Author/s:
Talagala, PD;Hyndman, RJ;Leigh, C;Mengersen, K;Smith-Miles, K

Title:
A Feature-Based Procedure for Detecting Technical Outliers in Water-Quality Data From In Situ Sensors

Date:
2019-11-01

Citation:
Talagala, P. D., Hyndman, R. J., Leigh, C., Mengersen, K. & Smith-Miles, K. (2019). A Feature-Based Procedure for Detecting Technical Outliers in Water-Quality Data From In Situ Sensors. Water Resources Research, 55 (11), pp.8547-8568. https://doi.org/10.1029/2019WR024906.

Persistent Link:
https://hdl.handle.net/11343/286581

# A feature-based procedure for detecting technical outliers in water-quality data from in situ sensors

**Priyanga Dilini Talagala[1,2], Rob J. Hyndman[1,2], Catherine Leigh[1,3,4], Kerrie Mengersen[1,4], Kate Smith-Miles[1,5]**

[1]ARC Centre of Excellence for Mathematics and Statistical Frontiers (ACEMS), Australia
[2]Department of Econometrics and Business Statistics, Monash University, Australia
[3]Institute for Future Environments, Science and Engineering Faculty, Queensland University of Technology, Australia
[4]School of Mathematical Sciences, Science and Engineering Faculty, Queensland University of Technology, Australia
[5]School of Mathematics and Statistics, University of Melbourne, Australia

**Key Points:**

- Our feature-based procedure starts by applying different statistical transformations to water-quality data to highlight outliers in high dimensional space
- Density and distance-based unsupervised outlier scoring techniques were applied to detect outliers due to technical issues with the sensors
- An approach based on extreme value theory was then used to calculate outlier thresholds



This is the author manuscript accepted for publication and has undergone full peer review but has not been through the copyediting, typesetting, pagination and proofreading process, which may lead to differences between this version and the Version of Record. Please cite this article as doi: 10.1029/2019WR024906

Corresponding author: Priyanga Dilini Talagala, dilini.talagala@monash.edu

**Abstract**

Outliers due to technical errors in water quality data from *in situ* sensors can reduce data quality and have a direct impact on inference drawn from subsequent data analysis. However, outlier detection through manual monitoring is infeasible given the volume and velocity of data the sensors produce. Here we introduce an automated procedure, named oddwater, that provides early detection of outliers in water-quality data from *in situ* sensors caused by technical issues. Our oddwater procedure is used to first identify the data features that differentiate outlying instances from typical behaviours. Then, statistical transformations are applied to make the outlying instances stand out in a transformed data space. Unsupervised outlier scoring techniques are applied to the transformed data space and an approach based on extreme value theory is used to calculate a threshold for each potential outlier. Using two datasets obtained from *in situ* sensors in rivers flowing into the Great Barrier Reef lagoon, Australia, we show that oddwater successfully identifies outliers involving abrupt changes in turbidity, conductivity and river level, including sudden spikes, sudden isolated drops and level shifts, while maintaining very low false detection rates. We have implemented this oddwater procedure in the open source R package `oddwater`.

## 1 Introduction

Water-quality monitoring traditionally relies on water samples collected manually. The samples are then analyzed within laboratories to determine the water-quality variables of interest. This type of rigorous laboratory analysis of field-collected samples is crucial in making natural resources management decisions that affect human welfare and environmental conditions. However, with the rapid advances in hardware technology, the use of *in situ* water-quality sensors positioned at different geographic sites is becoming an increasingly common practice used to acquire real-time measurements of environmental and water-quality variables. Though only a subset of the required water-quality variables can be measured by these sensors, they have several advantages. Their ability to collect large quantities of data and to archive historic records allows for deeper analysis of water-quality variables to improve understanding about field conditions and water-quality processes (Glasgow, Burkholder, Reed, Lewitus, & Kleinman, 2004). Near-real-time monitoring also allows operators to identify and respond to potential issues quickly and thus manage the operations efficiently. Further, the use of *in situ* sensors can greatly reduce the labor involved in field sampling and laboratory analysis.

Water-quality sensors are exposed to changing environments and extreme weather conditions, and thus are prone to errors, including failure. Automated detection of outliers in water-quality data from *in situ* sensors has therefore captured the attention of many researchers both in the ecology and data science communities (Archer, Baptista, & Leen, 2003; Hill, Minsker, & Amir, 2009; Koch & McKenna, 2010; McKenna, Hart, Klise, Cruz, & Wilson, 2007; Raciti, Cucurull, & Nadjm-Tehrani, 2012). This problem of outlier detection in water-quality data from *in situ* sensors can be divided into two sub-topics according to their focus: (1) identifying errors in the data due to issues unrelated to water events per se, such as technical aberrations, that make the data unreliable and untrustworthy; and (2) identifying real events (e.g. rare but sudden spikes in turbidity associated with rare but sudden high-flow events). Both problems are equally important when making natural resource management decisions that affect human welfare and environmental conditions. Problem 1 can also be considered as a data preprocessing phase before addressing Problem 2.

In this work we focus on Problem 1, i.e. detecting unusual measurements caused by technical errors that make data unreliable and untrustworthy, and affect performance of any subsequent data analysis under Problem 2. According to Yu (2012), the degree of confidence in the sensor data is one of the main requirements for a properly defined

environmental analysis procedure. For instance, researchers and policy makers are unable to use water-quality data containing technical outliers with confidence for decision making and reporting purposes because erroneous conclusions regarding the quality of the water being monitored could ensue, leading, for example, to inappropriate or unnecessary water treatment, land management or warning alerts to the public (Kotamäki et al., 2009; Rangeti, Dzwairo, Barratt, & Otieno, 2015). Missing values and corrupted data can also have an adverse impact on water-quality model building and calibration processes (Archer et al., 2003). Early detection of these technical outliers will limit the use of corrupted data for subsequent analysis. For instance, it will limit the use of corrupted data in real-time forecasting and online applications such as on-line drinking water-quality monitoring and early warning systems (Storey, Van der Gaag, & Burns, 2011), predicting algal bloom outbreaks leading to fish kill events and potential human health impacts, forecasting water level and currents, etc. (Archer et al., 2003; Glasgow et al., 2004; Hill & Minsker, 2006). However, because data arrive near continuously at high speed in large quantities, manual monitoring is highly unlikely to be able to capture all the errors. These issues have therefore increased the importance of developing automated methods for early detection of outliers in water-quality data from *in situ* sensors (Hill et al., 2009).

Different statistical approaches are available to detect outliers in water-quality data from *in situ* sensors. For example, Hill and Minsker (2006) addressed the problem of outlier detection in environmental sensors using regression-based time series models. In this work they addressed the scenario as a univariate problem. Their prediction models are based on four data-driven methods: naive, clustering, perceptron, and Artificial Neural Networks (ANN). Measurements that fell outside the bounds of an established prediction interval were declared as outliers. They also considered two strategies: anomaly detection (AD) and anomaly detection and mitigation (ADAM) for the detection process. ADAM replaces detected outliers with the predicted value prior to the next predictions whereas AD simply uses the previous measurements without making any alteration to the detected outliers. These types of data-driven methods develop models using sets of training examples containing a feature set and a target output. Later, Hill et al. (2009) addressed the problem by developing three automated anomaly detection methods using dynamic Bayesian networks (DBN) and showed that DBN-based detectors, using either robust Kalman filtering or Rao-Blackwellized particle filtering, outperformed that of Kalman filtering.

Another common approach for detecting outliers in environmental sensor data is based on residuals (the differences between predicted and actual values). Due to the ability of ANNs to model a wide range of complex non-linear phenomena, Moatar, Fessant, and Poirel (1999) used ANN techniques to detect anomalies such as abnormal values, discontinuities and drifts in pH readings. After developing the pH model, the Student t-test and the cumulative Page–Hinkley test were applied to detect changes in the mean of the residuals to detect measurement error occurring over short periods of time. The work was later expanded to a multivariate scenario with some additional water-quality variables including dissolved oxygen, electrical conductivity, pH and temperature (Moatar, Miquel, & Poirel, 2001). Their proposed algorithm used both deterministic and stochastic approaches for the model building process. Observed data were then compared with the model forecasts using a set of classical statistical tests to detect outliers, demonstrating the effectiveness and advantages of the multimodel approach. Later, Archer et al. (2003) proposed a method to detect failures in the water-quality sensors due to biofouling based on a sequential likelihood ratio test. Their method also had the ability to provide estimates of biofouling onset time, which was useful for the subsequent step of outlier correction.

A common feature of all of the above methods is that they are usually employed in a supervised or semi-supervised context and thus require training data pre-labeled with known outliers or data that are free from the anomalous features of interest. In many

cases, however, not all the possible outliers are known in advance and can arise spontaneously as new outlying behaviors during the test phase. In such situations, supervised methods may fail to detect those outliers. Semi-supervised methods are also unsuitable for certain applications due to the unavailability of training data containing only typical instances that are free from outliers (Goldstein & Uchida, 2016). The datasets that we consider in this paper suffer from both of these limitations highlighting the need for a more general approach.

This paper develops a method for detecting technical outliers in water-quality data derived from *in situ* sensors. Prior work by Leigh et al. (2019) emphasises the importance of different anomaly types and end-user needs and provides the starting point for constructing a framework for automated anomaly detection in high frequency water-quality data from *in situ* sensors. Their work briefly introduced unsupervised feature based methods for detecting technical-outliers in such data. The present paper differs substantially from Leigh et al. (2019) as (1) the unsupervised feature based procedure we present for detecting technical-outliers in high frequency water-quality data measured by *in situ* sensors is its sole focus (2) the unsupervised feature based procedure is fully elaborated in both details and depth and (3) the experimental results are enhanced through emphasis on the multivariate capabilities of the unsupervised feature based procedure. Furthermore, we focus on outliers involving abrupt changes in value, including sudden spikes, sudden isolated drops and level shifts (high priority outliers as described in Leigh et al. (2019)) rather than the broader suite considered by Leigh et al. (2019).

First, we present in detail our unsupervised feature based procedure that provides early detection of technical outliers in water-quality data from *in situ* sensors. Rule-based methods are also incorporated into the procedure to flag occurrences of impossible, out-of-range, and missing values. Second, we provide a comparative analysis of the efficacy and reliability of both density- and nearest neighbor distance-based outlier scoring techniques. Third, we introduce an R (R Core Team, 2018) package, `oddwater` (Talagala & Hyndman, 2019b) that implements the feature-based procedure and related functions. Further, to facilitate reproducibility and reuse of the results presented in this paper, we have made all of the code and associated datasets available on zenodo (Talagala & Hyndman, 2019a).

Our feature-based procedure has many advantages: (1) it can take the correlation structure of the water-quality variables into account when detecting outliers; (2) it can be applied to both univariate and multivariate problems; (3) the outlier scoring techniques that we consider are unsupervised, data-driven approaches and therefore do not require training datasets for the model building process and can be extended easily to other time series from other sites; (4) the outlier thresholds have a probabilistic interpretation as they are based on extreme value theory; (5) the approach has the ability to deal with irregular (unevenly spaced) time series; and (6)it can easily be extended to streaming data. In contrast to a batch scenario, which assumes that the entire dataset is available prior to the analysis with the focus on detecting complete events, the streaming data scenario gives many additional challenges due to high velocity, unbounded, nonstationary data with incomplete events (Hill et al., 2009; Talagala, Hyndman, Smith-Miles, Kandanaarachchi, & Muñoz, 2019). In this paper, although our oddwater procedure is introduced as a batch method, it can easily be extended to streaming data such that it can provide near-real-time support using a sliding window technique.

## 2 Materials and Methods

Our unsupervised feature-based procedure for detecting outliers in water-quality data from *in situ* sensors has six main steps (Figure 1), and the structure of this section is organised accordingly. For easy reference, we named our unsupervised feature-based

procedure as oddwater procedure, which stands for **O**utlier **D**etection in **D**ata from **WA-TER**-quality sensors.

**Figure 1.** Unsupervised feature-based procedure, named oddwater procedure for outlier detection in water quality data from *in situ* sensors. Squares represents the main steps involved. Circles correspond to input and output.

### 2.1 Study region and data

To evaluate the effectiveness of our oddwater procedure we considered a challenging real-world problem of monitoring water-quality using *in situ* sensors in a natural river system. This is challenging because the system is susceptible to a wide range of environmental, biological and human impacts that can lead to variation in water-quality and affect the technological performance of the sensors. For comparison, we evaluated two study sites, Sandy Creek and Pioneer River (PR), both in the Mackay-Whitsunday region of northeastern Australia (Mitchell, Brodie, & White, 2005). These two rivers flow into the Great Barrier Reef lagoon and have catchment areas of 1466 $km^2$ and 326 $km^2$, respectively. In this region, the wet season typically occurs from December to April and is dominated by higher rainfall and air temperatures, whereas the dry season typically occurs from May to November with lower rainfall and air temperatures (McInnes et al., 2015). The sensors at these two sites are housed within monitoring stations on the river banks. Water is pumped from the rivers to the stations approximately every 60 or 90 minutes to take measurements of various water-quality variables that are logged by the sensors. Here we focused on three water-quality variables: turbidity(NTU), conductivity (strictly, specific conductance at $25^0$C; $\mu$S/cm) and river level (m).

The water-quality data obtained from *in situ* sensors located at Sandy Creek were available from 12 March 2017 to 12 March 2018. The data set included 5402 recorded points. These time series were irregular (i.e. the frequency of observations was not constant) with a minimum time gap of 10 minutes and a maximum time gap of around 4 hours. The data obtained from Pioneer River were available from 12 March 2017 to 12 March 2018, and included 6303 recorded points. Many missing values were observed during the initial part of all three series, i.e. turbidity, conductivity and river level, at Pioneer River. With the help of a group of water-quality experts who were familiar with the study region and with over 40 years of combined knowledge of river water quality, observations were labeled as outliers or not, with the aim of evaluating the performance of the procedure. Our Shiny web application available through the *oddwater* R package was used during the labeling process to pinpoint observations and provide greater visual insight into the data. Using this interactive visualization tool and expert knowledge, the ground-truth labels were decided by consensus vote.

### 2.2 Apply rule-based approaches

Following Thottan and Ji (2003), we incorporated simple rules into our oddwater procedure to detect outliers such as out-of-range values, impossible values (e.g. negative values) and missing values, and labeled them prior to applying the statistical transformations introduced in Section 2.4.

If a sensor reading was outside the corresponding sensor detection range, it was marked as an outlier. Negative readings are also inaccurate and impossible for river turbidity, conductivity and level. We therefore imposed a simple constraint on the algorithm to filter these values and mark them as outliers. Missing values are also frequently encoun-

tered in water-quality sensor data (Rangeti et al., 2015). We detected missing values by calculating the time gaps between readings. If a gap exceeded the maximum allowable time difference between any two consecutive readings, the corresponding time stamp was then marked as an outlier due to missingness. Here the maximum allowable time difference was set at 180 minutes, given that the water-quality measurements were set to be taken at most every 90 minutes (measurements were often taken at higher frequencies during high-flow events, e.g. every 10-15 minutes, and occasionally as one-off measurements at times of interest to water managers).

### 2.3 Identify data features

After labeling out-of-range, impossible and missing values as outliers, further investigation was done with the remaining observations. We initiated this investigation by identifying common characteristics or patterns of the possible types of outliers in water-quality data that would differentiate them from typical instances or events. For turbidity, for example, "extreme" deviations upward are more likely than deviations downward (Panguluri, Meiners, Hall, & Szabo, 2009). The opposite is true for conductivity (Tutmez, Hatipoglu, & Kaymak, 2006). Further, in a turbidity time series, a sudden isolated upward shift (spike) is a point outlier (a single observation that is surprisingly large, independent of the neighboring observations (Goldstein & Uchida, 2016)), but if the sudden upward shift is followed by a gradually decaying tail then it becomes part of the typical behavior. For river level, rates of rise are often fast compared with fall rates. In general, isolated data points that are outside the general trend are outliers. Further, natural water processes under typical conditions generally tend to be comparatively slow; sudden changes therefore mostly correspond to outlying behaviors. Hereafter, these characteristics will be referred to as 'data features'.

### 2.4 Apply statistical transformations

After identifying the data features, different statistical transformations were applied to the time series to highlight different types of outliers focusing on sudden isolated spikes, sudden isolated drops, sudden shifts, and clusters of spikes (Table 1) that deviate from the typical characteristics of each variable (Leigh et al., 2019).

In this work, we considered the outlier detection problem in a multivariate setting. By applying different transformations on water-quality variables, we converted our original problem of outlier detection in the temporal context to a non-temporal context through a high dimensional data space with three dimensions defined by the three variables: turbidity, conductivity and river level. Different transformations were applied on different axes of the three dimensional data space resulting in different data patterns. We evaluated the performance of the transformations (Dang & Wilkinson, 2014) using the maximum separability of the two classes: outliers and typical points in the three dimensional data space. To provide a better visual illustration, in Figure 2 we present only the two dimensional data space defined by turbidity and conductivity; however, our actual data space is three dimensional. In this work our focus was to evaluate whether each point in time is an outlier or not such that an alarm could be triggered in the presence of an outlier. However, it was not our interest to investigate which variable(s) is (are) responsible for the outlier in time. Therefore, in Figure 2, a point is marked as an outlier in the two dimensional space if at least one variable corresponding to that point was labelled as an outlier by the water-quality experts.

When the transformation involves both the current value, $Y_t$, and the lagged value, $Y_{t-1}$, (as in the first difference and first derivative) both the outlier and immediate neighbour are highlighted in the transformed space. For example, if an outlier occurs at time point $t$, then the two values derived from the first derivative transformation ($(y_t - y_{t-1})$ and $(y_{t+1} - y_t)$) are highlighted as outlying values, because they both involve $y_t$. There-

**Table 1.** Transformation methods used to highlight different types of outliers in water-quality sensor data. Let $Y_t$ represent an original series from one of the three variables: turbidity, conductivity and level at time $t$.

| **Data Feature** | **Requirement** | **Possible Transformation** | **Formula** |
|---|---|---|---|
| High variability of the data. | Stabilize the variance across time series and make the patterns more visible (e.g. level shifts) | Log transformation | $\log(y_t)$ |
| Isolated spikes (in both positive and negative directions) that are outside the general trend are considered as outliers. Under typical behavior, sudden upward (downward) shifts are possible for turbidity (conductivity), but their rate of fall (rise) is generally slower than the rate of rise (fall). | Separate isolated spikes from the general upward/downward trend patterns. | First difference | $\log(y_t/y_{t-1})$ |
| Missing values in the data. the maximum allowable time difference between observations is 180 minutes. | Identify missing values. | Time gap | $\Delta t$ |
| Data are unevenly spaced time series. | Handle irregular time series. | First derivative (Data points with large gaps will get small value. Large gaps indicate the lack of information to make a claim regarding the points.) | $x_t = \log(y_t/y_{t-1})/\Delta t$ |
| One sided derivative | | | |
| Extreme upward trend in turbidity and level under typical behavior. | Separate spikes from typical upward trends. | *Turbidity or level* | $\min\{x_t, 0\}$ |
| Extreme downward trend in conductivity under typical behavior. | Separate isolated drops from typical downward trends. | *Conductivity* | $\max\{x_t, 0\}$ |
| High or low variability in the data. | Detect change points in variance. | Rate of change | $(y_t - y_{t-1})/y_t$ |
| Natural processes are comparatively slow. Sudden changes (upward or downward movements) typically correspond to outlying instances. | Detect sudden changes (both upward and downward movements) | Relative difference | $y_t - (1/2)(y_{t-1} + y_{t+1})$ |

fore, each outlying instance is now represented by two consecutive values under the first derivative or first difference transformation. As a result, one outlying instance is now represented by two points in the transformed data space (Figure 2(c, d)). The goal of the one sided derivative transformation is to select only one high value as a representative point for each outlying instance. However, the high values obtained could correspond to either the actual outlying time point or the neighboring time point, because each transformed value is derived from two consecutive observations. For example, in the data obtained from Sandy Creek, the one sided derivative transformation (Figure 2(e)) clearly separates all of the target outlying instances from the typical points using only one point for each outlying instance, shown as either red triangles (corresponding to outliers) or green squares (corresponding to the immediate neighbours of outliers). The second representative member of each outlying instance mingles with the typical points, allowing only one point to standout on behalf of the corresponding outlying instance. If the primary focus of detecting technical outliers is to alert managers of sensor failures, then it will be inconsequential if the alarm is triggered either at the actual time point corresponding to the outlier or at the next immediate time point. However, if the purpose is different, such as producing a trustworthy dataset by labeling or correcting detected outliers, then additional conditions should be imposed to ensure that the time points declared as outliers correspond to the actual outlying points and not to their immediate neighboring points.

