Raport Badawczy
Research Report

RB/28/2008

On online algorithms for piecewise linear segmentation

A. Wilbik

Instytut Badań Systemowych
Polska Akademia Nauk

Systems Research Institute
Polish Academy of Sciences

POLSKA AKADEMIA NAUK

Instytut Badań Systemowych

ul. Newelska 6

01-447 Warszawa

tel.: (+48) (22) 3810100

fax: (+48) (22) 3810105

Kierownik Pracowni zgłaszający pracę:
Prof. dr hab. inż. Janusz Kacprzyk

Warszawa 2008

On online algorithms for piecewise linear segmentation

Anna Wilbik

Systems Research Institute Polish Academy of Sciences,

Newelska 6, 01-447 Warsaw, Poland

wilbik@ibspan.waw.pl

Abstract

Recently, the more and more data are available, as e.g. time series data. Therefore there is a need for algorithms, that could process those data. Many methods for time series segmentation were proposed. In this paper we give our interest only to on-line algorithms for piecewise linear approximation of the series. Those methods are often used, they are simple, fast and intuitively appealing.

Keywords: time series segmentation, piecewise linear approximation

1 Introduction

A time series is a sequence of data points, measured typically at successive times, spaced at (often uniform) time intervals. There are two basic types of time series data: (1) continuous, where we have an observation at every instant of time, e.g. lie detectors, electrocardiograms, and (2) discrete, where we have an observation at (usually regularly)

spaced intervals. Time series data are often encountered in many fields such as: finance – weekly share prices, monthly profits, etc. [5], [6], [23]; medicine – e.g ECG or others [9], [18], [19]; meteorology – daily rainfall, wind speed, temperature [10], [17]; hydrology – river floods, etc. [15] sociology – crime figures (number of arrests, etc), employment figures [8], [11], and others. Therefore time series is an important class of temporal data.

The analysis of time series data involves different elements, notably (cf. Batyrshin and Sheremetov [2], [3]):

1. segmentation, i.e. splitting a time series into a number of meaningful or relevant segments, and exemplified by include approximating lines, perceptual patterns, words, etc.,
2. clustering, i.e. finding some natural groupings of time series or time series patterns,
3. classification, i.e. assigning given time series or time series patterns to one of more predefined classes,
4. indexing to secure an efficient execution of some queries,
5. summarization, i.e. providing a short description of a time series to capture its essential features of interest,
6. anomaly detection, i.e. finding some surprising, unexpected patterns,
7. motif discovery, i.e. finding frequently occurring patterns,
8. forecasting, i.e. finding possible future values,
9. discovery of association rules, i.e. finding rules relating patterns in time series that occur frequently in the same or close time periods.

In the time series data, what is usually interesting, is not the exact value at a certain time point, but a relation between the values, or a pattern. Such time series data may be very long, and if they need to be stored, they may take a lot of space [20].

Naturally, the first step in the analysis of time series data is segmentation, or the identification of the consecutive parts of the sequence of data within which the data exhibit some uniformity as to their behavior equated with the variability of values.

Piecewise linear approximation methods are one of the most frequency used, as they are simple and intuitively appealing. Those algorithms present the time series data as a set of straight pieces that fit best to the covered data points. We can divide those methods into the following groups: (1) on-line sliding window algorithms, (2) top-down algorithms, (3) bottom-up algorithms, and (4) segmentation via a genetic algorithm.

In this paper we will focus only on the on-line algorithms for piecewise linear segmentation, its modifications and applications. We will present three methods of construcitng the segment and try to compare them.

In time series segmentation we should find a compromise between the error value (goodness of fit) and the number of segments. If we have many segments, then the error is small, and the segments well approximate the data points. But then, the short-term behavior predominates and this might be caused just by noise. However if we have a few segments, we concentrate on longer-term behavior; therefore we might reduce the impact of noise. On the other hand, from the point of applicability of analytic methods for analysis of entire time series, small number of segments can make the use of those methods questionable.

