Outlier Detection for ARM Data

Yuping Lu[1], Jitendra Kumar[2], Nathan Collier[2] and Michael A. Langston[1]

***Abstract*— The Atmospheric Radiation Measurement (ARM) Data Center collects data from either permanent or mobile facilities around the globe. These data are then ingested and used to create high level scientific products which requires great accuracy. Multiple methods are available to detect these outliers from ARM time series data. As outliers are common in the collected data which could be either an instrument failure or extreme weather event, Pearson correlation coefficient was first examined to measure the pairwise correlations between variables. New version of Singular Spectrum Analysis (SSA) was also introduced to detect outliers. K-means was applied in a different manner to filter out the abnormal records as well. Pearson correlation coefficient, SSA and K-means methods were later combined together as a whole framework to track down these outliers. Compared to the current data quality reports stored in the DQR database, our results showed this framework is promising.**

I. Introduction

The Atmospheric Radiation Measurement (ARM) user facility was founded by the U.S. Department of Energy (DOE) in 1989 [1]. Since then, its aim is to be the platforms for the observation and study of Earth's climate. Huge ARM datasets are collected from instruments deployed in different ground stations across the globe [2]. ARM Data Center is responsible for ingesting these collected data and creates high level scientific data products for distribution and the improvement of global climate models (GCMs) [3]. These high level data products, also called "Value Added Products" (VAPs) are highly dependent on the correctness of the raw data. Thus it is crucial to detect those outliers in the raw data and correct them.

Outlier detection, also called anomaly detection or intrusion detection, is a common task in many application domains which include time series data, streaming data, distributed data, spatio-temporal data, and network data [4]. Common techniques for outlier detection include signal processing, classification, clustering, nearest neighbor, density, statistical, information theory, spectral decomposition, and visualization. Among all these techniques, time series data outlier detection and temporal network outlier detection are especially useful for ARM data.

Outlier detection in time series data was first studies by Fox in 1972 [5]. Common types of outlier are additive outliers, level shifts, temporary changes, and innovative outliers. One common approach is the discriminative method which is based on a similarity function. For example, normalized longest common subsequence (NLCS) is a similarity measurement widely used in the field of data mining [6]–[8]. Commonly used clustering methods such as K-means [9], dynamic clustering [8], single-linkage clustering [10], Principal component analysis (PCA) [11], and self-organizing map (SOM) [12] are also popular. The choice of the clustering algorithm depends on the problem itself as each has different size and complexity. Three unsupervised parametric models, Finite state automata (FSA), Markov models, and Hidden Markov Models (HMMs), are often seen in outlier detection as well. An outlier is detected if the FSA in the current state couldn't reach the final state [7]. The history size in the Markov model could be either fixed or flexible. HMMs are easy to interpret but not function well with big datasets [7]. Researchers also tried supervised methods such as neural networks [13], Support vector machines (SVMs) [14], and decision tree [15] to detect outliers.

Temporal data is a broad concept which include commercial transactions, sensor data, astronomy data, computer network traffic, medical records, judicial records, social network data and many others. Many challenges exist for outlier detection for temporal data. First, the algorithm or model needs to understand the properties of the data and network. Second, the temporal data has space and time dimensions which make it complex to analysis. Third, its scale is massive and efficient algorithm is crucial for fast outlier detection. One common problem for temporal data is to detect outlier graph snapshots from a series graph snapshots in temporal networks. Spearmans correlation coefficient is the rank correlation between two sorted lists of graph vertices which are ordered by PageRank or other properties [16]. Similar to Spearmans correlation coefficient, Pearson correlation coefficient is also commonly used. Jaccard similarity is the size of intersection vertex set divided by the union vertex set [17]. Graph edit distance describes the necessary changes to make graph G_1 isomorphic to graph G_2. It can defined as $d(G_1, G_2) = |V_{G_1}| + |V_{G_2}| - 2|V_{G_1} \cap V_{G_2}| + |E_{G_1}| + |E_{G_2}| - 2|E_{G_1} \cap E_{G_2}|$ papadimitriou2010web. The spectral distance is the difference between the adjacency spectrum of graph G_1 and G_2, written as $\sigma(G_1, G_2) = \sum_{i=1}^{n} |\lambda_i(G_1) - \lambda_i(G_2)|$ [18]. Entropy distance is defined by the entropy-like measurement between two graphs [19].

Different from the traditional methods mentioned above, subspace base transformation is a novel method for univariate time series data [20].

Wind Speed Measurement Errors [21], [22] Rain, Precipitation, Wave Heights [23], [24] Hurricanes, Floods and Rainfall Anomalies [25]–[27].

[1]University of Tennessee, Knoxville, TN, USA
[2]Oak Ridge National Laboratory, Oak Ridge, TN, USA

II. DATASETS

ARM data are stored in Network Common Data Form (NetCDF) format which is self-describing and machine-independent [28], [29]. NetCDF format also has good performance and data compression. It is commonly used to handle scientific data, especially those from the climatology, meteorology, oceanography and GIS projects. ARM data is publicly available and can be downloaded from ARM Data Archive (http://www.archive.arm.gov). Kinds of raw data are stored in ARM Data Center. It ranges from *Atmospheric Profiling* to *Satellite Observations*. All these data are measured at different locations using different instruments. Each instrument may only work on a specified time range. For the raw NetCDF dataset collected from each instrument, it contains multiple variables.

