THESIS

MATHEMATICAL MODELING OF GROUNDWATER ANOMALY DETECTION

Submitted by

Jianli Gu

Department of Mathematics

In partial fulfillment of the requirements

For the Degree of Master of Science

Colorado State University

Fort Collins, Colorado

Fall 2016

Master's Committee:

    Advisor: Jiangguo (James) Liu

    Kenneth H Carlson
    Yongcheng Zhou

ABSTRACT

MATHEMATICAL MODELING OF GROUNDWATER ANOMALY DETECTION

Public concerns about groundwater quality have increased in recent years due to the massive exploitation of shale gas through hydraulic fracturing which raises the risk of groundwater contamination. Groundwater monitoring can fill the gap between the public fears and the industrial production. However, the studies of groundwater anomaly detection are still insufficient. The complicated sequential data patterns generated from subsurface water environment bring many challenges that need comprehensive modeling techniques in mathematics, statistics and machine learning for effective solutions. In this reseach, Multivariate State Estimation Technique (MSET) and One-class Support Vector Machine (1-SVM) methods are utilized and improved for real-time groundwater anomaly detection. The effectiveness of the two methods are validated based upon different data patterns coming from the historic data of Colorado Water Watch (CWW) program. Meanwhile, to ensure the real-time responsiveness of these methods, a groundwater event with contaminant transport was simulated by means of finite difference methods (FDMs). The numerical results indicate the change of contaminant concentration of chloride with groundwater flow over time. By coupling the transport simulation and groundwater monitoring, the reliability of these methods for detecting groundwater contamination event is tested. This research resolves issues encountered when conducting real-time groundwater monitoring, and the implementation of these methods based on Python can be easily transfered and extended to engineering practices.

ACKNOWLEDGEMENTS

It is with genuine gratitude that I would like to acknowledge to the incredible support, guidance, feedback and encouragement of my advisor Dr. James Liu in the past two years. His clear thoughts and meticulous work were a great source of inspiration during the research and thesis writing.

I would like to give my special acknowledgements to Dr. Kenneth H Carlson for the financial and academic support while I am pursuing the master's degree. His vision and insight helped me keep in right directions while working and studying.

I also wish to thank my committee member Dr. Yongcheng Zhou for bringing his view and ideas, as well as his patience and advice in mathematical fields, especially numerical analysis.

Many thanks as well to Bryan Elder for his administrative help without which I would not have possibly graduated as an international.

I wish to thank all my friends at CSU for the encouragement and support.

I am deeply thankful to my family. Thank you mom, dad, and younger sisters. And, of course, I am grateful for the love and encouragement of my wife Huishu, she has been always supportive of my study in the United States.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 Groundwater Anomaly

Public concerns about the groundwater quality have increased with the fast-growing of shale oil and gas extraction activities in the United States. Shale gas extraction activities have grown exponentially since 2008 [46]. The transformed energy market results from the development of hydraulic fracturing technique that involves drilling deep wellbores into the earth and injecting high-pressure fracking fluid (mixture of water, sand and chemicals) to create cracks in the shale-rock formations through which the oil and gas flow back to the ground surface. Figure 1.1 illustrates the hydraulic fracturing.
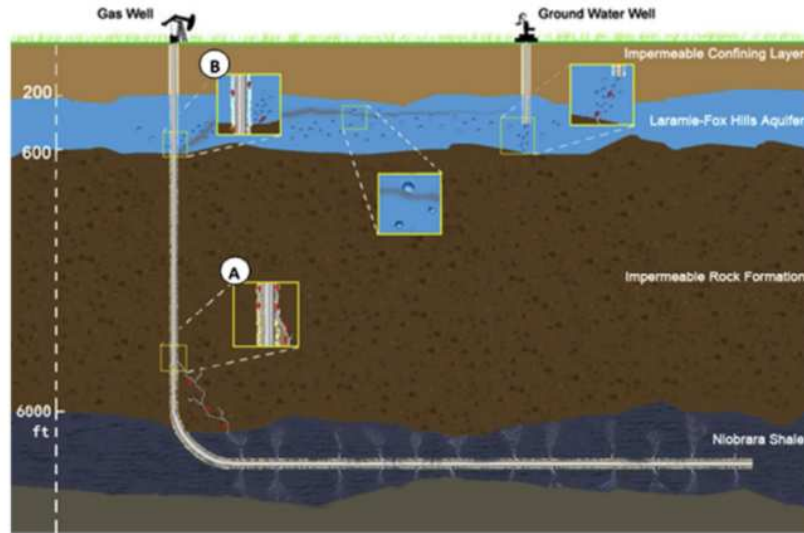
Figure 1.1: Hydraulic fracturing and simplified pathways of potential pollutants (A: stray gas; B: fracking fluid) into the groundwater aquifer from adjacent faulty oil-gas wells[26]
.

However, the hydraulic fracturing technique is controversial. The critics argue that drilling and fracking can cause uncertain environmental issues. As shown in Figure 1.1, the risks brought by the oil-gas well drilling include the natural gas escaping from producing wells through circumferential fractures and/or from improperly sealed casing strings

1

along the wellbore, and fracking fluid leakage along casing breaches into the groundwater aquifer [26]. The natural gas become explosive and flammable when the methane concentration in water is greater than its solubility, and the produced fluids are full of hazardous compounds. Both of them will bring severe contamination to the shallow groundwater in regional areas[46, 48]. Recent studies have shown elevated levels of chloride and potassium in wells around extensive shale gas operation in Pavillion, Wyoming[13, 47].

To bridge the gap between fears about public health impacts caused by oil and gas development and the assumption that industry environmental and health practices are reliable, Colorado Water Watch (CWW) designed by Colorado State University (CSU) has started to serve as a platform to support the states regulatory agency and provide real-time groundwater quality information to both industry and the public in the northern area of Wattenberg field. Wattenberg field locates north Denver Basin of Colorado and was historically one of the largest oil and gas production fields in the United States, owning more than 22,000 producing wells according to Colorado Oil and Gas Conservation Commission (COGCC), 2014. At first decades, petroleum companies started with oil production, but recently massive hydraulic fracturing was performed for natural gas routinely on thousands of wells.
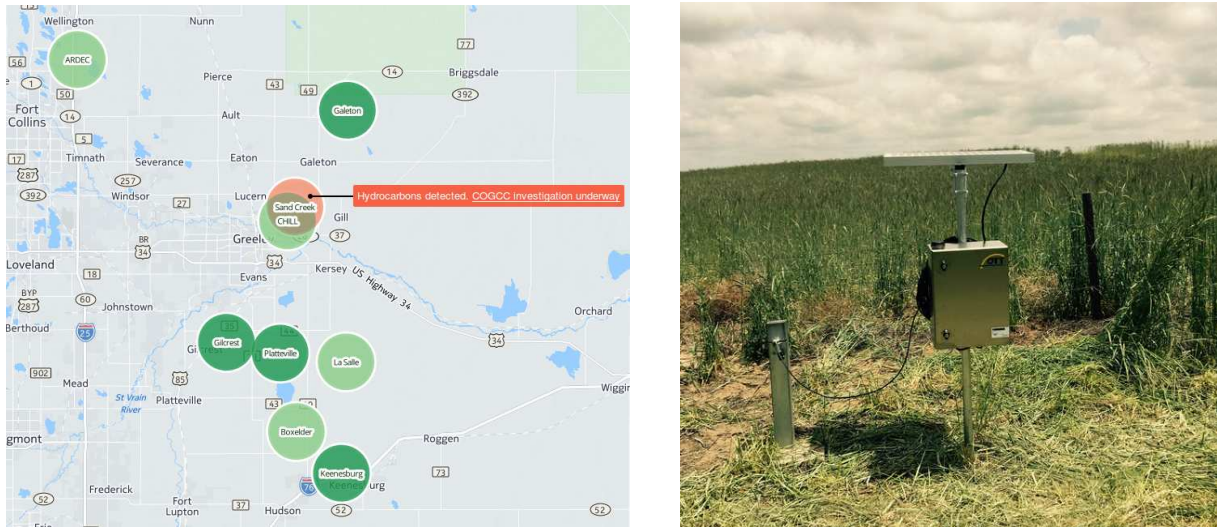
Figure 1.2: *Left*: the monitoring locations in Wattenberg field. *Right*: the sensor facility of one monitoring station [10]
.

CWW adopts contaminant-surrogate sensing technology to gather and provides ground-water information at intervals of one hour. The surrogate parameters include temperature, pH, conductivity, oxidation reduction potential (ORP), and dissolved oxygen (DO). These five parameters fall into two categories: fracturing fluid surrogates and natural gas surrogates. The surrogates are designed based on the correlation between conductivity and total dissolved solid (TDS), and the expected close relationship between ORP, DO, pH and dissolved methane gas in water under certain conditions. Normally, the surrogate parameters are kept in their reasonable ranges, but sometimes will get changed dramatically due to some external reasons. The sequence observations with anomalies are shown in Table 1.1.

Table 1.1: Observation Samples with Anomalies

Measurement Datetime	Temperature	pH	Condutivity	ORP	DO
2015-08-12 09:00:00	10.34	7.17	2.41	581	2.58
2015-08-12 10:00:00	10.34	7.17	2.4	581	2.6
2015-08-12 11:00:00	10.34	7.18	2.41	581	2.63
2015-08-12 12:00:00	16.51	7.95	0.001	551	8.26
2015-08-12 13:00:00	11.31	7.28	2.38	542	3.35
2015-08-12 14:00:00	10.61	7.14	2.4	572	3.2
2015-08-12 15:00:00	10.44	7.15	2.4	579	3.05
2015-08-12 16:00:00	10.4	7.15	2.4	581	2.9

Here, we firstly clarify two concepts following in this context.

Definition 1. *Groundwater event is particularly refered to the event which leads to the significant change of water quality, with a duration one or more time steps.*

Definition 2. *Groundwater anomaly refers to all phenomena of surrogate parameter's deviation from the normal signal tracks of sequential observations.*

All groundwater events are groundwater anomalies, but not all anomalies are groundwater events. For instance, the baseline change triggered by seasonal shifts and sensor calibration,

or the dramatic fluctuation of monitoring parameters resulted from the occurrences of precipitation and stream charging. In Table 1.1, the parameter values change dramatically and become abnormal at 12:00 pm. This anomaly results from sensor failure or some uncertain transmission error. As mentioned above, the hydraulic fracturing during oil and gas production operations may cause natural gas or fracking fluid leakages which bring disaster events to groundwater. No matter whatever the event is, CWW should get response to it and raise alarm in a real-time if the surrogate parameters are out of normal range, just in case the events result from stray gas or fracking fluid contamination which needs to take actions immediately. Therefore, anomaly detection plays an essential role in groundwater monitoring work in the oil and gas production fields. Mathematical methods for detecting anomalies in the non-stationary time series should be developed and adapted to the complex monitoring environment. Event discrimination from different anomalies is out of this research's scope.

1.2 Anomaly Detection

In general, anomaly refers to the abnormal observation or pattern which does not confirm with most of the behaviors in a system [39]. The anomalies can be translated to failure observations [17], system intrusions [35, 40], back frauds [34], structural defects [31] or environment pollutants [14] in specific application domains. Anomaly detection aims to find such anomalies by means of mathematical approaches. The analysts have great interest about the unusual and useful information behind the anomalous behaviors all the time [5], and give extensive attention to anomaly detection in recent years. Different models or methods have been employed based on diverse techniques, including statistical, clustering, support vector machine, neural network, as well as information and spectral theoretic[5, 39].

First, statistical approaches traditionally employ parametric or non-parametric models to describe the data distributions, where normal observations occur in high probability regions, but anomalies appear in low probability ones. These methods heavily rely on the assumption that the data is generated from a particular distribution, like the chi-square based approach

4

[40]. Normally, it's not a straightforward task to choose a good hypothesis test statistic, especially for complex distributions.

Second, typical machine learning approaches, like neural networks (NNs) [12, 22] and support vector machines (SVMs) [16, 44], have been reported for detection anomalies in many applications. These techniques perform either supervised or unsupervised anomaly detecting operations. The supervised methods usually consists of a training phase and a testing phase. Normal and abnormal instances are used to train a model in the training phase, based on which the state of newly coming observations are tested in the testing phase. In contrast, the unsupervised methods don't require the training dataset to be labeled or depend on any human intervention, and detect anomalies based on the similarity of internal structures or relationships between different observations in data. For example, one-class SVM proposed by Schölkop [43] only needs the one class (normal class) information for detecting anomaly and ignore the diversity of anomalies. Normally, the learning phases usually need sufficient data, and appears to be slow and expensive [35].

Third, data mining is another area which have proposed several methods for anomaly detection. Bayesian networks (BNs)[23] generally estimates the posterior probability of observing both normal and abnormal classes, then labels the test instance with largest posterior. Decision tree [33] requires to learn rules from the training dataset and then test whether the new instance is covered by such rules or not. Clustering methods [7, 52] do not require human beings to provide information about the normal or anomaly status of observations. Distance between data instances are important for measuring their similarity.

Besides, Multivariate State Estimation Technique (MSET) is an advanced pattern recognition technique developed and patented by Argonne National Laboratory (ANL) for fault monitoring applications in nuclear plant system [20, 45]. This method has been applied to detect anomaly for satellite components [37], video stream [51], enterprise server [19], et.al. All studies stated above provide possible directions for anomaly detection in groundwater.

