How to Predict the Spread of COVID-19

Eduard Dadyan (dadyan60@yandex.ru)
Financial University under the Government of the Russian Federation Moscow

Research Article

Keywords: time series, forecasting, neural networks, data preprocessing, training and control samples, coronavirus pandemics

Posted Date: September 29th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-941929/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License. Read Full License
Abstract

For analysis tasks, time counts are of interest – values recorded at some, usually equidistant, points in time. The calculation can be performed at various intervals: after a minute, an hour, a day, a week, a month, or a year, depending on how much detail the process should be analyzed. In time series analysis problems, we deal with discrete-time, when each observation of a parameter forms a time frame. The same can be said about the behavior of Covid-19 over time.

In this paper, we solve the problem of predicting Covid-19 diseases in the world using neural networks. This approach is useful when it is necessary to overcome difficulties related to non-stationarity, incompleteness, unknown distribution of data, or when statistical methods are not completely satisfactory. The problem of forecasting is solved with the help of the analytical platform Deductor Studio, developed by specialists of the company Intersoft Lab of the Russian Federation. When solving this problem, appropriate methods were used to clean the data from noise and anomalies, which ensured the quality of building a predictive model and obtaining forecast values for tens of days ahead. The principle of time series forecasting was also demonstrated: import, seasonal detection, cleaning, smoothing, building a predictive model, and predicting Covid-19 diseases in the world using neural technologies for 30 days ahead.

1. Introduction

Today, all over the world they are working to create mechanisms for detecting the spread of COVID-19 and its elimination. Predicting the spread of the disease can help solve this serious problem. Observation and analysis of the spread of coronavirus make it possible to assert that humanity is faced with a synchronized process. The data collected and used to develop forecasts is often a time series, that is, it describes the evolution of a process over time.

Therefore, to predict the process, it is possible to apply well-known forecasting methods with preliminary analysis and data processing, as well as using neural network technologies.

The goal of any forecast is to create a model that allows you to study the future and assess the trends of a factor. The quality of the forecast in this case depends on the presence of the background variable factor, the measurement error of the considered value, and other factors. Formally, the forecasting problem is formulated as follows: find a function f that allows us to estimate the value of the variable x at time \((t+d)\) by its \(N\) previous values, so that:

\[x(t+d) = f(x(t), x(t-1), ..., x(t-N+1)).\]

Usually, \(d\) is assumed to be equal to one, i.e., the function \(f\) predicts the next value of \(x\).

It is already clear that the coronavirus pandemic has affected the economies of all countries of the world. On the one hand, there is a need to solve the problems associated with reducing the consumption of almost all resources that form the basis of the country’s export potential. On the other hand, it is necessary to solve the problem of stimulating the production and consumption of goods and services in the country. In this situation, it is important to obtain predicted values of the COVID-19 coronavirus infection process for specific dates.

When analyzing data, forecasting can predict some unknown quantity from a set of related values. Hence, forecasting is performed using data mining tasks such as regression, classification, and clustering.

Predicting the spread of coronavirus is essential in developing protective measures and behavioral measures for the population. The problem with modeling such a system is that every day COVID-19 and the number of new potential cases cannot be determined in a simple mathematical equation. There are many reasons for such problems. The spread of human filaments generally depends on various features, depending on both human behavior and the coronavirus’s biological structure. In any case, research needs to be done to biologically describe the coronavirus to develop a medical treatment and model the spread that will help prevent new cases and focus on the places with the greatest potential needs. According to [1], predicting the spread of coronavirus is very important for operational action planning. Unfortunately, coronaviruses are not easy to control, as the speed and reach of their spread depend on many factors, from environmental to social. In [1], the research results on developing a neural network model for predicting the spread of COVID-19 are presented. The prediction process itself is based on the classical approach of training a neural network with a deep architecture using the NAAdam training model. For training, the authors of the article used official data from the government and open repositories.

In [2], deep learning was used to identify and diagnose patients with COVID-19 using x-ray images of the lungs. The authors presented two algorithms to diagnose the disease: the deep neural network (DNN) on the fractal feature of images and neural network (SNN) methods using lung images directly. The results show that the presented methods allow detecting infected areas of the lungs with high accuracy-83.84%.