TABLE I
SGPMET DATASETS TESTED

Instrument	E1	E3	E4	E5	E6	E7
Begin Year	1996	1997	1996	1997	1997	1996
End Year	2008	2008	2010	2008	2010	2011
Instrument	E8	E9	E11	E13	E15	E20
Begin Year	1994	1994	1996	1994	1994	1994
End Year	2008	2017	2017	2017	2017	2010
Instrument	E21	E24	E25	E27	E31	E32
Begin Year	2000	1996	1997	2004	2012	2012
End Year	2017	2008	2001	2009	2017	2017
Instrument	E33	E34	E35	E36	E37	E38
Begin Year	2012	2012	2012	2012	2012	2012
End Year	2017	2017	2017	2017	2017	2017

In this paper, we only tested Surface Meteorology Systems (MET) data collected from the Southern Great Plains (SGP). There were total 24 instruments in SGP area and we chose 5 typical variables which are *temp_mean*, *vapor_pressure_mean*, *atmos_pressure*, *rh_mean* and *wspd_arith_mean* from multiple variables. Table 1 contains the detail of these datasets.

III. METHODOLOGY

We have introduced many kinds of outlier detection method in the first section. We carefully picked three algorithms and tested them on ARM data. We also did necessary preprocessing before running these algorithms. The first level raw data is stored in minute level. It is normalized for pairwise comparison algorithm. Some algorithms may not need so much detail information to extract outliers. Thus we created a second level data by averaging the 1440 minute data points into one day point from the raw data. The second level data can save a lot of running time and is easier for Plotly [30] and Matplotlib [31] to visualize. The third level data was especially created for multivariate method by standardization all the 5 variables into the same scale based on the second level data. Below we will talk about each algorithm in detail.

A. Pearson correlation coefficient

Pearson correlation coefficient was first introduced by Karl Pearson [32]. It is used to measure the linear correlation between two variables. Pearson correlation coefficient is calculated from the covariance of two variables divided by the multiplication of the standard deviation of those two variables. Thus the value falls in [-1, 1]. If the value is close to -1, it means those two variables are highly negatively related. On the other hand, then the two variables are strongly positively related. If the value is near 0, it means those two variables don't have linear relation.

We performed pairwise comparison of the 5 variables using Pearson correlation on all the instruments in a seasonal level. The result in figure 1 makes sense and all the correlations in this violin plot are normally distributed. For example, x5 the Pearson correlation between *temp_mean* and *vapor_pressure_mean* is positively correlated with correlation mean close to 0.75. x1 is negatively correlated with correlation mean close to -0.60. We used this correlation as base knowledge. If a pairwise pearson correlation of two variables from a specific season of a instrument falls out of that range, we treated that seasonal data as outliers.

B. Singular Spectrum Analysis

Singular Spectrum Analysis (SSA) is a popular method for time series data analysis [33], [34]. The general idea is to use a subset of the decomposition of trajectory matrix to approximate it. Many applications can be found in [33]. For example, SSA can be applied to monitor volcanic activity [35]. It can also be used to extract trend [36]. Different from the classic SSA method, we defined our own version of SSA to best work on ARM data. Figure 2 is a demonstration of the workflow of SSA. Below is the formal description of the algorithm.

Assume we have an ARM time series data Y of length T.

$$Y = (y_1, \ \ldots, \ y_T)$$

Here $T > 2$ and y_i is not empty. Let L $(1 < L \leq T/2)$ be the window size and $K = T-L+1$. In general, the algorithm contains two main parts: decomposition and reconstruction.

1) The first step is to form trajectory matrix $\mathbf{X}$ from vector Y by embedding subsets of Y. These subsets of Y X_i are lagged vectors of length L.

$$X_i = (y_i, \ \ldots, \ y_{L+i-1})^T \quad (1 \leq i \leq K)$$
$$\mathbf{X} = [X_i, \ \ldots, \ X_K]$$

Thus the trajectory matrix is

$$\mathbf{X} = (x_{ij})_{i,j=1}^{L,K} = \begin{pmatrix} y_1 & y_2 & y_3 & \cdots & y_K \\ y_2 & y_3 & y_4 & \cdots & y_{K+1} \\ y_3 & y_4 & y_5 & \cdots & y_{K+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_L & y_{L+1} & y_{L+2} & \cdots & y_T \end{pmatrix} \quad (1)$$

where $x_{ij} = y_{i+j-1}$. We can see from equation 1 that matrix $\mathbf{X}$ has equal elements on anti-diagonals and therefore it is Hankel matrix.

