

Improving Quality of Service Parameter Prediction with Preliminary Outlier Detection and Elimination

L. Kovács, D. Vass and A. Vidács

*Budapest University of Technology and Economics
Dept. of Telecommunications and Media Informatics
Magyar tudósok körútja 2,
Budapest, H-1117 Hungary*

E-mail: {kovacs.l,vass,vidacs}@alpha.tmit.bme.hu

Abstract

Wide-spread real-time applications make it necessary for service providers to guarantee QoS parameters. This requires precise forecast of network traffic. A possible method of the forecast is measuring traffic then analyzing it and fitting model to the measured data, finally predicting the observed parameter using the fitted model. The efficiency of the prediction is decreased by outlying samples (so called outliers) found in the time series data. We developed a new tool that is able to detect and eliminate the outliers from time series data. This tool is capable to handle large sets of time series data fast and efficiently. We also propose a method to predict QoS parameters using the ARIMA (Auto-Regressive Integrated Moving Average) model, which is based on a preliminary detection and elimination of outliers. We have proven that this method increases the efficiency of the prediction significantly by forecasting real measurement data.

1. Introduction

Nowadays appropriate Quality of Service parameters guaranteed by Service Providers are more and more important for end users. Service Providers need to forecast Quality of Service parameters as precisely as possible based on real measurement data in order to plan the assignment of resources, applications and users. The prediction has to consider that most real measurement data series contain outliers that are extreme fluctuations caused by local random events. Outliers are specific patterns (extreme values) that do not follow the characteristic distribution of the rest of the samples. Outliers can have significant impact on the statistical estimates in data analysis and modelling.

The method we propose combines the preliminary outlier detection and elimination with the forecasting of QoS parameter values based on the Auto-Regressive Integrated Moving Average (ARIMA) modeling technique [1]. It is used in different researches for modeling and forecasting traffic and QoS parameter values in telecommunication networks. ARIMA is a linear time series forecasting model, because it assumes that the dependency of a future value on the past values is linear.

Similar approach in economy for financial time series data is proposed by [4] and realized in the public domain economical software [10]. Being designed for economical purposes this software is capable to handle only a small set of data, but during measurements in telecommunication networks often large sets of data are produced. We have developed a new tool that is capable to handle large sets of data. Our software can detect and eliminate outliers in time series data fast and efficiently. It replaces outliers with more appropriate values for the forecasting. Our algorithm implemented in the software is based on the so called L.O.C.I. algorithm proposed in [13]. The program is able to set the parameters automatically so it does not require user interaction. The algorithm was validated using synthetic and measured data sets, and the results show that the detection and elimination of outliers is fast and efficient.

This paper is organized as follows. Section 2 is about the detection of outliers in time series data. We present our new algorithm (based on the so-called L.O.C.I. algorithm) in section 3. Section 4 introduces the tool we have developed, and validates it. We examine the effect of the preliminary outlier detection and elimination on the efficiency of the prediction in section 5. Section 6 concludes this paper with further outlook.

2. Outlier detection in time series data

Intuitively, outliers can be defined as given by Hawkins [5]: “An outlier is an observation that deviates so much from other observations as to arouse suspicion that it was generated by a different mechanism.” Outliers are outlying samples found in time series data which are useless for forecasting. These pieces of data are produced by random errors (e.g., route failures, misconfiguration, etc.). Outliers can have significant impact on the estimates of the model parameters of the time series data.

