**Response Letter of PAAA-D-16-00115**

# ***Response to Editor and Reviewers***

Dear Editor and Reviewers,

**Re:** Manuscript ID PAAA-D-16-0015 entitled “RLC: Ranking Lag Correlations with Flexible Sliding Windows in Data Streams”

**Authors:** Shanshan Wu ·Huaizhong Lin· Wenxiang Wang · Dongming Lu· Leong Hou U· Yunjun Gao

We were pleased to have been given the opportunity to revise our manuscript entitled “RLC: Ranking Lag Correlations with Flexible Sliding Windows in Data Streams”. We thank for your interest in our paper and appreciate the reviews for providing such insightful and constructive feedback on our original submission which will improve this manuscript. In the new manuscript, we have carefully considered your comments and suggestions, as well as those offered by the reviewers. Herein, after providing a brief overview of the major revisions, we reply to each comment of the reviewers in point-by-point fashion. After addressing all the issues, we feel the quality of the paper is much improved and hope these can meet your approval. Revised parts are marked in blue in our new manuscript. The main corrections in the paper and the responds to the reviewers’ comments are as follows:

# ***List of the major revisions***

Thanks very much for the constructive comments from the reviewers. Based on the reviewers’ suggestions, we have made following major revisions:

1. In Section 1 (Introduction), we have remodified the motivation to be better understood. In addition, we have also restated our proposals to make it more readable.
2. Section 5 (Discussions) is added in our new manuscript. Specifically, we have added a new subsection about the discussion over the minimal length (Please see Section 5.1 in detail). And we have added a new subsection about the discussion over the variant of the flexible window (Please see Section 5.2 in detail).
3. Section 6 (Experiments) has been significantly improved by supplementing experimental comparison with larger datasets (Please see Table 4 and Figures 1-4) and more parameters (Please see effect of k shown in Figure 3) and more explanations (Please see Section 6.2 marked in blue). Moreover, we have also tested the fixed-length algorithm under different fixed-size windows to compare the performance of the comparison algorithms in our new manuscript (Please see Section 6.4 marked in blue and Tables 6-8).
4. We have tried our best to improve the presentation of this manuscript carefully. In order to make the new manuscript better understood, we have revised and re-edited the whole manuscript according to the Reviewer’s instruction. Meanwhile, the newly added authors who are skilled for academic writing to check the English. We hope that the language will be acceptable for the following review.

# ***Detailed response***

## Response to reviewer #1:

### **General Comments 1.1**: *This manuscript described a method for computational efficient lag correlation analysis. In addition, the proposed method does not need the user input of defined length. On many test data, the proposed algorithm is shown to be very effective. Overall, this manuscript is well written.*

**Response 1.1:** Many thanks for your positive comments and constructive suggestions. We have carefully addressed all your concerns in our new manuscript. Moreover, we have supplemented experimental comparison with larger datasets and more parameters and tested the fixed-length algorithm under different fixed-size windows to compare the performance of the comparison algorithms (Please see Section 6 marked in blue). Please refer to our responses to each of your comments below for details.

### **Comments 1.2:** *In section 1.3, it is not clear what Observation 1 and 2 are.*

**Response 1.2:** Thanks a lot for pointing this out. We have improved two observations in this new manuscript as follows: (Please see 1.3 in detail):

Appro\_RLC Algorithm(cf. Section 4): Instead of computing lag correlation for every possible value of the lag, i.e., $O(m)$, we propose to keep track of a geometric progression sequence of values, where each value after the first value is found by multiplying the previous one by a fixed, non-zero number. In this algorithm, we adapt a geometric progression with the lags, $0, 1, 2, 4, ..., 2^i, ...$, which only needs to compute $O(\log m)$ lag correlations (cf. Observation 1 of Section 4). This technique can achieve a dramatic reduction in computation time and a small error when the lag is small, because we have many points to interpolate when the lag is small. However, it leads to larger error when the lag is large. It is obvious that a new maximum lag correlation may emerge at maximum lag values of every sliding window. That is because that the effect caused by the new data point is largest when the length computed is shortest, hence a new maximum lag correlation may emerge at the largest lag values of every sliding window, which has the shortest length to compute. Fortunately, the estimation error caused by Observation 1 can be reduced by considering this technique (cf. Observation 2 of Section 4). We propose an approximate algorithm, Appr\_RLC based on these two observations and running sum technique.