2) Assume matrix $\mathbf{S} = \mathbf{X}\mathbf{X}^T$, thus we perform the singular value decomposition (SVD) on $\mathbf{S}$. The eigenvalues of S are denoted by $\lambda_1, \ldots, \lambda_L$ in the decreasing order of magnitude $(\lambda_1 \geq \ldots \geq \lambda_L \geq 0)$ and corresponding eigenvectors are denoted by $P_1, \ldots, P_L$. Let $d = rank\ \mathbf{X}$

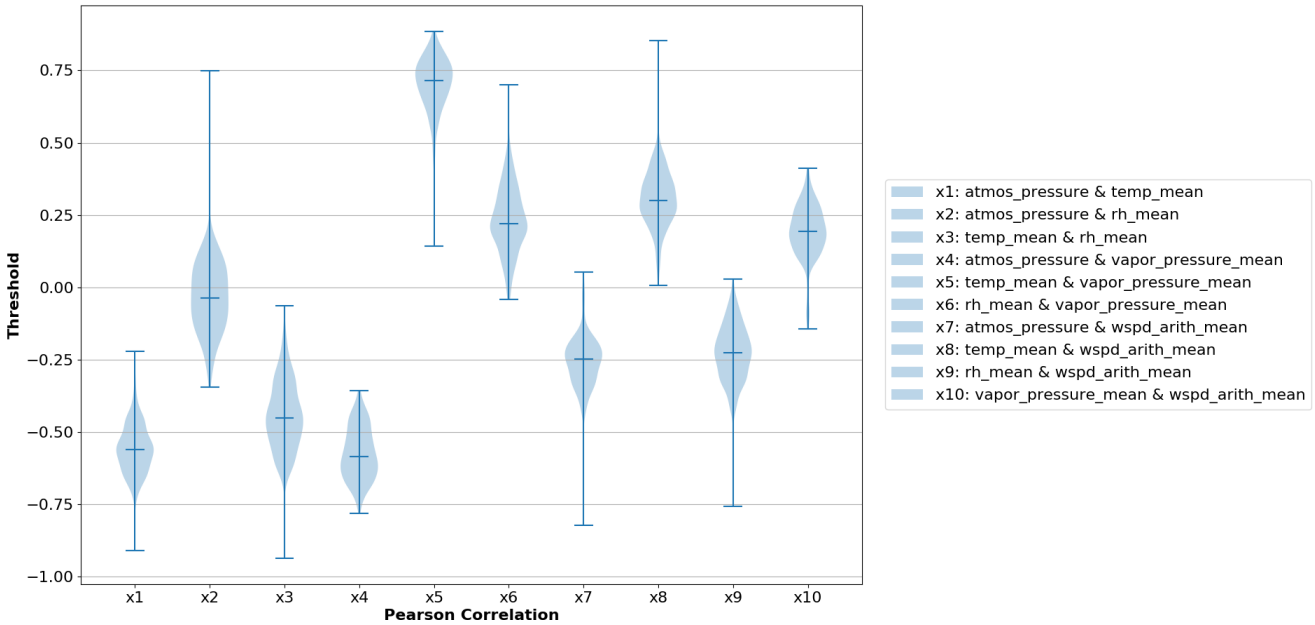

Fig. 1. Violin plot: Spring 5 variables from SGPMET

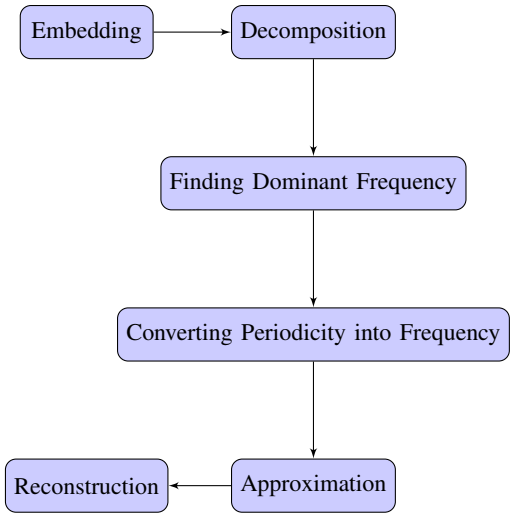

Fig. 2. Flowchart of SSA

and $V_i = \mathbf{X}^T P_i / \sqrt{\lambda_i} (i = 1, \ldots, d)$. Thus, the trajectory matrix X can also be written as

$$\mathbf{X} = \mathbf{X_1} + \ldots + \mathbf{X_d} \tag{2}$$

where $\mathbf{X_i} = \sqrt{\lambda_i} P_i V_i^T$.

3) In this step, we use Fast Fourier transform (FFT) to find the dominant frequency of each eigenvector [37] . Algorithm 1 shows the whole process.

4) Let G be the vector of user specified periodicity. We then convert G into a vector of targeted frequency TF for the next step. Here 0 is also added to TF. We use M to denote the length of TF.

5) As mentioned in step 2, there are d $\mathbf{X_i}$. The goal is to build an approximation of X by taking a subset of the decomposition $\mathbf{X_i}$. This approximation is formed by taking eigenvectors whose dominant frequency is close to the targeted frequency. Thus we have an approximation matrix $\mathbf{Xt}$ of size $M \times L \times K$.