2 General Framework

The on-line algorithms determine the segments while the data points are collected, based only on the past observations and not on all available data points. In general those algorithms work by checking if it is possible to add the newly observed point to the currently constructed segment, and if yes, then the segment is elongated. However, when it is not possible, e.g. because of exceeding the threshold value of accepted error, ε, then the currently constructed segment is terminated and this newly observed point creates a new segment.

The on-line algorithms are based mainly on sliding a window, therefore they are called sometimes the sliding window algorithms [14]. The on-line algorithms are attractive, because they are simple, intuitively appealing and quite fast.

To present details of the algorithm, let us first introduce the following notation:

`p_0` – a point starting the current segment,

`p_1` – the last point checked in the current segment,

`p_2` – the next point to be checked,

`s_1` – representation of the current segment (covering points from `p_0` to `p_1`),

`s_2` – representation of the newly created segment (covering points from `p_0` to `p_2`),

function `read_point()` reads a next point of data series,

function `find_segment(p_0, p_2)` finds the representation of the segment starting at the point `p_0` and ending at the point `p_2`,

function `termination_criteria_fullfield()` checked if the new segment `s_2` is well defined, e.g. the error is small enough,

function `save_found_segment()` saves the obtain segment, so no further changes are possible.

The pseudocode of the procedure that segments the time series is depicted on Fig. 1.

```
read_point(p_0);
read_point(p_1);
while(1)
{
  p_2 = p_1;
  s_1 = find_segment(p_0,p_2);
  s_2 = s_1;
  do
  {
    p_1 = p_2;
    s_1 = s_2;
    read_point(p_2);
    s_2 = find_segment(p_0,p_2);
  } while (termination_criteria_fullfield()= true);
  save_found_segment();
  p_0 = p_1;
  p_1 = p_2;
}
```

Figure 1: General framework of the on-line algorithm for the piecewise linear time series segmentation

This framework is quite general and it does not determine the essential details, e.g. how to construct the segments.

3 Modifications and Applications

In this section we present three methods how to construct the segments, as well as their modifications and applications.

3.1 Algorithm Based on Linear Interpolation

Sklansky and Gonzalez [22] proposed that a segment link two points: the first and the last covered by the segment. That is the segment is defined as:

$$(y_{last} - y_{first})x - (x_{last} - x_{first})y + y_{first}x_{last} - y_{last}x_{first} = 0,$$

where (x_{first}, y_{first}) is the first point belonging to the segment and (x_{last}, y_{last}) is the last point belonging to the segment. A new point can be added to the segment, only when the distance (might be vertical but not necessary) of each previous point belonging to this segment is smaller than some user-defined threshold value, ε. That is, for every point (x_i, y_i) belonging to the segment

$$\frac{|(y_{last} - y_{first})x_i - (x_{last} - x_{first})y_i + y_{first}x_{last} - y_{last}x_{first}|}{\sqrt{(y_{last} - y_{first})^2 + (x_{last} - x_{first})^2}} < \varepsilon$$

The idea of this algorithm is presented on the Figure 2:

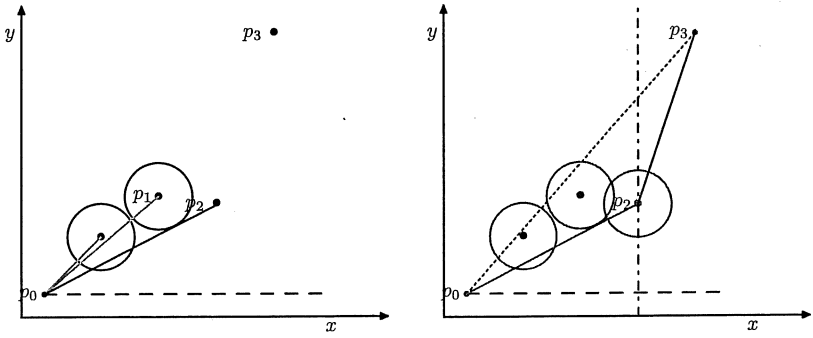

Figure 2: An illustration of the algorithm by Sklansky and Gonzalez

3.2 Algorithm Based on the Intersection of Cones

In Kacprzyk, Wilbik, Zadrożny [13] there is proposed a different method of constructing segments. The algorithm constructs the intersection of cones starting from point p_i of

the time series and including a circle of radius ε around the subsequent data points p_{i+j}, $j = 1, 2, \ldots$, until the intersection of all cones starting at p_i (indicated by the dark grey area on Fig. 3) is empty. If for p_{i+k} the intersection is empty, then we construct a new cone starting at p_{i+k-1}. Figure 3 presents the idea of the algorithm. The family of possible solutions is indicated as a gray area.