The existing approaches to outlier detection can be classified into the following five categories:

- *Distribution-based approach*: These algorithms contain some standard distribution model (Normal, Poisson, etc.) and recognized as outliers those objects which deviate from the model [14]. Their greatest disadvantage is that the distribution of the measurement data is unknown in practice. Often a large number of tests are required in order to decide which distribution model the measurement data follow (if there is any).
- *Depth-based approach*: This is based on computational geometry and computes different layers of k -d convex hulls [7]. Objects in the outer layer are detected as outliers. However, it is a well-known fact that the algorithms employed suffer from the dimensionality curse and cannot cope with large k .
- *Clustering approach*: These algorithms classify the input data. They detect outliers as by-products [6]. However, since the main objective is clustering, they are not optimized for outlier detection.
- *Distance-based approach*: This was originally proposed by Knorr and Ng [8, 9]. An object in a data set P is a distance-based outlier if at least a fraction b of the objects in P are further than r from it. This outlier definition is based on a single, global criterion determined by the parameters r and b . Problems may occur if the parameters of the data are very different from each other in different regions of the data set.
- *Density-based approach*: This algorithm was proposed by Breunig [2]. This procedure assigns a Local Outlier Factor (LOF) to each sample based on their local neighborhood density. Samples with high LOF value are identified as outliers. The disadvantage of this solution is that it is very sensitive to parameters defining the neighborhood.

Our algorithm described in section 3 combines the advantages of the distance-based and density-based approaches.

3. Our algorithm

Our algorithm implemented in the software is based on the so called L.O.C.I. algorithm proposed in [13]. We combined the advantages of the distance-based and density-based approaches. We examine only a small neighborhood of the given point, this way problems will not occur if the parameters of the data are very different from each other in different regions of the data set. Within the neighborhood we decide whether the sample is outlier or not based on the statistical characteristics of the members of the neighborhood. Our algorithm works with two different kinds of neighborhood.

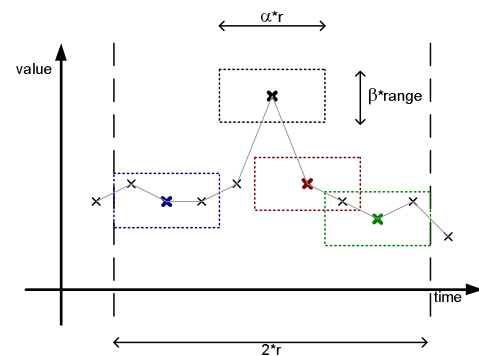


Figure 1. Neighborhoods.

The sampling neighborhood or r -neighborhood (see Figure 1) is a $2r$ wide interval. In Figure 1 the actually examined point is denoted by a bold x . The dashed lines are the borders of the sampling neighborhood. For each sample of this region a smaller neighborhood is defined. This is the counting neighborhood. For some points on the figure the counting neighborhood is shown (dotted line). This neighborhood is characterized by two parameters: α , β . In this region the density of the neighborhood is examined. If the neighborhood-density of the actually examined point is significantly different from the average neighborhood-density over the sampling neighborhood the point is recognized as an outlier. The neighborhood-density is basically determined by the number of the samples in the counting neighborhood. So far the algorithm has three main parameters. The r parameter is the radius of the sampling neighborhood. The α and β parameters are necessary for the definition of the counting neighborhood. Besides we need one more parameter (k), which determines the “strictness” of the detection. Table 1 contains the basic definitions needed to define the k parameter.

For any p_i , r , α and β we define the deviation factor (DF) as:

$$DF(p_i, r, \alpha, \beta) = \frac{\hat{n}(p_i, r, \alpha, \beta) - n(p_i, r, \alpha, \beta)}{\hat{n}(p_i, r, \alpha, \beta)}. \quad (1)$$

P, p_i	The data set. $P = (p_1, p_2, \dots, p_N)$
$N(p_i, r)$	The set of r -neighbors of p_i . $N(p_i, r) = \{p_x \in P (i - r) \leq x \leq (i + r)\}$
$n(p_i, r)$	The number of r -neighbors. $n(p_i, r) \equiv N(p_i, r) $
$n(p, \alpha, \beta)$	The number of samples in the counting neighborhood of p .
$\hat{n}(p_i, r, \alpha, \beta)$	Average of $n(p, \alpha, \beta)$ over the set of r -neighbors of p_i $\hat{n}(p_i, r, \alpha, \beta) = \frac{\sum_{p \in N(p_i, r)} n(p, \alpha, \beta)}{n(p_i, r)}$
$\sigma_{\hat{n}}(p_i, r, \alpha, \beta)$	Standard deviation of $n(p, \alpha, \beta)$ over the set of r -neighbors of p_i $\sigma_{\hat{n}}(p_i, r, \alpha, \beta) = \sqrt{\frac{\sum_{p \in N(p_i, r)} (n(p, \alpha, \beta) - \hat{n}(p_i, r, \alpha, \beta))^2}{n(p_i, r)}}$
$DF(p_i, r, \alpha, \beta)$	Deviation Factor for p_i . In details see below.
$\sigma_{DF}(p_i, r, \alpha, \beta)$	Normalized deviation (thus, directly comparable to DF). In details see below.