Algorithm 1: Dominant Frequency Finder

Input : λ of **S** and corresponding eigenvectors **P**
Output: Dominant frequency of each eigenvector

```
fftfreq ← Discrete Fourier Transform sample frequencies
fft ← Discrete Fourier Transform
len ← size of λ
frequencies ← zero vector of size len
fs ← fftfreq(λ)
ix ← indices that sort fs
fs ← fs[ix]
for i in range(len) do
    p1 ← abs(fft(P[:,i]))
    ps ← p1**2
    ps ← ps[ix]
    frequencies[i] ← fs[index of the maximum value in ps]
end
return abs(frequencies)
```

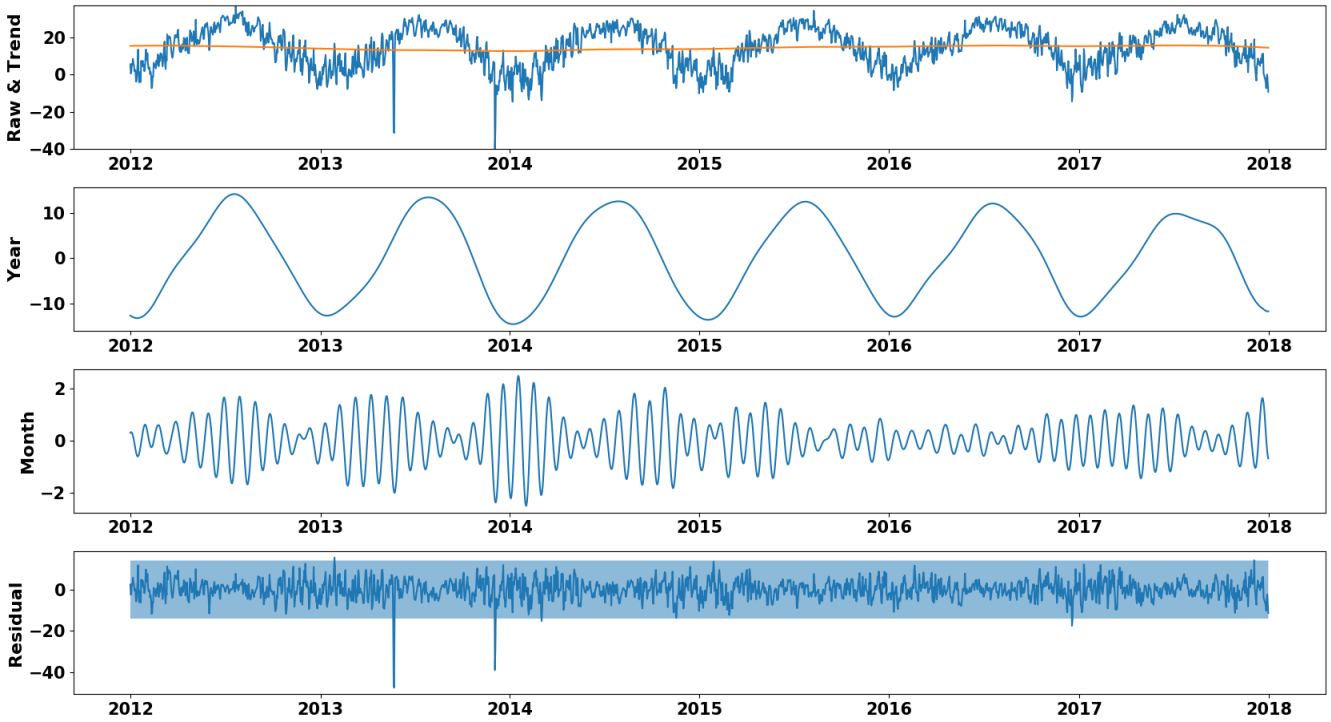

Fig. 3. Example of SSA application on ARM data. The full decomposition of *temp_mean* data from instrument E33.

6) Now we reconstruct the signal $\hat{Y}$ by taking a mean of all the approximations. The generated matrix $\mathbf{Yt}$ with size $M \times T$ can be used to approximate Y.

$$\hat{Y} = \sum_{i=1}^{M} \mathbf{Yt}[i] \tag{3}$$

In this paper, we chose the temp_mean data from instrument E33 as Y to illustrate SSA. Because SSA requires the time series data to be continuous, we replaced the empty points with the average temp_mean value for that day in a year.

We set L = 400 and picked year and month as the periodicity groups G = [365, 30]. Thus TF = [0, 0.00273973, 0.03333333]. The generated matrix $\mathbf{Yt}$ has 3 rows after performing SSA. And $\hat{Y}$ = $\mathbf{Yt}[0]$ + $\mathbf{Yt}[1]$ + $\mathbf{Yt}[2]$. The residual is then extracted from the raw data R = Y - $\hat{Y}$. As $\hat{Y}$ is the approximation which is a "perfect" representation of Y. We then extracted the extreme values from the residual. Those extracted outliers are outliers. Figure 3 is a visualization of the result. The first row is the raw data Y. The orange line $\mathbf{Yt}[0]$ is the trend. As we can see, the trend is pretty flat from 2012 to 2017. The second row and third row are $\mathbf{Yt}[1]$, $\mathbf{Yt}[1]$ respectively. The Year data matches the pattern of the raw data. The last row is the residual. Those peak values outside the blue shaded area are outliers.