1.3 MOTIVATION AND OBJECTIVES

CWW system collects groundwater information through surrogate technique at intervals of one hour, hence generating a multivariate time series which include temperature, pH, conductivity, ORP and DO. Figure 1.3 shows the historic surrogate parameter values.
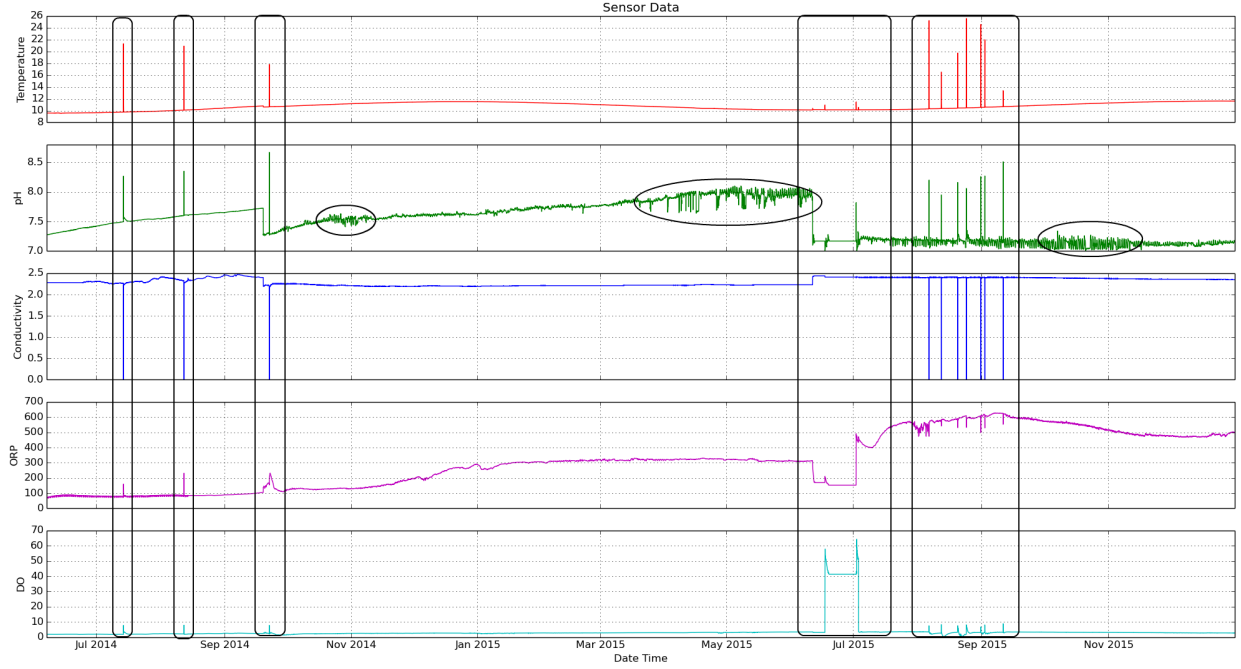
Figure 1.3: Historic surrogate parameters and anomalies patterns in black boxes

Due to the complexity of the groundwater environment, monitoring signals appear to be non-stationary which means the mean and/or variance are changing over time, and generate various anomaly patterns during this process, as shown in Figure 1.3. The characteristics during the monitoring process throw up many challenges to this research.

(i) The monitoring process is non-stationary which means the mean and/or variance of the signals are changing over time. Anomaly detection methods should be suitable for non-stationary process.

(ii) The anomalies patterns appear to be various, which implies the anomaly detection methods should be flexible enough to capture all anomaly patterns, instead of just only one or two situations.

6

(iii) The boundary between normal and anomaly is not precise locally. Some noisy data, as circled in figure, may appear to be anomaly in local. To avoid this, anomaly detection methods should have the capability to discriminate them.

(iv) The anomaly detection methods should also be adapted to the baseline changes over time. Baseline changes are not groundwater events.

(v) All observations have no "normal" or "anomaly" label, which implies the anomaly detection methods need to tell the anomaly observations from the majority normal ones based on their prior knowledge.

(vi) Anomaly detection methods must be implemented in real-time.

As a decision-making platform target at groundwater quality in the oil and gas field, CWW system continuously transmits the real-time monitoring data into CANARY which is an open-source software for water quality event detection developed by Environmental Protection Agency (EPA) associated with Sandia National Libraries (SNL). CANARY provides two algorithms named Linear Prediction Coefficient Filter (LPCF) and Multivariate Nearest Neighbor (MVNN) for prediction which is an essential step for anomaly detection. Based on the characteristics of surrogate parameters, CANARY shows some drawbacks. Firstly, LPCF, commonly known as Autoregression (AR), is an traditional statistical model whose performance highly relies upon the choice of data distribution [39], and it employs the statistical inference test to detect the observations whether they do or do not belong to this model. However, when the data generation process is more complex, like non-linearity or non-Gaussianity, than the model used, then the model's performance would be poor during the anomaly detection process. Secondly, MVNN is distribution free and measure the similarity between observations based on Euclidean distance. However, it may not be appropriate since the distance becomes complicated in high dimensions. Third, various anomaly patterns couldn't be captured by only one method due to the complexity of anomaly patterns. Multiple anomaly detection methods are expected to incorporate into CWW.

Due to the limitations stated above, the algorithms in CANARY couldn't solve the challenging problems effectively, which therefore motivates us to develop new anomaly detection methods for this particular groundwater monitoring application. The potential methods we studied including two methods: Multivariate State Estimation Technique (MSET) and One-class Support Vector Machine (1-SVM) which have not been applied to real-time groundwater anomaly detection. MSET is a general technique that can estimate both stationary and non-stationary states. 1-SVM detect anomaly based on observation density in feature space, even the distribution is unknown. In the particular context of groundwater monitoring, they are introduced for exploration by this research. The historic dataset with anomalies will be used for method testing. Meanwhile, as we can see in Figure 1.3, there actually no groundwater contamination events happened before. In order to validate the sensitivity of newly-developed methods to groundwater contamination, we will simulate an groundwater contamination event based on flow and transport equations and test the effectiveness of these methods for capturing the environmental anomalies.

1.4 Contributions of This Thesis

This thesis contributes to the area of groundwater monitoring. Specifically, it introduces novel thinking and methods to the field of real-time groundwater anomaly detection. Chapter 1 begins with the description about the current conditions of CWW, states the shortcomings of LPCF and MVNN algorithms used for groundwater anomaly detection, and points out the corresponding challenges encountered in this real world application. Chapter 2 explores and explains two potential methods, i.e. MSET and 1-SVM, with detailed procedures, as well as the related techniques such as Sequential Probability Ratio Test (SPRT), cross validation and grid search. Chapter 3 gives a MSET-based real-time anomaly detection framework and a 1-SVM-based real-time anomaly detection framework for practical implementation. The corresponding results are shown and discussed. Chapter 4 introduces flow and transport equations, finite element methods for simulating groundwater contamination event that

is used to test the robustness of MSET and 1-SVM algorithms for groundwater anomaly detection. Finally, we give the concluding remarks about the research.

The main contributions of this thesis can be identified as follows.

(i) We introduce two methods for real-time groundwater anomaly detection, including MSET and 1-SVM, solving the challenging problems with expectations.

(ii) According to the practical engineering application, two real-time groundwater anomaly detection frameworks are constructed with implementation details.

(iii) The groundwater contamination event is simulated based on transport and flow equations, and finite element methods. This experiment is used to validate the effectiveness of the two anomaly detection models.

As a final remark, the considerations with respect to MSET-based and 1-SVM-based real-time anomaly detection are likely to be applicable to other domains. Thus, even though the thoughts are motivated from real-time groundwater anomaly detection, its impact is possibly to transcend beyond that specific environment.

CHAPTER 2

METHODS FOR REAL-TIME ANOMALY DETECTION

2.1 Spaces, Mappings and Pseudo-inverse

In order to reveal the hidden information behind the dataset, many pattern recognition algorithms will employ mappings (or transformations) about the data, such as novelty filtering, dimensional reduction and so on. The mapping generalizes the idea of a function that maps the data from one space into another. The nature of the mapping depends on the mathematical representation of the data [51]. In this section, it gives some formal look definitions about the data, the space, the mapping and the operation.

Generally, a time series data can be expressed as a set of time-ordered infinite measurements, noted as χ

$$\chi = \{ \langle t_1, \mathbf{x}_1 \rangle, \langle t_2, \mathbf{x}_2 \rangle, \ ..., \ \langle t_i, \mathbf{x}_i \rangle \ ... \ \}, \ i \in \mathbb{N}^+ \tag{2.1}$$

and we call $\langle t_i, \mathbf{x}_i \rangle$ as a measurement, where t_i is a timestamp, and $\mathbf{x}_i$ is an observation, mathematically a vector.

$$\mathbf{x}_i = \begin{bmatrix} x_{i1} \\ x_{i2} \\ ... \\ x_{im} \end{bmatrix} \tag{2.2}$$

Definition 3. *Input space is defined as an inifite space X which contains all possible observations $\mathbf{x}_i$ from a certain process, such that*

$$X = \{\mathbf{x}_1, \ \mathbf{x}_2, \ ..., \ \mathbf{x}_i, \ ...\}, \ i \in \mathbb{N}^+ \tag{2.3}$$

where $\mathbf{x_i} \in \mathbb{R}^m$.

10

Definition 4. *Let Φ be the mapping from X to H, i.e. $\Phi : X \rightarrow H$, H is so-called the feature space which contains the points mapped from the data elements of the input space, such that*

$$H = \{\Phi(\mathbf{x}_1),\ \Phi(\mathbf{x}_2),\ ...,\ \Phi(\mathbf{x}_n)\} \tag{2.4}$$

where $\mathbf{x}_i \in X, i = 1, 2, ..., n$.

For any two points $\Phi(\mathbf{x})$ and $\Phi(\mathbf{x}')$ in H,

(i) $\langle \Phi(\mathbf{x}), \Phi(\mathbf{x}') \rangle$ is defined as the dot product.

(ii) The norm $\|\Phi(\mathbf{x})\|$ is defined as $\sqrt{\langle \Phi(\mathbf{x}), \Phi(\mathbf{x}) \rangle}$.

(iii) The distance between $\Phi(\mathbf{x})$ and $\Phi(\mathbf{x}')$ is defined as

$d_\Phi(\mathbf{x}, \mathbf{x}') = \sqrt{\langle \Phi(\mathbf{x}) - \Phi(\mathbf{x}'), \Phi(\mathbf{x}) - \Phi(\mathbf{x}') \rangle}$

For example, $\Phi(\mathbf{x}_1, \mathbf{x}_2) = (\mathbf{x}_1, \mathbf{x}_2, \frac{\mathbf{x}_1}{\mathbf{x}_2})$ is a mapping from $\mathbb{R}^2$ to $\mathbb{R}^3$. In practice, however, it's impossible to compute the feature space H and find the mapping Φ explicitly, especially when the dimension becomes large [17]. The practical approach is to deal with the mapping Φ implicitly via a kernel function.

Definition 5. *A kernel function k has the form as below,*

$$k(\mathbf{x}, \mathbf{x}') = \langle \Phi(\mathbf{x}), \Phi(\mathbf{x}') \rangle \tag{2.5}$$

where $\mathbf{x}, \mathbf{x}' \in X$, with Φ being the mapping from X to H.

Typically the feature space is a real vector space with some high dimension d, $\mathbb{R}^d$, or more generally a Hilbert space, in which the kernel function has the following properties[15, 38]:

(i) Symmetric, $k(\mathbf{x}, \mathbf{x}') = \langle \Phi(\mathbf{x}), \Phi(\mathbf{x}') \rangle = \langle \Phi(\mathbf{x}'), \Phi(\mathbf{x}) \rangle = k(\mathbf{x}', \mathbf{x})$.

(ii) Cauchy-Schwarz Inequality, $k(\mathbf{x}, \mathbf{x}')^2 = \langle \Phi(\mathbf{x}), \Phi(\mathbf{x}') \rangle^2 \leq \|\Phi(\mathbf{x})\|^2 \|\Phi(\mathbf{x}')\|^2 = $
$\langle \Phi(\mathbf{x}), \Phi(\mathbf{x}) \rangle \langle \Phi(\mathbf{x}'), \Phi(\mathbf{x}') \rangle = k(\mathbf{x}, \mathbf{x})k(\mathbf{x}', \mathbf{x}')$.

(iii) Positive semi-definite, the symmetry of kernel function guarantees that $k(\mathbf{x}, \mathbf{x}') \geq 0$.

11

There are many kernel functions in research, like the linear kernel, polynomial kernel, Gaussian (or radius basis function, i.e. rbf) kernel, et.al. The Gaussian kernel is most common for choice,

$$K_\gamma(\mathbf{x}, \mathbf{y}) = \langle \Phi(\mathbf{x}), \Phi(\mathbf{x}') \rangle = exp\left(-\frac{\|\mathbf{x} - \mathbf{y}\|^2}{\gamma}\right) \tag{2.6}$$

where γ is the bandwidth. After mapping the data points from the input space X into the feature space H via the kernel, then people can formalize the problem of anomaly detection easily, for example, by detecting some point is distant from the most other points in the feature space, or is living in the non-sparse region as a normal one.

Definition 6. *In mathematics, a pseudo-inverse (or Moore-Penrose pseudo-inverse) A^+ of a matrix A is the generalization of the inverse matrix [3], and can be expressed from the singular value decomposition (SVD) of $A = U\Sigma V^T$,*

$$A^+ = V\Sigma^{-1}U^T \tag{2.7}$$

where U, V are orthogonal matrices, and Σ is a diagonal matrix.