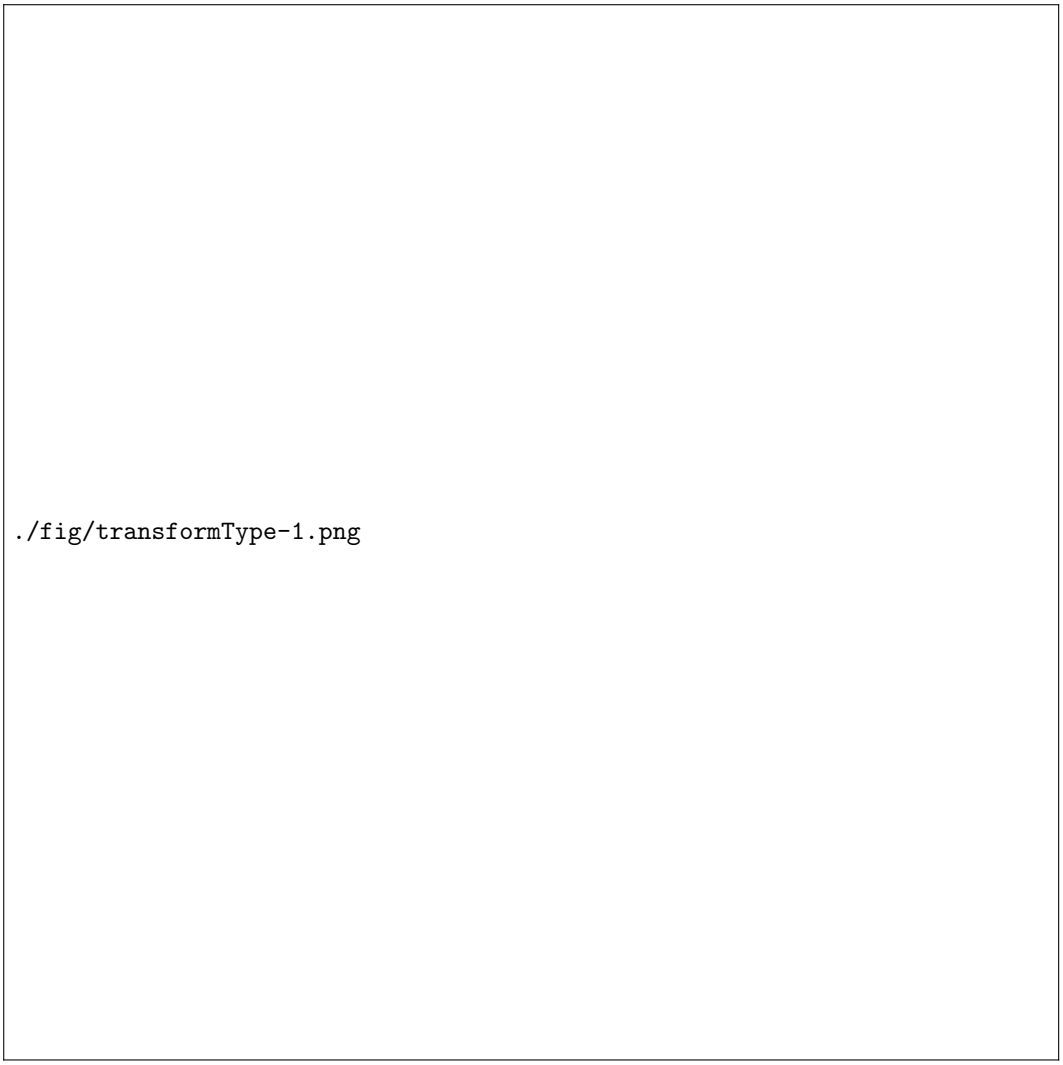



**Figure 2.** Bivariate relationships between transformed series of turbidity and conductivity measured by *in situ* sensors at Sandy Creek. In each scatter plot, outliers determined by water-quality experts are shown in red, while typical points are shown in black. Neighboring points are marked in green. (a) Original series, (b) Log transformation, (c) First difference, (d) First derivative, (e) One sided derivative, and (f) Rate of change, (g) Relative difference (for original series), (h) Relative difference (for log transformed series). In each scatter plot, data are normalised such that they are bounded by the unit hypercube.

### 2.5 Calculate outlier scores

We considered eight commonly used, unsupervised outlier scoring techniques for high dimensional data involving nearest neighbor distances or densities of the observations and applied them to the three dimensional data space defined by the three variables: turbidity, conductivity and river level. Methods based on $k$-nearest neighbor distances (where $k \in Z^+$) were the NN-HD algorithm (details of this algorithm, which was inspired by HDoutliers algorithm (Wilkinson, 2018) are provided in Supporting Information), KNN-AGG and KNN-SUM algorithms (Angiulli & Pizzuti, 2002; Madsen, 2018) and Local Distance-based Outlier Factor (LDOF) algorithm (Zhang, Hutter, & Jin, 2009), which calculate the outlier score under the assumption that any outlying point (or out-

lying clusters of points) in the data space is (are) isolated; therefore the outliers are those points having the largest $k$-nearest neighbor distances. In contrast, the density based Local Outlier Factor (LOF) (Breunig, Kriegel, Ng, & Sander, 2000), Connectivity-based Outlier Factor (COF) (Tang, Chen, Fu, & Cheung, 2002), Influenced Outlierness (INFLO) (Jin, Tung, Han, & Wang, 2006) and Robust Kernel-based Outlier Factor (RKOF) (Gao, Hu, Zhang, Zhang, & Wu, 2011) algorithms calculate an outlier score based on how isolated a point is with respect to its surrounding neighbors, and therefore, the outliers are those points having the lowest densities (see Supporting Information for detail). Each algorithm assigns outlier scores for all of the data points in the high dimensional space that describe the degree of outlierness of the individual data points such that outliers are those points having the largest scores (Kriegel, Kröger, & Zimek, 2010; Shahid, Naqvi, & Qaisar, 2015). This step allowed us to set a data driven threshold (Section 2.6) for the outlier scores to select the most relevant outliers (Chandola, Banerjee, & Kumar, 2009).

### 2.6 Calculate outlier threshold

Following Schwarz (2008), Burridge and Taylor (2006) and Wilkinson (2018), we used extreme value theory (EVT) to calculate a separate outlier threshold for each set of outlier scores calculated using a given unsupervised outlier scoring technique (introduced in Section 2.5) and assign a bivariate label for each point either as an outlier or typical point. Thus, 8 outlier scoring techniques resulted 8 different thresholds for a given dataset. The threshold calculation process started from a subset of data containing 50% of observations with the smallest outlier scores, under the assumption that this subset contained the outlier scores corresponding to typical data points and the remaining subset contained the scores corresponding to the possible candidates for outliers. Following Weissman's spacing theorem (Weissman, 1978), the algorithm then fit an exponential distribution to the upper tail of the outlier scores of the first subset, and computed the upper $1-\alpha$ (in this work $\alpha$ was set to 0.05) points of the fitted cumulative distribution function, thereby defining an outlying threshold for the next outlier score. From the remaining subset, the algorithm then selected the point with the smallest outlier score. If this outlier score exceeded the cutoff point, all the points in the remaining subset were flagged as outliers and searching for outliers ceased. Otherwise, the point was declared as a non-outlier and was added to the subset of the typical points. The threshold was then updated by including the latest addition. The searching algorithm continued until an outlier score was found that exceeded the latest threshold (Schwarz, 2008). We performed this threshold calculation under the assumption that the distribution of outlier scores produced by each of the eight unsupervised outlier scoring techniques for high dimensional data was in the maximum domain of attraction of the Gumbel distribution, which consists of distribution functions with exponentially decaying tails including the exponential, gamma, normal and log-normal (Embrechts, Klüppelberg, & Mikosch, 2013).

### 2.7 Performance evaluation

In this paper, we focused on high priority outliers as described in Leigh et al. (2019) in which importance ranking of different outlier types was done by taking into account the end-user goals and the potential impact of outliers going undetected. However, it is beyond the scope of this paper to discuss in detail the different types of outliers and their importance ranking. For more detail, we refer the reader to Leigh et al. (2019). We performed an experimental evaluation on the accuracy and computational efficiency of our oddwater procedure with respect to the eight outlier scoring techniques using the different transformations (Table 1) and different combinations of variables (turbidity, conductivity and river level). These experimental combinations were evaluated with respect to common measures for binary classification based on the values of the confusion matrix, which summarizes the false positives (FP; i.e. when a typical observation is misclassified as an outlier), false negatives (FN; i.e. when an actual outlier is misclassified

as a typical observation), true positives (TP; i.e. when an actual outlier is correctly classified), and true negatives (TN; i.e. when an observation is correctly classified as a typical point). In this work, false positives and false negatives are equally undesirable as false positives may demand unnecessary and/or expensive actions for corrections and refinement, and false negatives greatly reduce confidence in the data and results derived from them. The measures we considered include accuracy

$$accuracy = (TP + TN)/(TP + FP + FN + TN), \tag{1}$$

which explains the overall effectiveness of a classifier; and geometric-mean

$$GM = \sqrt{TP * TN}, \tag{2}$$

which explains the relative balance of TP and TN of the classifier (Sokolova & Lapalme, 2009). According to Hossin and Sulaiman (2015), these measures are not enough to capture the poor performance of the classifiers in the presence of imbalanced datasets where the size of the typical class (positive class) is much larger than the outlying class (negative class). The datasets obtained from *in situ* sensors were highly imbalanced and negatively dependent (i.e. containing many more typical observations than outliers). Therefore, we used three additional measures that are recommended for imbalanced problems with only two classes (i.e. typical and outlying) by Ranawana and Palade (2006): the negative predictive value

$$NPV = TN/(FN + TN), \tag{3}$$

which measures the probability of a negatively predicted pattern actually being negative; positive predictive value

$$PPV = TP/(TP + FP), \tag{4}$$

which measures the probability of a positively predicted pattern actually being positive; and optimized precision, which is a combination of accuracy, sensitivity and specificity metrics (Ranawana & Palade, 2006). The optimized precision is calculated as

$$OP = P - RI, \tag{5}$$

where

$$P = S_p N_n + S_n N_p \tag{6}$$

$$RI = |S_p - S_n|/(S_p + S_n) \tag{7}$$

$$S_p = TN/(TN + FP) \tag{8}$$

$$S_n = TP/(TP + FN) \tag{9}$$

and $N_p$ and $N_n$ represent the proportion of positives (outliers) and negatives (typical) within the entire dataset).

To evaluate the performance of our oddwater procedure, we incorporated additional steps after detecting the outlying time points using the outlying threshold based on EVT. This was done because the time points declared as outliers by the outlying threshold could correspond to either the actual outlying points or to their neighbors. Once the time points were declared as outliers, the corresponding points in the three dimensional space were further investigated by comparing their positions with respect to the median of the typical points declared by the oddwater procedure. This step allowed us to find the most influential variable for each outlying point. For example, in Figure 2(e) the isolated point in the first quadrant is an outlier in the two dimensional space due to the outlying behavior of the conductivity measurement. This allowed us because the deviation of this point from the median of the typical points (around $(0, 0)$) happens primarily along the conductivity axis. In contrast, the four isolated points in the third quadrant are outliers due to the outlying behavior of the turbidity measurement because the deviations of the

four points from the median of the typical points (around $(0, 0)$) happen primarily along the turbidity axis. After detecting the most influential variable for each outlying instance in the three dimensional space, further investigations were carried out separately for each individual outlying instance with respect to the most influential variable detected. This allowed us to see whether the outlying instance was due to a sudden spike or a sudden drop by comparing the direction of the detected points with respect to the mean of its two immediate surrounding neighbors and itself. These additional steps in the oddwater procedure allowed us to trigger an alarm at the actual outlying point in time if the neighboring points were declared as outliers instead of the actual outliers. However, we acknowledge that these additional steps select only the most influential variable, not all of the influential variables in the presence of more than one influential variable. The additional steps were incorporated solely to measure the performance of the oddwater procedure. In practice, because the goal is to trigger an alarm in an occurrence of a technical outlier, it is inconsequential if the alarm is triggered either at the actual time point or at the immediate neighbouring time points corresponding to the actual outlier. As such, users of the oddwater procedure can ignore these additional steps.

Using the outlier threshold, our oddwater procedure assigns a bivariate label (either as outlier or typical point) to each observed time point and thereby creates a vector of predicted class labels. That is, if a time point is declared as an outlier by oddwater procedure, then that could be due to at least one variable in the dataset. We also declared each time point as an outlier or not based on the labels assigned by the water-quality experts. At a given time point, if at least one variable was labeled as an outlier by the water-quality experts then the corresponding time point was marked as an outlier, thereby creating a vector of ground-truth labels. Then, the performance measures were calculated based on these two vectors of ground-truth labels and predicted class labels. Thus, this performance evaluation was done with respect to the algorithm's ability to label a point in time as an outlier or not (i.e. a point in time is an outlier if the observed value for any one or more of the three variables measured at that point in time are outliers).

### 2.8 Software implementation

The oddwater procedure was implemented in the open source R package `oddwater` (Talagala & Hyndman, 2019b), which provides a growing list of transformation and outlier scoring methods for high dimensional data together with visualization and performance evaluation techniques. In addition to the implementations available through `oddwater` package, `DDoutlier` package (Madsen, 2018) was also used for outlier score calculations. We measured the computation time (mean execution time) using the `microbenchmark` package (Mersmann, 2018) for different combinations of algorithms, transformations and variable combinations on 28 core Xeon-E5-2680-v4 @ 2.40GHz servers. We also developed an R Shiny web application (available via `oddwater` R package) to provide interactive visual analytic tools to gain greater insight into the data and perform preliminary investigations of the relationships between water-quality variables at different sites. To facilitate reproducibility of the results presented herein, we have archived a snapshot of version 0.7.0 of the R package on zenodo (Talagala & Hyndman, 2019a) along with the code and datasets used. The latest version and on-going development of the `oddwater` R package are available from Github (`https://github.com/pridiltal/oddwater`).

## 3 Results

### 3.1 Analysis of water-quality data from *in situ* sensors at Sandy Creek

A negative relationship was clearly visible between the water-quality variables turbidity and conductivity and also between conductivity and river level measured by *in situ* sensors at Sandy Creek (Figures 3 (a–i, b–i, c–i) and 4(a,c)). Further, no clear separa-

tion was observed between the target outliers and the typical points in the original data space (Figure 4(a–c)). However, a clear separation was apparent between the two sets of points once the one sided derivative transformation (an appropriate transformation for unevenly spaced data) was applied to the original series (Figures 4(d–f) and 3 (a–ii, b–ii, c–ii) ).

KNN-AGG and KNN-SUM algorithms performed on all three water-quality variables together using the one sided derivative transformation gave the highest OP (0.83) and NPV values(0.9996), which are the most recommended measurements for negatively dependent data where the focus is more on sensitivity (the proportion of positive patterns being correctly recognized as being positive) than specificity (Ranawana & Palade, 2006).

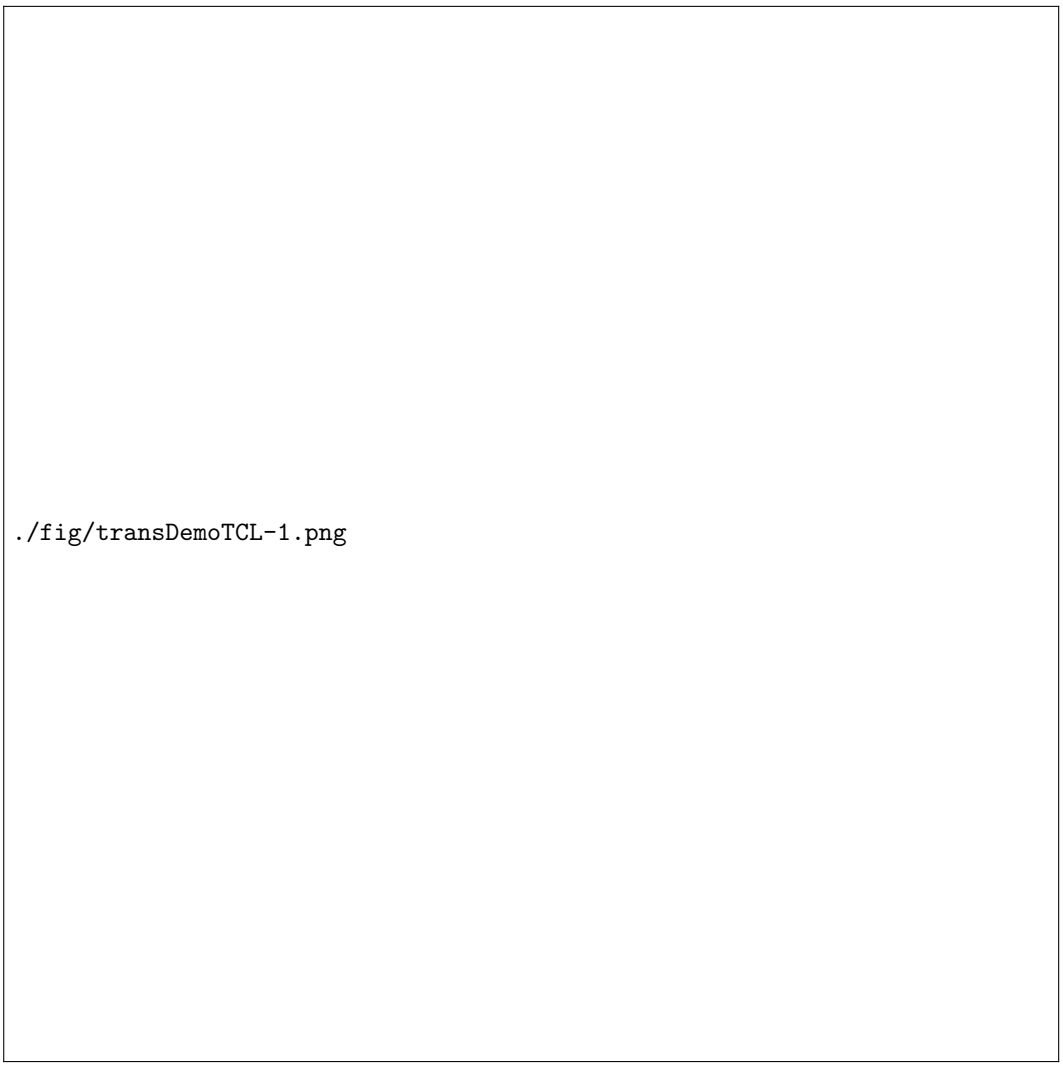



**Figure 3.** Time series for turbidity (NTU) (a–i), conductivity ($\mu$S/cm) (b–i) and river level (m) (c–i) measured by *in situ* sensors at Sandy Creek. Transformed series (one sided derivatives) of turbidity (NTU) (a–ii), conductivity ($\mu$S/cm) (b–ii) and river level (m) (c–ii) measured by *in situ* sensors at Sandy Creek. In each plot outliers determined by water-quality experts are shown in red, while typical points are shown in black. Neighboring points are marked in green.