C. K-means

K-means is a partitioning clustering algorithm [9], [38]. It starts with the k centroids user specified, and assigns the points to the nearest centroid. Then it computes new k centroids and assign the rest points to these centroids again. The process repeats until it converges.

Algorithm 2: K-means Outlier Detection

Input : ARM time series data
Output: Outliers

```
outliers ← ∅
df ← ARM time series data
data ← df['atmos_pressure','temp_mean',
  'rh_mean','vapor_pressure_mean','wspd_arith_mean']
number_of_clusters ← 4
clusters ← K-means(data, number_of_clusters)
distances ← Distance between each point and its
  centroid
mean ← arithmetic mean of distances
sigma ← standard deviation of distances
threshold ← mean + 3 * sigma
for i in range(size of distances) do
    if distances[i] > threshold then
        outliers ← outliers ∪ distances[i]
    end
end
return outliers
```

In this paper, we didn't stop after clustering ARM data with K-means. We transformed the generated clusters into a vector of distance between each point and its corresponding centroid. Algorithm 2 describes the whole process. Unlike SSA, we used all the 5 variables mentioned in Datasets

section together to extract outliers.

Again, we used data from instrument E33 as an example for K-means. Here we set k to 4 as each year has 4 seasons. Figure 4 visualized the outliers detected from E33. Y axes in this figure is the distance metric. The pattern of these points is close to the raw data.

IV. Results and Discussion

The three algorithms and visualizations are implemented in Python in this paper. All codes and results are available on GitHub (https://github.com/YupingLu/arm-pearson and https://github.com/YupingLu/arm-ssa). Multiple methods are available to set a threshold for extreme values as outliers. We used the three sigma rule to extract outliers [39]. For example, if the distance one point is larger than three sigmas, we treat this point as an outlier in algorithm 2.

Pearson correlation coefficient is a pairwise comparison method which is used to detect abnormality of correlation between two variables. However if the two variables suddenly change in the same direction, their correlation may still be normal similar to their "supposed" value. It is the same case if only a few outlier points inside a big quantity of data points. As we performed the Pearson correlation coefficient on the seasonal level, it's not possible to track down to the exact day. SSA is a univariate method to detect outliers for each variable in the ARM data. It can quickly catch those high peak and drop points. But it requires the time series data to be continuous with no missing points. K-means is a commonly used multivariate method for clustering. Here we used it for outlier detection. The problem is that the detected outliers could be just one type of variable or multiple types of variable. It's hard to tell which is the case and get the detail for future correction.

One outlier may only be detected by SSA or Pearson correlation coefficient or K-means. Thus we combined all the three methods together as a whole framework. SSA and K-means are used directed to detect outliers. Pearson correlation coefficient can mainly be used to detect the main variables caused the anomaly from the SSA results by comparing the pairwise correlations. Figure 5 shows the result of detected outliers for *temp_mean* from E33. The red squares stand for the common outliers detected by both K-means and SSA. The orange diamonds are the ones detected by K-means excluding the common outliers. And the black stars represents the outliers detected by SSA excluding the common outliers. We can see from the figure that more outliers have been detected compared to figure 3 and figure 4. Thus we applied this framework on all the test data. Table 2 shows the number of detected outliers. The size of common detected outliers is 378 by this framework.

The current data quality or outlier detection is maintained as data quality reports (DQRs) stored in the DQR database with each entry manually entered [40]. A description of an event which changed the normal data is included in these DQRs. The event could be temporary operating conditions such as power failures and frozen and snow covered sensors, instrument degradation, and contamination. It could also be an extreme weather event that has never been observed before. Each DQR entry also contains a specific time range affected, list of data projects, and specific measurements. And these entries are usually submitted by either the Data Quality Office [41] or the instrument mentor [42]. It is easy to notice that this method is not efficient as it requires a lot of labor. It is nearly also impossible to detect all the outliers due to the complexity and high volume of the ARM data.

TABLE II
Comparison of SSA and K-means Outlier Set Size

	Outlier Set Size
SSA	922
K-means	508
Intersection	378
Symmetric Difference	674

TABLE III
Precision and Recall of SSA and K-means

Method	Variable	Precision	Recall
SSA	temp_mean	16.00%	1.20%
SSA	vapor_pressure_mean	20.70%	1.40%
SSA	atmos_pressure	0.00%	0.00%
SSA	rh_mean	14.80%	0.50%
SSA	wspd_arith_mean	0.60%	1.50%
Kmeans	5 together	12.90%	1.90%
Combined	5 together	11.10%	4.10%

Currently, not many outliers entries stored in DQR database. Here we used detected outliers in the DQR database as the ground truth to compare with the results from our framework. Precision and recall which were first defined in [43] were used as the comparison metric. They are commonly used to measure the quality of classification tasks [44]. Precision is calculated from True Positives divided by the sum of True Positives and False Positives. On the other hand, recall is measured from True Positives divided by the sum of True Positives and False Negatives. We treated outliers in DQR database as True Positives. Thus detected outliers not in the DQR database are False Positives. Undetected values which in the DQR database are False Negatives, and which not in the DQR database are True Negatives. Table 3 contains the statistics of the comparison.

