictors leads to better predictions and thus to more reliable models. The basic results of all the classification learners are based on the ROC curves, confusion matrices, accuracy and precision. These results are not enough to demonstrate the role of each predictor individually. In simple words, the classification learners are based on the correct choice of the "predictors" to obtain the most accurate "response". To explore this quest, programmers have developed libraries such as LIME, ELI5, INTERPRET and SHAP. In this paper, we will focus on the application of LIME on different SOMs, which are used to classify the data, to develop the relationship between the predictors and to understand the impact of the response variable on a given statistical trial/experiment.

3. RESULTS AND DISCUSSION

In this section, we will implement the tools described in the previous section to understand the emergency data obtained from the COVID-19 cases reported in Italy.

Our purpose is to group patients who have the most similar features and to visualize this high-dimensional data set. Next, we aim to highlight the important attributes that can help to improve the hospital management. For these purposes we choose the SOMs. It helps us to visualize this high-dimensional data into two-dimensional grid, and form the clusters of patients based on their similarities.

The data were analyzed using MATLAB as well as Python and the most accurate models were finally selected. Various factors/characteristics of patients are chosen as inputs. The variables were selected turn by turn to select the most important attributes based on their mappings and clustering behavior.

Table 1 Week-Wise Grouping of Attributes for 54 Weeks.

Group	Members	Accuracy Achieved Using SOM (%)
1	T, D, ICU cases, COVID-19-P	75
2	D, ICU cases, COVID-19-P, Cardio	72
3	COVID-19-P, Cardio	83
4	T, COVID-19-P, Cardio	80
5	T, COVID-19-P	75
6	ICU cases, Cardio	71
7	S, COVID-19-P	84
8	T, COVID-19-P	80
9	T, S	71
10	ICU cases, S	81

3.1. Grouping the Data

Some of the cases reported of SARS-2 infection had other health issues as well, such as cardiac and other pulmonary diseases. For the current study, we have selected eight data variables: the total number of cases reported during 54 weeks (T), since February 2020, the total deaths reported (D), the patients registered in ICU (ICU cases), the patients admitted to the hospital (P), total tests (TT), stringency index (it is defined as the response indicator of the lockdown measures and is denoted as S), cardiovascular deaths reported (Cardio) and positive tests (COVID-19-P).

We made ten different groups to understand the linkage between different types of cases (see Table 1), with different medical history. For example, in group 1, we want to check during each week, frequency of the patients that were reported, died (D), frequency of patients taken to the ICU (ICU cases) and frequency of the patients that were tested COVID-19 positive (T). Similarly in group 6, we have week-wise statistics of ICU cases and of patients with cardiac disease.

SOM Neighbor Weight Distances Analysis

For the higher-dimensional spaces, one cannot easily visualize the weights at the same time. The SOM neighbor weight distances indicate the distance between neighboring neurons. The gray circles (hexagons when zoomed) represent the neurons. These neurons are connected with the aid of red lines. The shades given around the neurons range from light yellow to orange, red and at last black. To interpret the SOM graphs, darker colors present larger distance and lighter color shows that there is smaller distance between the neurons. From Figs. 5 and 6, we can see that the neural network behind

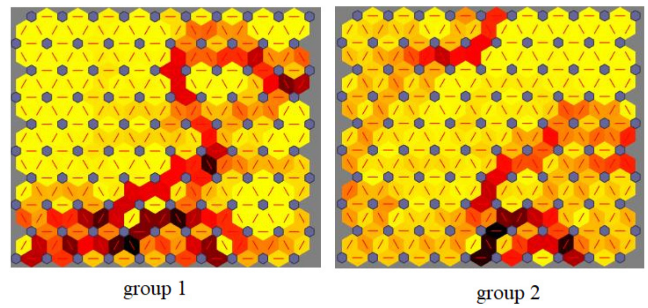

Fig. 5 SOM neighbor weight distances for groups 1 and 2.