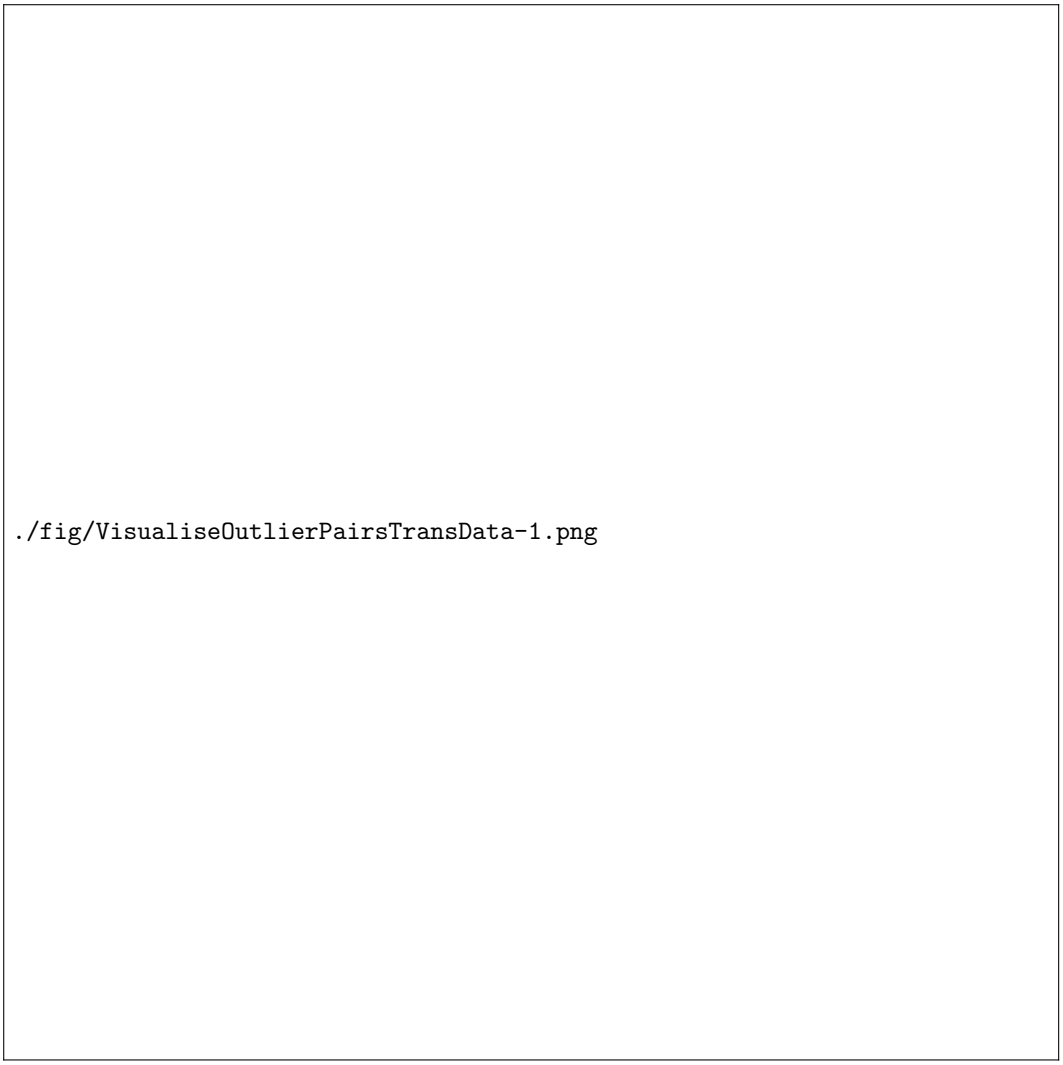



**Figure 4.** Top panel (a–c): Bi-variate relationships between original water-quality variables (turbidity (NTU), conductivity ($\mu$S/cm) and river level (m)) measured by *in situ* sensors at Sandy Creek. Bottom panel (d–f): Bi-variate relationships between transformed series (one sided derivative) of turbidity (NTU), conductivity ($\mu$S/cm) and river level (m) measured by *in situ* sensors at Sandy Creek. In each scatter plot, outliers determined by water-quality experts are shown in red, while typical points are shown in black. Neighboring points are marked in green.

Based on OP values, the one sided derivative transformation outperformed the first derivative transformation (Table 2, rows 1–2 compared to rows 3–4). Further, the distance-based outlier detection algorithms NN-HD, KNN-AGG and KNN-SUM outperformed all others (Table 2, rows 1–10 compared to rows 11–48). Among the three methods, the performance of $k$-nearest neighbor distance-based algorithms were only slightly higher (OP = 0.83) than the NN-HD algorithm (OP= 0.80), which is based only on the nearest neighbor distance. The algorithm combinations with the two highest OP values also had highest NPV (0.9996) and PPV (approximately 0.83). Furthermore, considering river level for the detection of outliers in the water-quality sensors slightly improved the performance (OP = 0.83). Among the analysis with transformed series, LOF with the first derivative transformation performed the least well (OP= 0.25). For most of the outlier detection algorithms (KNN-SUM, KNN-AGG, NN-HD , COF, LOF and INFLO) the poor-

**Table 2.** Performance metrics of outlier detection algorithms performed on multivariate water-quality time series data (T, turbidity; C, conductivity; L, river level) from *in situ* sensors at Sandy Creek, arranged in descending order of OP values. See Sections 2.7-8 for performance metric codes and details.

| i | Variables | Transformation | Method | Accuracy | GM | OP | PPV | NPV | Time (mean) |
|---|---|---|---|---|---|---|---|---|---|
| 1 | T-C-L | One sided Derivative | KNN-AGG | 0.9994 | 164.23 | 0.83 | 0.83 | 0.9996 | 404.0 |
| 2 | T-C-L | One sided Derivative | KNN-SUM | 0.9994 | 164.23 | 0.83 | 0.83 | 0.9996 | 186.8 |
| 3 | T-C | First Derivative | NN-HD | 0.9991 | 146.87 | 0.80 | 0.57 | 0.9996 | 45.0 |
| 4 | T-C | First Derivative | KNN-AGG | 0.9989 | 146.86 | 0.80 | 0.50 | 0.9996 | 415.8 |
| 5 | T-C | One sided Derivative | NN-HD | 0.9996 | 146.91 | 0.80 | 1.00 | 0.9996 | 112.9 |
| 6 | T-C | One sided Derivative | KNN-AGG | 0.9994 | 146.90 | 0.80 | 0.80 | 0.9996 | 411.7 |
| 7 | T-C | One sided Derivative | KNN-SUM | 0.9994 | 146.90 | 0.80 | 0.80 | 0.9996 | 190.4 |
| 8 | T-C-L | First Derivative | KNN-AGG | 0.9993 | 127.22 | 0.60 | 1.00 | 0.9993 | 404.4 |
| 9 | T-C-L | First Derivative | KNN-SUM | 0.9993 | 127.22 | 0.60 | 1.00 | 0.9993 | 188.9 |
| 10 | T-C | First Derivative | KNN-SUM | 0.9993 | 103.88 | 0.50 | 1.00 | 0.9993 | 189.5 |
| 11 | T-C | First Derivative | LDOF | 0.9991 | 103.87 | 0.50 | 0.67 | 0.9993 | 17444.7 |
| 12 | T-C | One sided Derivative | LDOF | 0.9991 | 103.87 | 0.50 | 0.67 | 0.9993 | 17253.8 |
| 13 | T-C-L | First Derivative | NN-HD | 0.9991 | 103.87 | 0.44 | 1.00 | 0.9991 | 52.5 |
| 14 | T-C-L | First Derivative | INFLO | 0.9965 | 103.74 | 0.44 | 0.12 | 0.9991 | 1107.9 |
| 15 | T-C-L | First Derivative | COF | 0.9987 | 103.86 | 0.44 | 0.50 | 0.9991 | 5939.8 |
| 16 | T-C-L | First Derivative | RKOF | 0.9963 | 103.73 | 0.44 | 0.12 | 0.9991 | 369.7 |
| 17 | T-C-L | One sided Derivative | NN-HD | 0.9991 | 103.87 | 0.44 | 1.00 | 0.9991 | 118.2 |
| 18 | T-C-L | One sided Derivative | INFLO | 0.9985 | 103.85 | 0.44 | 0.40 | 0.9991 | 1113.6 |
| 19 | T-C-L | One sided Derivative | COF | 0.9987 | 103.86 | 0.44 | 0.50 | 0.9991 | 5787.4 |
| 20 | T-C-L | One sided Derivative | LDOF | 0.9985 | 103.85 | 0.44 | 0.40 | 0.9991 | 17261.9 |
| 21 | T-C-L | One sided Derivative | LOF | 0.9985 | 103.85 | 0.44 | 0.40 | 0.9991 | 516.9 |
| 22 | T-C-L | One sided Derivative | RKOF | 0.9976 | 103.80 | 0.44 | 0.20 | 0.9991 | 370.5 |
| 23 | T-C-L | Original series | KNN-AGG | 0.9989 | 103.87 | 0.44 | 0.67 | 0.9991 | 391.6 |
| 24 | T-C-L | Original series | INFLO | 0.9974 | 103.79 | 0.44 | 0.18 | 0.9991 | 1070.7 |
| 25 | T-C-L | Original series | LDOF | 0.9987 | 103.86 | 0.44 | 0.50 | 0.9991 | 17156.9 |
| 26 | T-C-L | Original series | RKOF | 0.9985 | 103.85 | 0.44 | 0.40 | 0.9991 | 354.0 |
| 27 | T-C | First Derivative | INFLO | 0.9983 | 73.43 | 0.28 | 0.20 | 0.9991 | 1194.9 |
| 28 | T-C | First Derivative | COF | 0.9991 | 73.46 | 0.28 | 1.00 | 0.9991 | 5991.8 |
| 29 | T-C | First Derivative | LOF | 0.9987 | 73.44 | 0.28 | 0.33 | 0.9991 | 512.3 |
| 30 | T-C | First Derivative | RKOF | 0.9983 | 73.43 | 0.28 | 0.20 | 0.9991 | 363.2 |
| 31 | T-C | One sided Derivative | INFLO | 0.9987 | 73.44 | 0.28 | 0.33 | 0.9991 | 1207.0 |
| 32 | T-C | One sided Derivative | COF | 0.9987 | 73.44 | 0.28 | 0.33 | 0.9991 | 5880.8 |
| 33 | T-C | One sided Derivative | LOF | 0.9969 | 73.38 | 0.28 | 0.08 | 0.9991 | 511.3 |
| 34 | T-C | One sided Derivative | RKOF | 0.9961 | 73.35 | 0.28 | 0.06 | 0.9991 | 368.3 |
| 35 | T-C | Original series | KNN-AGG | 0.9989 | 73.45 | 0.28 | 0.50 | 0.9991 | 405.1 |
| 36 | T-C | Original series | INFLO | 0.9974 | 73.40 | 0.28 | 0.10 | 0.9991 | 1143.6 |
| 37 | T-C | Original series | LDOF | 0.9987 | 73.44 | 0.28 | 0.33 | 0.9991 | 17022.9 |
| 38 | T-C | Original series | RKOF | 0.9985 | 73.44 | 0.28 | 0.25 | 0.9991 | 351.8 |
| 39 | T-C-L | First Derivative | LDOF | 0.9989 | 73.45 | 0.25 | 1.00 | 0.9989 | 17323.2 |
| 40 | T-C-L | First Derivative | LOF | 0.9989 | 73.45 | 0.25 | 1.00 | 0.9989 | 517.1 |
| 41 | T-C-L | Original series | NN-HD | 0.9987 | 73.44 | 0.25 | 0.50 | 0.9989 | 48.6 |
| 42 | T-C-L | Original series | KNN-SUM | 0.9989 | 73.45 | 0.25 | 1.00 | 0.9989 | 177.3 |
| 43 | T-C-L | Original series | COF | 0.9989 | 73.45 | 0.25 | 1.00 | 0.9989 | 5931.7 |
| 44 | T-C-L | Original series | LOF | 0.9989 | 73.45 | 0.25 | 1.00 | 0.9989 | 505.0 |
| 45 | T-C | Original series | NN-HD | 0.9987 | 0.00 | 0.00 | 0.00 | 0.9989 | 41.7 |
| 46 | T-C | Original series | KNN-SUM | 0.9989 | 0.00 | 0.00 | NaN | 0.9989 | 184.6 |
| 47 | T-C | Original series | COF | 0.9989 | 0.00 | 0.00 | NaN | 0.9989 | 5896.4 |
| 48 | T-C | Original series | LOF | 0.9989 | 0.00 | 0.00 | NaN | 0.9989 | 502.7 |

est performances were associated with the untransformed original series, having the lowest OP and NPV values, highlighting how data transformation can improve the ability of outlier detection algorithms while maintaining low false detection rates.

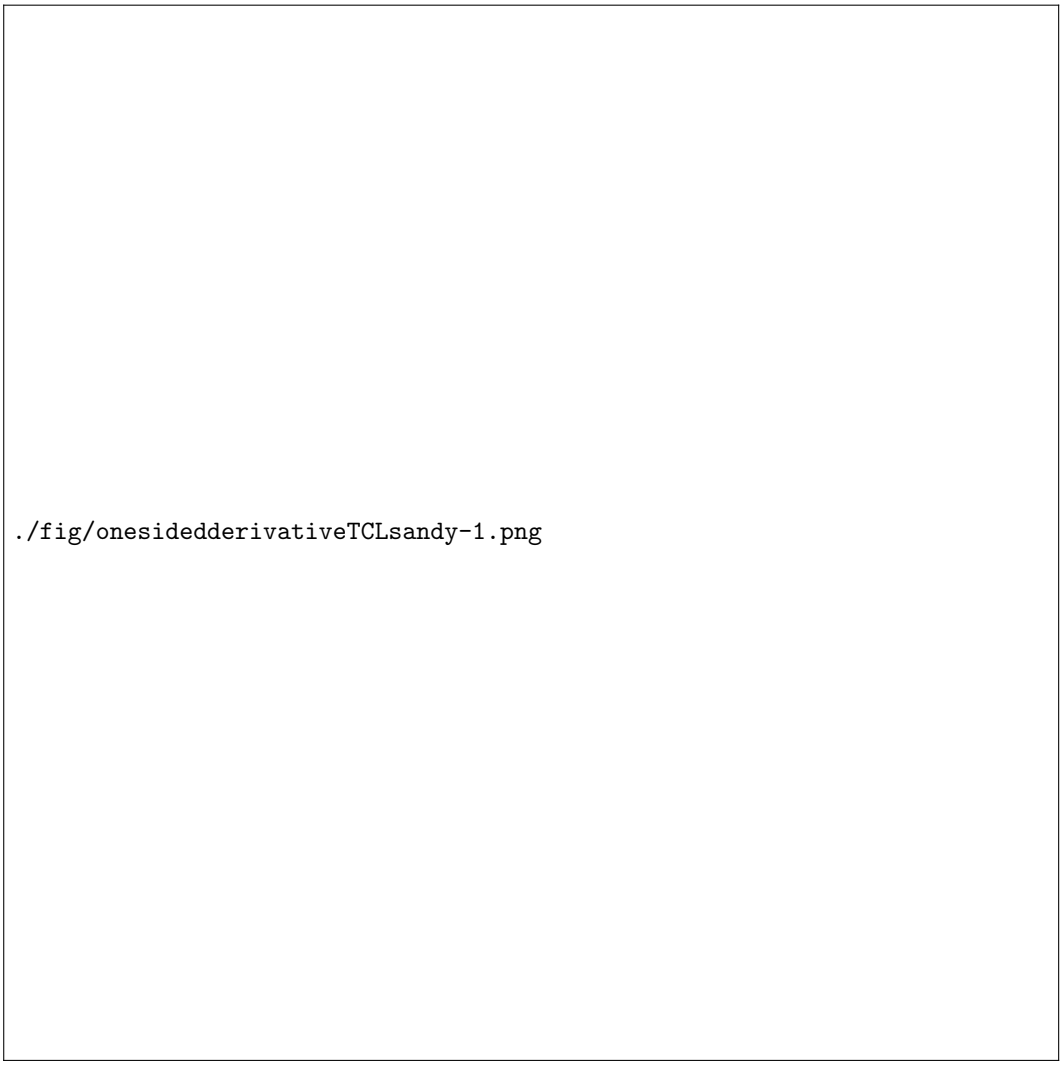



**Figure 5.** Classification of outlier scores produced from different algorithms as true negatives (TN), true positives (TP), false negatives (FN), false positives (FP). The top three panels (i, ii, iii) correspond to the original series (turbidity, conductivity and river level) measured by *in situ* sensors at Sandy Creek. The target outliers (detected by water-quality experts) are shown in red, while typical points are shown in black. The remaining panels (a–h) give outlier scores produced by different outlier detection algorithms for high dimensional data when applied to the transformed series (one sided derivative) of the three variables: turbidity, conductivity and level. Through different outlier scoring algorithms (Panel a - h), we are evaluating whether each point in time is an outlier or not. Therefore, from Panel a-h, if the outlier scoring algorithm is effective, then there should be either TP or TN at each point in time when either a red triangle is plotted in at least one of the three panels (i- iii), or black dots are plotted in all of the top three panels (i - iii), respectively. Because outlier scores are non negative and are mostly clustered near zero, with some occasional high values, a square root transformation was applied to reduce skewness of the data in Panel (a) to (h).

The three outlier detection algorithms that demonstrated the highest level of accuracy (NN-HD, KNN-AGG and KNN-SUM) also outperformed the others with respect to computational time. NN-HD algorithm required the least computational time. Among the remaining two, the mean computational time of KNN-AGG ($\approx$ 400 milliseconds) was twice that of KNN-SUM's ($<$ 200 milliseconds). LOF and its extensions (INFLO, COF and LDOF) demonstrated the poorest performance with respect computational time ($>$ 500 milliseconds on average).

Only KNN-SUM and KNN-AGG assigned high scores to most of the targeted outliers in turbidity, conductivity and level data transformed using the one-sided derivative (Figure 5(a,b)). For each outlying instance, however, the next immediate neighboring point was assigned the high outlier score instead of the true outlying point. After determining the most influential variable using the additional steps of the algorithm (Section 2.7), adjustments were made to correct this to the actual outlier. Because of this correction, the first orange triangle for the True Positive in Figure 5(a – h), for instance, is always plotted next to the high outlier score (corresponding to the neighboring point), pointing to the actual outlier instead of the neighbouring point. The outlier scores produced by LOF and COF (Figure 5(d,e)) were unable to capture the outlying behaviors correctly and demonstrated high scattering. In comparison to other outlier scoring algorithms, KNN-SUM algorithm displayed a good compromise between accuracy and computational efficiency ( Table 2).