Precision attempts to answer What proportion of positive identifications was actually correct. The Combined precision is 11.10% which shows that many outliers detected by the framework are not in the DQR database. Recall tries to solve What proportion of actual positives was identified correctly. The number is 4.10% which is even smaller than precision. One reason is the same as precision that the size of True Positives is much small. The other reason is that DQR database records the whole possible affected time range which makes the size of False Negatives large. It could possible only a few days the data recorded are wrong.

V. Conclusions

In this paper we tested pairwise Pearson correlation coefficient, univariate SSA and multivariate K-means and combined them as a framework to detect outliers in the ARM

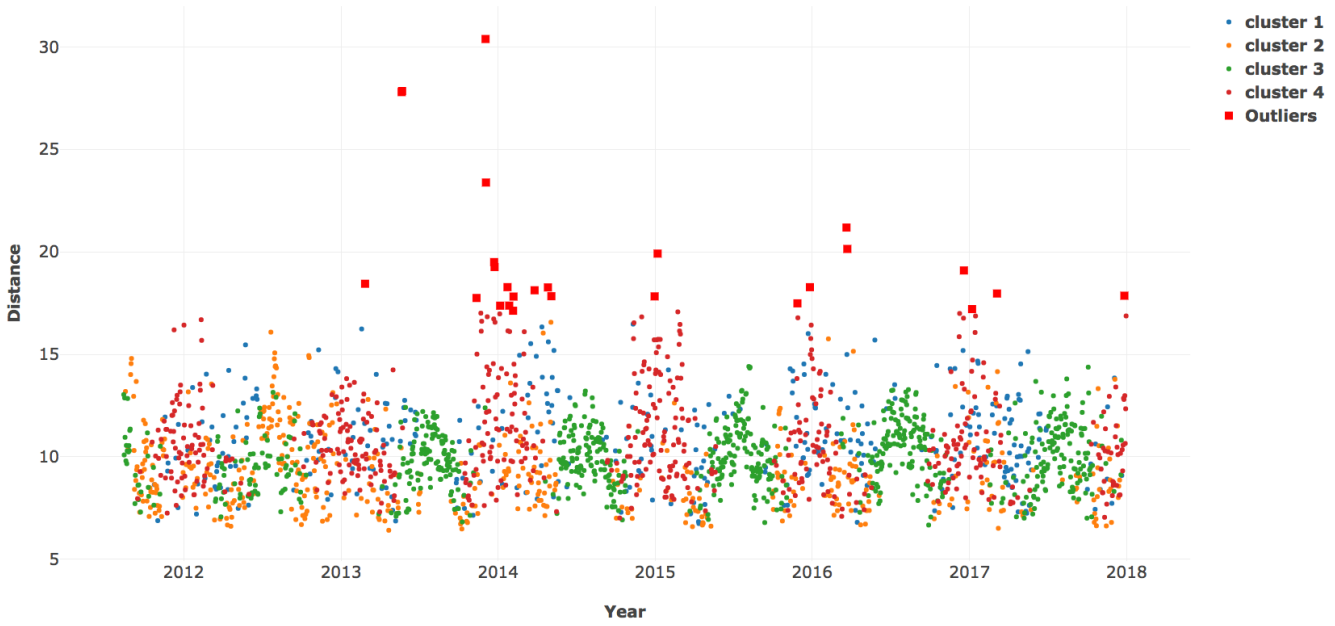

Fig. 4. Outliers detected using K-means for E33

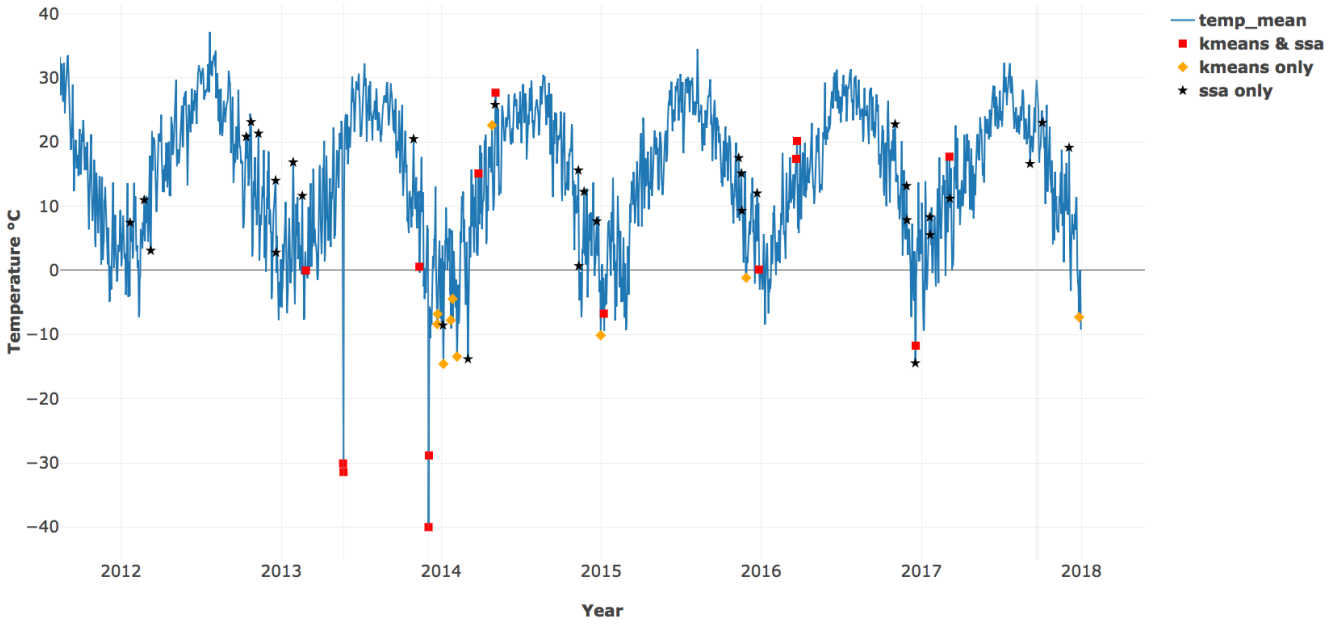

Fig. 5. Outliers detected for E33 *temp_mean* using combined algorithms

data. Each method has its own drawbacks. But our experiments showed that this framework works well compared to the manually Data Quality Report method. Currently, we only tested MET data from SGP. And we analyzed data from each instrument independently. We'll apply this framework on other types of data from other facilities in the future. Meanwhile, other methods will be examined to test data from multiple instruments together such as graph theory methods [45] and machine learning methods.

ACKNOWLEDGMENT

This research was supported by the Atmospheric Radiation Measurement (ARM) user facility, a U.S. Department of

Energy (DOE) Office of Science user facility managed by the Office of Biological and Environmental Research.

REFERENCES

[1] "Arm research facility," https://www.arm.gov/, accessed: 2018-06-22.

[2] G. M. Stokes and S. E. Schwartz, "The atmospheric radiation measurement (arm) program: Programmatic background and design of the cloud and radiation test bed," *Bulletin of the American Meteorological Society*, vol. 75, no. 7, pp. 1201–1222, 1994.

[3] K. Gaustad, T. Shippert, B. Ermold, S. Beus, J. Daily, A. Borsholm, and K. Fox, "A scientific data processing framework for time series netcdf data," *Environmental modelling & software*, vol. 60, pp. 241–249, 2014.

[4] M. Gupta, J. Gao, C. C. Aggarwal, and J. Han, "Outlier detection for temporal data: A survey," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 9, pp. 2250–2267, 2014.