Table 1. Symbols and definitions

Note that the r -neighborhood for an object p_i always contains p_i . This implies that the denominator of the above fraction is always greater than zero and so the above quantity is always defined.

We define $\sigma_{DF}(p_i, r, \alpha, \beta)$ as the normalized standard deviation of $n(p, \alpha, \beta)$ for $p \in N(p_i, r)$ as

$$\sigma_{DF}(p_i, r, \alpha, \beta) = \frac{\sigma_{\hat{n}}(p_i, r, \alpha, \beta)}{\hat{n}(p_i, r, \alpha, \beta)}. \quad (2)$$

Points are flagged as outliers if

$$DF(p_i, r, \alpha, \beta) \geq k \cdot \sigma_{DF}(p_i, r, \alpha, \beta). \quad (3)$$

Thus, as we mentioned above, parameter k determines the “strictness” of the detection.

Let k be about 3 and let us suppose that the distribution of the samples is normal. In this case less than 1% is the chance that a sample (which is not outlier) lies outside the triple deviation (and is flagged as outlier). Automatically the parameter k is set to 2.8. If the value of parameter r is too low, we do not have enough data to determine the deviation exactly. If the value is too high then the detection

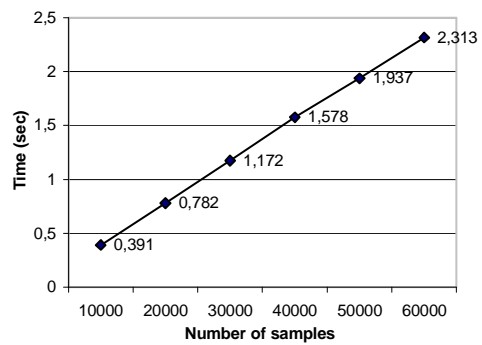
might be incorrect because of the different local characteristics. In default settings the value of r is 10% of the number of the samples but maximum 50.

For the results presented below the values of parameters α and β are the same. If the values of these parameters are too low it can happen that even correct samples will not have neighbors, while if the values are too high almost every sample belongs to the neighborhood so the outlier will not be outside the triple deviation. In default settings the values of α and β are between 0.1 and 0.05 depending on the value of r .

The automatic parameter setting are not optimal yet, but we still achieved good results (see below). However, there are problems in fast changing time series that needs to be examined in more detail.

4. Outlier Detection and Remove Tool

We have developed a new tool (Outlier Detection and Remove Tool) that is able to handle large sets of data fast and efficiently. This program runs under Microsoft Windows. The software is able to set the parameters of the algorithm automatically, so it can handle large sets of data efficiently without user interactions. In order to validate the algorithm we have made a large number of tests. During the tests we used the automatic parameter settings. We used many ARIMA time series synthetic data with different parameters. To these time series data additive outliers were added by a special program made for this purpose. So we could examine how efficient the algorithm is in the different cases, what percent of the added outliers is recognized. During the validation we examined the speed of the algorithm (see Figure 2), the effect of the number of outliers on the detection, the effect of the distribution of the outliers on the detection, and some special cases (e.g., negative outliers, level-shift outliers, etc.).


Figure 2. The speed of the algorithm.