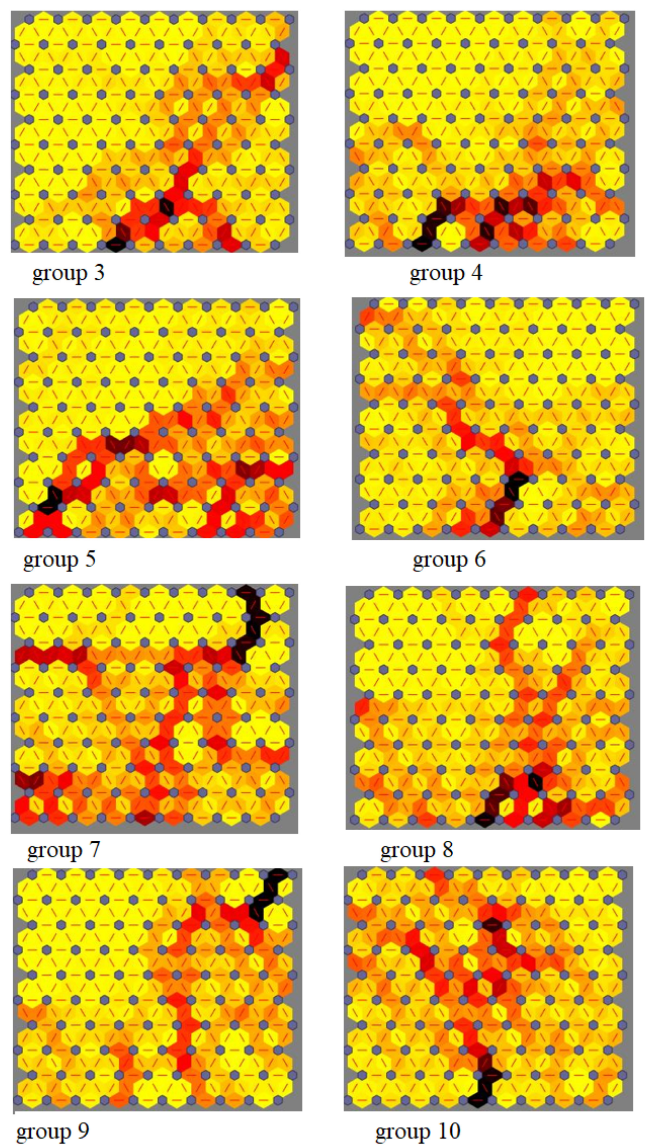

Fig. 6 SOM neighbor weight distances for groups 3–10.

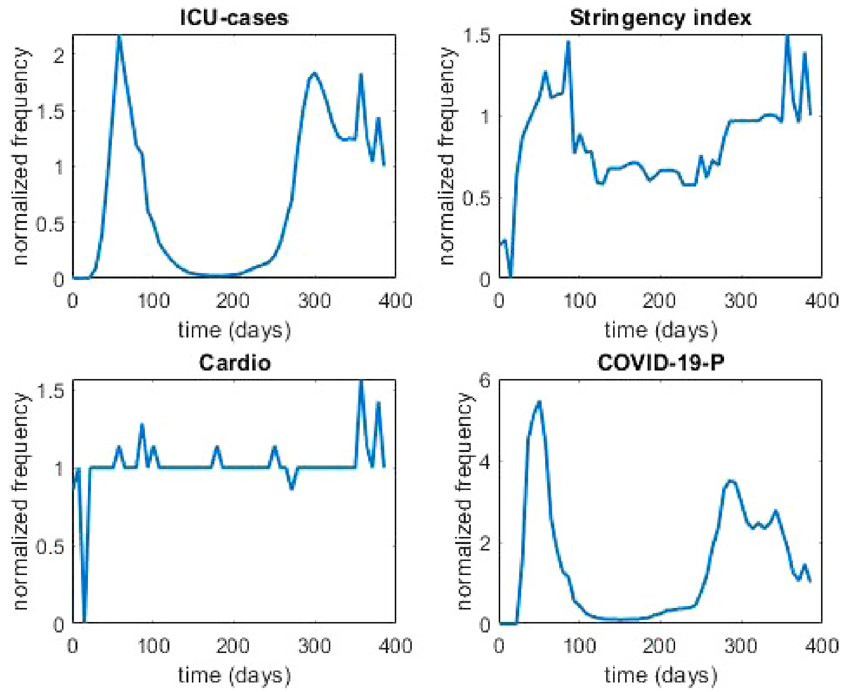

Fig. 7 Data after scaling.

the SOM has clustered the data into parts, tightly clustered data points with light yellow shade and farther apart data points with darker shades. The data dimension was based on the weeks. We can see that in each group, the response varies in 10×10 (100 neurons). This variation is obvious from the change in colors in images relative to each group.