Several works are devoted to COVID-19 disease detection using neural networks. The authors of [3], [4], [5] propose a method based on a convolutional neural network (CNN) developed using the EfficientNet architecture for automated COVID-19 diagnostics. The architecture of a computerized medical diagnostics system is also proposed to support healthcare professionals in the decision-making process to diagnose diseases. Several important models have been introduced in recent months. In [6], machine learning was applied to evaluate how this stream's flash will take place. However, predicting the situation in the case of COVID-19 is not easy since many factors determine rapid changes [7]. Therefore, many approaches have been used to help. In [8], the flow prediction was performed using a mathematical model that evaluated undetected Chinese infections. Sometimes even elementary techniques are used. When a solution is needed immediately, we can start predicting based on preprocessing, in which some cases are simply removed for the applied model on the Euclidean network [9]. In Japan, prognostic models also evaluated the first symptoms of the disease [10]. One of Italy’s first models was the use of the Gauss error function and Monte Carlo simulation on registered cases [11]. Stochastic predictors also provide potential help in the early days when not much data is available for machine learning approaches [12]. Such stochastic models also seem to work even for huge societies, such as India [13]. Therefore, when artificial intelligence is applied in the first days of forecast periods, the results are mostly related to a single region or country. One of the first approaches for
China was presented in [14]. An interesting discussion of the principles of using mathematical modeling was presented in [15]. Some methodologies predict the number of new cases and make some assumptions about growth dynamics [16]. There are many sources of information for predicting the situation. As reported in [17], social networks can bring valuable information about confirmed cases of the disease and further spread. The relationship between new cases and the rate or coverage of growth can be transformed into a prediction elsewhere, as shown in [18]. This transfer of knowledge to model another region was carried out between Italy and Hunan province in China. The case of the ship "Diamond Princess" was discussed in [19]. Some models assess the situation in larger regions or more than one country. In [20] and [21], and applied forecasting model was defined for working with data from China, Italy, and France. Some models only consider the total number of cases worldwide as a whole [22]. The model proposed in [23] is a complex solution. The proposed neural network architecture was developed to forecast new cases in various countries and regions. The architecture consists of seven layers, and the output predicts the number of new cases. In [24], a shallow long short-term memory (LSTM) based neural network was used to predict the risk category by country. The results show that the proposed pipeline outperforms state-of-the-art methods for data of 180 countries and can be a useful tool for such risk categorization. In [25], a combination of the LSTM-SAE network model, clustering of the world's regions, and Modified Auto-Encoder networks were used to predict future COVID-19 cases for Brazilian states. A comprehensive review of artificial intelligence and nature-Inspired computing models is presented in [26]. To predict time-dependent processes, one can, among other things, use an adaptive network based on the fuzzy inference system (adaptive neuro-fuzzy inference system) ANFIS — an artificial neural network based on a fuzzy inference system that was developed in the early 1990s [27]. ANFIS integrates the principles of neural networks with the principles of fuzzy logic. To use ANFIS most efficiently and optimally, some authors recommend using the parameters obtained using the genetic algorithm [28].

2. Methods

The solution of the forecasting problem using a trained neural network presupposes, first of all, the availability of statistical data on the spread of this disease by day, provided by the Federal Service for Surveillance on Consumer Rights Protection and Human Well-being (https://yandex.ru/covid19/stat?utm_source=mainnotif & geoid = 1) for the world as a whole (Table 1).
### Table 1
Statistics of Covid-19 infections in the world by days from May 6, 2020, to March 27, 2021.