We examined the speed of the algorithm using automatic parameter settings. If the number of the samples is more

than 500 the parameters are fix values, so the speed of the algorithm is linear. (In this scenario the hardware was: Intel P4 2,4GHz processor, ABIT BD7II motherboard, 256 MB RAM.)

When we examined the effect of the number of outliers on the detection, the number of the added outliers was 4-13% of the number of the samples. We made 200 tests, the results are shown in Figure 3.

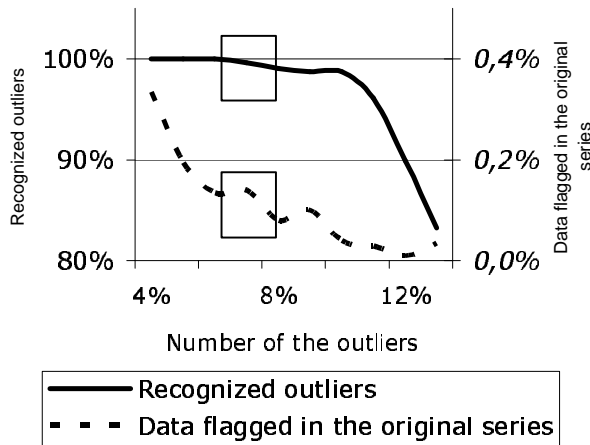


Figure 3. The effect of the number of outliers.

Solid line and the left hand side ordinate denote the recognized outliers, dotted line and the right hand side ordinate denote the data flagged in the original series. Based on the results we concluded that in the interval 5-10% of added outliers the algorithm recognizes almost 100% of the outliers while less than 0.2% of the original series data is flagged as outliers.

5. QoS parameter prediction

Since the last decade, the use of real-time applications is rapidly spreading. To fulfill the request of the users, the network operators are required to provide the adequate service quality (QoS). To do this, it is important for internet service providers to forecast traffic parameters as accurate as possible based on real network measurements to plan resource distributions.

5.1. ARIMA modeling and forecasting

ARIMA models are useful for a wide variety of problems including forecasting, spectral elimination, as well as providing a summary of the data. Box and Jenkins (1976) give a comprehensive account of ARIMA modelling, and discussions of ARIMA models can be found in many recent textbooks for time series (see, for example [1, 3]). A

stationary autoregressive moving-average model (ARMA) combines the autoregressive (AR) and the moving average (MA) processes. An autoregressive process of order p (AR(p)) is defined by

$$x_t = c_1x_{t-1} + c_2x_{t-2} + \dots + c_px_{t-p} + \varepsilon_t, \quad (4)$$

where c_1, \dots, c_p are constants, and ε_t denotes a series of i.i.d. random variables with zero mean and variance σ^2 . A moving average process of order q (MA(q)) is defined by

$$x_t = d_1\varepsilon_{t-1} + d_2\varepsilon_{t-2} + \dots + d_q\varepsilon_{t-q} + \varepsilon_t, \quad (5)$$

with the d_1, \dots, d_q constants being the model parameters.

An ARMA(p,q) process is thus defined by

$$x_t = c_1x_{t-1} + c_2x_{t-2} + \dots + c_px_{t-p} + d_1\varepsilon_{t-1} + d_2\varepsilon_{t-2} + \dots + d_q\varepsilon_{t-q} + \varepsilon_t \quad (6)$$

Many time series encountered in practice are nonstationary. For these series, simple ARMA models are typically inadequate. However the differenced series may be stationary. Box and Jenkins (1976) developed a methodology for fitting ARMA models to differenced data. These are known as autoregressive integrated moving average (ARIMA) models [11]. For an ARIMA(p,d,q) process its d th difference is an ARMA(p,q) process. Yule-Walker equations algorithm is to be used in fitting the autoregression model. The automatic model identification procedure was realized using S-PLUS 2000 [11, 17].

5.2. QoS prediction with outlier removal

Most of data sets contain outliers as the consequences of local, irregular random events which are extreme, non interpretable values. With the detection and removal of these abnormal data, we can eliminate their influence on our forecast. Figure 4 illustrates the co-operation between Outlier Detection and Remove Tool (ODRT) and ARIMA based forecasting.