3.2. Interpretable Results with Improved Explainability

The entire data were again cleaned, and the peaks, which were really difficult to handle during the grouping, were managed with the aid of scaling (see Fig. 7). Each of the attributes listed above was scaled by the last value of the study period. This helped us to obtain the value of each variable to range from 0 to 1 and for the positive tested cases (COVID-19-P) to range from 0 to 5. We used LIME for this purpose as discussed in the previous section.

From Figs. 8 and 9, we analyze that the groups of cases can be further divided into two sets. In Fig. 8, the scattering with vertical trend is collected. For example, in group 3, the number of COVID-19 positively tested cases per week and cases of patients with COVID-19 and cardiac deaths were grouped together. The clustering helps to interpret the distance between the weights. As discussed in detail, in the previous section, each node has a topological position in x and y coordinates, and a weight vector of two dimensions (each weight designated to each dimension), the SOM algorithm organizes the weight vectors in a manner that represents similarities with the input data. Thus following the BMU approach, the COVID-19 cases data were distributed into horizontal, vertical and inclined clustering.

From all the SOM analysis based on the SOM neighbor weight distance and SOM weight positions in two dimensions, we conclude that people with positive COVID-19 test report and cardiac issues required more ICU support.

3.3. Testing of Analysis via Time Series Modeling

With the help of clustering, we have selected the groups that showed best approximation. The number of cases for a period of one year were analyzed, and thus this time-dependent data can be forecast for the upcoming times with the help of the time series AI tools. The next step is to apply the time series modeling.[28] Based on our previous research findings,[22,29,30] we have used the Bayesian regularization approach to analyze the data. The key variables selected in the SOM analysis were forecast with the aid of the input variables. These input

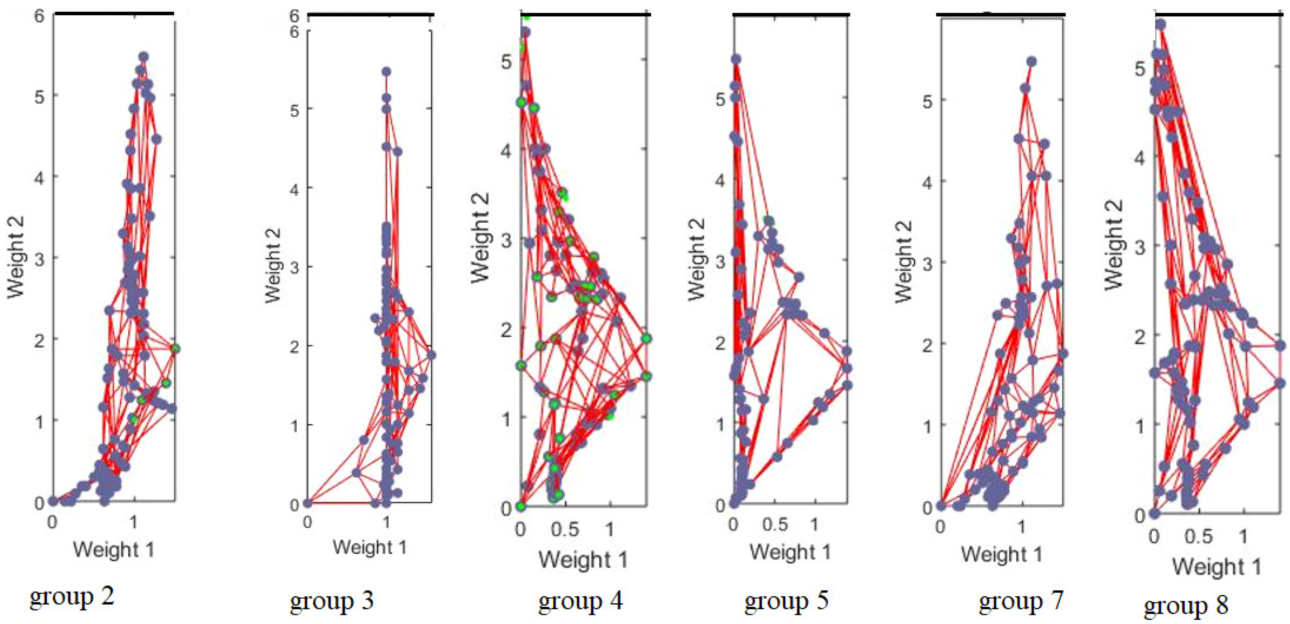

Fig. 8 SOM neighbor weight distances for groups with vertical scattering.