### 3.2 Analysis of water-quality data from *in situ* sensors at Pioneer River

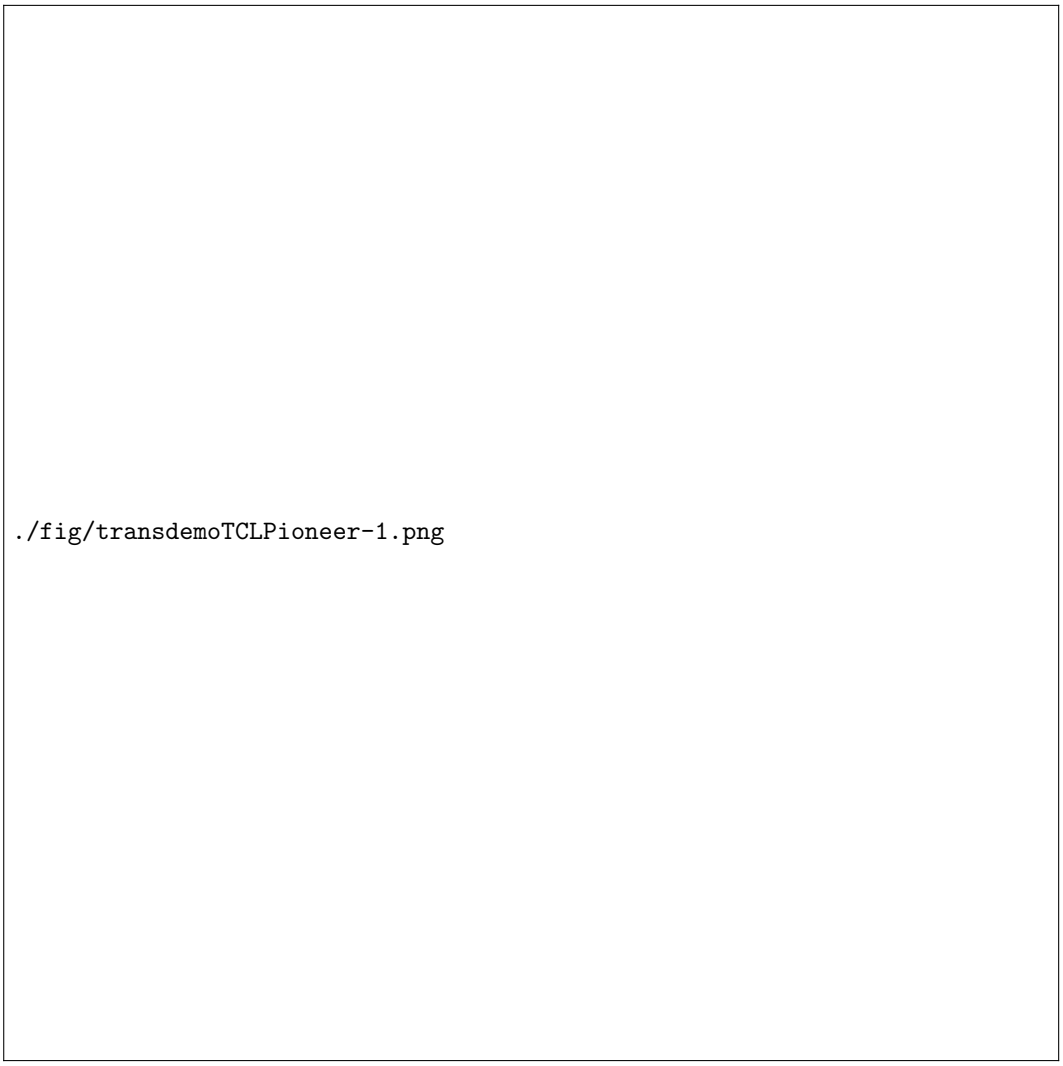
Compared to Sandy Creek where the river level is mostly less than 1 meter with occasional bursts of atypical spikes and flow events resulting in levels up to 14.8 meters (Figure 3 (c–i)), Pioneer River is much deeper with the river level ranging between 13.9 and 16.5 metres during the period of study (Figure 6 (c–i)). Two small dense clusters of points gathered around zero were observed for all three variables from late March to mid April in 2017 (Figure 6). These co-occurrences of values around zero are atypical behaviour and may have been due to technical issues with the sensor equipment. These type of anomalies can be easily detected by incorporating rule based methods.

Some of the target outliers in the data obtained from the *in situ* sensors at Pioneer River only deviated slightly from the general trend (Figure 6 (a–i)), making outlier detection challenging. A negative relationship was clearly visible between turbidity and conductivity (Figure 7(a)), however, the relationship between level and conductivity was complex (Figure 7(c)). Most of the target outliers were masked by the typical points in the original space (Figure 7(a–c)). Similar to Sandy Creek, data obtained from the sensors at Pioneer River showed good separation between outliers and typical points under the one sided derivative transformation (Figures 7(d–f) and 6 (a–ii, b–ii, c–ii). However, the sudden spikes in turbidity labeled as outliers by water-quality experts could not be separated from the majority by a large distance and were only visible as a small group (micro cluster (Goldstein & Uchida, 2016)) in the boundary defined by the typical points (Figure 7(d, e)).

From the performance analysis, it was observed that turbidity and conductivity together produced better results (Table 3, rows 1–8) than when combined with river level, which tended to reduce the performance (i.e. generating lower OP and NPV values) while increasing the false negative rate (Table 3, rows 9–13). KNN-AGG and KNN-SUM (Table 3, rows 2–3) had the highest accuracy (0.9978), highest geometric means (492.8012), highest OP (0.88) and highest NPV (0.9984). Despite the challenge given by the small spikes which could not be clearly separated from the typical points, KNN-AGG, KNN-SUM and NN-HD with one sided derivatives of turbidity and conductivity still detected some of those points as outliers while maintaining low false negative and false positive rates. Similar to Sandy Creek, NN-HD ($<$ 200 milliseconds on average) and KNN-SUM

555 (< 230 milliseconds on average) demonstrated the highest computational efficiency for
556 the data obtained from Pioneer River.






557 **Figure 6.** Time series for turbidity (NTU) (a–i), conductivity ($\mu$S/cm) (b–i) and river level
558 (m) (c–i) measured by *in situ* sensors at Pioneer River. Transformed series (one sided deriva-
559 tives) of turbidity (NTU) (a–ii), conductivity ($\mu$S/cm) (b–ii) and river level (m) (c–ii) measured
560 by *in situ* sensors at Pioneer River. In each plot, outliers determined by water-quality experts are
561 shown in red, while typical points are shown in black. Neighboring points are marked in green.

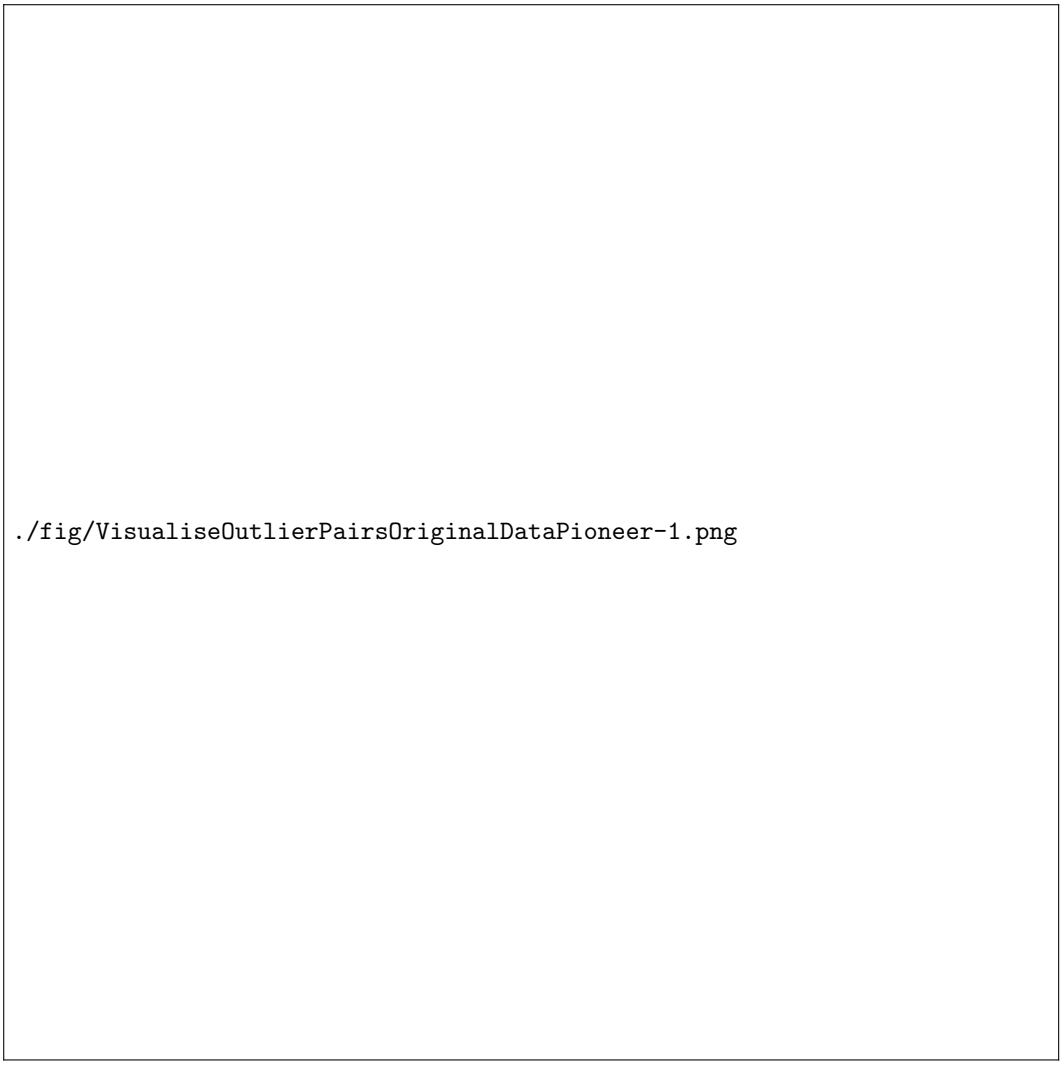



**Figure 7.** Top panel (a–c): Bi-variate relationships between original water-quality variables (turbidity (NTU), conductivity ($\mu$S/cm) and river level (m)) measured by *in situ* sensors at Pioneer River. Bottom panel (d–f): Bi-variate relationships between transformed series (one sided derivative) of turbidity (NTU), conductivity ($\mu$S/cm) and river level (m) measured by *in situ* sensors at Pioneer River. In each scatter plot, outliers determined by water-quality experts are shown in red, while typical points are shown in black. Neighboring points are marked in green.

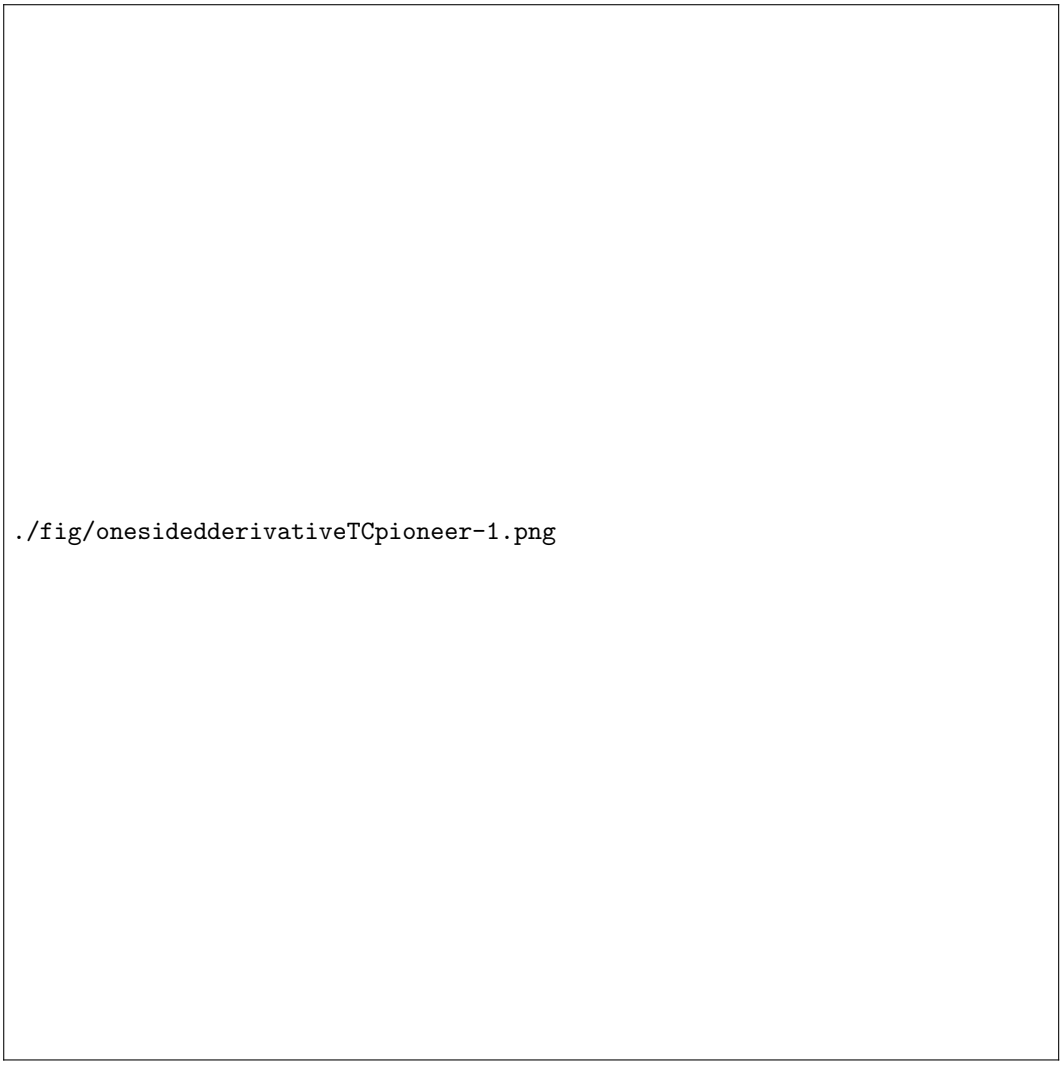



**Figure 8.** Classification of outlier scores produced from different algorithms as true negatives (TN), true positives (TP), false negatives (FN), false positives (FP). The top two panels (i and ii) correspond to the original series (turbidity and conductivity) measured by *in situ* sensors at Pioneer River. The target outliers (detected by water-quality experts) are shown in red, while typical points are shown in black. The remaining panels (a–h) give outlier scores produced by different outlier detection algorithms for high dimensional data when applied to the transformed series (one sided derivative) of the two variables: turbidity and conductivity. Through different outlier scoring algorithms (Panel a - h), we are evaluating whether each point in time is an outlier or not. Therefore, from Panel a-h, if the outlier scoring algorithm is effective, then there should be either TP or TN at each point in time when either a red triangle is plotted in at least one of the two panels (i- ii), or black dots are plotted in both of the top two panels (i - ii), respectively. Because outlier scores are non negative and are mostly clustered near zero, with some occasional high values, a square root transformation was applied to reduce skewness of the data in Panel (a) to (h).

582 **Table 3.** Performance metrics of outlier detection algorithms performed on multivariate water-
583 quality time series data (T, turbidity; C, conductivity; L, river level) from *in situ* sensors at
584 Pioneer River, arranged in descending order of OP values. See Sections 2.7-8 for performance
585 metric codes and details.

| i | Variables | Transformation | Method | Accuracy | GM | OP | PPV | NPV | Time (mean) |
|---|---|---|---|---|---|---|---|---|---|
| 1 | T-C | One sided Derivative | NN-HD | 0.9976 | 492.76 | 0.88 | 0.89 | 0.9984 | 136.5 |
| 2 | T-C | One sided Derivative | KNN-AGG | 0.9978 | 492.80 | 0.88 | 0.91 | 0.9984 | 478.8 |
| 3 | T-C | One sided Derivative | KNN-SUM | 0.9978 | 492.80 | 0.88 | 0.91 | 0.9984 | 222.2 |
| 4 | T-C | First Derivative | NN-HD | 0.9978 | 480.08 | 0.86 | 0.95 | 0.9981 | 182.0 |
| 5 | T-C | First Derivative | KNN-AGG | 0.9978 | 480.08 | 0.86 | 0.95 | 0.9981 | 488.5 |
| 6 | T-C | First Derivative | KNN-SUM | 0.9978 | 480.08 | 0.86 | 0.95 | 0.9981 | 225.3 |
| 7 | T-C | First Derivative | INFLO | 0.9971 | 479.92 | 0.86 | 0.86 | 0.9981 | 1525.0 |
| 8 | T-C | First Derivative | RKOF | 0.9970 | 479.88 | 0.86 | 0.84 | 0.9981 | 430.4 |
| 9 | T-C-L | One sided Derivative | KNN-AGG | 0.9975 | 492.72 | 0.86 | 0.91 | 0.9981 | 465.2 |
| 10 | T-C-L | One sided Derivative | KNN-SUM | 0.9975 | 492.72 | 0.86 | 0.91 | 0.9981 | 214.5 |
| 11 | T-C-L | First Derivative | RKOF | 0.9951 | 485.82 | 0.85 | 0.68 | 0.9979 | 425.9 |
| 12 | T-C-L | First Derivative | KNN-AGG | 0.9975 | 480.00 | 0.84 | 0.95 | 0.9978 | 478.0 |
| 13 | T-C-L | First Derivative | KNN-SUM | 0.9975 | 480.00 | 0.84 | 0.95 | 0.9978 | 220.0 |
| 14 | T-C | First Derivative | COF | 0.9978 | 473.58 | 0.84 | 0.97 | 0.9979 | 7908.2 |
| 15 | T-C | First Derivative | LDOF | 0.9978 | 473.58 | 0.84 | 0.97 | 0.9979 | 23435.7 |
| 16 | T-C | First Derivative | LOF | 0.9975 | 473.51 | 0.84 | 0.92 | 0.9979 | 594.4 |
| 17 | T-C | One sided Derivative | INFLO | 0.9973 | 473.47 | 0.84 | 0.90 | 0.9979 | 1559.9 |
| 18 | T-C | One sided Derivative | COF | 0.9976 | 473.54 | 0.84 | 0.95 | 0.9979 | 7505.5 |
| 19 | T-C | One sided Derivative | LDOF | 0.9975 | 473.51 | 0.84 | 0.92 | 0.9979 | 22986.0 |
| 20 | T-C | One sided Derivative | LOF | 0.9975 | 473.51 | 0.84 | 0.92 | 0.9979 | 596.9 |
| 21 | T-C | One sided Derivative | RKOF | 0.9960 | 473.16 | 0.84 | 0.75 | 0.9979 | 419.7 |
| 22 | T-C | Original Series | INFLO | 0.9973 | 473.47 | 0.84 | 0.90 | 0.9979 | 1498.5 |
| 23 | T-C-L | First Derivative | COF | 0.9975 | 473.51 | 0.83 | 0.97 | 0.9976 | 7910.7 |
| 24 | T-C-L | First Derivative | LDOF | 0.9975 | 473.51 | 0.83 | 0.97 | 0.9976 | 23357.7 |
| 25 | T-C-L | One sided Derivative | NN-HD | 0.9975 | 473.51 | 0.83 | 0.97 | 0.9976 | 131.9 |
| 26 | T-C | Original Series | NN-HD | 0.9976 | 466.96 | 0.83 | 0.97 | 0.9978 | 171.0 |
| 27 | T-C | Original Series | KNN-AGG | 0.9970 | 466.81 | 0.83 | 0.88 | 0.9978 | 468.7 |
| 28 | T-C | Original Series | KNN-SUM | 0.9970 | 466.81 | 0.83 | 0.88 | 0.9978 | 211.6 |
| 29 | T-C | Original Series | COF | 0.9978 | 467.00 | 0.83 | 1.00 | 0.9978 | 7617.6 |
| 30 | T-C | Original Series | LDOF | 0.9978 | 467.00 | 0.83 | 1.00 | 0.9978 | 22910.4 |
| 31 | T-C | Original Series | LOF | 0.9978 | 467.00 | 0.83 | 1.00 | 0.9978 | 579.1 |
| 32 | T-C | Original Series | RKOF | 0.9963 | 466.66 | 0.83 | 0.80 | 0.9978 | 401.9 |
| 33 | T-C-L | First Derivative | NN-HD | 0.9973 | 473.47 | 0.82 | 0.95 | 0.9976 | 167.1 |
| 34 | T-C-L | One sided Derivative | INFLO | 0.9971 | 473.43 | 0.82 | 0.92 | 0.9976 | 1418.8 |
| 35 | T-C-L | One sided Derivative | COF | 0.9973 | 473.47 | 0.82 | 0.95 | 0.9976 | 7497.9 |
| 36 | T-C-L | One sided Derivative | LDOF | 0.9973 | 473.47 | 0.82 | 0.95 | 0.9976 | 23090.7 |
| 37 | T-C-L | One sided Derivative | RKOF | 0.9952 | 472.97 | 0.82 | 0.71 | 0.9976 | 422.1 |
| 38 | T-C-L | First Derivative | INFLO | 0.9975 | 466.92 | 0.81 | 1.00 | 0.9974 | 1398.3 |
| 39 | T-C-L | First Derivative | LOF | 0.9975 | 466.92 | 0.81 | 1.00 | 0.9974 | 600.7 |
| 40 | T-C-L | One sided Derivative | LOF | 0.9965 | 466.70 | 0.81 | 0.85 | 0.9974 | 596.1 |
| 41 | T-C-L | Original Series | NN-HD | 0.9973 | 466.88 | 0.81 | 0.97 | 0.9974 | 163.0 |
| 42 | T-C-L | Original Series | KNN-AGG | 0.9967 | 466.73 | 0.81 | 0.88 | 0.9974 | 456.3 |
| 43 | T-C-L | Original Series | KNN-SUM | 0.9967 | 466.73 | 0.81 | 0.88 | 0.9974 | 201.4 |
| 44 | T-C-L | Original Series | INFLO | 0.9975 | 466.92 | 0.81 | 1.00 | 0.9974 | 1372.8 |
| 45 | T-C-L | Original Series | COF | 0.9975 | 466.92 | 0.81 | 1.00 | 0.9974 | 7707.2 |
| 46 | T-C-L | Original Series | LDOF | 0.9975 | 466.92 | 0.81 | 1.00 | 0.9974 | 127337.1 |
| 47 | T-C-L | Original Series | LOF | 0.9975 | 466.92 | 0.81 | 1.00 | 0.9974 | 580.9 |
| 48 | T-C-L | Original Series | RKOF | 0.9955 | 466.47 | 0.81 | 0.74 | 0.9974 | 406.8 |