In the following we will demonstrate the improvement in our prediction with former outlier detection and removal. We performed our analysis on real network measured data sets. A part of the time series we have applied is based on real traffic measurement between two computers on the department. The other part of time series was provided by Salzburg Research, Austria. End-to-end delay is monitored between Salzburg Research and the network provider (Austria Telekom). The end-to-end delay measurement approach is based on GPS clock synchronisation [12]. In order to prove the efficiency of the method we have analyzed several data sets, each containing 1000 elements. Based on the first 95% of the data we have forecasted the last 50 values. During the outlier detection and the forecasting,

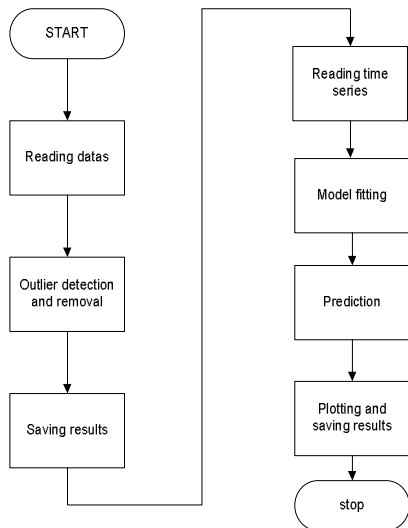


Figure 4. Co-operation between ODRT and ARIMA forecasting.

we used the automatic parameter settings. For specifying the order of the model, we received a relatively high (25-35) value. To evaluate the measurement results we compared three series in one chart: the original, the forecasted upon the original and the forecasted upon the cleaned series. First we demonstrate a case when the values, beside the outliers, are fluctuating in small range. The original and the filtered series are depicted in Figure 5.

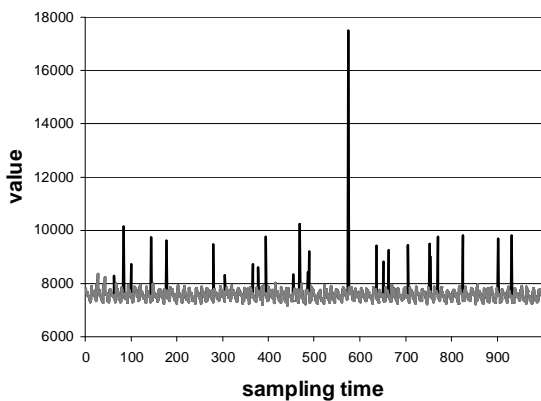


Figure 5. The original and the filtered series.

It can be seen from Figure 6 that even after 50 data, the results of the outlier detection method follows the original values well. Figure 6 shows the original series (thick grey line), the prediction without preliminary outlier detection (thin black line) and the prediction with preliminary out-

lier detection (thick black line). If we use the original data series as input of the method, the results do not follow the values of the original time series.

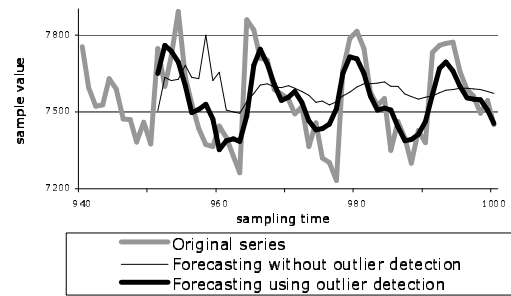


Figure 6. Influence of the outlier detection (1).

Figure 7 shows the original series (thick grey line), the prediction without preliminary outlier detection (thin black line) and the prediction with preliminary outlier detection (thick black line). The chart describes, that the prediction without preliminary outlier removal becomes a flat line, while if the outliers are removed in advance, the forecast follows the pattern of the original time series well.