The real world data is not always consistent and might contains many repetitions. For instance, the observations of groundwater quality might keep constant during a period time. In this situation, the singular matrix generated and has no inverse matrix. To deal with this problem, pseudo-inverse takes an effect, since A^+ exists for any matrix and is unique. For instance, to solve the least square problem

$$\min_x \|Ax - y\|^2 \tag{2.8}$$

The pseudo-inverse A^+ can be applied if A^{-1} does not exist, having the solution $x^* = A^+y$. In practice, many softwares, like `Matlab`, `R` and `Python` package `NumPy`, provide the calculation of pseudo-inverse. Take `NumPy` as an example,

```
import numpy as np

A = np.random.randn(11, 8)

B = np.linalg.pinv(A)
```

2.2   Sequential Probability Ratio Test

The Sequential Probability Ratio Test (SPRT) was introduced by Wald [49] by combining the classical hypothesis test process and Neyman-Pearson theorem [30]. It tests the alternative hypothesis against the null hypothesis through examining the random variables sequentially, and makes a decision about the acceptance between these verified hypothesis, or if extra-samples are required before giving an answer that the current observation is behaving normally or abnormally [9, 21]. The SPRT is an quantitative approach which constructs a conventional Logarithmic Likelihood Ratio (LLR) which permits the decision to be made. Notice that the probability density function, i.e. pdf, of the investigated variable should be known, and the type I error α and type II error β should be pre-assigned as well. Type I error refers to rejecting the null hypothesis while it's true, and type II error is to accept the null hypothesis while it's wrong, as shown in Figure 2.1.
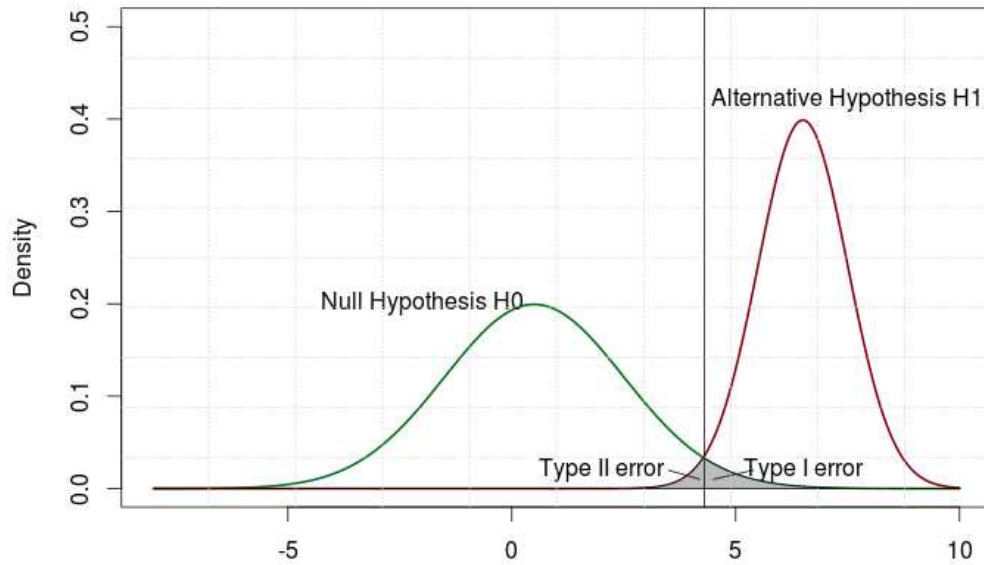
Figure 2.1: The null hypothesis, alternative hypothesis, type I error, and type II error

The SPRT works as follows. Given the successive random variable generated from a discrete process $Y \in \mathbb{R}^n$, so

$$Y_n = \{y_1, \ y_2, \ ..., \ y_n\} \tag{2.9}$$

where y_i presents a sample from the process Y at time t_i, and they are independent and identically distributed (i.i.d). Without loss generality, assume the variable follows the normal distribution, informally "the bell curve".

$$f(y \mid \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left(-\frac{(y-\mu)^2}{2\sigma^2}\right) \tag{2.10}$$

Therefore, the mean μ and/or the variance σ would be the parameters for statistical testing here. The binary hypothesis of SPRT contains the null hypothesis and one or more alternative hypotheses. The null hypotheses of SPRT is $H_0 : Y_n \sim N(0, \sigma^2)$, which the healthy states have zeros mean and σ standard deviation. The alternative hypothesis have four instances, as shown in Figure 2.2.
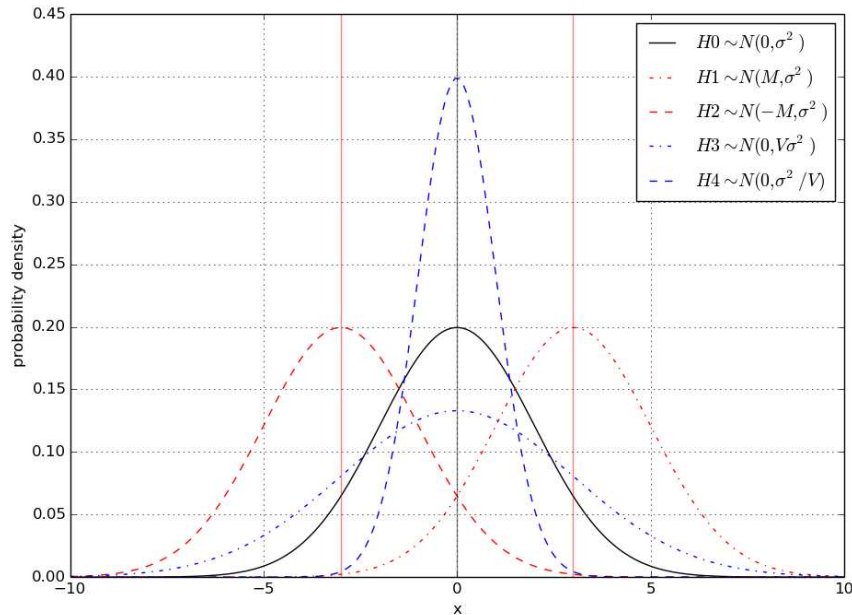
Figure 2.2: The null hypothesis and the alternative hypotheses; M and V are pre-determined thresholds by the user.

Literally, the four alternatives are stated as:

- $H_1 : Y_n \sim N(M, \sigma^2)$, the mean of states shifts to M with unchanged standard deviation.

- $H_2 : Y_n \sim N(-M, \sigma^2)$, the mean of states shifts to $-M$ with unchanged standard deviation.

- $H_3 : Y_n \sim N(0, V\sigma^2)$, the standard deviation changes to $V\sigma$ with non-shifted mean.

- $H_4 : Y_n \sim N(0, \sigma^2/V)$, the standard deviation changes to σ/V with non-shifted mean.

Based upon the null hypothesis and one alternative hypothesis, the SPRT index, i.e. LLR, is calculated for checking the state of observations. Let's take H_0 and H_1 for example,

$$I_1 = \log \frac{P(y_1, y_2, ..., y_n \mid H_1)}{P(y_1, y_2, ..., y_n \mid H_0)} = \log \frac{\prod_{i=1}^{n} f_1(y_i)}{\prod_{i=1}^{n} f_0(y_i)} = \frac{M}{\sigma^2} \sum_{i=1}^{n} (y_i - \frac{M}{2})^2 \qquad (2.11)$$

Similarly, for the other three alternative hypotheses, we can obtain the following equations for SPRT index [21]:

$$I_2 = \frac{M}{\sigma^2} \sum_{i=1}^{n} (-y_i - \frac{M}{2})^2 \qquad (2.12)$$

$$I_3 = \frac{1}{2\sigma^2}(1 - \frac{1}{V}) \sum_{i=1}^{n} y_i^2 - \frac{n}{2} \ln(V) \qquad (2.13)$$
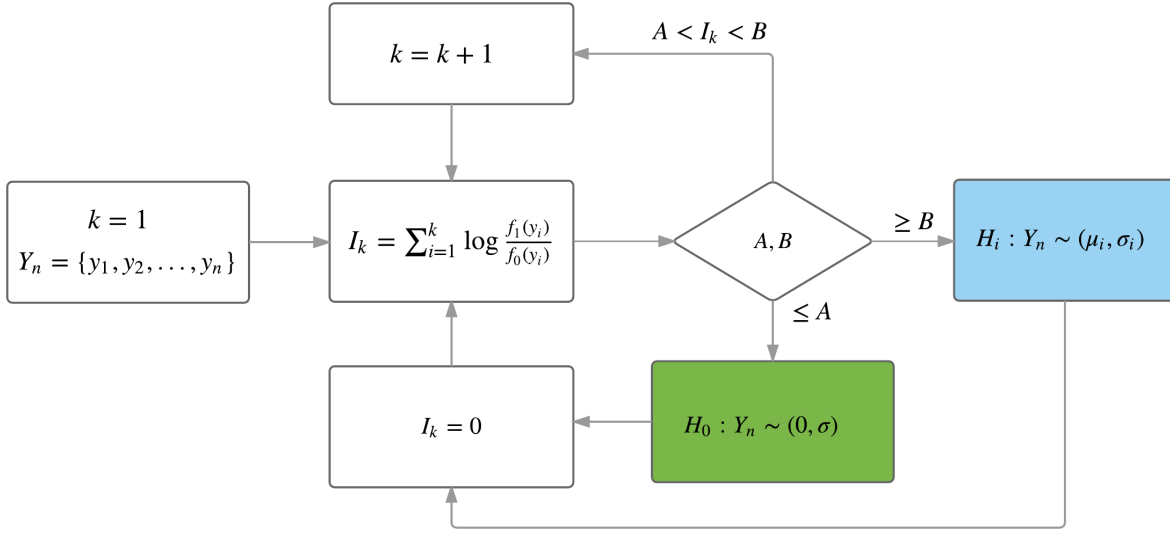
$$I_4 = \frac{1}{2\sigma^2}(1 - V) \sum_{i=1}^{n} y_i^2 + \frac{n}{2} \ln(V) \qquad (2.14)$$

Then according to the user-defined type I error α and type II error β, two stopping boundaries are formulated, and the approximations by Wald [49] are:

$$A \cong \log \frac{\beta}{1 - \alpha} \qquad (2.15)$$

$$B \cong \log \frac{1 - \beta}{\alpha} \qquad (2.16)$$

The hypothesis test operates as below while making a decision, as shown in Figure 2.3.

Figure 2.3: SPRT Procedure

- If $I_n <= A$, then accept H_0. mark the observing state as normal.

- If $I_n >= B$, then accept H_1, mark the observing state as anomaly and raise alarm.

- If $A < I_n < B$, additional sampling is required, set $n = n + 1$.

The SPRT framework provides great advantages for continuous surveillance work. On the one hand, it attains high sensitivity to the subtle changes of the monitoring signal, no matter whether the signal follows normal distribution or not. On the other hand, it allows the user to define the probability of missed alarm and false alarm, and to interpret the result without depending on the expert knowledge. Apparently, the SPRT method described above is a promising algorithm for quick anomaly determination.

2.3 MULTIVARIATE STATE ESTIMATION TECHNIQUE (MSET)

In short, Multivariate State Estimate Technique (MSET) comprises two essential modules named the estimation module and the anomaly detection module [8]. It firstly uses the non-linear and non-parametric technique to train a group of weighting values over the

historic observations, and then applies the SPRT for the residual mean and variance test. Mathematically, the MSET method operates as follows.

In MSET's estimation module, the time series data of sensor readings are generally organized in a matrix form denoted by $X \in \mathbb{R}^{m \times n}$, where the number of rows m equals to the number of sensor signals, and the number of columns n is the number of observations over time. Each column vector of the matrix, i.e. an observation $\mathbf{x}_{obs}$, lists the values of all sensor parameters $x_1, x_2, ..., x_M$ and represents a particular state. For a series of time $t_1, t_2, ..., t_j, ...$, the matrix X takes the shape.

$$
X = \begin{bmatrix}
x_{11} & x_{21} & ... & x_{j1} & ... \\
x_{12} & x_{22} & ... & x_{j2} & ... \\
... & ... & ... & ... & ... \\
x_{1m} & x_{2m} & ... & x_{jm} & ...
\end{bmatrix}
\tag{2.17}
$$

where x_{ji} is the value of the ith sensor signal at time j. The observation at time j can be represented as

$$
X(t_j) = \begin{bmatrix}
x_{j1} \\
x_{j2} \\
... \\
x_{jm}
\end{bmatrix}
\tag{2.18}
$$

So there are numerous of observations in X whose states are labeled as normal or anomaly, only the states with normal labels can be selected as members of the training dataset.

For the training dataset $T \in \mathbb{R}^{m \times n}$, where $T \subset X$. since it just contains the observations with healthy state.

$$
T = \begin{bmatrix}
x_{11} & x_{21} & ... & x_{n1} \\
x_{12} & x_{22} & ... & x_{n2} \\
... & ... & ... & ... \\
x_{1m} & x_{2m} & ... & x_{nm}
\end{bmatrix}
\tag{2.19}
$$

The selection of observations used for training is decided by the researchers. While doing the training work, the training dataset T would be split into two datasets: the memory dataset D and the remaining dataset L, i.e. $T = D \cup L$, where $D \in \mathbb{R}^{m \times nd}, L \in \mathbb{R}^{m \times nl}$ and $nd + nl = n$.

The memory dataset D can be regarded as the dataset where to learn experience. The selection of observations from T to create D is essential. The rules obeyed here include two parts. On the one hand, the observations with extreme features should be included, that is, selecting the states with maximum or minimum signal value. This rule ensures that the memory dataset D covers the whole healthy range of each sensor signal. On the other hand, the additional selection based on the norm of the observations. After calculation and sorting the norm of observations, the left states are selected with an equally spaced interval. The remaining dataset L is composed by the non-selection observations of training dataset T. This dataset is used for calculating the healthy estimates and healthy residuals for validating the state of the new observation.

The estimate of $\mathbf{x}_{est} \in \mathbb{R}^m$ is based the memory dataset D and current observation $\mathbf{x}_{obs}$. The equations take the form below.

$$\mathbf{x}_{est} = D \cdot W_{\mathbf{x}} \tag{2.20}$$

$$W_{\mathbf{x}} = \left(D^T \otimes D\right)^{-1} \cdot \left(D^T \otimes \mathbf{x}_{obs}\right) \tag{2.21}$$

where the weight vector $W_{\mathbf{x}} \in \mathbb{R}^{nd}$ can be regarded as the measurement of similarity between the estimate and the memory dataset. Similarly, the estimate of remaining dataset $L_{est} \in \mathbb{R}^{m \times nl}$ can be calculated based on the following equations.

$$L_{est} = D \cdot W_L \tag{2.22}$$

$$W_L = \left(D^T \otimes D\right)^{-1} \cdot \left(D^T \otimes L\right) \tag{2.23}$$

where the weight matrix $W_L \in \mathbb{R}^{nd \times nl}$, and each column vector is a weight vector for the corresponding observation in L. Pseudo-inverse is applied for calculating $\left(D^T \otimes D\right)^{-1}$ in order to solve the general problem similar with least square, see section 2.1 for more details.