| Date       | Infections | Date | Infections | Date       | Infections | Date       | Infections | Date       | Infections | Date       | Infections | Date       | Infections |
|------------|------------|------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|
| 6-May      | 89958      | 10-Jun | 134573     | 15-Jul     | 231222     | 19-Aug     | 247589     | 23-Sep     | 267074     | 28-Oct     | 516386     | 2-Dec      | 646584     |
| 7-May      | 89075      | 11-Jun | 138182     | 16-Jul     | 252668     | 20-Aug     | 267333     | 24-Sep     | 361019     | 29-Oct     | 541755     | 3-Dec      | 690367     |
| 8-May      | 91280      | 12-Jun | 128598     | 17-Jul     | 242254     | 21-Aug     | 269856     | 25-Sep     | 330375     | 30-Oct     | 570759     | 4-Dec      | 678344     |
| 9-May      | 84934      | 13-Jun | 135456     | 18-Jul     | 236885     | 22-Aug     | 265492     | 26-Sep     | 277478     | 31-Oct     | 476619     | 5-Dec      | 640731     |
| 10-May     | 75215      | 14-Jun | 132757     | 19-Jul     | 214165     | 23-Aug     | 215236     | 27-Sep     | 240257     | 1-Nov      | 438987     | 6-Dec      | 533695     |
| 11-May     | 76919      | 15-Jun | 119817     | 20-Jul     | 206667     | 24-Aug     | 226410     | 28-Sep     | 277088     | 2-Nov      | 449556     | 7-Dec      | 517609     |
| 12-May     | 84295      | 16-Jun | 141900     | 21-Jul     | 233628     | 25-Aug     | 241987     | 29-Sep     | 284038     | 3-Nov      | 513543     | 8-Dec      | 634452     |
| 13-May     | 84847      | 17-Jun | 142557     | 22-Jul     | 280833     | 26-Aug     | 289000     | 30-Sep     | 327023     | 4-Nov      | 597627     | 9-Dec      | 668873     |
| 14-May     | 96766      | 18-Jun | 140396     | 23-Jul     | 283095     | 27-Aug     | 279244     | 1-Oct      | 317377     | 5-Nov      | 593225     | 10-Dec     | 703294     |
| 15-May     | 96428      | 19-Jun | 179899     | 24-Jul     | 280585     | 28-Aug     | 281663     | 2-Oct      | 295469     | 6-Nov      | 641152     | 11-Dec     | 696219     |
| 16-May     | 94222      | 20-Jun | 157326     | 25-Jul     | 255698     | 29-Aug     | 260990     | 3-Oct      | 329002     | 7-Nov      | 511574     | 12-Dec     | 623311     |
| 17-May     | 77424      | 21-Jun | 128345     | 26-Jul     | 213171     | 30-Aug     | 226389     | 4-Oct      | 247877     | 8-Nov      | 564822     | 13-Dec     | 547656     |
| 18-May     | 88731      | 22-Jun | 137933     | 27-Jul     | 226293     | 31-Aug     | 262534     | 5-Oct      | 330031     | 9-Nov      | 500379     | 14-Dec     | 523657     |
| 19-May     | 97233      | 23-Jun | 165369     | 28-Jul     | 252445     | 1-Sep      | 264298     | 6-Oct      | 323342     | 10-Nov     | 558531     | 15-Dec     | 625889     |
| 20-May     | 102811     | 24-Jun | 170976     | 29-Jul     | 289688     | 2-Sep      | 282813     | 7-Oct      | 349254     | 11-Nov     | 644668     | 16-Dec     | 734370     |
| 21-May     | 106605     | 25-Jun | 178360     | 30-Jul     | 280380     | 3-Sep      | 280936     | 8-Oct      | 359470     | 12-Nov     | 646835     | 17-Dec     | 735615     |
| 22-May     | 108651     | 26-Jun | 191430     | 31-Jul     | 290323     | 4-Sep      | 312839     | 9-Oct      | 360872     | 13-Nov     | 648440     | 18-Dec     | 717653     |
| 23-May     | 104687     | 27-Jun | 178576     | 1-Aug      | 250657     | 5-Sep      | 263292     | 10-Oct     | 330884     | 14-Nov     | 593817     | 19-Dec     | 616228     |
| 24-May     | 94566      | 28-Jun | 161892     | 2-Aug      | 228443     | 6-Sep      | 222126     | 11-Oct     | 268476     | 15-Nov     | 470256     | 20-Dec     | 534444     |
| 25-May     | 86395      | 29-Jun | 156229     | 3-Aug      | 222445     | 7-Sep      | 233799     | 12-Oct     | 326210     | 16-Nov     | 535993     | 21-Dec     | 541155     |
| 26-May     | 93312      | 30-Jun | 174257     | 4-Aug      | 258666     | 8-Sep      | 241048     | 13-Oct     | 328389     | 17-Nov     | 609487     | 22-Dec     | 646791     |
| 27-May     | 103060     | 1-Jul  | 217244     | 5-Aug      | 272290     | 9-Sep      | 284383     | 14-Oct     | 380616     | 18-Nov     | 623420     | 23-Dec     | 693002     |
| 28-May     | 119582     | 2-Jul  | 284877     | 6-Aug      | 284877     | 10-Sep     | 299295     | 15-Oct     | 406660     | 19-Nov     | 650567     | 24-Dec     | 671130     |
| 29-May     | 121042     | 3-Jul  | 202722     | 7-Aug      | 281070     | 11-Sep     | 319908     | 16-Oct     | 411337     | 20-Nov     | 666146     | 25-Dec     | 466942     |
| 30-May     | 137017     | 4-Jul  | 195833     | 8-Aug      | 259436     | 12-Sep     | 277345     | 17-Oct     | 341688     | 21-Nov     | 586304     | 26-Dec     | 512292     |
| 31-May     | 106656     | 5-Jul  | 182990     | 9-Aug      | 223626     | 13-Sep     | 236124     | 18-Oct     | 284809     | 22-Nov     | 483799     | 27-Dec     | 432576     |
| 1-Jun      | 95594      | 6-Jul  | 165649     | 10-Aug     | 227334     | 14-Sep     | 279748     | 19-Oct     | 439890     | 23-Nov     | 521723     | 28-Dec     | 490741     |
|            |            |       |            |            |            |            |            |            |            |            |            |            |            |
2.2. SMOOTHING AND REMOVAL OF ANOMALIES: SPECTRAL DATA PROCESSING