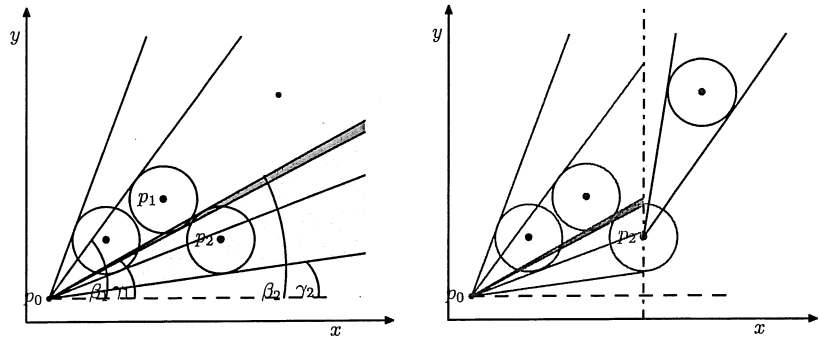

Figure 3: An illustration of the algorithm by Kacprzyk, Wilbik and Zadrożny

The bounding values of (γ_2, β_2), correspond to the slopes of two lines that:

- are tangent to the circle of radius ε around point $p_2 = (x_2, y_2)$,
- start at point $p_0 = (x_0, y_0)$

Thus

$$\gamma_2 = \arctan\left(\frac{\Delta x \cdot \Delta y - \varepsilon\sqrt{(\Delta x)^2 + (\Delta y)^2 - \varepsilon^2}}{(\Delta x)^2 - \varepsilon^2}\right) \tag{1}$$

and

$$\beta_2 = \arctan\left(\frac{\Delta x \cdot \Delta y + \varepsilon\sqrt{(\Delta x)^2 + (\Delta y)^2 - \varepsilon^2}}{(\Delta x)^2 - \varepsilon^2}\right) \tag{2}$$

where $\Delta x = x_0 - x_2$ and $\Delta y = y_0 - y_2$.

The resulting linear ε-approximation of a group of points `p_0, ... ,p_1` is either a single line segment exemplified by a bisector of the cone, or a line segment that minimizes

the distance (e.g., the sum of squared errors) from the approximated points, or the whole family of possible solutions, i.e. rays of the cone.

3.3 Algorithm Based on Regression

Another possibility of defining the segments is to employ the simple linear regression. Generally, this method chooses the straight line that minimizes the sum of the squares of the errors of the fit (vertical distance between the line and the data point) [7].

The segment over n points (x_i, y_i), $i = 1, \ldots, n$ may be represented as $y = ax + b$ for $x \in < x_1; x_n >$. The coefficent may be calculated as

$$a = \left(\frac{(\sum x_i y_i) - (\sum x_i)(\sum y_i)/n}{(\sum x_i^2) - (\sum x_i)^2/n} \right)$$

and

$$b = \frac{\sum y_i}{n} - a\frac{\sum x_i}{n}$$

The brief idea of the algorithm is presented on Figure 4.

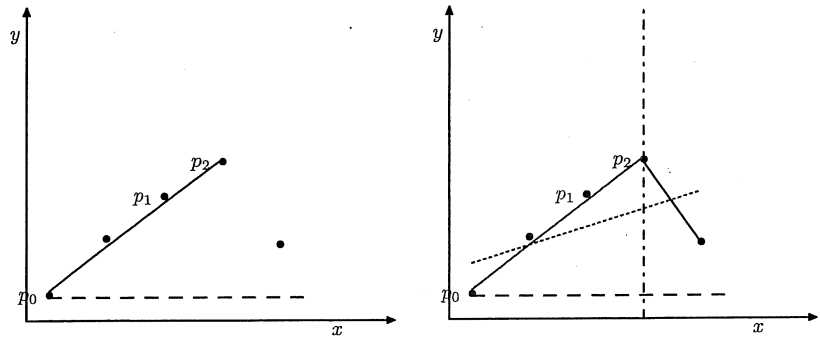

Figure 4: An illustration of the algorithm employing simple linear regression

The segment is terminated, when after adding the next point to the segment, the sum of the squares of the errors of the fit is bigger than a given threshold, ε.