In the equations 2.21 and 2.23, they have the form of least square minimization, i.e. $\left(D^T \cdot D\right)^{-1} \cdot \left(D^T \cdot L\right)$, but with a non-linear operator $\otimes$ termed a kernel function, instead of the linear operator - dot product. The non-linear operator $\otimes$ chosen here is used for fitting the input data, and therefore expected to preserve the properties of the linear operator [19, 32]. First, $\left(D^T \otimes D\right)^{-1}$ should be non-singular. Second, the difference between $\mathbf{x}_{est}$ and $\mathbf{x}_{obs}$ should be minimized. Third, if $\mathbf{x}_{obs}$ is equal to the column vector in D, then the difference must be zero. Forth, if $\mathbf{x}_{obs}$ falls out the range of signal in D, then the estimate should be optimal. Many non-linear operators exist [4, 37, 51], but the choice of the non-linear operator does not have major impact on the performance of MSET [24]. The common choice is Gaussian kernel.

$$k_\gamma(\mathbf{x}, \mathbf{x}') = \langle \Phi(\mathbf{x}), \ \Phi(\mathbf{x}') \rangle = exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{\gamma} \right) \tag{2.24}$$

where the kernel takes an effect on the observations $\mathbf{x} \in \mathbb{R}^m$ and $\mathbf{x}' \in \mathbb{R}^m$ for comparison, and γ is the bandwidth. A narrow bandwidth leads to an over-fitting solution for depending only on a few extreme states, but a large bandwidth will bring over-smooth solution which is also not good [24].

In MSET's SPRT module, the procedure uses the residuals of estimates to validate sensor signals. The residuals include the actual residual $R_\mathbf{x}$ and the healthy residual R_L, which will be used as the input of SPRT for anomaly detection.

$$r_\mathbf{x} = \|\mathbf{x}_{est}\| - \|\mathbf{x}_{obs}\|, \ R_L = \left\{ \ r_{\mathbf{x}_1}, \ r_{\mathbf{x}_2}, \ ..., \ r_{\mathbf{x}_{nl}} \ \right\} \tag{2.25}$$

$$R_{LX} = \left\{ \ r_\mathbf{x}, \ r_{\mathbf{x}_1}, \ r_{\mathbf{x}_2}, \ ..., \ r_{\mathbf{x}_{nl}} \ \right\} \tag{2.26}$$

where $r_{\mathbf{x}_i}$ is the healthy residual of the ith remaining observation, and $i \leq nl$. Then, the null hypothesis and alternative hypotheses can be constructed with the assumption that the residuals follow the normal distribution.

- $H_0 : R_{LX} \sim N(0, \sigma^2)$, where the healthy residual R_L has mean $= 0$ and standard deviation $= \sigma$, which is the expected distribution.

- $H_1 : R_{LX} \sim N(M, \sigma^2)$, the distribution pattern of residuals will be changed and the mean shifts to M after adding in the actual residual.

- $H_2 : R_{LX} \sim N(-M, \sigma^2)$, the distribution pattern will be changed and the mean shifts to $-M$ after adding in the actual residual.

- $H_3 : R_{LX} \sim N(0, V\sigma^2)$, the standard deviation of the residuals will be changed to $V\sigma^2$ after adding in the actual residual.

- $H_4 : R_{LX} \sim N(0, \sigma^2/V)$, the standard deviation of the residuals will be changed to σ^2/V after adding in the actual residual.

Please refer to the previous section for the detailed SPRT procedure based upon these hypotheses, the final SPRT index gives indication about the state of the new observation, i.e. normal or anomaly. This procedure is triggered whenever a new observation is coming, which we call real-time anomaly detection. Since the training dataset is constructed according to current observation, the system works dynamically to adapt to the changing sensor signals and ensures the prediction to be accurate with time going.

2.4 Cross Validation and Grid Search

2.4.1 Cross Validation

In general, all pattern recognition algorithms have one or more free parameters which need to be optimized for a given classification problem. Cross validation is a popular technique for model selection and performance estimation. Given the values of parameters, then the model can be determined through training and the corresponding performance can be evaluated based upon the classification error rate on the entire population. The easiest way is to divided

the dataset into two subsets randomly: the training subset and used to fit the model and the test subset used to estimate the error rate of the trained model. This is also known as Hold-out method. However, the error estimate of this method can be highly variable, depending precisely on observations while dividing groups. And, just a part of the observations are utilized for model training, the validation error may tend to overestimate the error for the model fit on the entire observations. To overcome the limitations, some other validation techniques are developed like K-Fold cross validation and Leave-one-out cross validation.

K-Fold cross validation is such a technique that estimate the performance of a classifier after dividing the training dataset into k subsets. A single run of K-fold cross validation has the following steps:

(i) Given a training dataset with n observations, so to arrange them in an random order.

(ii) Divide the training dataset into k folds, each fold has approximately n/k observations.

(iii) Cross-over the k folds in successive rounds, take one fold as the validation set and the left $k - 1$ folds as the training sets, and count the wrongly classified observations n_i.

(iv) Estimate the error rate of the classifier, such that $e = (\sum_{i=1}^{k} n_i)/n$.

To obtain an accurate estimate about the error rate, the K-folds cross validation may need to run several times.

Leave-one-out cross validation is the degenerate case of K-folds cross validation, where K is the total number of observations, the steps as below:

(i) Given a training dataset with n observations, the method will perform n times of experiments.

(ii) For each experiment, use $n - 1$ observations for training and the left one for testing. If the test passes, then $n_i = 1$, otherwise $n_i = 0$.

(iii) Estimate the error rate of the classifier, such that $e = 1 - (\sum_{i=1}^{n} n_i)/n$

How many folds should we divide? It depends on the number of observations in the training dataset. In practice, 5-folds or 10-folds are the common choices. To obtain the highest accuracy estimate, Leave-one-out cross validation should be the best choice, but will pay price for the large computation.

2.4.2 Grid Search

Model selection is actually the problem of selecting optimal parameters for the desired algorithm. To perform parameter optimization, the common technique for choice is grid search which is an exhaustive searching through a manually specified a subset of the parameter space and is guided by the cross validation on the training dataset. The specified parameter intervals of manually input should be investigated for performing valid searching which is usually conducted in the log-space. The final outputs of grid search is the settings that achieved the highest performance in the validation procedure. To ensure that the optimized parameters do not over-fit the training dataset, a tuning trick plays the role, i.e. regularization.

$$2.5 \quad \text{One-class Support Vector Machine (1-SVM)}$$

One-class Support Vector Machine (1-SVM) is an unsupervised anomaly detection technique. The basic idea behind is that it maps the data into a higher dimensional space, estimates the high density regions of the data sample, constructs a decision boundary or hyperplane which is maximumly distant from the origin, and finally labels one side of the boundary as positive (normal) and other side as negative (anomaly).

Mathematically, the 1-SVM has the following procedures. Consider a training dataset X which belongs to a single class.

$$X = \{ \mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n \}, \ \mathbf{x}_i \in \mathbb{R}^m, \ m, n \in \mathbb{N} \tag{2.27}$$

The data points are distributed in the input space under some unknown probability

distribution P. For any test sample $\mathbf{x}$, how do people know whether $\mathbf{x}$ follows such a unknown distribution or not? It can be determined through a decision function $f : \mathbb{R}^m \to \mathbb{R}$ which returns $+1$ for the most data points that falls in the decision boundary D, and -1 if the point falls out of the boundary [44], as shown in Figure 2.4.

$$f(\mathbf{x}) = \begin{cases} +1 & if\ \mathbf{x} \in D \\ -1 & if\ \mathbf{x} \notin D \end{cases} \qquad (2.28)$$
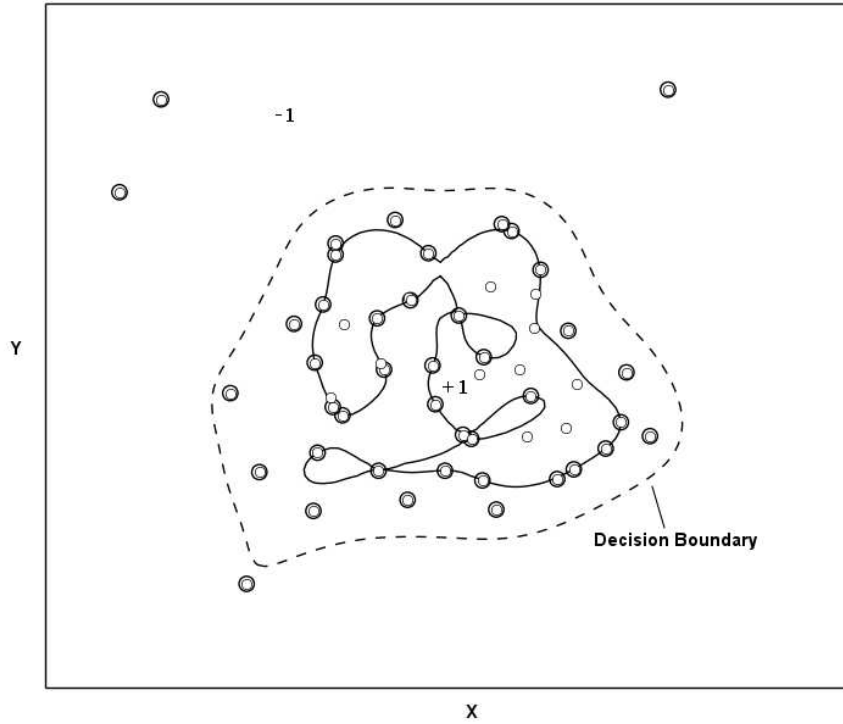
Figure 2.4: Decision Boundary

However, it's hard to directly find such a function with the desirable region in the input space. The strategy is to map the input space into a feature space via a kernel function, and to separate them from the origin with maximum margin. Define a mapping $\Phi : X \to H$, where H is the Reproducing Kernel Hilbert Space (RKHS) with kernel $k : R^n \times \mathbb{R}^m \to \mathbb{R}$,

let's say if Gaussian kernel, then

$$k(\mathbf{x}, \mathbf{x}') = \langle \Phi(\mathbf{x}), \ \Phi(\mathbf{x}') \rangle = exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{\gamma}\right) \qquad (2.29)$$

where γ is the bandwidth. Since $k(\mathbf{x}, \mathbf{x}') = 1, \forall \mathbf{x} \in X$, all data points are mapped onto the unit-radius hypersphere centered at the origin of H. Then the hyperplane is defined as

$$h(\mathbf{x}) = \mathbf{w}^T \Phi(\mathbf{x}) - \rho \qquad (2.30)$$

where $\mathbf{w} \in H$ is the weight vector and $\rho \in \mathbb{R}$ is the bias (or offset). This hyperplane characterized by $\mathbf{w}$ and ρ is used to separate the mapped vectors $\{\Phi\mathbf{x}_i\}_{i=1}^n$ in H, and we hope the margin $\frac{\rho}{\|\mathbf{w}\|}$ is maximum, as illustrated in Figure 2.5.
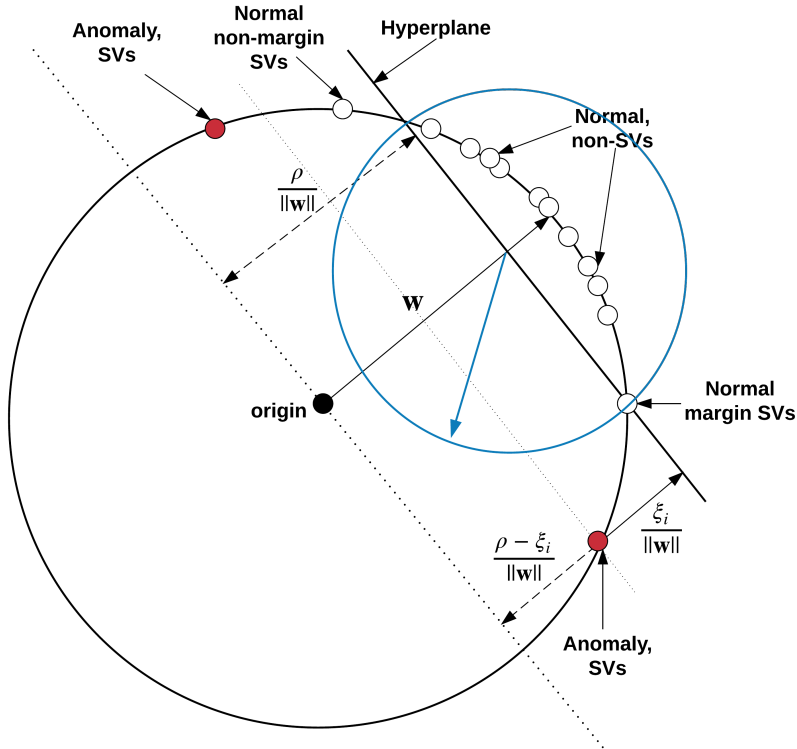
Figure 2.5: 1-SVM illustration

24

To find the maximum margin, we need solve to the following quadratic equation,

$$\min_{\mathbf{w},\xi,\rho} \frac{1}{2}\|\mathbf{w}\|^2 - \rho + \frac{1}{\nu n}\sum_i \xi_i$$

$$\text{s.t.} \quad \langle \mathbf{w}, \Phi(\mathbf{w})\rangle \geq \rho - \xi_i, \tag{2.31}$$

$$\xi_i \geq 0$$

where $\xi_i \in \mathbb{R}, \nu \in (0,1]$. In this equation, ξ_i is the slack variables which penalizes the objective function and tolerant the error of data points that appears on the wrong side of the hyperplane, as the green point shown in Figure 2.5. And, ν has the meaning of an upper bound on the fraction of outliers or training examples regarded out-of-class, and the meaning of a lower bound on the number of training examples used as the support vectors. Let $(\alpha_1, \alpha_2, ..., \alpha_n)$ be the Lagrangian multipliers associated with the constraints, the solution to problem 2.31 will be:

$$\max_{\alpha} \frac{1}{2}\sum_{i,j=1}^{n} \alpha_i\alpha_j k(\mathbf{x}_i, \mathbf{x}_j)$$

$$\text{s.t.} \quad 0 \leq \alpha_i \leq \frac{1}{\nu n} \tag{2.32}$$

$$\sum_{i=1}^{n} \alpha_i = 1$$

Once we got the optimized parameter ν and γ through grid search, we are able to calculate the optimal solution α. After α is obtained, the constant ρ can be achieved by $\rho = \langle \mathbf{w}, \Phi(x_i)\rangle$, where $\mathbf{w} = \sum_{i=1}^{n} \alpha_i\Phi(x_i)$ and $\mathbf{x}_i$ is some sample whose corresponding $\alpha_i \in (0, \frac{1}{\nu n})$. By using kernel function for dot-product calculations, the decision function in input space becomes:

$$f(\mathbf{x}) = sign\left(\langle \mathbf{w}, \Phi(\mathbf{x})\rangle - \rho\right) = sign\left(\sum_{i=1}^{n} \alpha_i k(\mathbf{x}_i, \mathbf{x}) - \rho\right) \tag{2.33}$$

All training data that has $f(\mathbf{x}) \leq 0$ are so-called support vectors (SVs): margin SVs with $f(\mathbf{x}) = 0$ and non-margin vectors with $f(\mathbf{x}) < 0$, as shown in Figure 2.5. We can finally obtain two group of points: normal points if $f(\mathbf{x}) > 0$, then normal, otherwise, anomalies.