## 4 Discussion

We introduced a new procedure, named oddwater procedure for the detection of outliers in water-quality data from *in situ* sensors, where outliers were specifically defined as due to technical errors that make the data unreliable and untrustworthy. We showed that our oddwater procedure, with carefully selected data transformation methods derived from data features, can greatly assist in increasing the performance of a range of existing outlier detection algorithms. Our oddwater procedure and analysis using data obtained from *in situ* sensors positioned at two study sites, Sandy Creek and Pioneer River, performed well with outlier types such as sudden isolated spikes, sudden isolated drops and level shifts, while maintaining low false detection rates. As an unsupervised procedure, our approach can be easily extended to other water-quality variables, other sites and also to other outlier detection tasks in other application domains. The only requirement is to select suitable transformation methods according to the data features that differentiate the outlying instances from the typical behaviors of a given system.

Studies have shown that transforming variables affects densities, relative distances and orientation of points within the data space and therefore can improve the ability to perceive patterns in the data which are not clearly visible in the original data space (Dang & Wilkinson, 2014). This was the case in our study where no clear separation was visible between outliers and typical data points in the original data space but a clear separation was obtained between the two sets of points once the one-sided derivative transformation was applied to the original series. Having this type of a separation between outliers and typical points is important before applying unsupervised outlier detection algorithms for high dimensional data because the methods are usually based on the definition of outliers in terms of distance or density (Talagala, Hyndman, Smith-Miles, Kandanaarachchi, & Muñoz, 2019). Most of the outlier detection algorithms (KNN-SUM, KNN-AGG, NN-HD, COF, LOF and INFLO) performed least well with the untransformed original series, demonstrating how data transformation methods can assist in improving the ability of outlier detection algorithms while maintaining low false detection rates.

In our modified algorithm, the NN-HD algorithm, we did not incorporate the clustering step of the HDoutliers algorithm because the data obtained from the two study sites are free from micro clusters (Talagala, Hyndman, & Smith-Miles, 2019) and therefore free from the masking problem. Because the datasets have only local and global outliers, incorporating a clustering step that forms small clusters using a small ball with a fixed radius (the Leader Algorithm in Wilkinson (2018)) does not significantly change the structure of the data points in the high dimensional data space. Furthermore, because NN-HD has the additional requirement of isolation in addition to clear separation between outlying points and typical points, it performed poorly in comparison to the two KNN distance-based algorithms (KNN-AGG and KNN-SUM) which are not restricted to the single most nearest neighbor (Talagala, Hyndman, & Smith-Miles, 2019). For the current work, $k$ was set to 10, the maximum default value of $k$ in Madsen (2018), because too large a value of $k$ could skew the focus towards global outliers (points that deviates significantly from the rest of the dataset) alone (Zhang et al., 2009) and make the algorithms computationally inefficient. On the other hand, too small a value of $k$ could incorporate an additional assumption of isolation into the algorithm, as in the NN-HD algorithm where $k = 1$. Among the analyses using transformed series, LOF with the first derivative transformation performed the least well, which could also be due to its additional assumption of isolation (Tang et al., 2002). However, using the same $k$ across all algorithms may bias direct comparison because the performance of the algorithms can depend on the value of $k$ and algorithms can reach their peak performance for different choices of $k$ (Campos et al., 2016). Therefore, performing an optimisation to select the best $k$ is non trivial and we leave it for future work.

We took the correlation structure between the variables into account when detecting outliers given some were apparent only in the high dimensional space but not when

each variable was considered independently (Ben-Gal, 2005). A negative relationship was observed between conductivity and turbidity and also between conductivity and level for the Sandy Creek data. However, for Pioneer River, no clear relationship was observed between level and the remaining two variables, turbidity and conductivity. This could be one reason why the variable combination with river level gave poor results for the Pioneer River dataset, while results for other combinations were similar to those of Sandy Creek. The one-sided derivative transformation outperformed the derivative transformation. This was expected, because in an occurrence of a sudden spike or isolated drop, the first derivative assigns high values to two consecutive points, the actual outlying point as well as the neighboring point, and therefore increases the false positive rate (because the neighboring points that are declared to be outliers actually correspond to typical points in the original data space). Therefore, to detect technical outliers in water-quality data from Sandy Creek and Pioneer River, the one sided derivative transformation is recommended because it outperformed the other transformations during the comparative analysis. For Sandy Creek, all three water-quality variables together with the one-sided derivative transformation is recommended. However, for Pioneer River, the use of river level is not advisable due its complex relationships with the other variables and its temporal variability. For both rivers, the use of KNN-SUM algorithm is recommended because it provides a good compromise between accuracy and computational efficiency.

In this study, our goal was to detect suitable transformations, combinations of variables, and the algorithms for outlier score calculation for the data from two study sites. Results may depend on the characteristics of the time series (site and time dependent for example), and what is best for one site may not be the best for another site. Therefore, care should be taken to select transformations most suitable for the problem at hand. According to Dang and Wilkinson (2014), any transformation used on a dataset must be evaluated in terms of a figure of merit (i.e. a numerical quantity used to characterize the performance of a method, relative to its alternatives). For our work on detecting outliers, the figure of merit was the maximum separability of the two classes generated by outliers and typical points. However, we acknowledge that the set of transformations that we used for this work was relatively limited and influenced by the data obtained from the two study sites. Therefore, the set of transformations we considered (Table 1) should be viewed only as an illustration of our oddwater procedure for detecting outliers. We expect that the set of transformations will expand over time as the oddwater procedure is used for other data from other study sites and for applications to other fields.

For the current work, we selected transformation methods that could highlight abrupt changes in the water-quality data.We hope to expand the ability of oddwater procedure so that it can detect other outlier types not previously targeted but commonly observed in water-quality data (e.g. low/high variability, drift etc. as per Leigh et al. (2019)). One possibility is to consider the residuals at each point, defined as the difference between the actual values and the fitted values (similar to Schwarz (2008)) or the difference between the actual values and the predicted values (similar to Hill and Minsker (2006)), as a transformation and apply outlier detection algorithms to the high dimensional space defined by those residuals. Here the challenge will be to identify the appropriate curve fitting and prediction models to generate the residual series. In this way, continuous subsequences of high values could correspond to other kinds of technical outliers such as high variability or drift. However, the range of applications and the space of the transformations are extremely diverse, which makes it challenging to provide a structured formal vision that covers all of the possible transformations that could be considered. The transformations we present in this paper were mainly chosen as appropriate to the data collected from Sandy Creek and Pioneer River. We observed that different transformations can lead to entirely different data structures and that the selection of suitable transformations is directed by the data features and typical patterns imposed by a given application. Domain specific knowledge plays a vital role when selecting suitable transforma-

tions and, as such, defining structured guidelines for the selection of suitable transformations remains problematic.

Not surprisingly, NN-HD algorithm required the least computational time given the outlying score calculation only involves searching for the single most nearest neighbors of each test point (Wilkinson, 2018). The mean computational time of KNN-AGG was twice as high as that of KNN-SUM because the KNN-AGG algorithm has the additional requirement of calculating weights that assign nearest neighbors higher weight relative to the neighbors farther apart (Angiulli & Pizzuti, 2002). LOF and its extensions (INFLO, COF and LDOF) required the most computational time; all four algorithms involve a two step searching mechanism at each test point when calculating the corresponding outlying score. This means that at each test point each algorithm searches its $k$ nearest neighbors as well those of the detected nearest neighbors for the outlier score calculation (Breunig et al., 2000; Jin et al., 2006; Tang et al., 2002; Zhang et al., 2009).

Assessing performance of the detection methods based on the classification criteria, while traditional, has limitations . During performance evaluation, we observed that some outliers were detected by all the approaches, some were detected as outliers only by certain methods and some were identified by no method. Therefore, incorporating ensemble methods as proposed in Unwin (2019) would assist in selecting the best performing approaches for a particular outlier type and enable further insight into the results obtained from the oddwater procedure.

We hope to extend our multivariate outlier detection framework into space and time so that it can deal with the spatio-temporal correlation structure along branching river networks. Further, in the current paper, we have introduced our oddwater procedure as a batch method. However, due to the unsupervised nature of our oddwater procedure it can be easily extended to a streaming data scenario with the help of a sliding window of fixed length. A streaming data scenario always demands a near-real-time support. Therefore, one significant challenge is to find efficient methods that allow us to update outlier scores taking account of the newest observations and removing the oldest observations introduced by overlapping sliding windows, rather than recalculating scores corresponding to observations which are not affected by either new arrivals or the oldest observations (that are no longer covered by the latest window). Further work will be needed to investigate the efficient computation of regenerating nearest neighbours in a data streaming context.

## Notation

***FP*** False Positives (i.e. when a typical observation is misclassified as an outlier)
***FN*** False Negatives (i.e. when an actual outlier is misclassified as a typical observation)
***TP*** True Positives (i.e. when an actual outlier is correctly classified)
***TN*** True Negatives (i.e. when an observation is correctly classified as a typical point)

## Acknowledgments

Funding for this project was provided by the Queensland Department of Environment and Science (DES) and the ARC Centre of Excellence for Mathematical and Statistical Frontiers (ACEMS). The authors would like to acknowledge the Queensland Department of Environment and Science; in particular, the Great Barrier Reef Catchment Loads Monitoring Program for the data, and the staff from Water Quality and Investigations for their input. We thank Ryan S. Turner and Erin E. Peterson for several valuable discussions regarding project requirements and water quality characteristics. Fur-

ther, this research was supported in part by the Monash eResearch Centre and eSolutions-Research Support Services through the use of the MonARCH (Monash Advanced Research Computing Hybrid) HPC Cluster. We would also like to thank David Hill and other anonymous reviewers for their valuable comments and suggestions. The datasets used for this article are available in the open source R package `oddwater` (Talagala & Hyndman, 2019b).

## References

Angiulli, F., & Pizzuti, C. (2002). Fast outlier detection in high dimensional spaces. In *European conference on principles of data mining and knowledge discovery* (pp. 15–27).

Archer, C., Baptista, A., & Leen, T. K. (2003). Fault detection for salinity sensors in the columbia estuary. *Water Resources Research*, *39*(3).

Ben-Gal, I. (2005). Outlier detection. In *Data mining and knowledge discovery handbook* (pp. 131–146). Springer.

Breunig, M. M., Kriegel, H.-P., Ng, R. T., & Sander, J. (2000). Lof: identifying density-based local outliers. In *Acm sigmod record* (Vol. 29, pp. 93–104).

Burridge, P., & Taylor, A. M. R. (2006). Additive outlier detection via extreme-value theory. *Journal of Time Series Analysis*, *27*(5), 685–701.

Campos, G. O., Zimek, A., Sander, J., Campello, R. J., Micenková, B., Schubert, E., ... Houle, M. E. (2016). On the evaluation of unsupervised outlier detection: measures, datasets, and an empirical study. *Data Mining and Knowledge Discovery*, *30*(4), 891–927.

Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. *ACM computing surveys (CSUR)*, *41*(3), 15.

Dang, T. N., & Wilkinson, L. (2014). Transforming scagnostics to reveal hidden features. *IEEE transactions on visualization and computer graphics*, *20*(12), 1624–1632.

Embrechts, P., Klüppelberg, C., & Mikosch, T. (2013). *Modelling extremal events: for insurance and finance*. Springer Berlin Heidelberg. Retrieved from `https://books.google.com.au/books?id=BXOI2pICfJUC`

Gao, J., Hu, W., Zhang, Z. M., Zhang, X., & Wu, O. (2011). Rkof: robust kernel-based local outlier detection. In *Pacific-asia conference on knowledge discovery and data mining* (pp. 270–283).

Glasgow, H. B., Burkholder, J. M., Reed, R. E., Lewitus, A. J., & Kleinman, J. E. (2004). Real-time remote monitoring of water quality: a review of current applications, and advancements in sensor, telemetry, and computing technologies. *Journal of Experimental Marine Biology and Ecology*, *300*(1-2), 409–448.

Goldstein, M., & Uchida, S. (2016). A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data. *PloS one*, *11*(4), e0152173.

Hill, D. J., & Minsker, B. S. (2006). Automated fault detection for in-situ environmental sensors. In *Proceedings of the 7th international conference on hydroinformatics.*

Hill, D. J., Minsker, B. S., & Amir, E. (2009). Real-time bayesian anomaly detection in streaming environmental data. *Water Resources Research*, *45*(4).

Hossin, M., & Sulaiman, M. (2015). A review on evaluation metrics for data classification evaluations. *International Journal of Data Mining & Knowledge Management Process*, *5*(2), 1.

Jin, W., Tung, A. K., Han, J., & Wang, W. (2006). Ranking outliers using symmetric neighborhood relationship. In *Pacific-asia conference on knowledge discovery and data mining* (pp. 577–593).

Koch, M. W., & McKenna, S. A. (2010). Distributed sensor fusion in water quality event detection. *Journal of Water Resources Planning and Management*,

*137*(1), 10–19.

Kotamäki, N., Thessler, S., Koskiaho, J., Hannukkala, A. O., Huitu, H., Huttula, T., ... Järvenpää, M. (2009). Wireless in-situ sensor network for agriculture and water monitoring on a river basin scale in southern finland: Evaluation from a data users perspective. *Sensors*, *9*(4), 2862–2883.

Kriegel, H.-P., Kröger, P., & Zimek, A. (2010). Outlier detection techniques. *Tutorial at KDD*, *10*.

Leigh, C., Alsibai, O., Hyndman, R. J., Kandanaarachchi, S., King, O. C., McGree, J. M., ... others (2019). A framework for automated anomaly detection in high frequency water-quality data from in situ sensors. *Science of The Total Environment*, *664*, 885–898.

Madsen, J. H. (2018). Ddoutlier: Distance and density-based outlier detection [Computer software manual]. Retrieved from `https://CRAN.R-project.org/package=DDoutlier` (R package version 0.1.0)

McInnes, K., Abbs, D., Bhend, J., Chiew, F., Church, J., Ekstrm, M., ... Whetton, P. (2015). *Wet tropics cluster report: Climate change in australia projections for australia's nrm regions*. CSIRO.

McKenna, S. A., Hart, D., Klise, K., Cruz, V., & Wilson, M. (2007). Event detection from water quality time series. In *World environmental and water resources congress 2007: Restoring our natural habitat* (pp. 1–12).

Mersmann, O. (2018). microbenchmark: Accurate timing functions [Computer software manual]. Retrieved from `https://CRAN.R-project.org/package=microbenchmark` (R package version 1.4-4)

Mitchell, C., Brodie, J., & White, I. (2005). Sediments, nutrients and pesticide residues in event flow conditions in streams of the mackay whitsunday region, australia. *Marine Pollution Bulletin*, *51*(1-4), 23–36.

Moatar, F., Fessant, F., & Poirel, A. (1999). ph modelling by neural networks. application of control and validation data series in the middle loire river. *Ecological Modelling*, *120*(2-3), 141–156.

Moatar, F., Miquel, J., & Poirel, A. (2001). A quality-control method for physical and chemical monitoring data. application to dissolved oxygen levels in the river loire (france). *Journal of Hydrology*, *252*(1-4), 25–36.

Panguluri, S., Meiners, G., Hall, J., & Szabo, J. (2009). Distribution system water quality monitoring: Sensor technology evaluation methodology and results. *US Environ. Protection Agency, Washington, DC, USA, Tech. Rep. EPA/600/R-09/076*, *2772*.

R Core Team. (2018). R: A language and environment for statistical computing [Computer software manual]. Vienna, Austria. Retrieved from `https://www.R-project.org/`

Raciti, M., Cucurull, J., & Nadjm-Tehrani, S. (2012). Anomaly detection in water management systems. In *Critical infrastructure protection* (pp. 98–119). Springer.

Ranawana, R., & Palade, V. (2006). Optimized precision-a new measure for classifier performance evaluation. In *Evolutionary computation, 2006. cec 2006. ieee congress on* (pp. 2254–2261).

Rangeti, I., Dzwairo, B., Barratt, G. J., & Otieno, F. A. (2015). Validity and errors in water quality dataa review. In *Research and practices in water quality*. InTech.

Schwarz, K. T. (2008). *Wind dispersion of carbon dioxide leaking from underground sequestration, and outlier detection in eddy covariance data using extreme value theory*. University of California, Berkeley.

Shahid, N., Naqvi, I. H., & Qaisar, S. B. (2015). Characteristics and classification of outlier detection techniques for wireless sensor networks in harsh environments: a survey. *Artificial Intelligence Review*, *43*(2), 193–228.

Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance mea-

sures for classification tasks. *Information Processing & Management*, *45*(4), 427–437.

Storey, M. V., Van der Gaag, B., & Burns, B. P. (2011). Advances in on-line drinking water quality monitoring and early warning systems. *Water research*, *45*(2), 741–747.