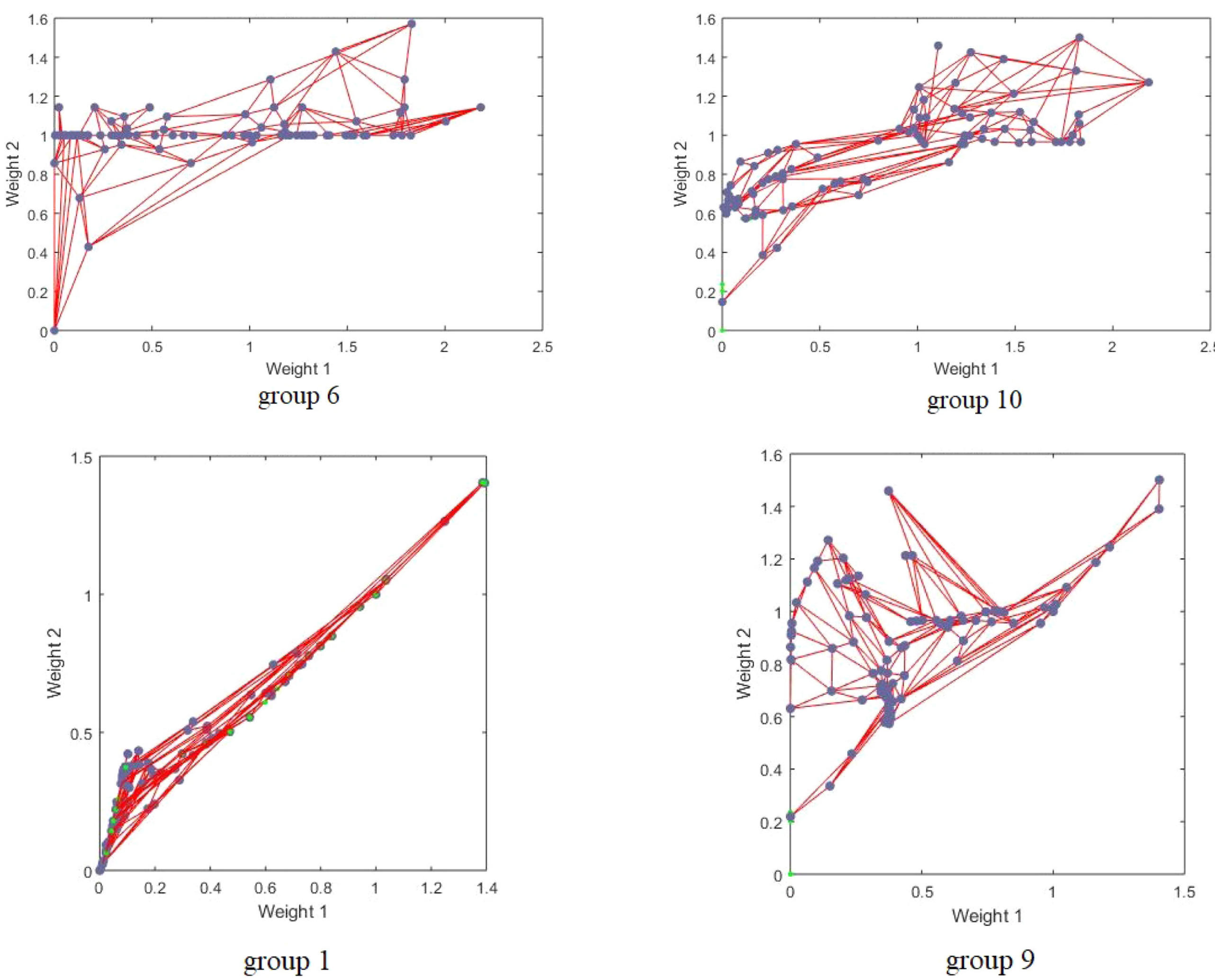

Fig. 9 SOM neighbor weight distances for groups with horizontal scattering (first row), diagonal scattering (second row).