The purpose of spectral processing is to smooth ordered data sets using a wavelet or Fourier transform. The principle of such processing is to decompose the original time series function into basic functions. It is most often used for preliminary data preparation in forecasting tasks.

The analysis of the above points and numerous experiments allowed us to propose a General scheme for analytical processing of statistical source data to obtain the necessary data for the operation of the neural network dataset. This process usually includes the following steps:

- Time-series adjustment – smoothing and removing anomalies.
- Study of the time series, highlighting its components (trend, seasonality, cyclicity, noise) – autocorrelation analysis.
- Data processing using the sliding window method.
- Data processing using a multi-layer neural network, neural network training.
- Selecting the appropriate forecasting method.
- Assessment of the accuracy of forecasting and the adequacy of the chosen forecasting method.

The statistical data obtained in the form of a time series require significant processing to form a training sample of the neural network and obtain the necessary data for the operation of the neural network dataset. This process usually includes the following steps:

- Assessment of the accuracy of forecasting and the adequacy of the chosen forecasting method.
- Selecting the appropriate forecasting method.
- Data processing using a multi-layer neural network, neural network training.
- Study of the time series, highlighting its components (trend, seasonality, cyclicity, noise) – autocorrelation analysis.

2.1. ADJUSTMENT OF TIME SERIES

The graph of identified cases of COVID-19 coronavirus infection in the world as of October 23, 2020, is shown in Fig. 1. To obtain a forecast on the required scale, it is necessary to change the time scale of the data series: Year + Day, we will receive the corresponding graph of the initial data in the indicated scale (Fig. 2):

2.2. SMOOTHING AND REMOVAL OF ANOMALIES: SPECTRAL DATA PROCESSING

The purpose of spectral processing is to smooth ordered data sets using a wavelet or Fourier transform. The principle of such processing is to decompose the original time series function into basic functions. It is most often used for preliminary data preparation in forecasting tasks.

At the "Spectral Processing" step of the processing wizard, the "Wavelet Transform" method was selected, and the decomposition depth and order of the wavelet were set. The depth of decomposition determines the "scale" of the parts be filtered out: the larger this value, the «larger» parts in the source data will be discarded. If the parameter values are large enough (about 7–9), the data is not only cleared of noise but also smoothed (sharp outliers are "cut off"). Using too many decomposition depth values can lead to a loss of useful information due to too much "coarsening" of the data. The wavelet's order determines the smoothness of the reconstructed data series: the lower the parameter value, the more pronounced the "outliers" will be, and, conversely, if the parameter values are large, the "outliers" will be smoothed. Figure 3 shows a plot of smoothing and removal of anomalies using spectral processing using the "Wavelet transform" method and setting the average values of the parameters of this method.

2.3. AUTOCORRELATION ANALYSIS OF DATA
The purpose of autocorrelation analysis is to determine the degree of statistical dependence between different values (counts) of a random sequence formed by the data sample field. In the process of autocorrelation analysis, correlation coefficients (a measure of mutual dependence) are calculated for two sample values that are separated by a certain number of samples, also called lag. The set of correlation coefficients for all lags is an autocorrelation function of the series (ACF): 

\[ R(k) = \text{corr}(X(t), X(t+k)) \], where \( k > 0 \) is an integer (lag).

