Title	L1ノルムを用いた点列の一連節折れ線近似アルゴリズムの開発に関する研究
Author(s)	角川，肇喜
Citation	
Issue Date	2005-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/1855
Rights	
Description	Supervisor:浅野　哲夫，情報科学研究科，修士

Algorithms for One-join Polyline Approximation of Points

Motoki Tsunokawa (310068)

School of Information Science,
Japan Advanced Institute of Science and Technology

February 10, 2005

Keywords: The Regression Analysis, The Linear Approximation, The Polyline Approximation, L_1norm, L_∞norm.

Recently, more and more works are processed by computers due to the progress of the information science. In this situation we demand faster algorithms for problems which deal with large amount of data. In the problems of regression analysis we have to deal with large amount of data, and they are studied rigorously and many algorithms have been obtained in this field. One of the good results of these researches is a linear-time algorithm for approximating a set of n points on the plane by a line, that minimizes the L_1 norm. Although, the efficient and practical algorithm for approximating a set of n points by a polyline using the L_1 norm has not been developed yet. We would like to develop practical algorithms that are able to solve the problem efficiently. The algorithm will become the basis of the L_1 k-joins polyline approximation.

The most famous method that approximates the point-set by a line is the least-squares method. Generally, when we approximate a point set by a line, we use norms which are represented by the vertical distances between points in the point set and the line. In this research, we develop algorithms for the polyline approximation of the point set depending on the L_1 norm. The L_1 norm is represented by the sum total of vertical distances between all points in the point set and the approximation line. The other well-known norms are the L_2 norm and the L_∞ norm. The L_2 norm is the norm used in the least-squares method, and the L_∞ norm is represented by the maximum of the distances between all points of the point set and the approximation line. The optimal linear approximation of each such method is obtained by finding a line minimizing the each norm. To compare the L_1 norm with the other norms, the method using the L_1 norm has more strong resistance against outliers and noises in the data than the methods using the other norms. But, the L_1 linear approximation problem is more complex than the problems depending on the other norm.

The case of the problem that approximates the point set by a k-joins polyline, the several algorithms depending on the L_∞ norm have already been developed. These methods are able to solve the problem efficiently. But, any efficient methods depending on the other norms have not been developed yet. In particular, speaking about the L_1 method, in the case of the approximation by one-join polyline, the $O(n^{1.3})$ time algorithm has already been developed. But, this algorithm is dynamic and depends on very complex data structure, and hence this method is not appropriate for practical use. In this research, we have developed the practical algorithms that can approximate the point-set by one-join polyline using the L_1 norm efficiently, and the algorithms are intended to apply the k-joins case in the future.

The simplest algorithm for L_1 one-join polyline algorithm is a brute force method. The method considers all of the given points as candidates for single join of the optimal polyline approximation. And, these candidates of the optimal join divide the point set into left and right point sets. And, we can get n pairs of lines which approximate the left point sets and the right point sets respectively by using linear-time L_1 linear approximation method that has been introduced above. Then, we can obtain n polylines which are candidates of the optimal one-join polyline approximation. Finally, we find the best polyline that minimizes the L_1 norm, and output it as the result. This method needs $O(n^2)$ time to get an optimal polyline. We have developed our algorithm using the above brute-force method, and there are two ways to improve the simple method. One is to reduce candidates of join of the optimal approximation polyline, and we apply the linear-time L_1 linear approximation algorithm toward the fewer candidates to construct polylines. Then, we find the best polyline that minimizes the L_1 norm from these polylines. In this way, we can improve the processing time. Another way is to improve the time complexity of L_1 linear approximation algorithm, and apply it to every candidate of the optimal join. In this research, we took the former way, and the latter way was taken by the $O(n^{1.3})$ algorithm which has been introduced above.

The largest problem for improving the algorithm is how to reduce candidates of the optimal join. To resolve the problem, we propose two methods. One method is to use the result of the L_∞ polyline approximation which has been introduced above. We apply joins of the L_∞ approximation polyline as candidates of the one-join. Another way is to use the one of characteristics of the linear L_1 approximation. The characteristic is that the approximation line divides the plane into the upper and the lower half planes. And, each half plane includes the same number of points. Using this characteristic, we obtain the candidates of join for approximating a polyline. We apply some points around the point that is obtained above as candidates of join of polyline for both methods. The advantage of the former method is that L_1 and L_∞ norms complement their disadvantages by their advantages each other. The advantage of the latter method is that the obtained candidate of join is closer to optimal join than the candidate by the former method. The common disadvantage of both methods is not to be able to guarantee the optimality of the obtained polyline. The latter method has another disadvantage that it is hard to extend the method for the case to approximate s point set by a k-joins polyline. The purpose of this research is to approximate a point set by one-join polyline, since we only apply these methods to the point-sets such that their forms are suitable for approximation by one-join polylines. In order to judge it, we use the result for the L_∞ approximate polyline. We use these methods to approximate the point set by one-join polyline, if and only if the point set fulfills every condition for judging the point-set suitable to be approximated by a one-join polyline.

We implemented algorithms mentioned above, and applied it to various point sets to evaluate the algorithms. Accordingly, we can conclude that the method using the result of linear L_1 approximation is better than the method using the result of L_∞ polyline approximation.