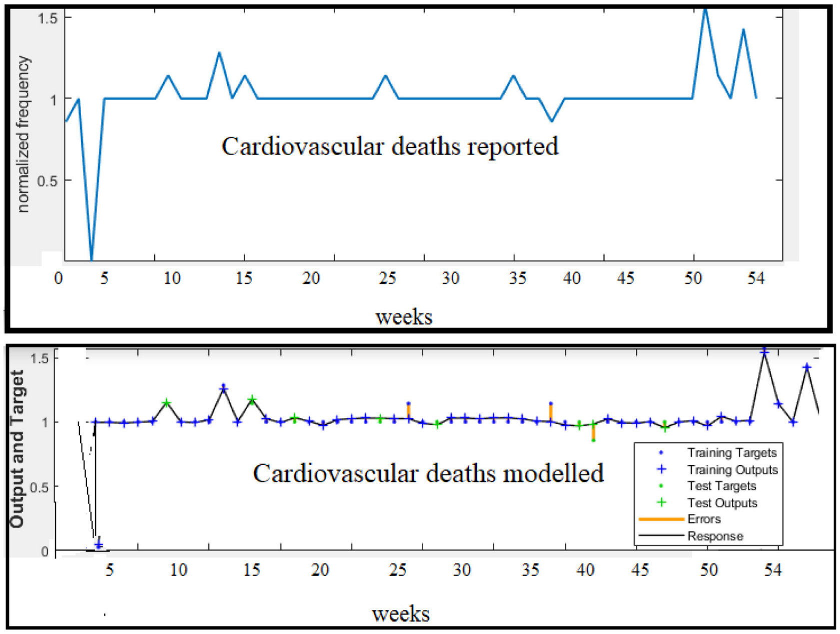

Fig. 10 Time series forecasting via the model developed for the cardiovascular deaths.

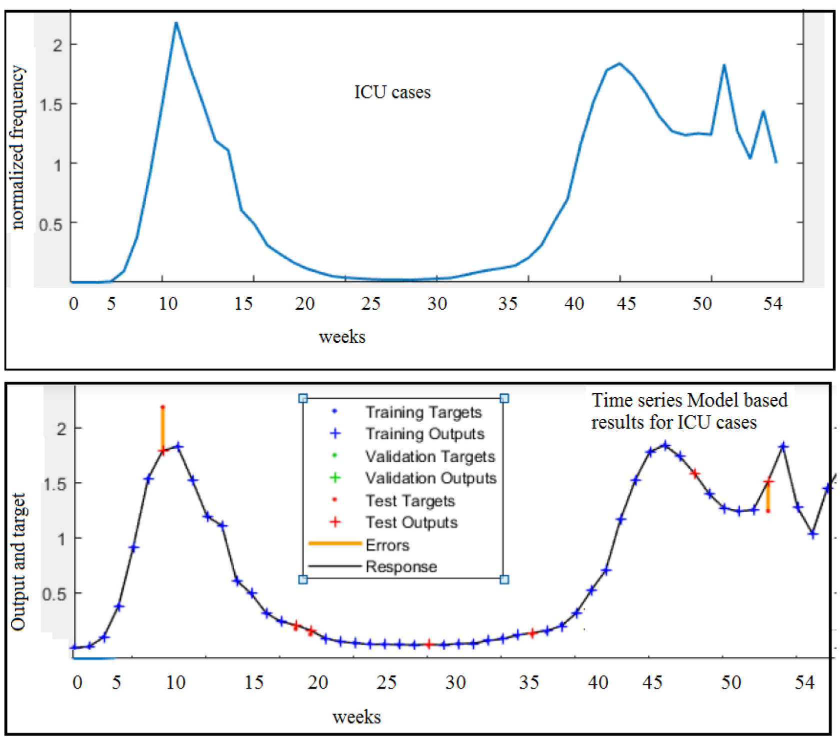

Fig. 11 Time series forecasting via the model developed for the ICU cases.

variables were the COVID-19 positive cases, the stringency index and the total cases. Figures 10 and 11 show that the models performed to better accuracy with training performance of 8×10^{-6} at 802 epochs.