The ACF behavior can be used to judge the nature of the analyzed sequence, i.e. the degree of its smoothness and the presence of periodicity (for example, seasonal) or a trend.

For \( k = 0 \), the autocorrelation function will be maximal and equal to 1 as the number of lags increases, i.e. the distance between two values for which the correlation coefficient is calculated increases, the ACF value will decrease due to a decrease in the statistical interdependence between these values (the probability of occurrence of one of them less affects the probability of occurrence of the other). At the same time, the faster the ACF decreases, the faster the analyzed sequence changes. Conversely, if the ACF falls slowly, then the corresponding process is relatively smooth. If there is a trend in the original sample (a smooth increase or decrease in the series), then a smooth change in the ACF will also occur. If there are seasonal fluctuations in the original data set, the ACF will also have periodic spikes. Figure 4 shows a graph of the autocorrelation function of detected COVID-19 cases in the world. Using this graph, you can visually determine the trend on the curve with lags of 290.

### 2.4. DATA PROCESSING BY A SLIDING WINDOW

Data processing using the sliding window method is used for preprocessing data in forecasting tasks when the values of several neighboring samples of the original dataset must be fed to the input of the neural network. The term "sliding window" reflects the essence of processing - a specific continuous piece of data is selected, called a window. The window, in turn, moves, "slides" over the entire set of initial data. This operation results in a selection, in which each record contains a field corresponding to the current selection (it will have the same name as in the original selection), and to the left and right of it there are fields containing selections shifted from the current selection to the past and future accordingly.

The sliding window processing has two parameters: immersion depth - the number of samples in the "past" and the forecast horizon - the number of samples in the "future". The article used a sliding window method to smooth the plots of detected COVID-19 cases in a world with a depth of 282 immersion using spectral processing. The forecast horizon was taken equal to one. The result was a dataset for training a neural network. The Deductor Studio analytical neural network platform was chosen to predict coronavirus in the Russian Federation and Moscow under the current conditions (www.basegroup.ru). Deductor Studio is the analytical core of the Deductor platform. Deductor Studio contains a complete set of data import, processing, visualization, and export mechanisms for fast and efficient information analysis. It focuses on state-of-the-art methods for extracting, cleaning, manipulating, and visualizing data. With it, you can use modeling, forecasting, clustering, pattern search, and many other technologies for Knowledge Discovery in Databases and Data Mining. Deductor Studio includes a full set of mechanisms that allow you to get information from any data source, perform the entire processing cycle (cleaning, transforming data, building models), display the results most conveniently (OLAP, tables, charts, decision trees...) and export the results (https://docplayer.ru/49814110-Naznachenie-i-osnovnye-vozmozhnosti-analiticheskoy-platformy-deductor-studio.html).

Data processing is performed using a multi-layer neural network. In this mode, the "Processing Wizard" of the Deductor analytical platform allows you to construct a neural network with a given structure, determine its parameters and train it using one of the training algorithms available in the system. The result will be a neural network emulator that can be used to solve problems of forecasting, classification, finding hidden patterns, data compression, and many other applications [29].

Configuring and training a neural network consists of the following steps:

- setting up field assignments,
- adjust the normalization of the fields,
- setting up a training sample,
- configuring the structure of the neural network,
- selecting the algorithm and configuring the training parameters,
- setting the conditions for stopping training,
- starting the training process,
- selecting the data display method.

When configuring the neural network, in the "Neurons in layers" section, you must specify the number of hidden layers, i.e., the layers of the neural network located between the input and output layers. The number of neurons in the input and output layers is automatically set according to the number of input and output fields of the training sample, and it cannot be changed here.

The choice of the number of hidden layers and the number of neurons for each hidden layer should be approached carefully. It is believed that a problem of any complexity can be solved using a two-layer neural network [29]. When choosing the number of neurons, the following rule should be followed: “the number of connections between neurons should be about an order of magnitude less than the number of examples in the training set.” The number of connections is calculated as the connection of each neuron with all the neurons of the neighboring layers, including the connections on the input and output layers. Too
many neurons can lead to the so-called “overfitting” of the network when it produces good results on the examples included in the training sample but practically does not work on other examples.