Talagala, P. D., & Hyndman, R. J. (2019a). A feature-based procedure for detecting technical outliers in water-quality data: R package oddwater v.0.7.0 [Computer software manual]. Zenodo. doi: 10.5281/zenodo.3378211

Talagala, P. D., & Hyndman, R. J. (2019b). oddwater: Outlier detection in data from water-quality sensors [Computer software manual]. Retrieved from `https://github.com/pridiltal/oddwater` (R package)

Talagala, P. D., Hyndman, R. J., & Smith-Miles, K. (2019). Anomaly detection in high dimensional data. *arXiv preprint arXiv:1908.04000*.

Talagala, P. D., Hyndman, R. J., Smith-Miles, K., Kandanaarachchi, S., & Muñoz, M. A. (2019). Anomaly detection in streaming nonstationary temporal data. *Journal of Computational and Graphical Statistics*(Accepted), 1–28.

Tang, J., Chen, Z., Fu, A. W.-C., & Cheung, D. W. (2002). Enhancing effectiveness of outlier detections for low density patterns. In *Pacific-asia conference on knowledge discovery and data mining* (pp. 535–548).

Thottan, M., & Ji, C. (2003). Anomaly detection in ip networks. *IEEE Transactions on signal processing*, *51*(8), 2191–2204.

Tutmez, B., Hatipoglu, Z., & Kaymak, U. (2006). Modelling electrical conductivity of groundwater using an adaptive neuro-fuzzy inference system. *Computers & geosciences*, *32*(4), 421–433.

Unwin, A. (2019). Multivariate outliers and the o3 plot. *Journal of Computational and Graphical Statistics*, 1–11.

Weissman, I. (1978). Estimation of parameters and large quantiles based on the k largest observations. *Journal of the American Statistical Association*, *73*(364), 812–815.

Wilkinson, L. (2018). Visualizing big data outliers through distributed aggregation. *IEEE transactions on visualization and computer graphics*, *24*(1), 256–266.

Yu, J. (2012). A bayesian inference based two-stage support vector regression framework for soft sensor development in batch bioprocesses. *Computers & Chemical Engineering*, *41*, 134–144.

Zhang, K., Hutter, M., & Jin, H. (2009). A new local distance-based outlier detection approach for scattered real-world data. In *Pacific-asia conference on knowledge discovery and data mining* (pp. 813–822).