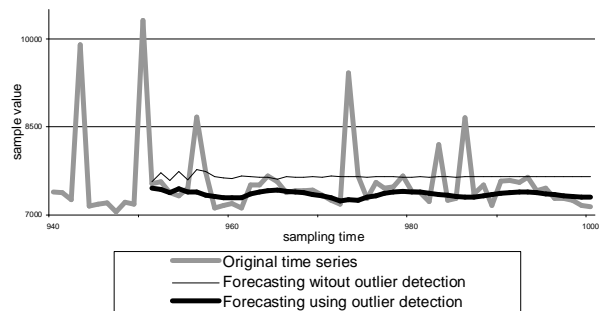


Figure 7. Influence of the outlier detection (2).

6. Conclusions and future work

The results show that the method of preliminary outlier detection significantly improves the prediction.

Until now we replace the value that was masked by the outlier with the mean of neighbour data. Instead, a linear prediction or other method could also be used to estimate the original masked value. To analyse the impact of this method is to be discussed in the future as well.

References

- [1] G. Box, G. Jenkins, and G. Reinsel. *Time Series Analysis: Forecasting and Control*. Prentice-Hall, New Jersey, 1994.
- [2] M. M. Breunig, H-P. Kriegel, R. T. Ng, and J. Sander. LOF: identifying density-based local outliers. In *Proc. of ACM SIGMOD Int. Conf. on Management of Data*, pages 93–104, Dallas, TX, 2000.
- [3] P. J. Brockwell and R. A. Davis. *Introduction to Time Series and Forecasting*. Springer Verlag, 2002.
- [4] V. Gómez and A. Maravall. Programs TRAMO (Time series Regression with Arima noise, Missing observations, and Outliers) and SEATS (Signal Extraction in Arima Time Series). Working Paper 9628, Servicio de Estudios, Banco de Espana, 1996.
- [5] D. Hawkins. *Identifications of Outliers*. Chapman & Hall, London, 1980.
- [6] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Computing Surveys*, 31(3):264–323, 1999.
- [7] T. Johnson, I. Kwok, and R. T. Ng. Fast computation of 2-dimensional depth contours. In *Knowledge Discovery and Data Mining*, pages 224–228, 1998.
- [8] E. M. Knorr and R. T. Ng. Algorithms for mining distance-based outliers in large datasets. In *Proc. 24th Int. Conf. Very Large Data Bases, VLDB*, pages 392–403, 24–27 1998.
- [9] E. M. Knorr, R. T. Ng, and V. Tucakov. Distance-based outliers: Algorithms and applications. *VLDB Journal: Very Large Data Bases*, 8(3–4):237–253, 2000.
- [10] A. Maravall and G. Caporello. A tool for Quality control of time series data, Program TERROR. In *Proc. of Challenges to Central Bank Statistical Activities*, Basel, August 2002.
- [11] MathSoft. *S-PLUS 2000: Guide to statistics*. 1999.
- [12] I. Miloucheva, A. Anzaloni, and E. Müller. A practical approach to forecast Quality of Service parameters considering outliers. In *Proc. 1st Int. Workshop on Inter-Domain Performance and Simulation (IPS2003)*, pages 163–172, Salzburg, Austria, February 2003.
- [13] S. Papadimitriou, H. Kitawaga, P. B. Gibbons, and C. Faloutsos. LOCI: Fast outlier detection using the Local Correlation Integral. Technical Report IRP-TR-02-09, Intel Research Laboratory, Pittsburg, July 2002.
- [14] P. J. Rousseeuw and A. M. Leroy. *Robust Regression and Outlier Detection*. John Wiley and Sons, 1987.
- [15] Adam Bernard Smith, Cristoph David Jones, and E F Roberts. Article title. *Journal*, 99(1):1–100, January 1999.
- [16] Adam Bernard Smith, Cristoph David Jones, and E F Roberts. *Book Title*. Publisher, Address, 1999.
- [17] W. N. Venables and B. D. Ripley. *Modern Applied Statistics with S-Plus*. Springer Verlag, 1998.