4. CONCLUSIONS

We conclude that data if interpreted correctly can lead to trustworthy forecasting model. In this paper, the main focus was to identify the groups of attributes which showed correlation with the

COVID-19 outbreak, the secondary focus during this research is to implement LIME, which is the modern programming tool, on different SOMs, that are used to classify the data, to develop the relationship between the predictors and to understand the impact of the response variable on a given statistical trial/experiment. In short, during this research, with the aid of the modern clustering tools, groups with higher correlations are selected, next, the time series model based on the time period of 54 weeks for each group is developed. The models for patients reported with cardiovascular disease and patients registered for ICU showed best accuracy with lowest error. Currently the threat of COVID-19 is not over, although, vaccines have reached the phase three trials. The clear understanding of the "need-based" ICU care and availability according to the patient's status with or without cardiac issues is really desired. With our step-by-step modeling approach, we aim that the work can contribute positively in upcoming waves of COVID-19 and cases emerging from other variants of SARS-2, such as the Delta variant.

ACKNOWLEDGMENTS

The authors received financial support from Taif University Researches Supporting Project No. (TURSP-2020/031), Taif University, Taif, Saudi Arabia. All authors contributed equally to this work. The authors would like to acknowledge the source of data ("https://ourworldindata.org/covid-vaccinations?country = ITA") and their cooperation.

REFERENCES

1. Y. Karaca, Y.-D. Zhang and K. Muhammad, Characterizing complexity and self-similarity based on fractal and entropy analyses for stock market forecast modelling, *Expert Syst. Appl.* **144** (2020) 113098.
2. Y. Karaca, M. Moonis and D. Baleanu, Fractal and multifractional-based predictive optimization model for stroke subtypes' classification, *Chaos Solitons Fractals* **136** (2020) 109820.
3. K. A. Al-Utaibi, A. Sohail, Z. Yu, R. Arif, A. Nutini, A.-S. G. Abdel-Salam and S. M. Sait, Dynamical analysis of the delayed immune response to cancer, *Results Phys.* **26** (2021) 104282.
4. A. Sohail, Genetic algorithms in the fields of artificial intelligence and data sciences, *Ann. Data Sci.* (2021) 1–12.
5. F. Alam, A.-S. G. Abdel-Salam, A. Sohail, M. Yousaf and S. Tunç, Exploration of the novel corona virus transition graphs with petrinet modeling, *Biomed. Eng. – Appl. Basis Commun.* **33**(4) (2021) 2150028.
6. Z. Yu, A.-S. G. Abdel-Salam, A. Sohail and F. Alam, Forecasting the impact of environmental stresses on the frequent waves of COVID19, *Nonlinear Dynam.* **106**(2) (2021) 1509–1523.
7. M. Idrees, A. Nutini and A. Sohail, Paradigms of COVID 19 and the cytokine therapy approaches, *Biomed. Eng. – Appl. Basis Commun.* **33** (2021) 2150035.
8. M. Idrees and A. Sohail, Bio-algorithms for the modeling and simulation of cancer cells and the immune response, *Bio-Algorithms Med-Syst.* **17**(1) (2021) 55–63.
9. A. Sohail and F. Arif, Supervised and unsupervised algorithms for bioinformatics and data science, *Progr. Biophys. Mol. Biol.* **151** (2020) 14–22.
10. L. Sherin *et al.*, Facile green synthesis of silver nanoparticles using terminalia bellerica kernel extract for catalytic reduction of anthropogenic water pollutants, *Colloid Interface Sci. Commun.* **37** (2020) 100276.
11. Z. Yu, R. Arif, M. A. Fahmy and A. Sohail, Self organizing maps for the parametric analysis of COVID-19 SEIRS delayed model, *Chaos Solitons Fractals* **150** (2021) 111202.
12. L. Yuan, *Implementation of Self-organizing Maps with Python* (University of Rhode Island, 2018).
13. M. R. Mahmoudi, D. Baleanu, Z. Mansor, B. A. Tuan and K.-H. Pho, Fuzzy clustering method to compare the spread rate of COVID-19 in the high risks countries, *Chaos Solitons Fractals* **140** (2020) 110230.
14. S. Yadav, D. Kumar, J. Singh and D. Baleanu, Analysis and dynamics of fractional order COVID-19 model with memory effect, *Results Phys.* **24** (2021) 104017.
15. M. R. Mahmoudi, D. Baleanu, S. N. Qasem, A. Mosavi and S. S. Band, Fuzzy clustering to classify several time series models with fractional brownian motion errors, *Alexandria Eng. J.* **60**(1) (2021) 1137–1145.
16. R. Roscher, B. Bohn, M. F. Duarte and J. Garcke, Explainable machine learning for scientific insights and discoveries, *IEEE Access* **8** (2020) 42200–42216.
17. A. Sohail, Inference of biomedical data sets using bayesian machine learning, *Biomed. Eng. – Appl. Basis Commun.* **31**(4) (2019) 1950030.
18. A. Sohail, M. Noor, R. Ellahi and S. M. Sait, Concentration gradients of turbulent flows of viscous fluid in a multi-chambered reactor: Application of solar energy system in oil industry, *Sustain. Energy Technol. Assess.* **45** (2021) 101140.