CHAPTER 3

APPLICATIONS TO GROUNDWATER MONITORING

3.1 Dataset Description

The dataset used for testing both MSET and 1-SVM comes from the monitoring station named Gilcrest, as shown in Figure 1.2. This monitor has been working for two more years gathering surrogate parameter values at intervals of one hour. We organize the observations in a matrix form as the inputs, see Table 3.1.

**Table 3.1:** Actual Observations with timestamps

Time	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	...
Temperature	9.591	9.581	9.6	9.588	9.591	9.591	9.596	9.598	9.6	...
pH	7.273	7.273	7.273	7.273	7.274	7.275	7.276	7.277	7.277	...
Conductivity	2.282	2.282	2.282	2.282	2.282	2.282	2.282	2.282	2.282	...
ORP	65.329	72.308	73.14	73.295	73.346	73.166	73.385	73.29	73.413	...
DO	1.838	1.841	1.807	1.831	1.831	1.831	1.805	1.817	1.816	...

In this dataset, some surrogate parameter values are missing due to the sensor problems and appear to be `NaN`. We did some data cleaning work and replaced the `NaN` with its previous observed value. It's reasonable, since the groundwater environment keep nearly unchanged within one hour interval, except for big groundwater events.

3.2 MSET Implementation and Results

MSET for real-time groundwater anomaly detection is implemented under the Linux environment by using Python language. As a high-level language, Python is a widely used since it supports multiple programming paradigms like object-oriented and procedural styles. It provides a comprehensive standard libraries and rich third-party libraries. Two fundamental scientific computing packages, `NumPy` and `SciPy`, are adopted in this implementation.
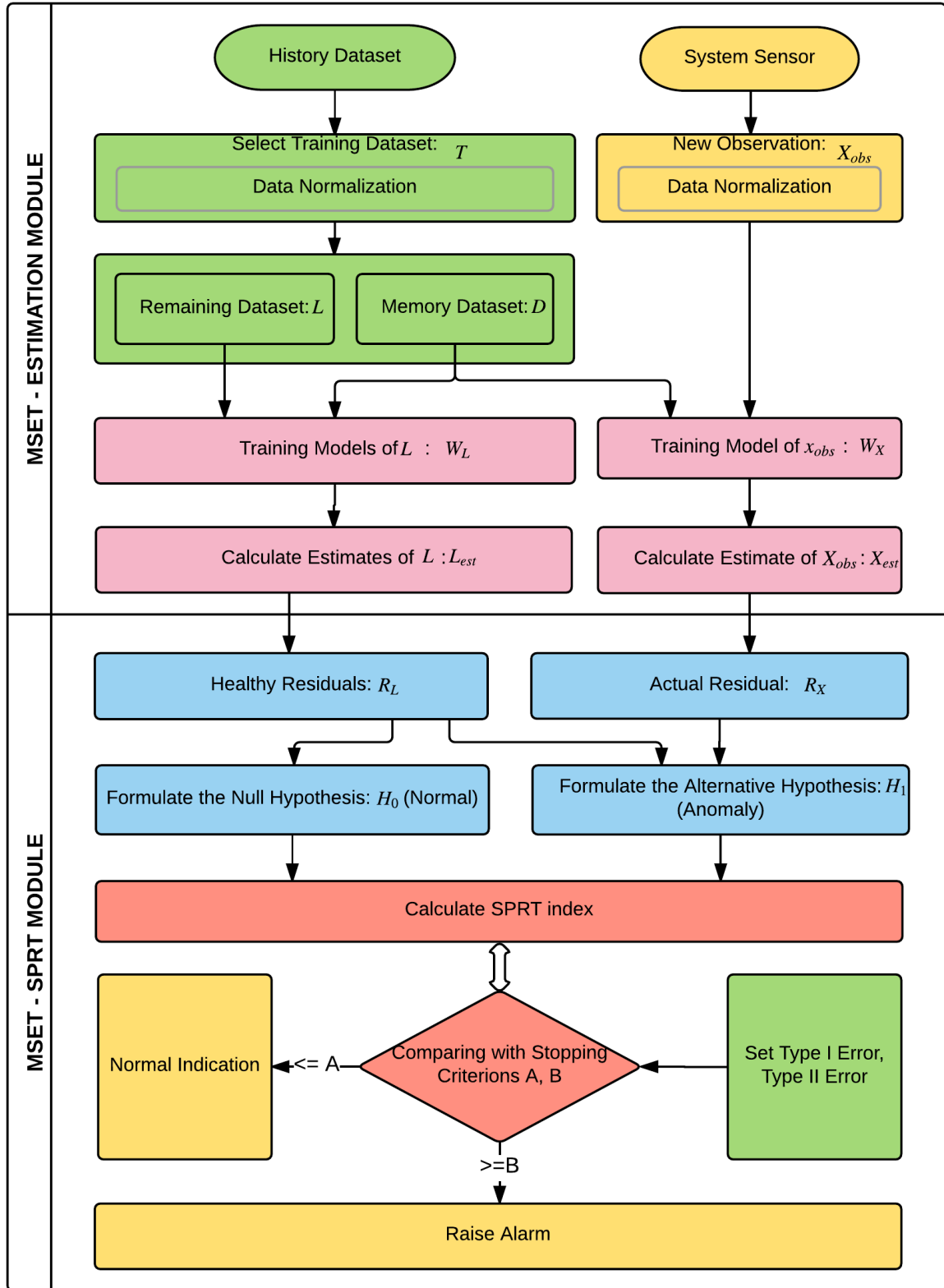
Figure 3.1: MSET-based framework for real-time groundwater anomaly detection

3.2.1 MSET-based Real-time Anomaly Detection Framework

This framework (Figure 3.1) contains two core modules: estimation module and SPRT module. Since it's real-time, the estimation module starts to work whenever the new observation comes in, calculating the estimates for the new observation and the reamining observations. And, SPRT module take the healthy residuals and the actual residual as input to validate the state of the new observation.

3.2.2 MSET Estimation

Suppose at time t_j, a new observation $\mathbf{x}_{obs}$ generates, then the latest n normal observations whose timestamp is smaller than t_j are selected from the historic data to compose the training data matrix $T \in \mathbb{R}^{m \times n}$, here $n = 720$, about one month. This guarantees that T covers the whole healthy range of each parameter. Since the magnitudes of these surrogate parameters are different, so matrix T should be normalized at first. Here, the values of each parameter are scaled into $[0,\ 1]$ according to the formulate below,

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{3.1}$$

The following table shows an example, i.e. normalized training data matrix.

Table 3.2: Normalized observations for model training

Time	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	...
Temperature	0.269	0.200	0.331	0.248	0.270	0.270	0.303	0.317	0.331	...
pH	0.000	0.000	0.000	0.000	0.005	0.011	0.016	0.021	0.021	...
Conductivity	0.400	0.400	0.400	0.400	0.400	0.400	0.400	0.400	0.400	...
ORP	0.000	0.286	0.320	0.326	0.329	0.321	0.330	0.326	0.331	...
DO	0.651	0.665	0.502	0.617	0.617	0.617	0.492	0.550	0.545	...

Then, the memory dataset D is created based on the two rules: i) the observations with extreme values in T are selected and put into D, and ii) the left observations in T are sorted based on their norms, and selected with equal-spaced interval 2. The non-selected

observations in T forms the remaining data matrix L. Based on D and L, the weight vectors $W_\mathbf{x}$ and W_L are calculated for $\mathbf{x}_{obs}$ and L respectively by using Gaussian kernel, as wells the $\mathbf{x}_{est}$ and L_{est}. Next, we calculate estimates for observations which appears in different data patterns, including the normal pattern, single-anomaly pattern, baseline change, and continuous-anomaly pattern.

First, normal patterns contain observations with normal state. The result shows that the estimates are very close to the true observation values, as shown in Figure 3.2, which implies that the series observations are normal and consistent with the fact.
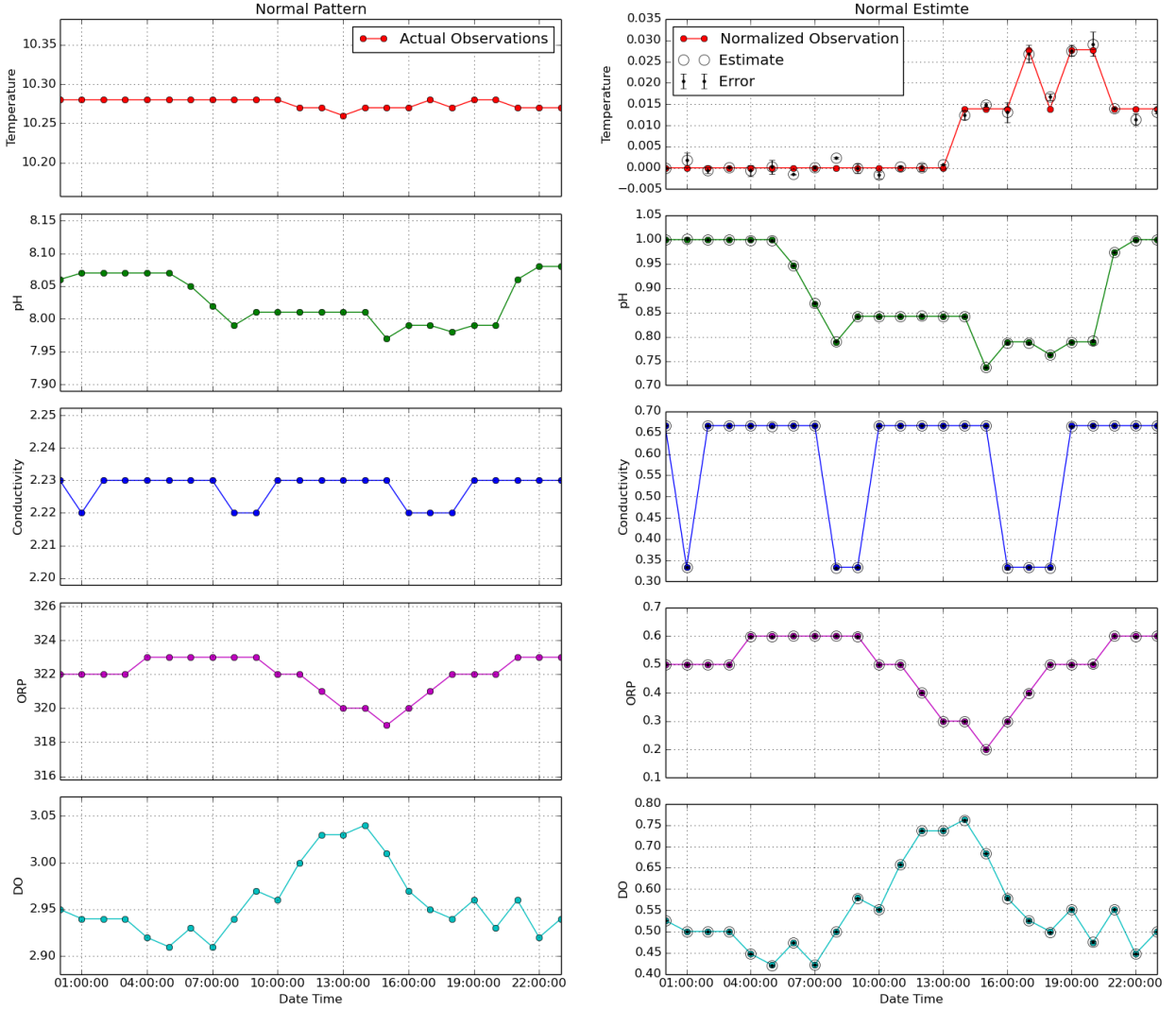
Figure 3.2: *Left:* normal pattern with 24 continuous actual observations; *Right:* observation estimates are calculated based on normalized observation data.