While comparing those three algorithms, they differ mainly by the method constructing the segments. We may prove that the algorithm based on the intersection of cones extract no more segments (usually much less) than the one, that is based on linear interpolation.

The comparison of the two previous algorithms with the one employing the linear regression is more difficult. A natural threshold value, ε, for this algorithm is the sum of the squares of the errors, that is the value minimized by the regression. This value shouldn't be compared with the maximal distance between the point and the line that is used in the previously discussed algorithms.

We should also notice that in the method using the linear regression, the newly added point is a high leverage point. (A high leverage point is an observation that i extreme in the predictor space, i.e. it takes on extremes values for the x-variable, without reference to the y-variable.) Therefore if the new point is quite far away from the previously calculated line than it can be influential and therefore its presence in the segment may alter the parameters of the regression (slope and the y-intercept) significantly.

In many cases, higher number of segment can give us much more insight into the behavior of time series data and can make it possible to apply more sophisticated analytical methods.

There were proposed also other modifications and optimizations, e.g. [24] where was proposed a method allowing not to test every data point. These algorithms are omnipresent in medical applications (eg. FAN, SAPA (Scan-Along Polygonal Approximation technique), [1], [4], [9], [12], [16], [21]), where the online data processing is often required for patient monitoring.

Unfortunately, those algorithms are not appropriate for all types of time series data. They have relative best performance on noisy data, therefore they may be used for pro-

cessing stock market data. Analysis of performance of such type of algorithms is shown at Shatkay [20].

4 Concluding Remarks

The first step in the analysis of time series data is segmentation, or the identification of the consecutive parts of the sequence of data within which the data exhibit some uniform behavior.

In this paper we have discussed the on-line piecewise linear approximation methods, its modifications and applications.

The algorithms presented here differ mainly by the method of constructing the segments. The algorithm based on cones intersection extract no more segments (usually much less) than the one, that is based on linear interpolation. The algorithm employing regression is incomparable with the two others, in respect to the number of segments for a given threshold value. The application of the particular algorithm nad the threshold value may be dependent from the users needs. The user should seek to find a compromise between the error value (goodness of fit) and the number of segments. Smaller number of segments usually means the bigger error, and contrary smaller error value result in higher number of segments. Also,small number of segments can make the use of analytic methods for analysis of entire time series questionable.

The on-line algorithms are often used due to its simplicity, and efficiency, however its effectiveness still require some improvement. It has been proved, that those algorithms can be applied in several real word applications.

References

[1] Barr R.C., Blanchard S.M., Dipersio D.A., 1985, *SAPA-2 is the Fan* IEEE Transactions on Biomedical Engineering, Vol. 32, No. 5, pp. 337–337.

[2] Batyrshin I., Sheremetov L., *Perception based functions in qualitative forecasting*, 2006, In: Batyrshin I., Kacprzyk J., Sheremetov L., Zadeh L.A. (Eds.), Perception-based Data Mining and Decision Making in Economics and Finance, Springer-Verlag, pp. 119–134.

[3] Batyrshin I., Sheremetov L., *Towards perception based time series data mining*, 2007, In: Nikravesh M., Kacprzyk J., Zadeh L.A. (Eds.), Forging New Frontiers. Fuzzy Pioneers I, Springer-Verlag, pp. 217–230.

[4] Bohs L.N., Barr R.C., 1988, *Prototype for real-time adaptive sampling using the fan algorithm*, Medical and Biological Engineering and Computing, Vol. 26, No.6, pp. 574–583.