**Figure 1.**

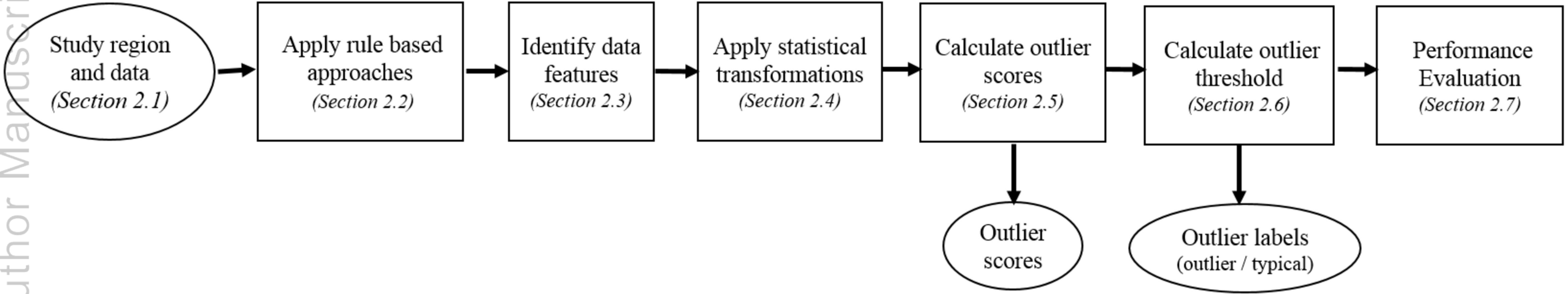


**Figure 2.**

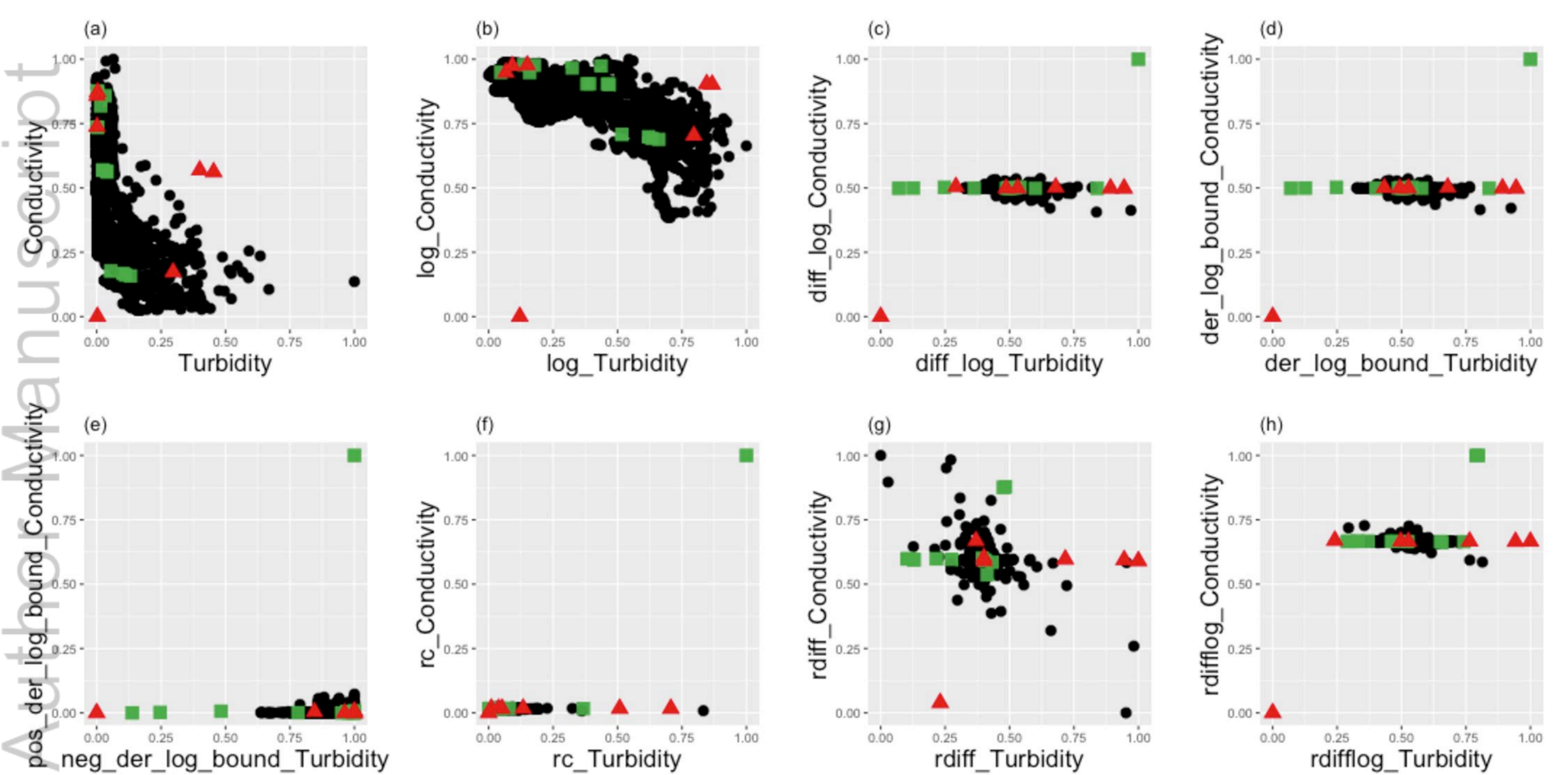


**Figure 3.**

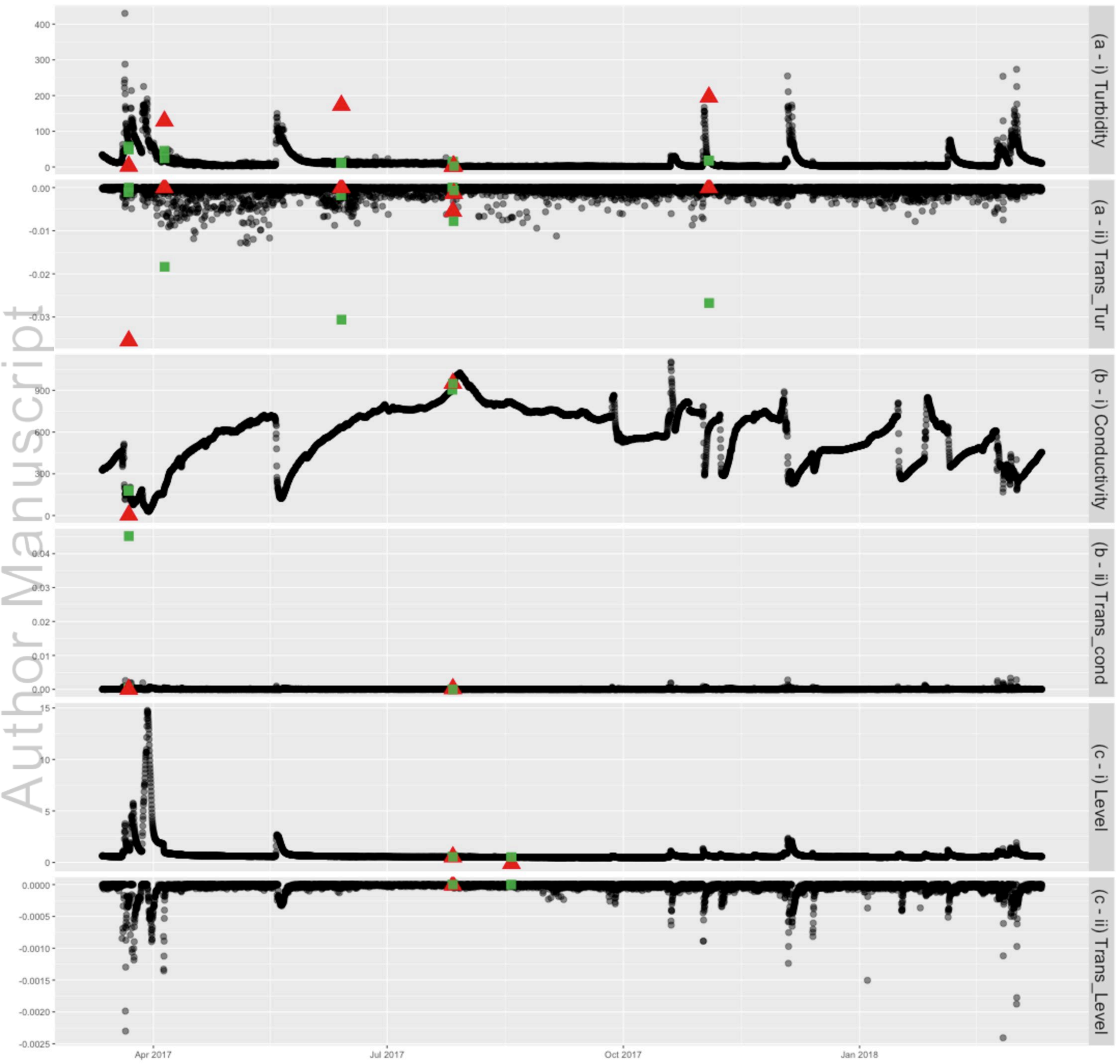


**Figure 4.**

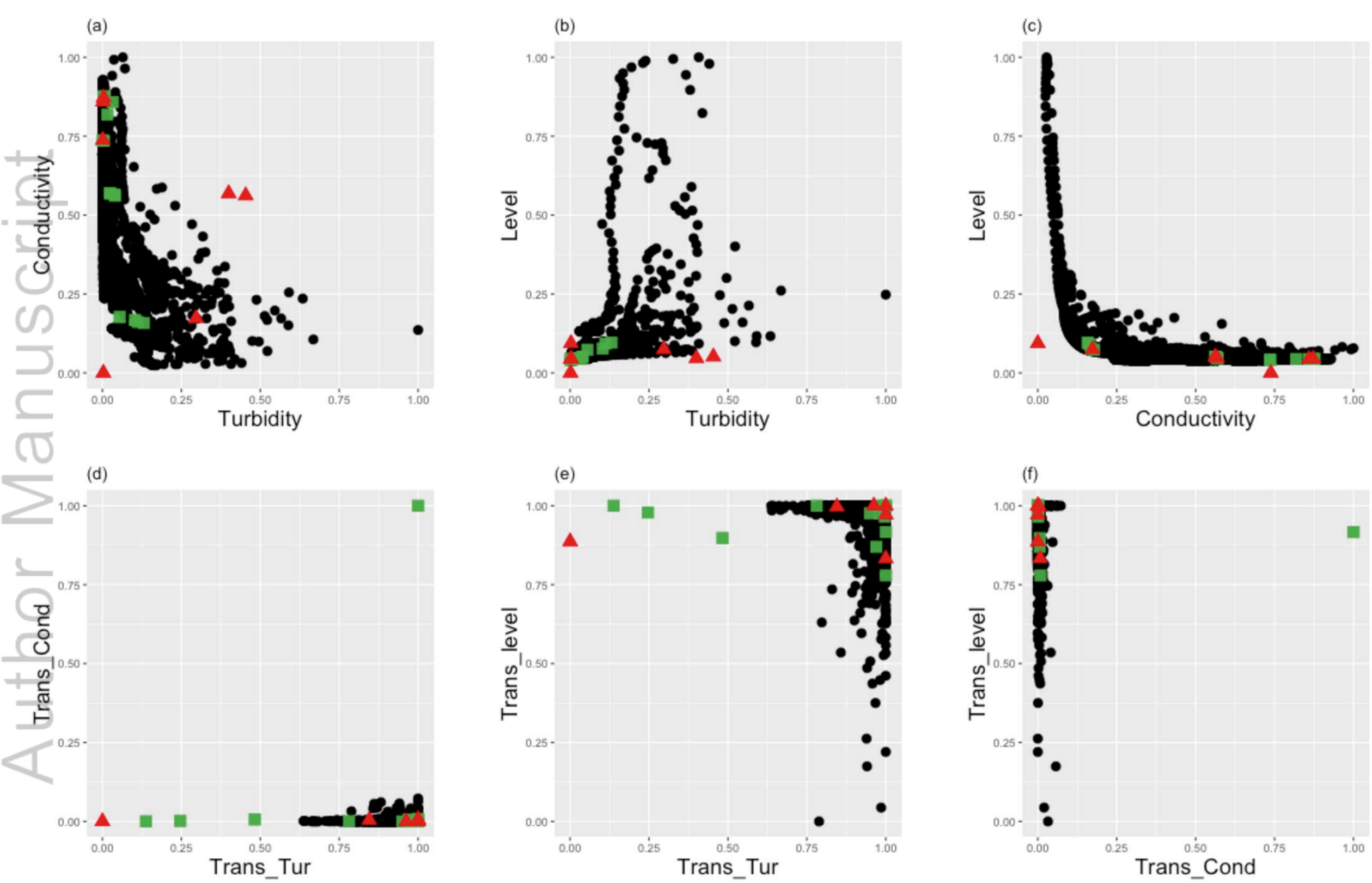


**Figure 5.**

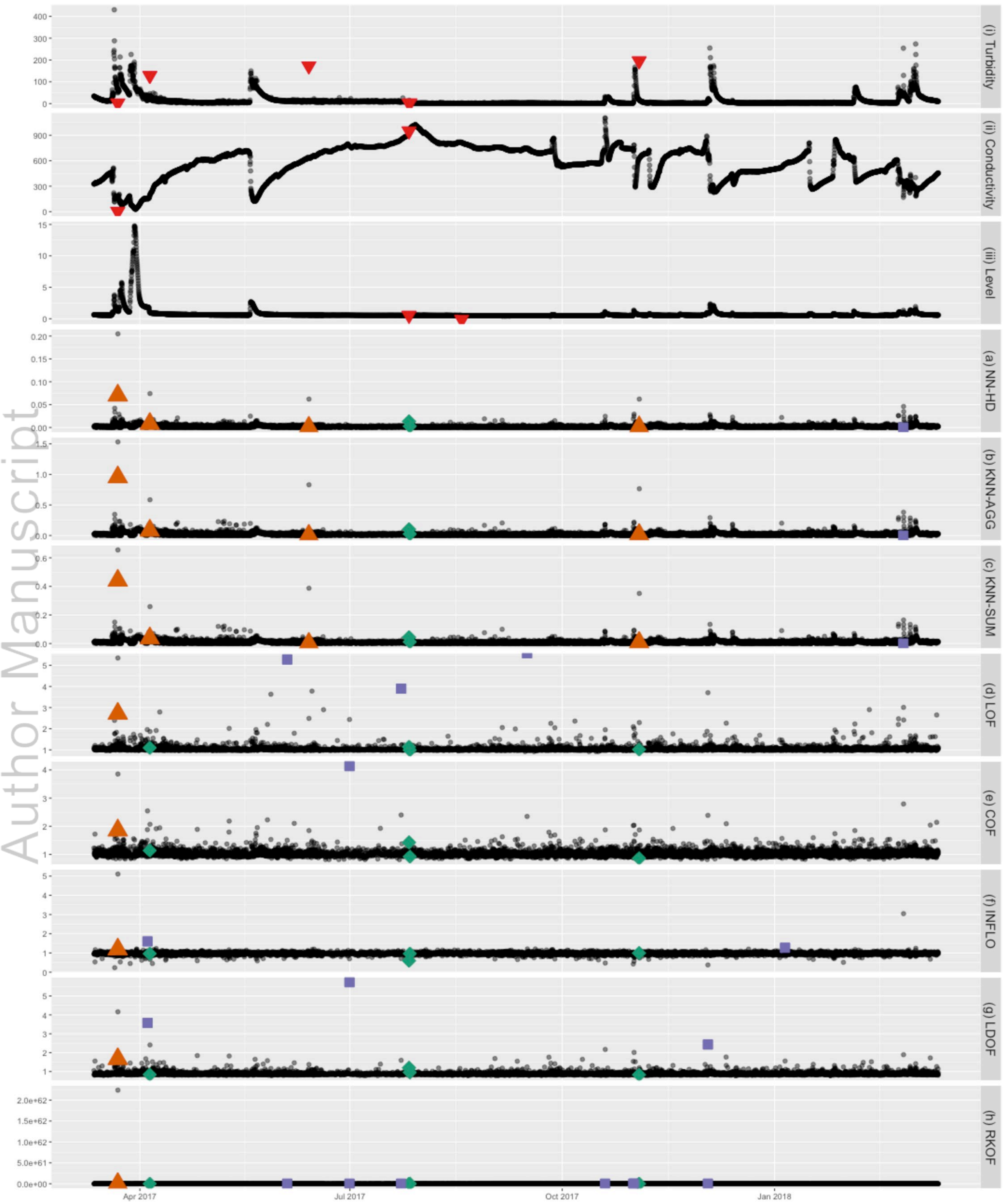


**Figure 6.**

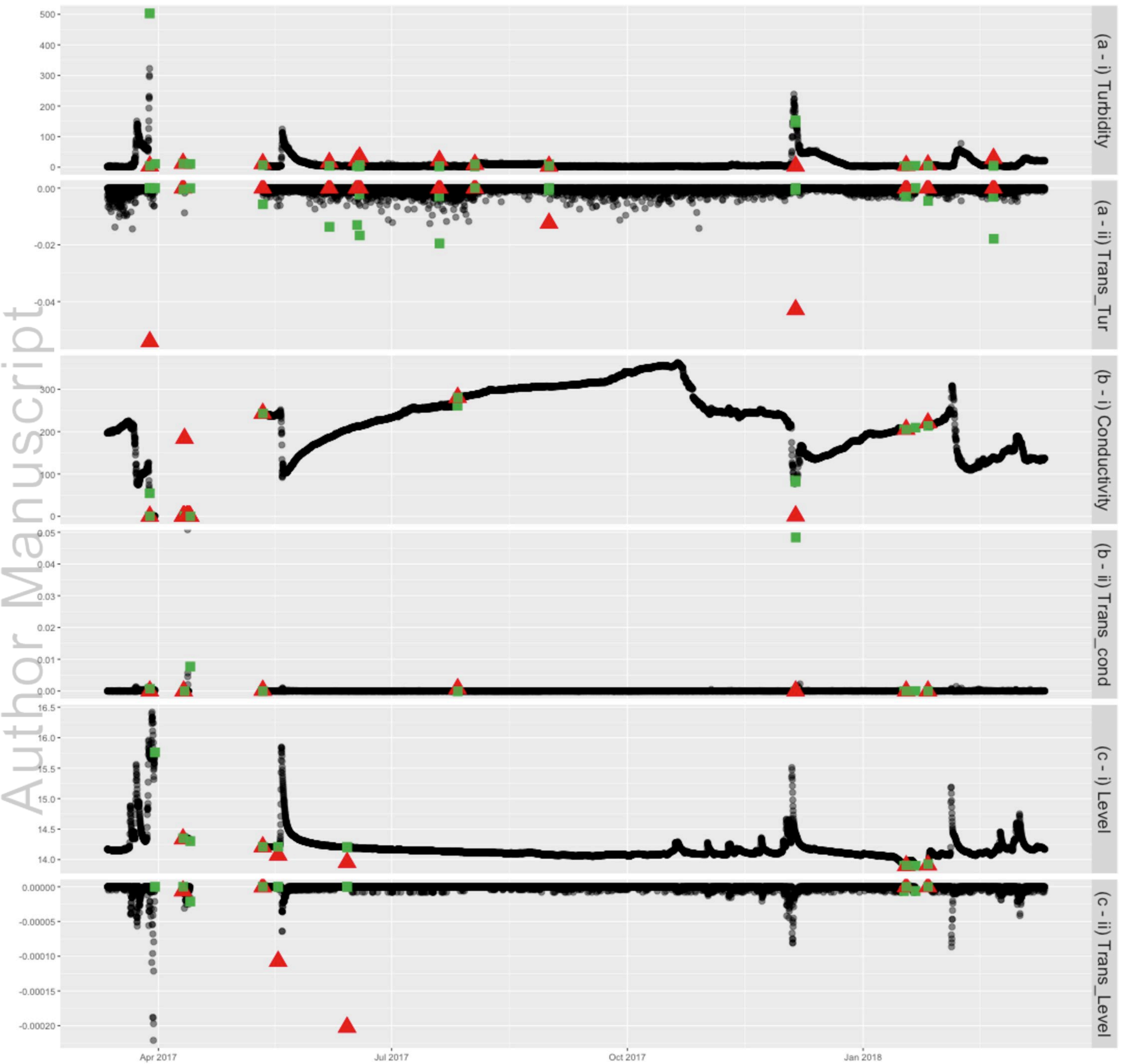


**Figure 7.**

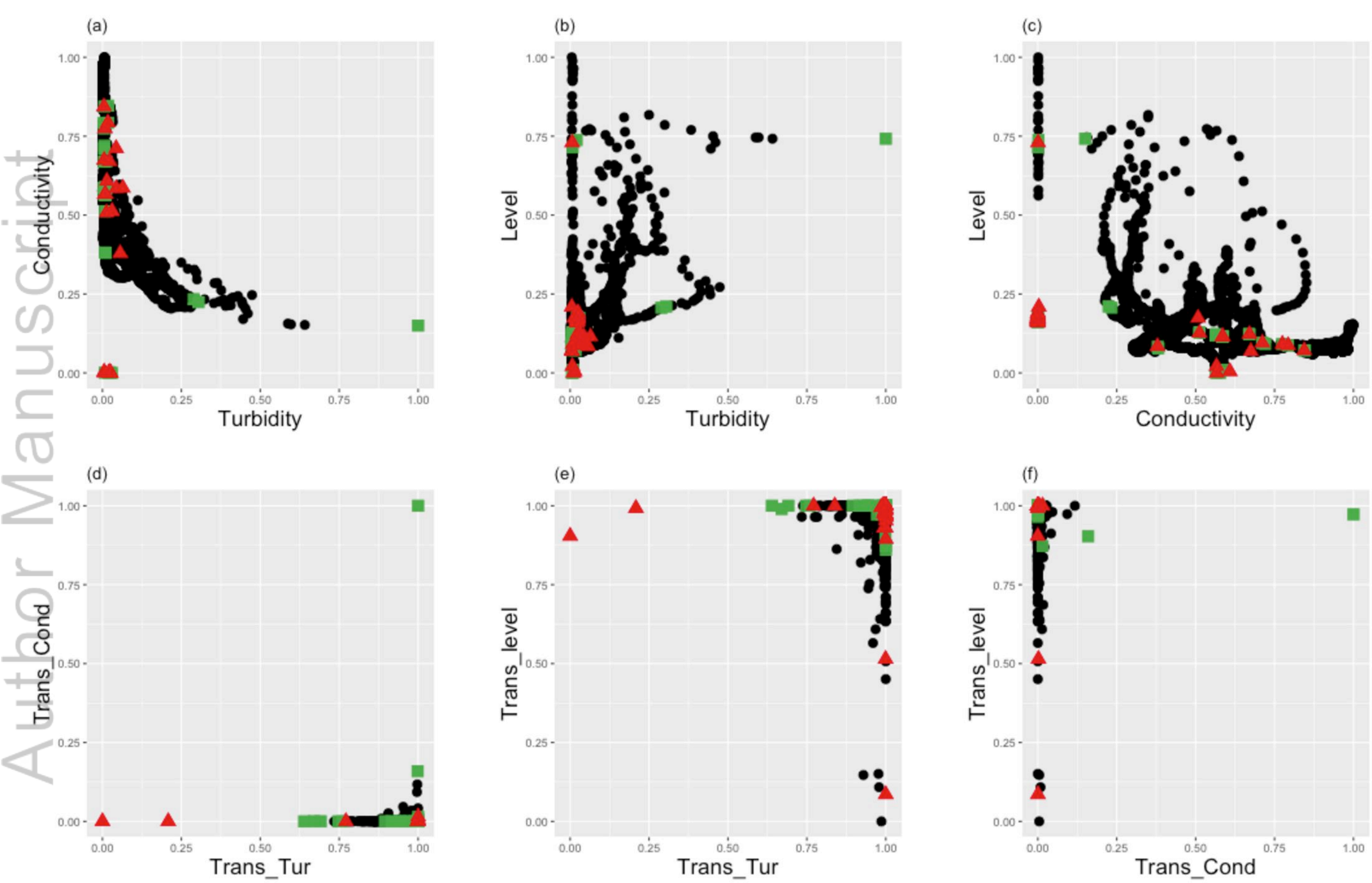
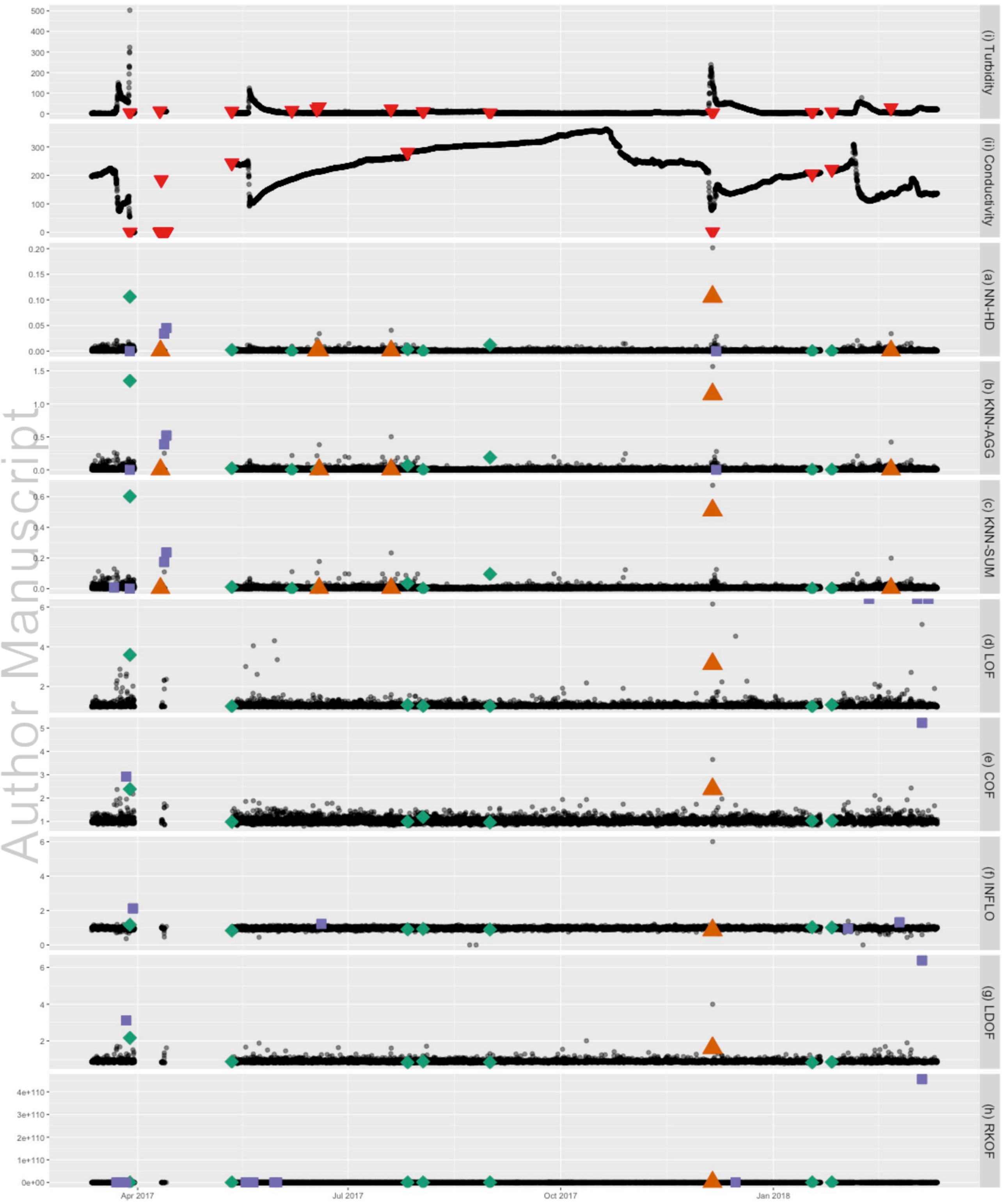
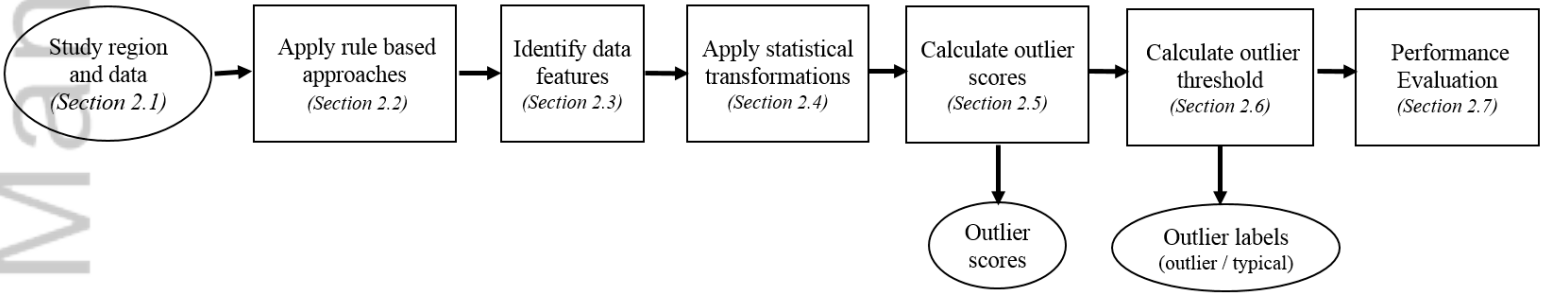
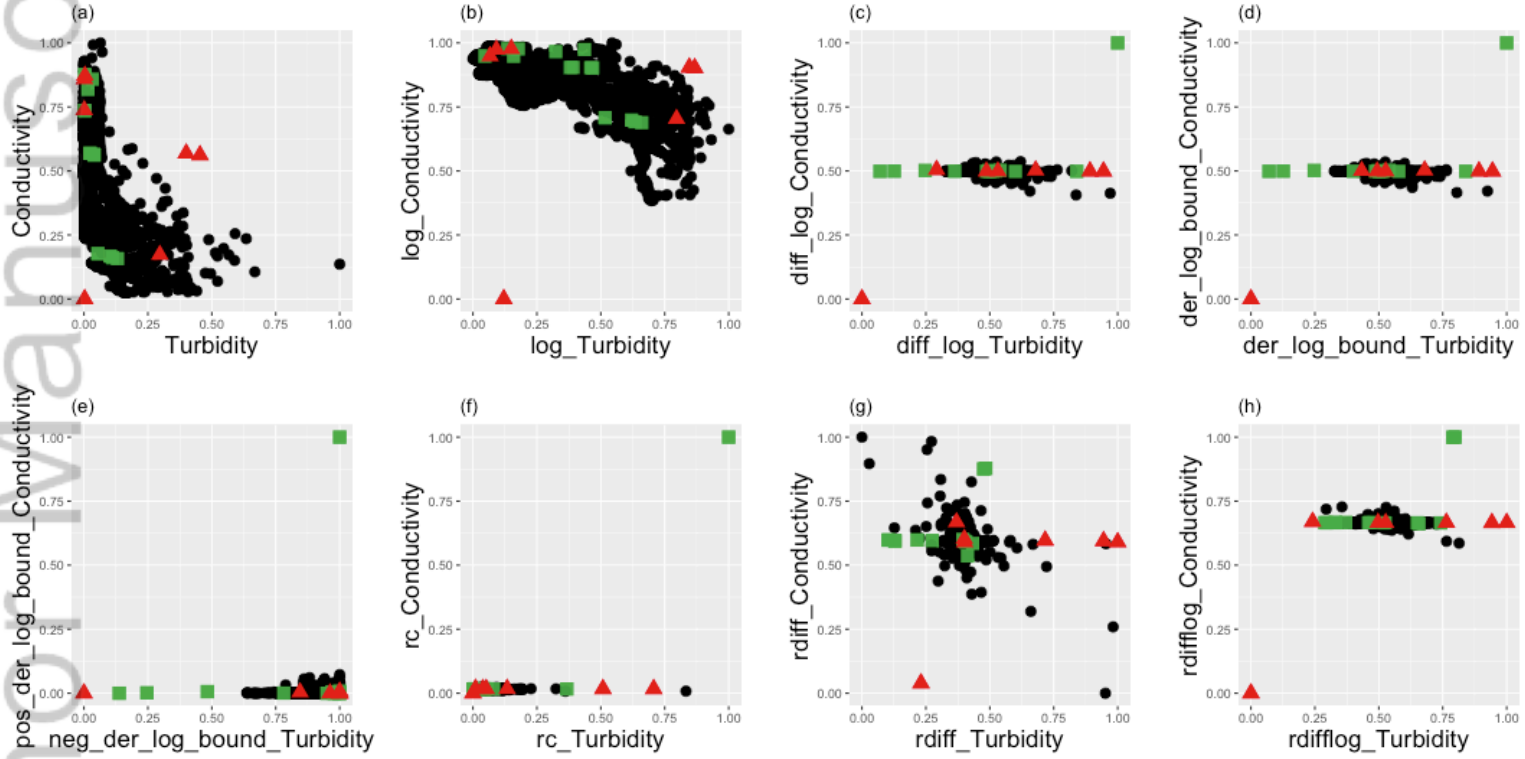