Second, a single-anomaly pattern contains only one anomaly observation during a period time. We send such observations into MSET for estimation sequentially, and get the estimates of these observations as shown in Figure 3.3.
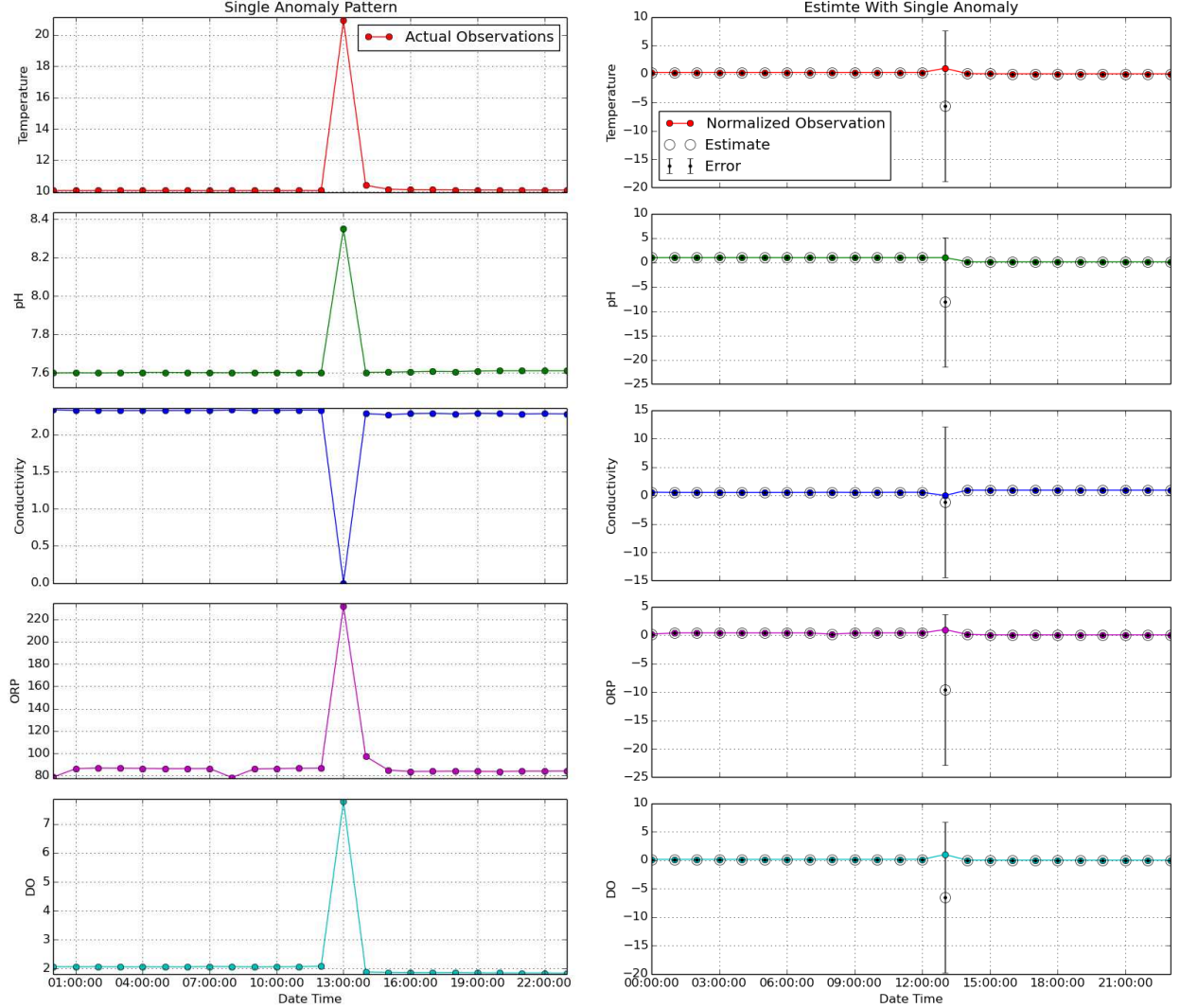
Figure 3.3: *Left:* single-anomaly pattern happens in 24 continuous actual observations; *Right:* observation estimates are calculated based on normalized observation data.

The anomaly occurs due to the sensor or data transmission fault, which brings serious error to the estimate, since the calculation of weight vector $W_{\mathbf{x}} = \left(D^T \otimes D\right)^{-1} \cdot \left(D^T \otimes \mathbf{x}_{obs}\right)$ is closely related to the observation, and so forth the estimate. So the error is large when anomaly happens at 13:00:00. For the normal observations, the estimates are still good.

Third, baseline changes are common phenomenon while monitoring groundwater, which are caused by the season shifts, perceptions or stream charges. Commonly, the estimates based upon observations in old baseline will be inaccurate, as shown in Figure 3.4.
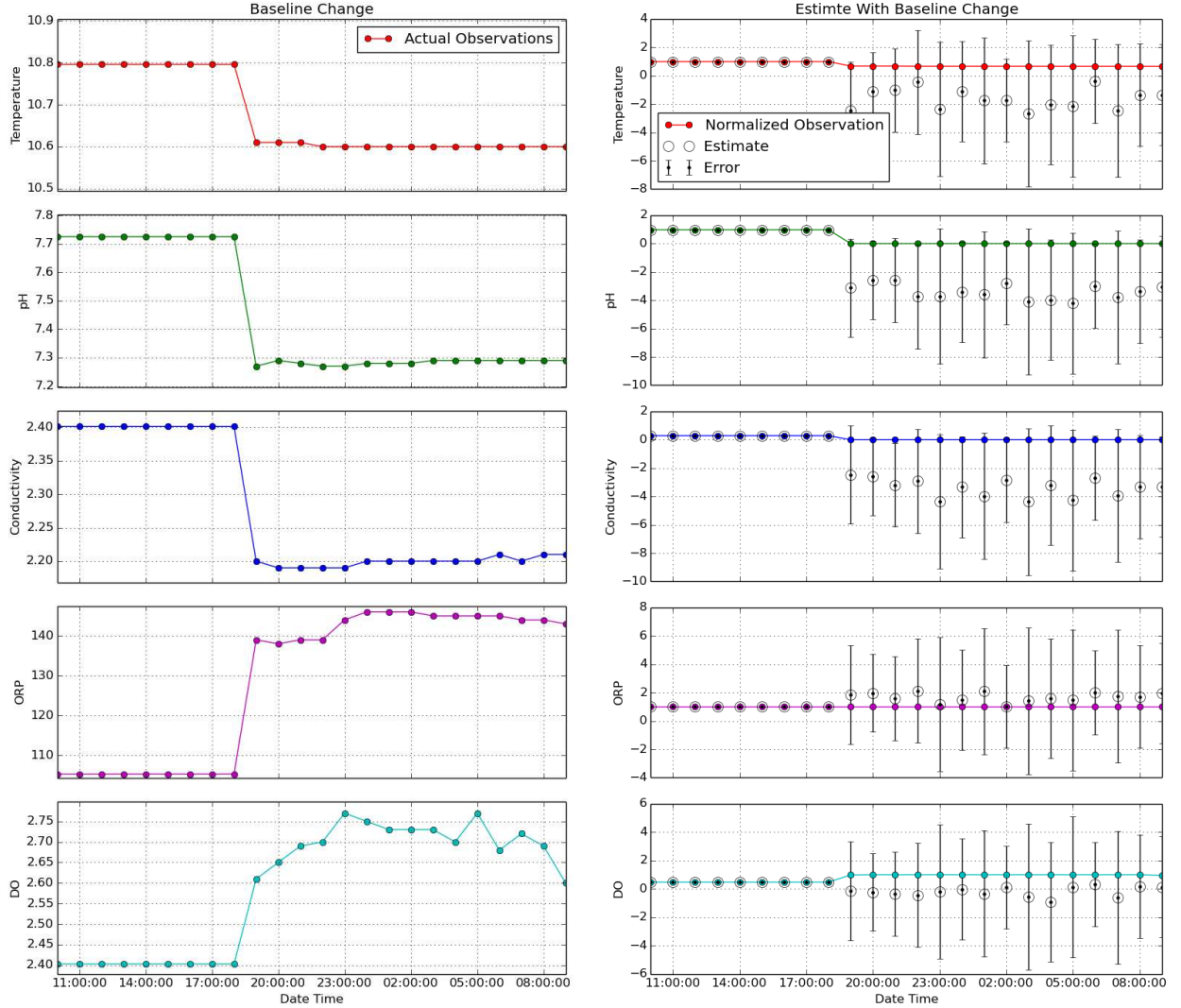
Figure 3.4: *Left:* baseline change happens in 24 continuous actual observations; *Right:* observation estimates are calculated based on normalized observation data.

When MSET encounters baseline changes, these new observations may appear to be anomalies according to the old observations, but they are not in fact. How to deal with this situation? Event recognition methods should be developed based on the knowledge of groundwater hydrology, which is out the scope of this research.

Forth, continuous-anomaly patterns appear due to groundwater contamination or some other events and will last for a period of time, an example is shown in Figure 3.5.
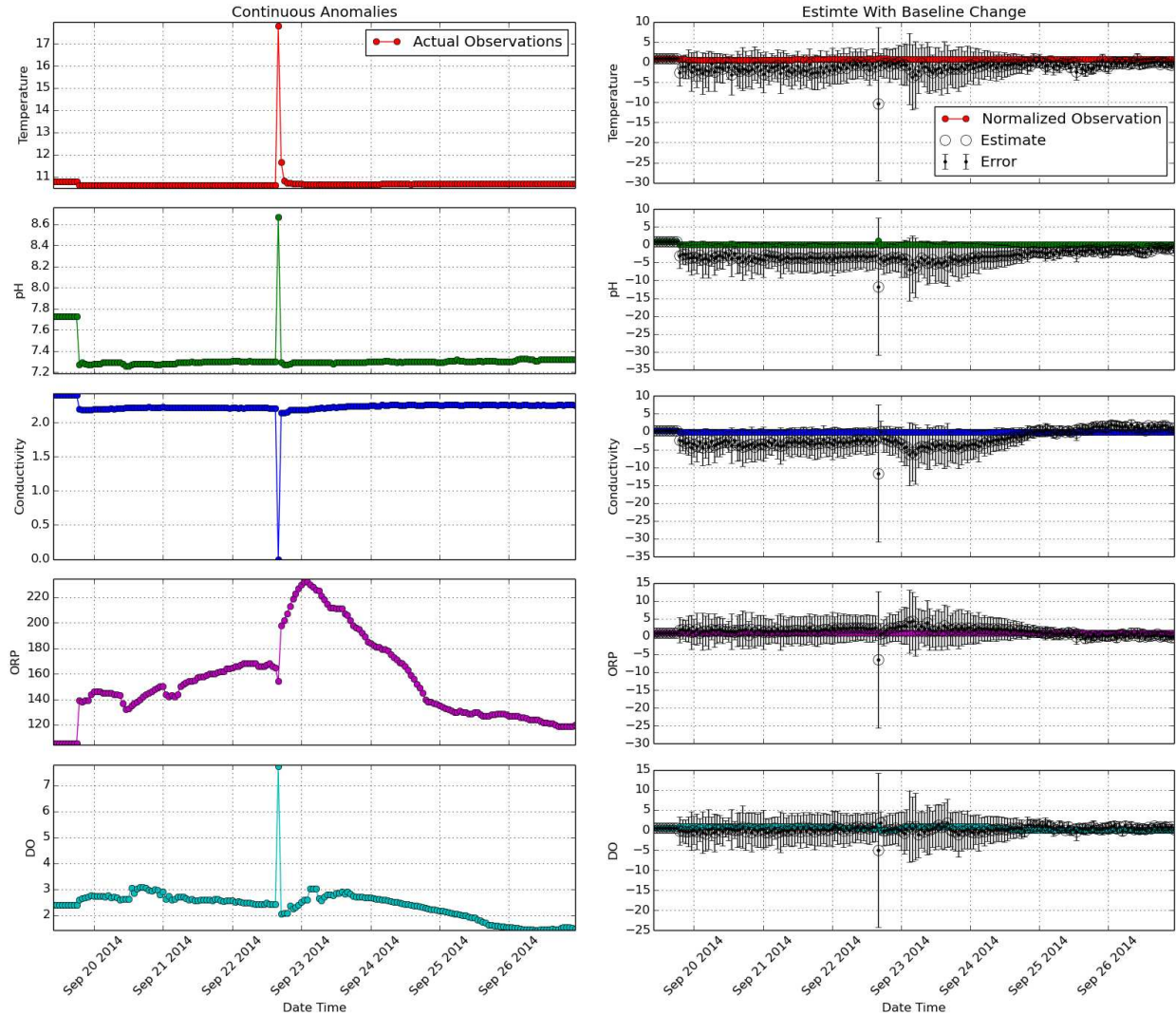

Figure 3.5: *Left:* continuous-anomaly pattern happens in 170 continuous actual observations; *Right:* observation estimates are calculated based on normalized observation data.

Baseline change is one kind of continuous-anomaly patterns, but they have some differences. Typically, the estimates in baseline change are nearly equals to each other, since the baselines are normally parallel. However, the estimates of continuous-anomaly observations appears to be various, because the observations along with the event looks like a wave. MSET has high sensitivity to these changes and generate different estimate results.