In the "Activation function" section, you need to determine the type of neuron activation function and its steepness. To do this, in the "Function type" list, select the desired activation function, and in the "Steepness" field, set its steepness.

Neural networks differ from traditional statistical methods but may have some similarities with them. For example, a traditional linear regression model can acquire knowledge through the least-squares method and express this knowledge in regression coefficients. In this sense, the regression model can be considered as a neural network. Then we can say that linear regression is a special case of neural networks of a certain type. However, linear regression has several assumptions that are imposed before the information is extracted from the data - the hypothesis of a preliminary determination of the relationship between the dependent and independent variables is put forward. Instead, in neural networks, the shape of the relationship is determined during the learning process.

In this mode, the processing wizard allows you to define the structure of the neural network, determine its parameters and train it using one of the algorithms available in the system.

Configuring and training a neural network consists of the following steps:

1. 1. Configure field assignments. Here you need to determine how the fields of the source data set will be used when training the neural network and working with it in practice.
2. 2. Setting the normalization fields. The goal of normalizing field values is to transform data to the form that is most suitable for processing using a neural network.
3. 3. Setting the training sample. Here you need to divide the training sample for building a model based on a neural network into two sets – training and test. The training set includes the records that will be used as input data, as well as the corresponding desired output values.

The test set includes records that contain input and desired output values but are used to test the results of the model, rather than to train it.

1. 4. Adjust the structure of the neural network. At this stage, parameters are set that determine the structure of the neural network, such as the number of hidden layers and neurons in them, as well as the activation function of neurons. In the "neurons in layers" section, you need to set the number of hidden layers, that is, the layers of the neural network located between the input and output layers.
2. 5. The choice of algorithm and parameters training. At this step, we select the neural network training algorithm and set its parameters.
3. 6. Setting the conditions for stopping training. At this step, we set the conditions under which training will be terminated: the condition that the discrepancy between the reference and real network output becomes less than the specified value or the number of epochs (training cycles) after which training stops, regardless of the error value.
4. 7. Starting the learning process. At this step, we start the actual process of training the neural network.
5. 8. Choosing a way to display data. At this step, we choose how the imported data will be presented. In our case, the following specialized visualizers are of interest:

6. 8.1. Conjugacy table, scattering diagrams. The choice of an appropriate forecasting method is to determine whether this method produces satisfactory forecast errors. In addition to calculating errors, their comparison is carried out in a special Visualizer – the "scatter diagram" (Fig. 5). The scatter plot shows the output values for the training sample set (dataset) for the entire world.

The x-axis is the output value of the training sample (reference), and the y-axis is the output value calculated by the trained model using the same example. A straight diagonal line is a reference point (a line of ideal values). The closer the point is to this line, the smaller the model error.

The scatter plot allowed us to compare several models to determine which model provides the best accuracy on the training set.

8.2. Diagram. The graph displays the dependence of the values of one field on another. The most used chart type is a 2D graph. Its horizontal axis is the independent column values, and the vertical axis is the corresponding dependent column values.

After building a model for assessing the quality of training, we present the obtained data in the form of a diagram for the current and reference values of the set of the whole world (Fig. 6).

An analysis of the scatter plot (Fig. 5) and the trained neural network diagram for the entire world dataset (Fig. 6) suggests that the neural network has been successfully trained.

3. Results

Forecasting allows you to get a prediction of the values of a time series for the number of samples corresponding to the specified forecast horizon.

What is the maximum forecast horizon? The following rule is recommended: the amount of statistical data should be 10–15 times greater than the forecast horizon. This means that in our case, the maximum forecast horizon can be 30 days.

When performing the actual forecast, we pre-configure several fields: forecast horizon (set 20 days), request the "forecast step" and "source data" fields, and set color and scale parameters. Adding the "forecast step" field (check the box) allows you to add the "forecast Step" field to the resulting selection, which will indicate the number of the forecast step that resulted in it for each record.
"Source data" – selecting this check box allows you to include in the resulting selection not only those records that contain the predicted values, but also all those that contain the source data. In this case, the records containing the forecast will be located at the end of the resulting selection.

The final graph for predicting the number of COVID-19 infections by date using neural technologies is shown in Figs. 7 (worldwide). The proposed model for predicting the number of COVID-19 infections by date using neural technologies, built once, cannot "work" indefinitely. There are new data on the number of infections in the world. Therefore, the model should be periodically reviewed and retrained.