19. T. Kohonen, E. Oja, O. Simula, A. Visa and J. Kangas, Engineering applications of the self-organizing map, *Proc. IEEE* **84**(10) (1996) 1358–1384.
20. Z. Ceylan, Estimation of COVID-19 prevalence in Italy, Spain, and France, *Sci. Total Environ.* **729** (2020) 138817.
21. D. Fanelli and F. Piazza, Analysis and forecast of COVID-19 spreading in China, Italy and France, *Chaos Solitons Fractals* **134** (2020) 109761.
22. Z. Yu, A. Sohail, A. Nutini and R. Arif, Delayed modeling approach to forecast the periodic behaviour of SARS-2, *Front. Mol. Biosci.* **7** (2020) 386.
23. Z. Yu, R. Ellahi, A. Nutini, A. Sohail and S. M. Sait, Modeling and simulations of CoViD-19 molecular mechanism induced by cytokines storm during SARS-CoV2 infection, *J. Mol. Liquids* **327** (2021) 114863.
24. M. Ghaseminezhad and A. Karami, A novel self-organizing map (SOM) neural network for discrete groups of data clustering, *Appl. Soft Comput.* **11**(4) (2011) 3771–3778.
25. M. Y. Kiang, Extending the Kohonen self-organizing map networks for clustering analysis, *Comput. Stat. Data Anal.* **38**(2) (2001) 161–180.
26. S. Papadimitriou, S. Mavroudi, L. Vladutu, G. Pavlides and A. Bezerianos, The supervised network self-organizing map for classification of large data sets, *Appl. Intelligence* **16**(3) (2002) 185–203.
27. F. M. Riese, S. Keller and S. Hinz, Supervised and semi-supervised self-organizing maps for regression and classification focusing on hyperspectral data, *Remote Sensing* **12**(1) (2020) 7.
28. L. Sherin, A. Sohail and S. Shujaat, Time-dependent AI-modeling of the anticancer efficacy of synthesized gallic acid analogues, *Comput. Biol. Chem.* **79** (2019) 137–146.
29. A. Sohail, M. Noor, S. Farwa, Z. Li and S. Akbar, Neural network forecasting of the flow regimens of viscous fluids in a 3D reactor, *Adv. Mech. Eng.* **13**(3) (2021) 1687814021999520.
30. Z. Yu *et al.*, SEI2RS malware propagation model considering two infection rates in cyber-physical systems, *Physica A* **597** (2022) 127207, doi: 10.1016/j.physa.2022.127207.

APPENDIX A. Selected Data Set

In the following table, selected data is presented (curtsy of (https://ourworldindata.org/covid-vaccinations?country = ITA), to give the reader insight of the frequencies of the variables. In the table below, T represents the total number of cases, ICU represents the total number of ICU cases and S represents the stingency index. The rows are taken from last 30 weeks (30×3 data set is presented from 54×10 data set).

T	ICU	S
245,331	46.2857	58.33
282,625	48.1429	55.0286
249,668	42.8571	49.3414
252,797	54.5714	51.4586
257,268	65.5714	54.63
265,337	76.4286	54.63
274,574	121.857	54.63
284,646	171.143	53.5714
294,803	213.857	47.22
306,190	248.286	47.22
320,122	294.143	47.22
391,139	426.571	62.3057
393,119	647	50.7943
486,825	1065.29	59.79
543,940	1446.57	57.1457
829,564	2404	71.0329
1,068,721	3147	79.2329
1,308,571	3684.71	79.63
1,509,356	3802.43	79.63
1,665,243	3594	79.63
1,790,372	3280	79.63
1,904,840	2889.29	79.63
2,008,118	2625	82.2757
2,105,806	2557.43	82.6743
2,220,345	2585.86	82.1457
2,335,719	2565.57	78.7
3,853,560	3788.43	123.671
2,880,403	2626.86	89.9429
2,610,007	2144.57	78.7
3,874,695	2984.29	114.549
2,780,207	2070.86	82.41