**Figure 8.**

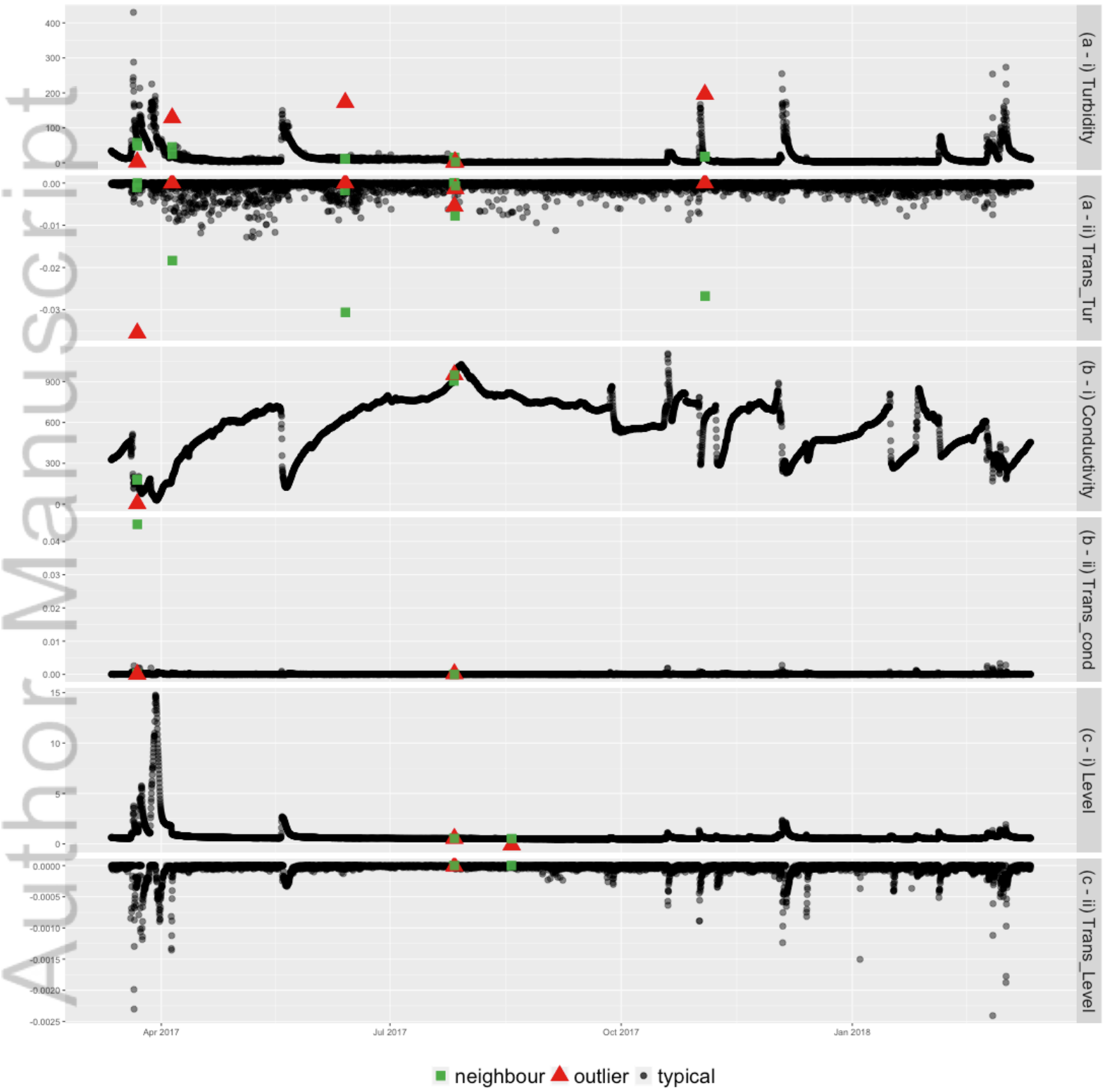




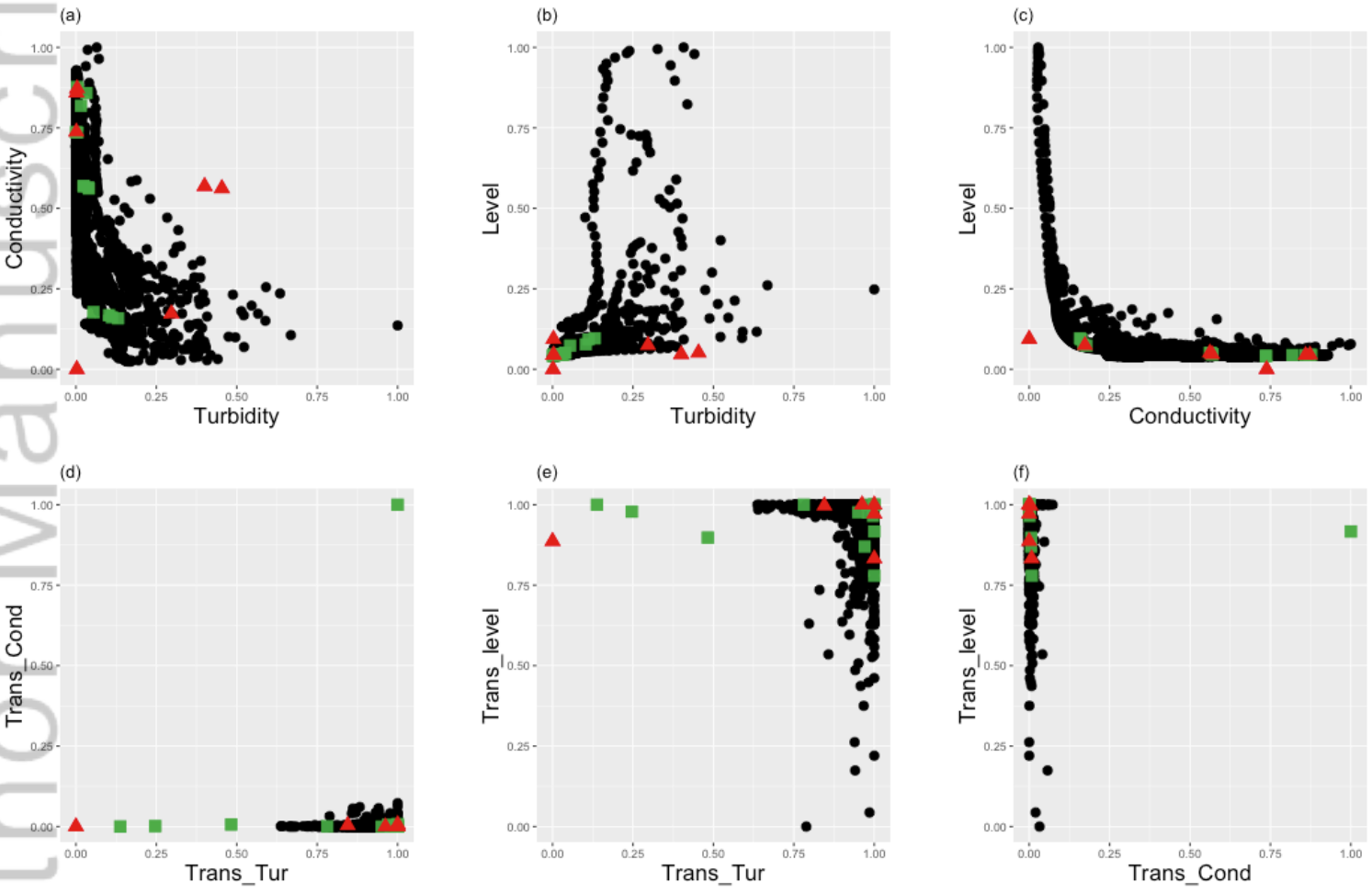

2019WR024906-f01-z-.png



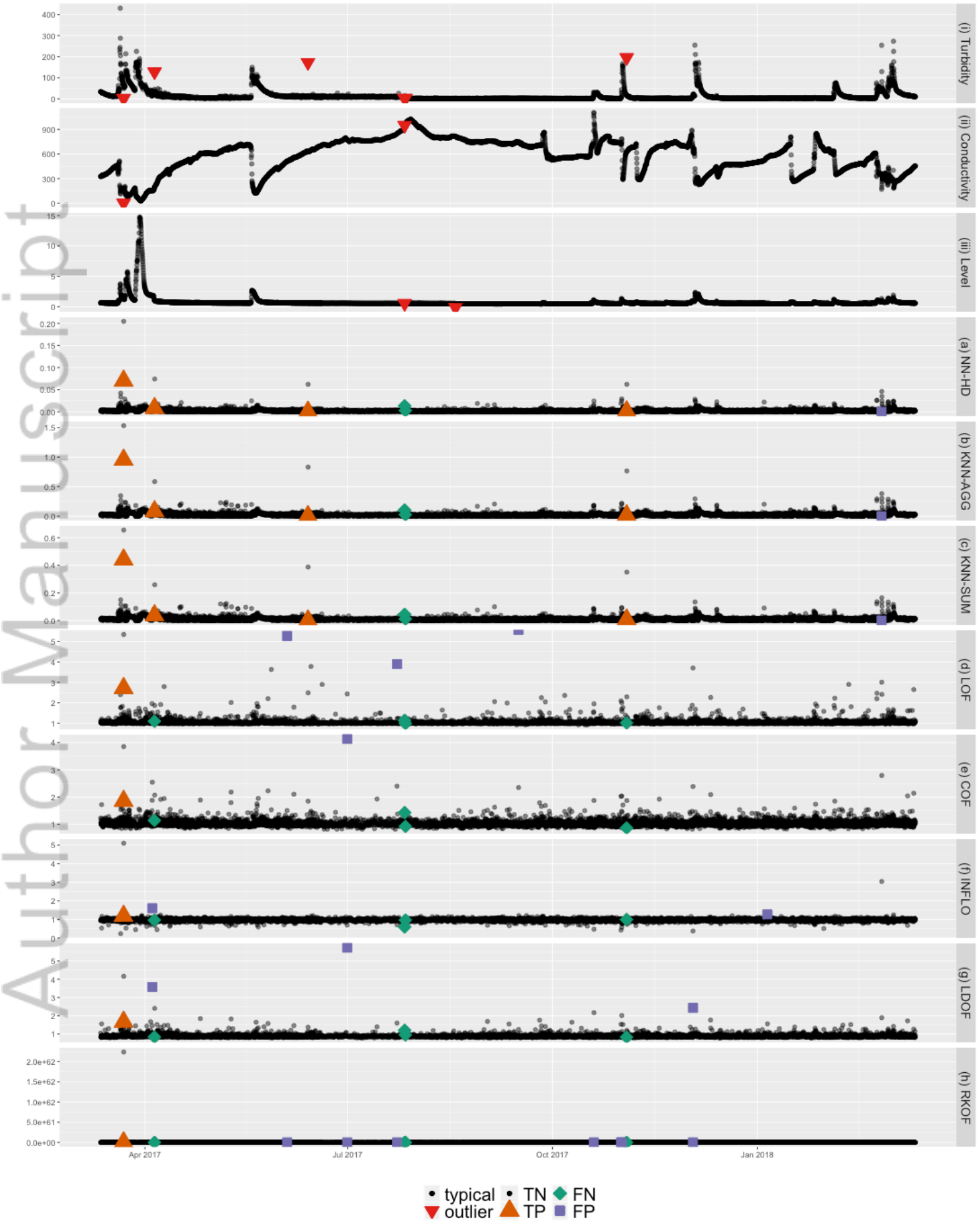

2019WR024906-f02-z-.png



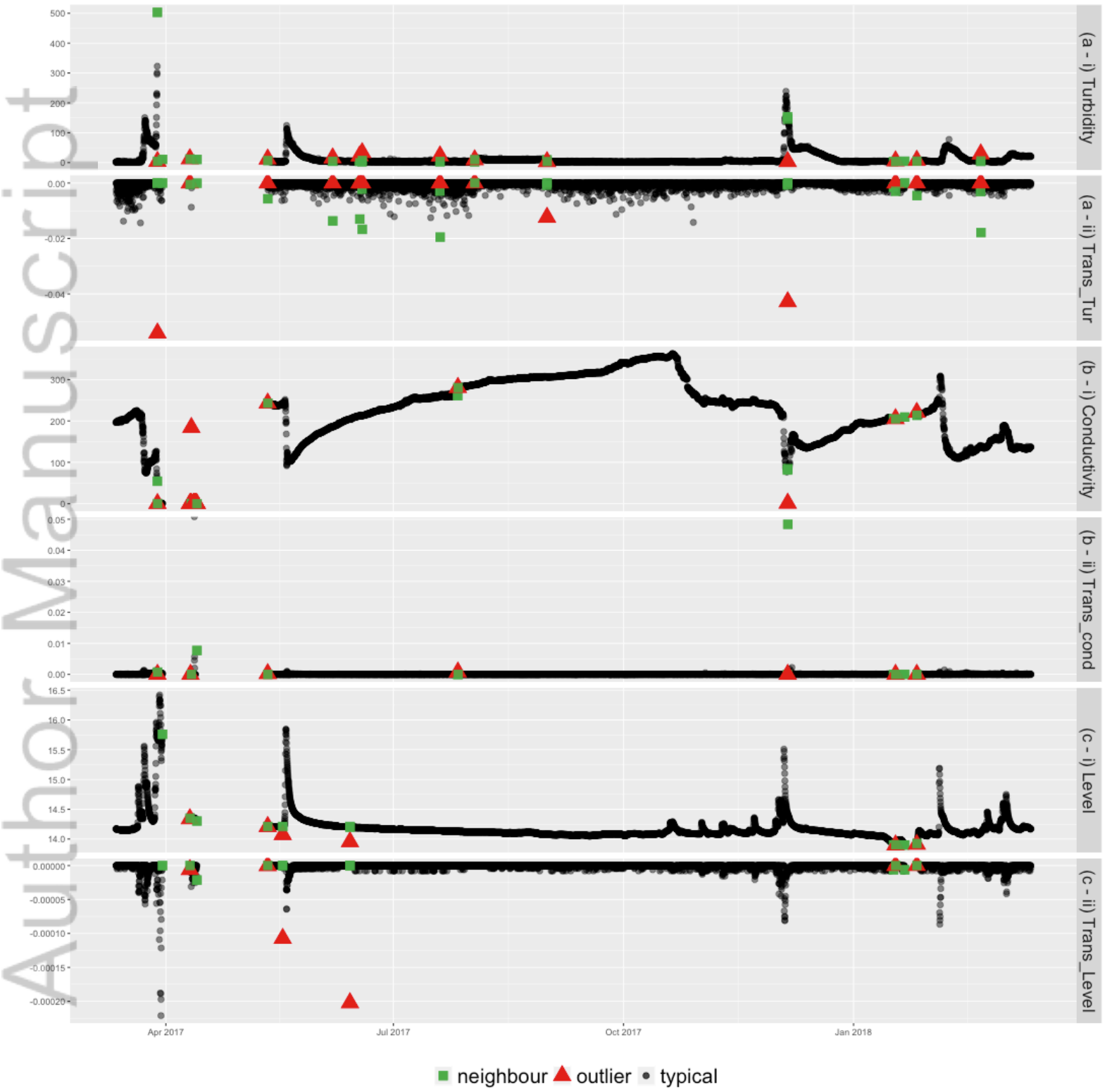

2019WR024906-f03-z-.png



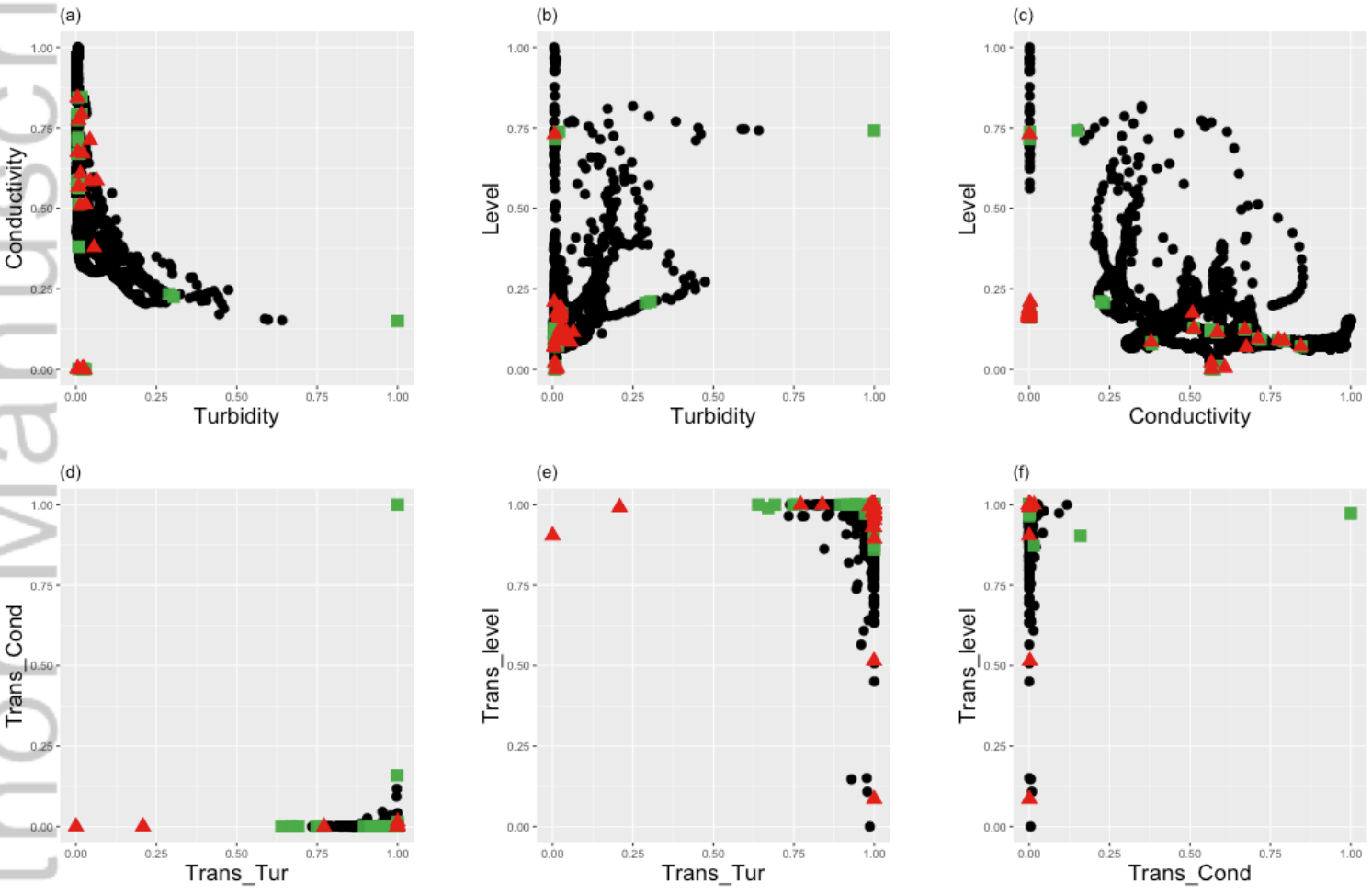

2019WR024906-f04-z-.png



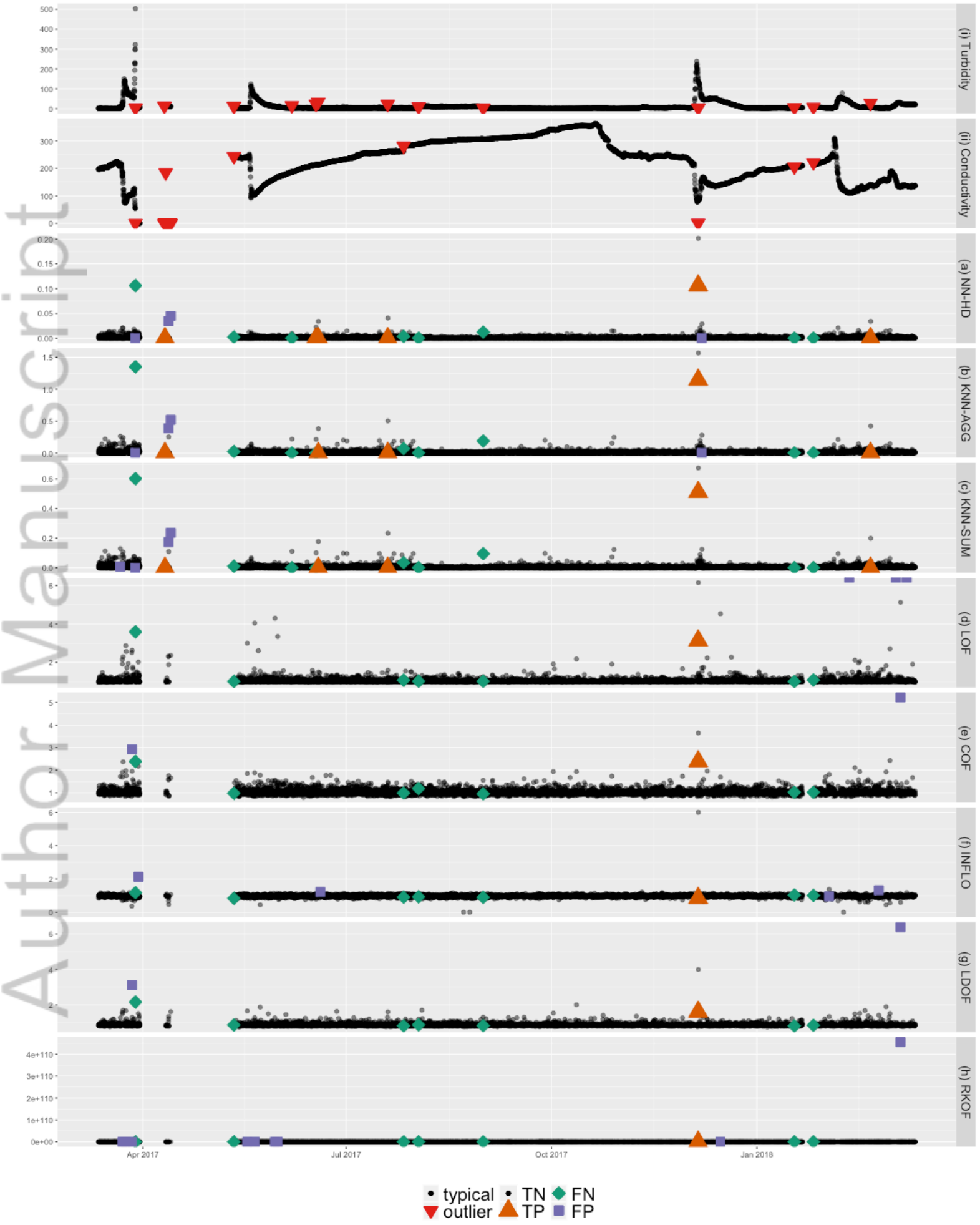

2019WR024906-f05-z-.png




2019WR024906-f06-z-.png




2019WR024906-f07-z-.png




2019WR024906-f08-z-.png