[5] Chung F., Fu T., Luk R., Ng V., 2002, *Evolutionary time series segmentation for stock data mining*, Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM'02), pp. 83–90.

[6] Fu T., Chung F., Ng C., 2006, *Financial time series segmentation based on specialized binary tree representation*, Crone S.F., Lessmann S., Stahlbock R. (Eds.), Proceedings of 2006 International Conference on Data Mining, pp. 3–9.

[7] Giudici P., 2003, *Applied Data Mining: Statistical Methods for Business and Industry*, Wiley.

[8] Greenberg D.F., 2001, *Time series analysis of crime rates*, Journal of Quantitative Criminology, Vol. 17, No. 4, pp. 291–327.

[9] Gurkan H., Guz U., Yarman B.S., 2005, *An efficient ECG data compression technique based on predefined signature and envelope vector banks*, Proceedings of the IEEE International Symposium on Circuits and Systems, ISCAS 2005, Vol. 2, pp. 1334–1337.

[10] Herath S., Ratnayake U., 2004, *Monitoring rainfall trends to predict adverse impacts – a case study from Sri Lanka (1964-1993)*, Global Environmental Change, Vol. 14, pp. 71–79.

[11] Huang J.T., 2003, *Unemployment and family behavior in Taiwan*, Journal of Family and Economic Issues, Vol. 24, No. 1, pp. 27–48.

[12] Ishijima M., Shin S.B., Hostetter G.H., Sklansky J., 1983, *Scan-along polygonal approximation for data compression of electrocardiograms*, IEEE Trans Biomed Eng., Vol. 30, No. 11, pp. 723–729.

[13] Kacprzyk J., Wilbik A., Zadrożny S., 2006, *On some types of linguistic summaries of time series*, Proceedings of the 3rd International IEEE Conference Intelligent Systems, IEEE Press, London, UK, pp. 373–378.

[14] Keogh E., Chu S., Hart D., Pazzani M., 2001, *An online algorithm for segmenting time series*, Proceedings of IEEE International Conference on Data Mining, pp. 289–296.

[15] Marques C.A.F., Ferreira J.A., Rocha A., Castanheira J.M., Melo-Gonalves P., Vaz N., Dias J.M., 2006, *Singular spectrum analysis and forecasting of hydrological time series*, Physics and Chemistry of the Earth, Vol. 31, No. 18, pp. 1172–1179.

[16] McKee J.J., Evans N.E., Owens F.J., 1994, *Efficient implementation of the Fan/SAPA-2 algorithm using fixed point arithmetic*, Automedica Vol. 16, pp. 109–117.

[17] Mikšovský J., Raidl A., 2006, *Testing for nonlinearity in European climatic time series by the method of surrogate data*, Theoretical and Applied Climatology, Vol. 83 No. 1-4, pp. 21–33.

[18] Portet F., Reiter E., Hunter J., Sripada S., 2007, *Automatic generation of textual summaries from neonatal intensive care data*, Proceedings of Artificial Intelligence in Medicine – AIME 2007, pp. 227–236.

[19] Sarkar M., Leong T.Y., 2001, *Top-down approaches to abstract medical time series using linear segments*, IEEE International Conference on Systems, Man, and Cybernetics, SMC 2001, pp. 765–770, Vol.2.

[20] Shatkay H., 1995, *Approximate queries and representations for large data sequences*, technical raport CS-95-03.

[21] Shu H.Z., Luo L.M., Zhou J.D., Bao X.D., 2002, *Moment-based methods for polygonal approximation of digitized curves*, Pattern Recognition, Vol.35, No. 2, pp. 421–434.

[22] Sklansky J., Gonzalez V., 1980, *Fast polygonal approximation of digitized curves*, Pattern Recognition Vol. 12, No. 5, pp. 327–331.

[23] Tsay R, 2002, *Analysis of Financial Time Series*, Willey.

[24] Vullings H.J.L.M., Verhaegen M.H.G., Verbruggen H.B., 1997, *ECG segmentation using time-warping*, Advances in Intelligent Data Analysis. Proceedings of Second International Symposium Intelligent Data Analysis, IDA-97, pp. 275–289.