[5] A. J. Fox, "Outliers in time series," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 350–363, 1972.

[6] S. Budalakoti, A. N. Srivastava, M. E. Otey *et al.*, "Anomaly detection and diagnosis algorithms for discrete symbol sequences with applications to airline safety," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications"*, vol. 39, no. 1, p. 101, 2009.

[7] V. Chandola, V. Mithal, and V. Kumar, "Comparative evaluation of anomaly detection techniques for sequence data," in *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*. IEEE, 2008, pp. 743–748.

[8] K. Sequeira and M. Zaki, "Admit: anomaly-based data mining for intrusions," in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2002, pp. 386–395.

[9] J. MacQueen *et al.*, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, no. 14. Oakland, CA, USA, 1967, pp. 281–297.

[10] L. Portnoy, E. Eskin, and S. Stolfo, "Intrusion detection with unlabeled data using clustering," in *In Proceedings of ACM CSS Workshop on Data Mining Applied to Security (DMSA-2001*. Citeseer, 2001.

[11] M. Gupta, A. B. Sharma, H. Chen, and G. Jiang, "Context-aware time series anomaly detection for complex systems," in *Workshop Notes*, vol. 14, 2013.

[12] F. A. González and D. Dasgupta, "Anomaly detection using real-valued negative selection," *Genetic Programming and Evolvable Machines*, vol. 4, no. 4, pp. 383–403, 2003.

[13] D. Dasgupta and F. Nino, "A comparison of negative and positive selection algorithms in novel pattern detection," in *Systems, man, and cybernetics, 2000 IEEE international conference on*, vol. 1. IEEE, 2000, pp. 125–130.

[14] X. Li, J. Han, and S. Kim, "Motion-alert: automatic anomaly detection in massive moving objects," in *International Conference on Intelligence and Security Informatics*. Springer, 2006, pp. 166–177.

[15] D.-K. Kang, D. Fuller, and V. Honavar, "Learning classifiers for misuse detection using a bag of system calls representation," in *International Conference on Intelligence and Security Informatics*. Springer, 2005, pp. 511–516.

[16] P. Papadimitriou, A. Dasdan, and H. Garcia-Molina, "Web graph similarity for anomaly detection," *Journal of Internet Services and Applications*, vol. 1, no. 1, pp. 19–30, 2010.

[17] J. J. Jay, J. D. Eblen, Y. Zhang, M. Benson, A. D. Perkins, A. M. Saxton, B. H. Voy, E. J. Chesler, and M. A. Langston, "A systematic comparison of genome-scale clustering algorithms," in *BMC bioinformatics*, vol. 13, no. 10. BioMed Central, 2012, p. S7.

