Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 108C (2017) 2363–2367

Procedia Computer Science

International Conference on Computational Science, ICCS 2017, 12-14 June 2017, Zurich, Switzerland

Influenza peaks forecasting in Russia: assessing the applicability of statistical methods

Vasiliy N. Leonenko[1], Klavdiya O. Bochenina[2], and Sergey A. Kesarev[3]

[1] ITMO University, Saint Petersburg, Russian Federation
vnleonenko@yandex.ru
[2] ITMO University, Saint Petersburg, Russian Federation
k.bochenina@gmail.com
[3] ITMO University, Saint Petersburg, Russian Federation
kesarevs@gmail.com

Abstract

This paper compares the accuracy of different statistical methods for influenza peaks prediction. The research demonstrates the performance of long short-term memory neural networks along with the results obtained by proved statistical methods, which are the Serfling model, averaging and point-to-point estimates. The prediction accuracy of the methods is compared with the accuracy of the modeling approaches. Possible applications of the methods and the ways to improve their accuracy are discussed.

Peer-review under responsibility of the scientific committee of the International Conference on Computational Science

Keywords: data analysis, mathematical epidemiology, acute respiratory infection, seasonal influenza, machine learning, Python

1 Introduction and motivation

The problem of predicting the outbreaks of acute respiratory infections (ARIs) and influenza is one of the most important in the field of mathematical epidemiology due to the worldwide spread of these diseases. In the late 1960's, the first mathematical models of influenza outbreaks were created. One of the most prominent studies of that time, accomplished by Baroyan and Rvachev, was connected with the flu propagation within the cities in the Soviet Union [1]. From early 1980's, the Soviet modeling complex for flu forecasting showed some incoherence with the epidemic outbreak patterns observed in Soviet cities [6]. The experiments with the SEIR models calibration to the contemporary Russian ARI incidence data available [11], [12] demonstrated that the models allow predicting an epidemic peak height with satisfactory accuracy. However, the accuracy is low for an epidemic peak day prediction.

The question arises whether it is better to rely on the statistical methods based on historical data rather than on the modeling approaches while making predictions in the situation of sophisticated and heterogeneous disease dynamics. As known, statistical approaches taking the

1877-0509
Peer-review under responsibility of the scientific committee of the International Conference on Computational Science
10.1016/j.procs.2017.05.196

last season peak data and the average peak data as the peak prediction for the current season, can demonstrate the same accuracy as the modeling prediction for the day peak [11]. In fact, the reason is not the high accuracy of such a naive approach, but the low accuracy of the modeling prediction of this parameter. The aim of this paper is to apply more sophisticated statistical techniques to the same ARI dataset and compare the resulting accuracy with the modeling approaches demonstrated in [11], [12].

2 Methods review

The two most popular families of non-parametric methods that can be used for epidemic peaks prediction are Bayesian and machine learning (ML) methods. The example of Bayesian approach is given in [2], [7], and the approach to outbreak forecasting on the basis of classification techniques (belong to ML methods) is presented in [13]. However, as we do not possess the type of input data which was used in the mentioned works, the techniques described cannot be applied to our case. The application of the Long-Short Term Memory (LSTM) neural networks seems more promising, as they do not require large synthetic data sets as an input. Moreover, they are capable of learning long-term dependencies without preliminary assumptions on the structure of underlying process. The example of LSTM application is [16], where the authors used LSTM networks to perform sequence-to-sequence weather forecasting. The approach used in [16] requires the input and output sequences to be of the same length. However, the limitation of fixed dimensionality can be avoided by using two LSTMs (one as encoder, and one as decoder) as it is shown in [15].

3 Numerical experiment

Algorithm details. The input dataset is a time series with daily ARI incidence observations within the period from 1985 to 2015. With the aim of synthesizing large training samples we slice the data using the running window with a small step s. This way yields the dataset of around four thousand rows for each city if $s = 1$. The main drawback of this idea is that the adjacent samples differ only in s observations. LSTM-based neural networks are sensitive to the scale of the input data [3], so we work with normalized data, using the inverse transform to return to the initial scale. Denote the whole dataset as $T = (t_1, \ldots, t_N)$ and set the step size s (in our implementation $s = 1$). Then kth sample in the training set is $x_k = (t_{sk}, t_{sk+1}, \ldots, t_{sk+364})$ and the output for x_k is $y_k = (t_{sk+365}, t_{sk+365+1}, \ldots, t_{sk+365+364})$. The test set contains one sample: the year before the last as x_1 and the last year as y_1. No observations from the test set are used in the training set.

The architecture of our network has the following structure. The input layer of the network takes a matrix of shape 1×365 (which is essentially x_k). The linear combinations of these inputs are sent to 128 LSTM cells. Fully-connected layer maps the time dependencies resolved by LSTM cells back to the space of scaled influenza levels per each day of the year. The fact that our output is within $[0; 1]$ range allows us to use sigmoid activation function for the outer layer. For the training procedure the batch size 32 was used. The network was implemented using Keras library with TensorFlow backend. Initial weights are set to random numbers. We employ three different loss functions. The first one is the classical mean square error (MSE): $\mathrm{MSE}(\hat{y}_k, y_k) = \frac{1}{2} \left\| \hat{y}_k - y_k \right\|^2$. The second one, softmax, enforces the network to predict peak positions more precisely by suppressing the values which are significantly below the maximum: $\mathrm{softmax}(x)_i = \frac{e^{x_i}}{\sum_j e^{x_j}}$, $\mathrm{SF\,LOSS}(\hat{y}_k, y_k) = \frac{1}{2} \left\| \mathrm{softmax}(\hat{y}_k) - \mathrm{softmax}(y_k) \right\|^2$. The third one is