3.2.3 SPRT Anomaly Detection

After obtaining the estimates $\mathbf{x}_{est}$ and L_{est}, we can calculate the actual residual $r_{\mathbf{x}}$, the healthy residuals R_L, and their combination R_{LX}. The distribution of R_L is $R_L \sim N(\mu, \sigma^2)$, where $\mu \neq 0$ is the mean of R_L and $\sigma > 0$ is the standard deviation of R_L. The distribution of R_{LX} is $R_{LX} \sim N(\mu', \sigma')$, where $\mu' \neq 0$ is the mean of $R_L X$ and $\sigma' > 0$ is the standard deviation of R_{LX}. For example, an observation comes in at 2015-06-01 03:00:00 and the residuals are obtained as shown in Figure 3.6.
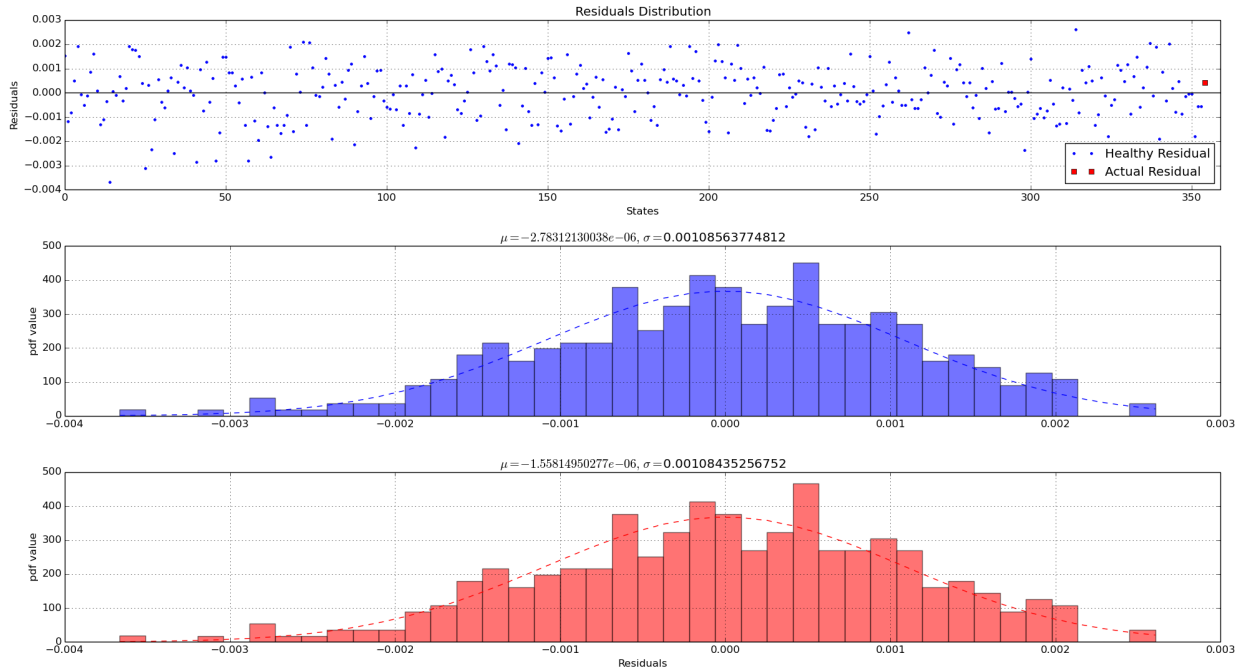
Figure 3.6: *Top:* the healthy and actual residuals distribute around 0; *Middle:* the histogram of healthy residuals and the fit appear to be nearly normal; *Bottom:* the histogram of all residuals and the fit are both nearly normal.

Based on these statistical information, we can construct the hypotheses, as follows.

- $H_0 : R_{LX} \sim N(0, \sigma^2)$, the null hypothesis.

- $H_1 : R_{LX} \sim N(M, \sigma^2)$ where $M = \mu + 3\sigma$ with $\mu > 0$.

- $H_2 : R_{LX} \sim N(-M, \sigma^2)$ where $M = -\mu + 3\sigma$ with $\mu < 0$.

- $H_3 : R_{LX} \sim N(0, V\sigma^2)$ where $V = \sigma'^2/\sigma^2$ with $\sigma' > \sigma$.

- $H_4 : R_{LX} \sim N(0, \sigma^2/V)$ where $V = \sigma'^2/\sigma^2$ with $\sigma' < \sigma$.

Based on these hypotheses, the SPRT index is calculated respectively. For this example, SPRT index I_1 based on H_0 and H_1, and SPRT index H_4 based on H_0 and H_2 are employed, having $I_1 = -8.088$ and $I_4 = -3.980$. Next, according to the user-defined type I error $\alpha = 0.01$ (false alarm) and type II error $\beta = 0.05$ (missing alarm), consequently the boundaries $A = ln(\frac{\beta}{1-\alpha}) = -2.986$ and $B = ln(\frac{1-\beta}{\alpha}) = 4.554$. Since both I_1 and I_4 are smaller than the lower boundary A, therefore H_0 is accepted and this observation is normal.

For anomalous observations, however, the situation will be different. It cause the mean and the standard deviation to change. Take the anomaly at 2014-07-14 10:00:00 for example, the residual result is shown in Figure 3.7.

Figure 3.7: *Top:* the healthy residuals distribute around 0, but the actual residual is nearly 15; *Middle:* the histogram of healthy residuals and the fit appear to be nearly normal; *Bottom:* the histogram of all residuals and the fit are obviously different with the middle ones.
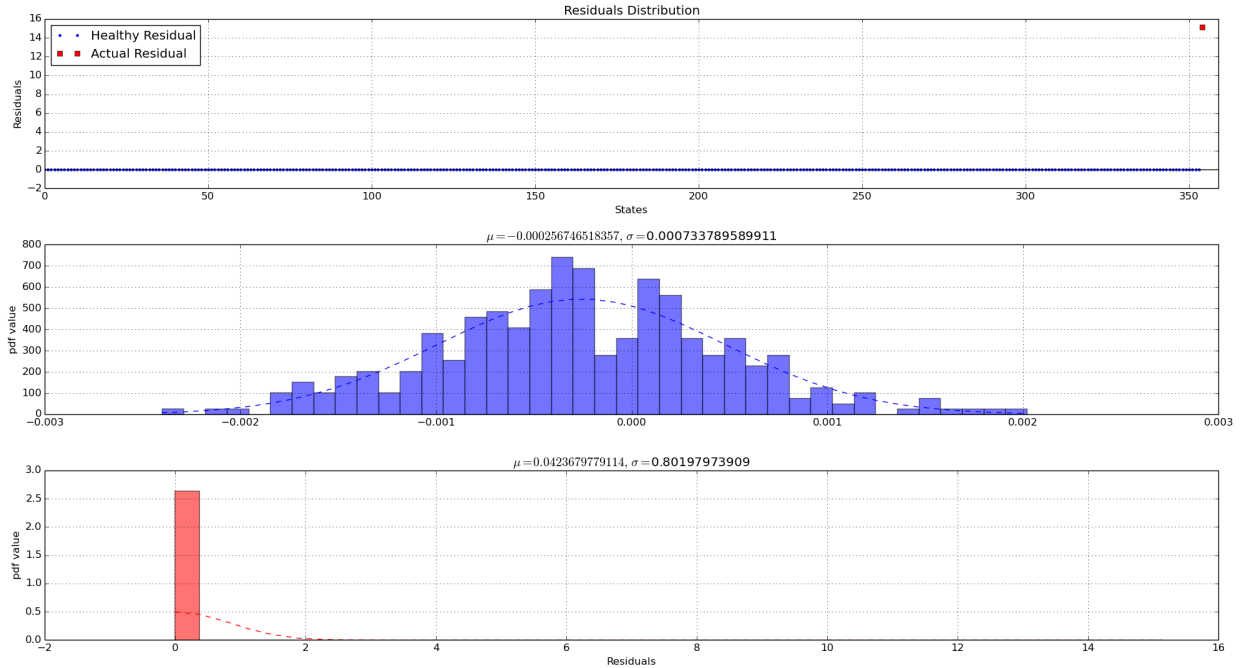
Here, SPRT I_2 based on H_0 and H_2, and I_3 based on H_0 and H_3 are calculated, having $I_2 = -69083.924 < A = -2.986$ but $I_3 = 212613908.322 > B = 4.554$. Therefore, the standard deviation appears to be obviously different with σ. Therefore H_3 is accepted and this observation is anomaly.

Table 3.3: State estimation of observations in baseline change based on SPRT index

Measurement Datetime	Temperature	pH	Condutivity	ORP	DO	SPRT Index	State
2014-09-19 10:00:00	10.796	7.725	2.401	105.352	2.403	-5.871	Normal
2014-09-19 11:00:00	10.796	7.725	2.401	105.352	2.403	-4.495	Normal
2014-09-19 12:00:00	10.796	7.725	2.401	105.352	2.403	-3.405	Normal
2014-09-19 13:00:00	10.796	7.725	2.401	105.352	2.403	-4.985	Normal
2014-09-19 14:00:00	10.796	7.725	2.401	105.352	2.403	-7.096	Normal
2014-09-19 15:00:00	10.796	7.725	2.401	105.352	2.403	-5.900	Normal
2014-09-19 16:00:00	10.796	7.725	2.401	105.352	2.403	-4.898	Normal
2014-09-19 17:00:00	10.796	7.725	2.401	105.352	2.403	-4.478	Normal
2014-09-19 18:00:00	10.796	7.725	2.401	105.352	2.403	-3.578	Normal
2014-09-19 19:00:00	10.61	7.27	2.2	139	2.61	2940353.983	Anomaly
2014-09-19 20:00:00	10.61	7.29	2.19	138	2.65	2068941.436	Anomaly
2014-09-19 21:00:00	10.61	7.28	2.19	139	2.69	2681018.974	Anomaly
2014-09-19 22:00:00	10.6	7.27	2.19	139	2.7	3433809.698	Anomaly
2014-09-19 23:00:00	10.6	7.27	2.19	144	2.77	6307929.153	Anomaly
2014-09-20 00:00:00	10.6	7.28	2.2	146	2.75	4865273.893	Anomaly
2014-09-20 01:00:00	10.6	7.28	2.2	146	2.73	4707069.328	Anomaly
2014-09-20 02:00:00	10.6	7.28	2.2	146	2.73	2593224.241	Anomaly
2014-09-20 03:00:00	10.6	7.29	2.2	145	2.73	4999670.978	Anomaly
2014-09-20 04:00:00	10.6	7.29	2.2	145	2.7	5157422.7236	Anomaly
2014-09-20 05:00:00	10.6	7.29	2.2	145	2.77	6706972.022	Anomaly
2014-09-20 06:00:00	10.6	7.29	2.21	145	2.68	2628710.126	Anomaly
2014-09-20 07:00:00	10.6	7.29	2.2	144	2.72	4260414.562	Anomaly
2014-09-20 08:00:00	10.6	7.29	2.21	144	2.69	3116289.917	Anomaly
2014-09-20 09:00:00	10.6	7.29	2.21	143	2.6	3361050.703	Anomaly

The continuous observations can contain both normal and anomalous observations, the SPRT index values will be smaller than $A = -2.986$ or larger than $B = 4.554$. Without lose generality, we gives the SPRT index result of baseline change here, as shown in Table 3.3. The result shows that MSET-based anomaly detection method works well.

3.3   1-SVM Implementation and Results

The implementation of 1-SVM for groundwater real-time anomaly detection is under the support of `Libvsm` (version 3.11) package [6]. `Libvsm` which was developed by National Taiwan University (NTU) is a C++ library for Support Vector Machines (SVMs) with a Python interface. This library has gained world-wide popularity for data classification and regression in the area of machine learning and data mining.

3.3.1 1-SVM based Real-time Anomaly Detection Framework

As shown in Figure 3.8, the 1-SVM framework for real-time anomaly detection is also divided into two modules: training module and detecting module. The former is used for training a classifier based the normal historic data, and the free parameters ν and γ need to be optimized by means of cross validation before training a model with expectation. The later is used for classify the new observation based on the trained classifier, the output is just the state information: normal $+1$ or anomaly -1.

3.3.2 Parameter Optimization

The accuracy of the 1-SVM classifier highly depends on free parameters ν and γ. To obtain an effective model, we need to find the optimal ν and γ as the training input. The basic procedure contains selecting normal observations from historic data and grid searching based on k-folds cross validation. Suppose a new observation $\mathbf{x}_{obs}$ generates at time t_j, then $n = 720$ normal observations whose timestamp is smaller than t_j are selected for model training, and we give each of these observations a label `+1` indicating its normal state. Then a mesh-grid is created based on the range of ν parameter $(0, 1]$ and the self-defined range
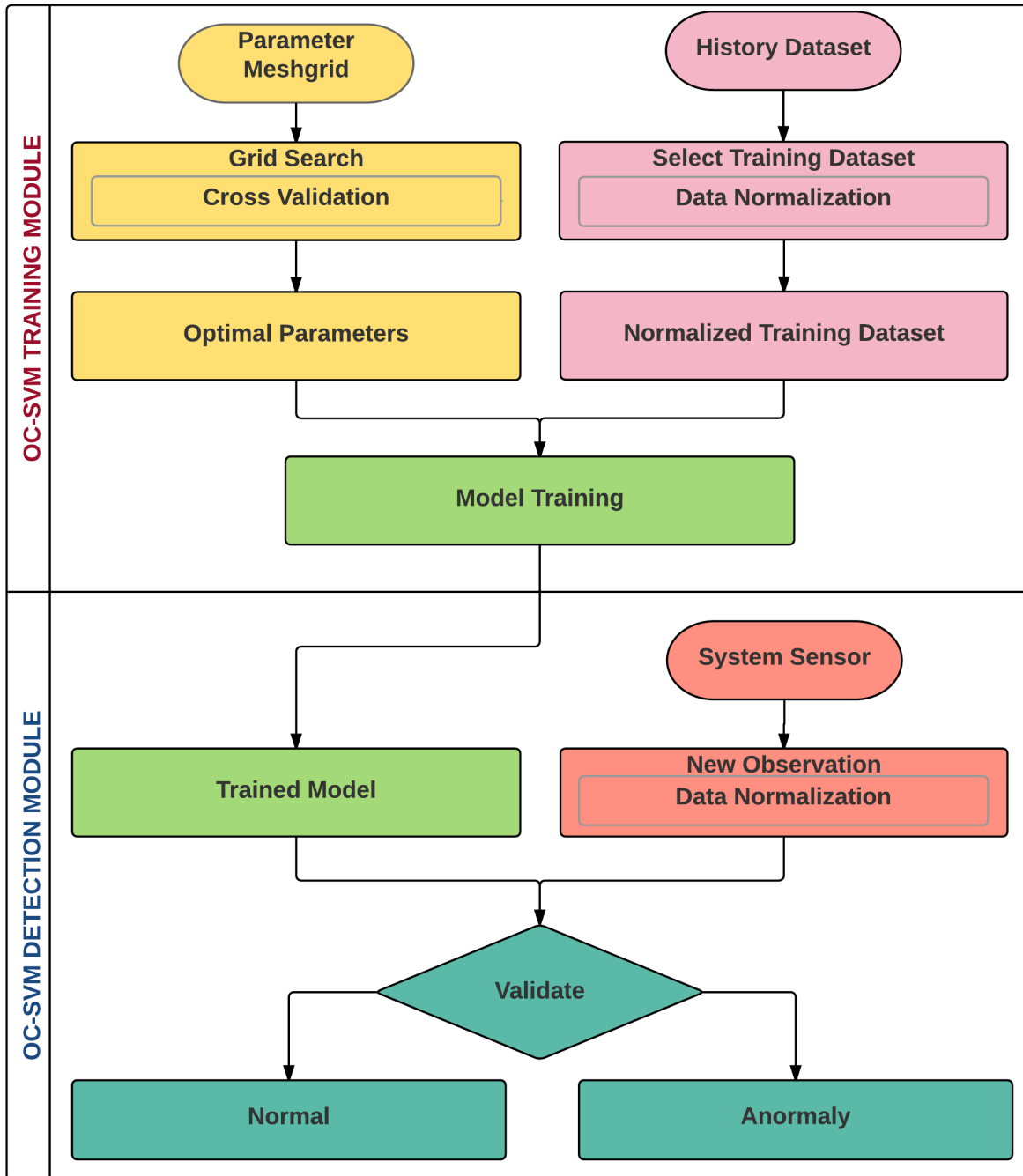
Figure 3.8: 1-SVM based framework for real-time groundwater anomaly detection