4. Conclusion

In this paper, we solve the problem of predicting COVID-19 diseases in the world using neural networks. This approach is useful when it is necessary to overcome difficulties related to non-stationarity, incompleteness, unknown distribution of data, or when statistical methods are not satisfactory. The forecasting problem is solved using the analytical platform Deductor Studio, developed by BaseGroup Labs (www.basegroup.ru, Russian Federation, Ryazan). When solving this problem, we used mechanisms for cleaning data from noise and anomalies, ensuring the quality of building a predictive model and obtaining forecast values for tens of days ahead. The principle of time series forecasting was also demonstrated: import, seasonal detection, cleaning, smoothing, building a predictive model, and predicting COVID-19 diseases in the world using neural technologies for thirty days.

Declarations

Ethics approval and consent to participation:
I approve of all the ethical standards of the magazine, and I am ready to take any part.

Consent to publication:
I am the sole author and I agree and ask you to publish my work.

Availability of data and materials:
All the initial data for training a neural network are presented in Table 1 and are obtained from the official link
https://yandex.ru/covid19/stat?utm_source=%20main_notif%20&%20geold%20=%201
and in section 2. METHODS.

Competing Interests - "Not applicable" for this section.

Funding - "Not applicable" for this section.

Authors’ contribution:
The entire article (collection of initial data, processing, writing of the text, drawings, table, study of literature and other works) was performed by one author Eduard Dadyan.

References

1. Judy Yong Jun Lin, Shaoyan Guo, Longshan Xia, Gu Xu, Multi-projection of unequal dimension optimal transport theory for Generative Adversary Networks, Neural Networks, Volume 128, August 2020, Pages 107-125. https://www.sciencedirect.com/science/article/pii/S089360802030157X

2. Yong Shi, Jiabin Liu, Bo Wang, Zhiquan Qi, Yingjie Tian, Deep learning from label proportions with labeled samples, Neural Networks, Volume 128, August 2020, Pages 73-81. https://doi.org/10.1016/j.neunet.2020.04.026

3. M.A. Rahman, M.S. Hossain, N.A. Alrajeh and N. Guizani, "5G and Explainable Deep Learning Assisted Healthcare Vertical at the Edge: COVID-19 Perspective," IEEE Network, vol. 34, no. 4, pp. 98-105, July/August 2020.

4. A. Zhao, G. Balakrishnan and F. Durand et al., Data augmentation using learned transformations for one-shot medical image segmentation, in Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 43-53.

5. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed and D. Anguelov et al., Going deeper with convolutions, in Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1-9.

6. P. Chudzik, S. Majumdar, F. Caliva, B. Al-Diri, A. Hunter, Exudate segmentation using fully convolutional neural networks and inception modules, Medical Imaging 2018: Image Processing, 10574, SPIE (2018), pp. 785-792.

7. R. Zhang, Y. Zheng, C.C. Poon, D. Shen, J.Y. Lau, Polyp detection during colonoscopy using a regression-based convolutional neural network with a tracker, Pattern Recognition, 83 (2018), pp. 209-219.

8. IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770-778.
9. R. Hadsell, S. Chopra, and Y. l'Cun, Dimensionality reduction by learning an invariant mapping, in Proc. of 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’06), New York, NY, USA, 2006, pp. 1735-1742.

10. F. Chollet, Xception: Deep learning with depthwise separable convolutions, in Proc. Of 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1800-1807, 2017.

11. M.S. Hossain, G. Muhammad, and N. Guizani, Explainable AI and mass surveillance system-based healthcare framework to combat COVID-19 like pandemics, IEEE Network, vol. 34, no. 4, pp.126-132, 2020.

12. Deng, W. Dong, R. Sucher, L.J. Li, K. Li et al., ImageNet: A large-scale hierarchical image database, in Proc. of 2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 248-255, 2009.