### **Comment 1.3:** *The window size is not user specified, and the proposed algorithm would find a 'proper' window size. The readers might be curious about how the returned window size would vary. Also, what would happen on extreme cases when there is no good match throughout the entire sequence or entirely identical sequences.*

**Response 1.3:** Thanks a lot for pointing these out. We conducted experiments to answer this question "how would the returned window size vary?". In this experiment, we perform this query "Finding the top-1 lag correlation of window size no less than 50".

|  |  |  |  |
| --- | --- | --- | --- |
| Steam size | Returned window size | Lag length | Lag correlation |
| 500 | 70 | 15 | 0.983 |
| 550 | 126 | 18 | 0.984 |
| 600 | 101 | 13 | 0.988 |
| 650 | 165 | 46 | 0.983 |
| 700 | 214 | 106 | 0.987 |
| 750 | 96 | 20 | 0.988 |
| 800 | 74 | 8 | 0.991 |
| 850 | 114 | 57 | 0.989 |
| 900 | 164 | 79 | 0.988 |
| 950 | 148 | 68 | 0.985 |
| 1000 | 60 | 30 | 0.988 |

From the above table, we find that the vary of the returned window size is irregular as the stream size increases. That's because that the correlation of a pair of subsequences does not any monotonicity for their lengths.

Also, there are two extreme cases in our query.

One is that there is no good match throughout the entire sequence. Because there is no limit for correlations, we still return top-k subsequences with the highest correlations.

The other is that all objective sequences are entirely identical to the query sequences. In our problem definition, we will return the longer subsequence when correlations are equal. Hence, in this case, we will return the entire sequence.

### **Comment 1.4:** *Would the algorithm be available to the public?*

**Response 1.4:** Yes, of course. The algorithm is available to the public. The source code and datasets will also be made available under a creative commons license as open source when the paper is accepted.

### **Comment 1.5:** *The name Agile (ranking lag correlation with flexible sliding windows between a given query stream and multiple objective streams) is a little far-fetched. If possible, a more suitable name would be preferred. This is not a major issue and unrelated to the algorithm itself. So it's only a suggestion for optional action.*

**Response 1.5:** Many thanks for your constructive suggestions. The name of our query has been modified to RLC (Ranking Lag Correlation with flexible sliding windows) in our new manuscript.

## Response to reviewer #2:

### **General Comments 2.1:** *This motivation of this paper is very clear. Lag correlation aims to analyze the correlation between time series shifted in time relative to one another. Existing research on lag correlation mainly considers the entire stream or the substream based on the sliding window with a fixed window size. However, it is rather difficult to discover the optimal fixed window size in practice. This paper thus proposes to adapt the existing sliding window based algorithms to consider the flexible window size. The contribution and novelty of this paper is sufficient enough for this well-known journal.*

**Response 2.1:** Many thanks for your positive comments and constructive suggestions. We have addressed carefully addressed all your concerns in our new manuscript. Please refer to our responses to each of all the comments below for details.

### **Comment 2.2:** *Flexible window seems to be more reasonable than fixed window size. The authors have given some examples as well. For example, a flexible window whose length is "no less than 50 days". Since here the parameter $50$ depends on some prior information as well, how to determine it in practice then? Moreover,* *the flexible window can be defined in diverse approach. For example, it may be "no longer than 50 days". Some discussions over this issue are thus necessary.*

**Response 2.2:** Thanks a lot for pointing this out. We have added a new subsection about the discussion over the minimal length in our new manuscript (Please see Section 5.1 in detail).

The minimal length is an important parameter which should be set based on the characteristic of data streams and applications. If the value is too large, some meaningful correlations will be ignored. The returned correlation may be very small, which is meaningless. On the other hand, if the value is too small, the trend of correlations between two sequences cannot be discovered. Moreover, the minimal length is a more natural parameter than a fixed length since analysts can evaluate the lower limit in their application domain.

Many thanks for your constructive suggestions. We have added a new subsection about the discussion over the variant of the flexible window in our new manuscript (Please see Section 5.2 in detail).

The flexible window can be defined in diverse approach. Not only the lag correlation of sequences no less than the given length can be of interest for an analyst, also the lag correlation of sequences *no more than* the given length can sometimes reveal interesting knowledge. However, ranking the lag correlation of sequences no more than the given length does not pose a new challenge. It is not difficult to find that the running sums and geometric progression technologies can also be applied to this problem, hence our proposed algorithms can still solve it.

Moreover, the minimal lag correlation with no less (more) than the given length maybe also meaningful. Nevertheless, this new problem does not also bring about a new challenge.

### **Comment 2.3:** *How to set the window sizes of different algorithms in experiments has not been clearly introduced.*

**Response 2.3:** Thanks a lot for pointing this out. In section 1.1 of the previous manuscript, we report our RLC query of minimal length 50 days, however, we report the fixed-size query of length 90 days. This inconsistency of two lengths may cause confusion, so we report the results of the same length in our new manuscript (Please see Table 1 and Table 2 in Section 1.1). Moreover, we have tested the fixed-length algorithm under different fixed-size windows to compare the performance of the comparison algorithms in our new manuscript (Please see Tables 6-8 in Section 6.4).

### **Comment 2.4:** *It would be more comprehensive if the performance of the comparison algorithms under different fixed-size windows are reported.*

**Response 2.4:** Many thanks for your constructive suggestions. We have tested the fixed-length algorithm under different fixed-size windows to compare the performance of the comparison algorithms in our new manuscript (Please see Section 6.4 marked in blue and Tables 6-8). .