[18] I. Jovanović and Z. Stanić, "Spectral distances of graphs," *Linear Algebra and its Applications*, vol. 436, no. 5, pp. 1425–1435, 2012.

[19] B. Pincombe, "Anomaly detection in time series of graphs using arma processes," *Asor Bulletin*, vol. 24, no. 4, p. 2, 2005.

[20] D. Cheboli. (2010) Anomaly detection of time series. [Online]. Available: http://hdl.handle.net/11299/92985

[21] D. J. Hill, B. S. Minsker, and E. Amir, "Real-time bayesian anomaly detection for environmental sensor data," in *Proceedings of the Congress-International Association for Hydraulic Research*, vol. 32, no. 2. Citeseer, 2007, p. 503.

[22] D. J. Hill and B. S. Minsker, "Anomaly detection in streaming environmental sensor data: A data-driven modeling approach," *Environmental Modelling & Software*, vol. 25, no. 9, pp. 1014–1022, 2010.

[23] A. Kut and D. Birant, "Spatio-temporal outlier detection in large databases," *Journal of computing and information technology*, vol. 14, no. 4, pp. 291–297, 2006.

[24] F. Angiulli and F. Fassetti, "Detecting distance-based outliers in streams of data," in *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*. ACM, 2007, pp. 811–820.

[25] W. Drosdowsky, "An analysis of australian seasonal rainfall anomalies: 1950–1987. ii: Temporal variability and teleconnection patterns," *International Journal of Climatology*, vol. 13, no. 2, pp. 111–149, 1993.

[26] C.-T. Lu and L. R. Liang, "Wavelet fuzzy classification for detecting and tracking region outliers in meteorological data," in *Proceedings of the 12th annual ACM international workshop on Geographic information systems*. ACM, 2004, pp. 258–265.

[27] R. Lasaponara, "On the use of principal component analysis (pca) for evaluating interannual vegetation anomalies from spot/vegetation ndvi temporal series," *Ecological Modelling*, vol. 194, no. 4, pp. 429–434, 2006.

[28] R. Rew and G. Davis, "Netcdf: an interface for scientific data access," *IEEE computer graphics and applications*, vol. 10, no. 4, pp. 76–82, 1990.

[29] Unidata. (2014) Network common data form (netcdf) version 4.1.1. Boulder, CO: UCAR/Unidata. [Online]. Available: https://doi.org/10.5065/D6H70CW6

[30] P. T. Inc. (2015) Collaborative data science. Montreal, QC. [Online]. Available: https://plot.ly

[31] J. D. Hunter, "Matplotlib: A 2d graphics environment," *Computing In Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007.

[32] K. Pearson, "Note on regression and inheritance in the case of two parents," *Proceedings of the Royal Society of London*, vol. 58, pp. 240–242, 1895.

[33] N. Golyandina and A. Zhigljavsky, *Singular Spectrum Analysis for time series*. Springer Science & Business Media, 2013.

[34] N. Golyandina and A. Korobeynikov, "Basic singular spectrum analysis and forecasting with r," *Computational Statistics & Data Analysis*, vol. 71, pp. 934–954, 2014.

[35] E. Bozzo, R. Carniel, and D. Fasino, "Relationship between singular spectrum analysis and fourier analysis: Theory and application to the monitoring of volcanic activity," *Computers & Mathematics with Applications*, vol. 60, no. 3, pp. 812–820, 2010.

[36] T. Alexandrov, "A method of trend extraction using singular spectrum analysis," *arXiv preprint arXiv:0804.3367*, 2008.

[37] J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculation of complex fourier series," *Mathematics of computation*, vol. 19, no. 90, pp. 297–301, 1965.

[38] J. A. Hartigan and M. A. Wong, "Algorithm as 136: A k-means clustering algorithm," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 28, no. 1, pp. 100–108, 1979.

[39] F. Pukelsheim, "The three sigma rule," *The American Statistician*, vol. 48, no. 2, pp. 88–91, 1994.

[40] R. McCord and J. Voyles, "The arm data system and archive," *Meteorological Monographs*, vol. 57, pp. 11–1, 2016.

[41] R. A. Peppler, K. E. Kehoe, J. W. Monroe, A. K. Theisen, and S. T. Moore, "The arm data quality program," *Meteorological Monographs*, vol. 57, pp. 12–1, 2016.

[42] T. S. Cress and D. L. Sisterson, "Deploying the arm sites and supporting infrastructure," *Meteorological Monographs*, vol. 57, pp. 5–1, 2016.

[43] J. W. Perry, A. Kent, and M. M. Berry, "Machine literature searching x. machine language; factors underlying its design and development," *Journal of the Association for Information Science and Technology*, vol. 6, no. 4, pp. 242–254, 1955.

[44] D. L. Olson and D. Delen, *Advanced data mining techniques*. Springer Science & Business Media, 2008.

[45] J. D. Phillips, W. Schwanghart, and T. Heckmann, "Graph theory in the geosciences," *Earth-Science Reviews*, vol. 143, pp. 147–160, 2015.