13. Marcos Eduardo Valle, Rodolfo Anibal Lobo, Quaternion-valued recurrent projection neural networks on unit quaternions, Theoretical Computer Science, Theoretical Computer Science, Volume 843, 2 December 2020, Pages 136-152. https://doi.org/10.1016/j.tcs.2020.101748

14. Carmen-Pilar Martí-Ballester, Do European renewable energy mutual funds foster the transition to a low-carbon economy? Renewable Energy, Volume 143, December 2019, Pages 1299-1309

15. S. Zhang, M. Diao, W. Yu, L. Pei, Z. Lin, D. Chen, Estimation of the reproductive number of novel coronavirus (COVID-19) and the probable outbreak size on the diamond princess cruise ship: a data-driven analysis, Int J Infect Dis, 93 (2020), pp. 201-204

16. A. Atangana, Fractal-fractional differentiation, and integration: connecting fractal calculus and fractional calculus to predict complex system, Chaos Solitons Fractals, 102 (2017), pp. 396-406

17. D. Baleanu, G.-C. Wu, S.-D. Zeng, Chaos analysis and asymptotic stability of generalized Caputo fractional differential equations, Chaos Solitons Fractals, 102 (2017), pp. 99-105

18. Cirillo P., Taleb N.N., Tail risk of contagious diseases Nature Physics, 16 (2020), pp. 606-613, 10.1038/s41567-020-0921-x

19. Van den Driessche P., Watmough J. Reproduction numbers and sub-threshold endemic equilibria for compartmental models of disease transmission Mathematical Biosciences, 180 (1) (2002), pp. 29-48, 10.1016/S0025-5564(02)00108-6 http://www.sciencedirect.com/science/article/pii/S0025556402001086

20. Elman, 1990, Elman J.L., Finding structure in time, Cognitive Science, 14 (2) (1990), pp. 179-211

21. Gers F.A., Schmidhuber J., LSTM Recurrent networks learn simple context-free and context-sensitive languages, IEEE Transactions on Neural Networks, 12 6 (2001), pp. 1333-1340

22. Gompertz B., On the nature of the function expressive of the law of human mortality, and on a new mode of determining the value of life contingencies, Philosophical Transactions of the Royal Society of London, 115 (1825), pp. 513-583

23. Kiskowski M., Chowell G., Modeling household and community transmission of Ebola virus disease: Epidemic growth, spatial dynamics, and insights for epidemic control, Virulence, 7 (2) (2016), pp. 163-173, 10.1080/21505594.2015.1076613 PMID: 26399855

24. Hyndman R.J., Koehler A.B., Another look at measures of forecast accuracy International Journal of Forecasting, 22 (4) (2006), pp. 679-688, 10.1016/j.ijforecast.2006.03.001 http://www.sciencedirect.com/science/article/pii/S0169207006000239

25. Paré PE., Liu J., Beck C.L., Kirwan B.E., Başar T., Analysis, estimation, and validation of discrete-time epidemic processes, IEEE Transactions on Control Systems Technology, 28 (1) (2020), pp. 79-93

26. Makridakis S., Spiliotis E., Assimakopoulos V., The m4 competition: 100,000-time series and forecasting methods International Journal of Forecasting, 36 (1) (2020), pp. 54-74, 10.1016/j.ijforecast.2019.04.014 M4 Competition. http://www.sciencedirect.com/science/article/pii/S0169207019301128

27. Jang, Jyh-Shing R (1991). "Fuzzy Modeling Using Generalized Neural Networks and Kalman Filter Algorithm" in Proceedings of the 9th National Conference on Artificial Intelligence, Anaheim, CA, USA, July 14–19. 2: 762–767.

28. Tahmasebi, P A hybrid neural networks-fuzzy logic-genetic algorithm for grade estimation // Computers & Geosciences: journal. – 2012. – Vol. 42. – P. 18–27

29. Kirichenko A.A., Neuropackets – a modern intellectual tool of the researcher, 2013. Online electronic edition of the textbook (https://publications.hse.ru/mirror/pubs/share/folder/wk780k9og9/direct/91940629)
Figure 1

Block diagram of the dataset generation algorithm for the neural network and prediction of cases of COVID-19 coronavirus infection
Figure 2

Schedule of detected cases of COVID-19 coronavirus infection in the world by dates as of March 27, 2021, in the "date: year + day" scale

Figure 3
Graph of detected cases of COVID-19 coronavirus infection in the world, smoothed using spectral processing using the "Wavelet transform" method

**Figure 4**

A graph of autocorrelation functions detected cases of coronavirus infection COVID-19 in the world

**Figure 5**

Scattering diagram of a trained neural network for the whole world dataset
Figure 6

Schematic of a trained neural network for a dataset the whole world.

Figure 7

Graph for predicting the number of COVID-19 infections by date using neural technologies (worldwide)

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- REFERENCE.pdf