of γ parameter $[\log(10e - 4), \log(10e + 4)]$ in the log-space with the mesh size $n \times n$, here $n = 50$. Since we can get a trained classifier regarding to each pair of ν and γ, 5-folds cross validation technique is applied to test the accuracy of this classifier. Finally, the pair of ν and γ based on which the trained classifier has the highest classification accuracy are taken as the optimal ν and σ. Here, we gives several examples of optimized parameter ν and σ, as shown in Figure 3.9.

(a)

(b)

(c)

(d)
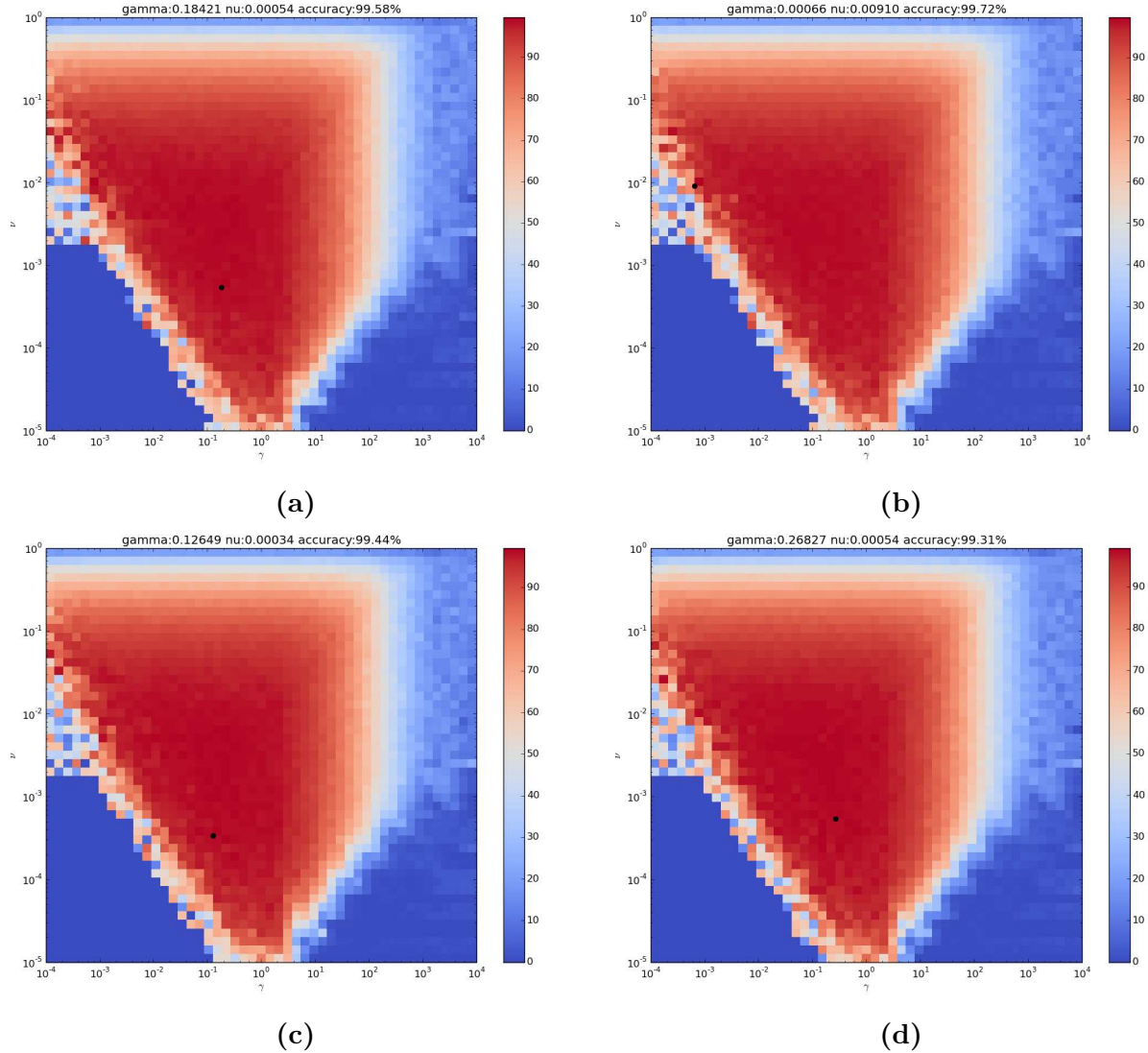
Figure 3.9: Grid search for 1-SVM parameter optimization: the colorbar indicates the accuracies of models trained based on different pairs of ν and γ, the dark point contains the optimal μ and γ.

Grid search generates similar result for parameter ν and γ each time. On the one hand,

the range of the parameters are valid, especially for γ range $[\log(10e-4), \log(10e+4)]$ in which the validated model shows high classification accuracy by combining a desirable ν. Otherwise, the accuracy of the classifier will be nearly zero. On the other hand, due to the randomness of fold-dividing, the subsets used for training and testing will be different when performing cross validation each time. Therefore, the optimal parameter values are not exactly identical even that the training data is the same. However, the randomness might cause that the final parameters are not optimal. In this situation, we need to perform k-folds cross validation several times or use Leave-one-out cross validation as the alternative.

3.3.3 Model Training

With the optimal parameters ν and γ obtained through grid search, then a 1-SVM classifier will be trained based on the training dataset in which all observations are normal. The observations are mapped from the input space R^5 into a feature space H through Gaussian kernel in order to find a hyperplane which maximize the margin. The process of producing a model is easy to be implemented under Libsvm, like this:

```
from libsvm.python import svmutil
def train(observations, labels, optimal_gamma, optimal_nu):
    # -s 2: one-class svm
    # -t 2: gaussian kernel
    # -n : nu
    # -g : gamma
    prob = svmutil.svm_problem(labels, observations)
    param = svmutil.svm_parameter("-s 2 -t 2 -n 0.1 -g 0.1")
    param.gamma, param.nu = optimal_gamma, optimal_nu
    model = svmutil.svm_train(prob, param)
    return model
```

For example, with $\nu = 0.32$ and $\gamma = 0.0061$, the train function returns the model below,

```
optimization finished, #iter = 12
obj = 6.267336, rho = 2.880065
nSV = 6, nBSV = 2
```

where `obj` is the optimal objective value (equation 2.33). `rho` is the bias term in the decision function $sign(\mathbf{w}^T\mathbf{x}_{obs}-\rho)$. `nSV` and `nBSV` are number of support vectors and bounded support vectors.

3.3.4 Anomaly Detecting

Whenever a new observation generates, a model will be trained based upon the historic normal observations for predicting the label, i.e. the state, of the new observation. The prediction process performs like this:

```
from libsvm.python import svmutil
def predict(model, observation):
    # model : the trained model
    # observation: a single observation
    pred_label = svmutil.libsvm.svm_predict(model, observation)
```

One should notice that since the model is trained based on normalized data, the input observation for prediction should also be normalized. The `pred_label` will be either `+1` or `-1`. The former indicates that the tested observation is normal, but the later implies the tested observation is anomalous. Unlike MSET, 1-SVM doesn't provide any estimate for each single surrogate parameter, and generate only a state label for a whole vector.

As what we do for MSET, four main data patterns (normal pattern, single-anomaly pattern, baseline change, and continuous-anomaly pattern) with the same observations are used to test the capability of 1-SVM for abnormal behavior detection while conducting groundwater monitoring. Similarly, 1-SVM here can only predict the state of the single observation sequentially, and has no ability of recognizing event patterns. Event recognition algorithm should be developed independently based on the type of events defined in the groundwater field. However, that development has close relationship with the sequential states of observations. Event recognition is beyond the scope of this research and won't be discussed much in this paper. Here, we focus on the real-time anomaly detection of 1-SVM as stated below.

First, for the normal pattern with 24 continuous normal observations, the performance of 1-SVM is good and all states of these observations are indicated as normal without wrong classification, as shown in Figure 3.10.
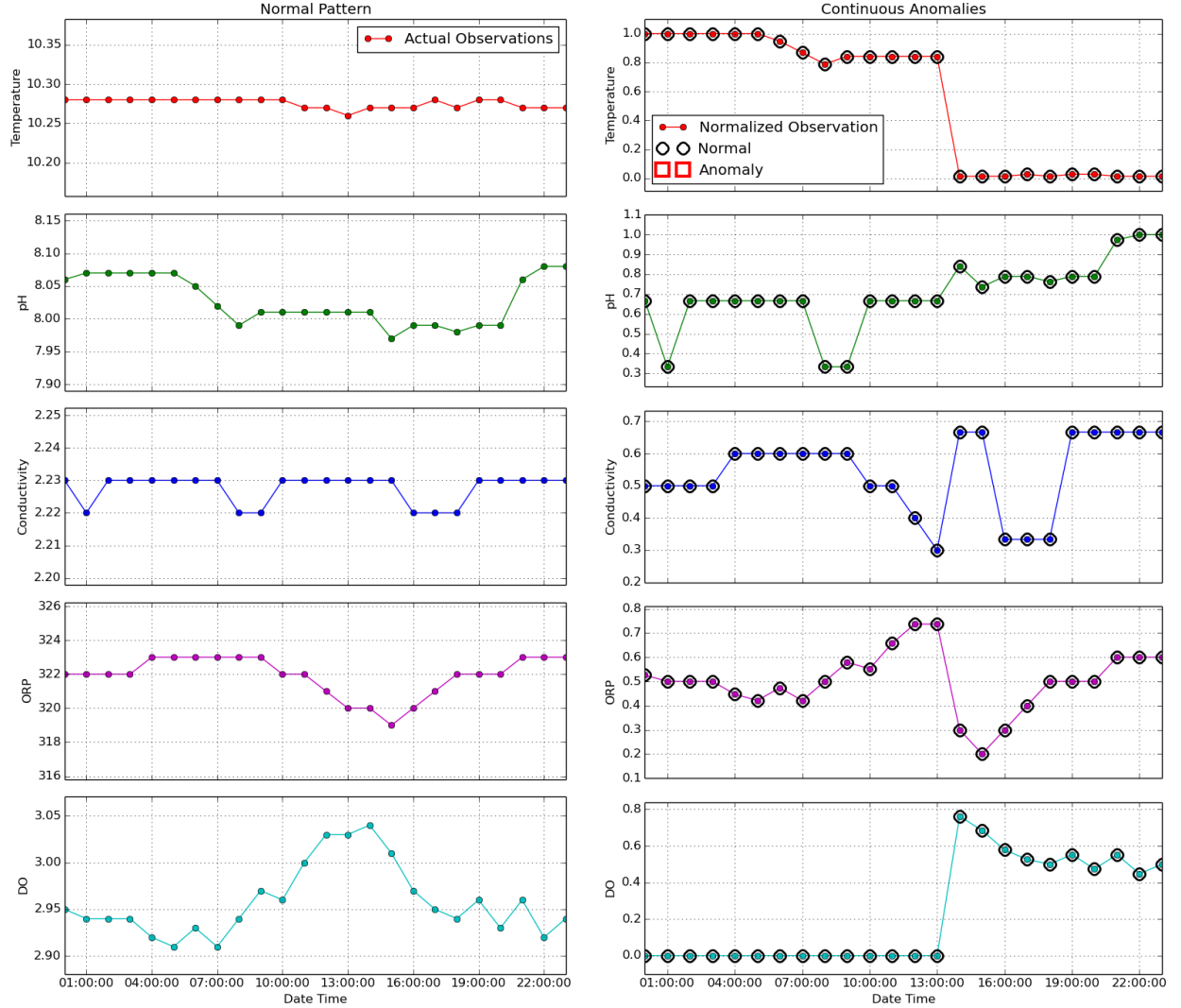
Figure 3.10: *Left:* normal pattern with 24 continuous actual observations; *Right:* the observation states are predicted orderly based upon the normalized training observation and marked with symbols (normal: dark circle, anomaly: red square).

From the result we can see that all states are marked with dark circle, implying the predicted states are same with their own status, therefore 1-SVM works well in regular mode.

Second, single-anomaly pattern contains 24 observations with an abnormal one. These observation is normalized and sent to 1-SVM algorithm in order. Finally, we get the predicted state as shown in Figure 3.11.
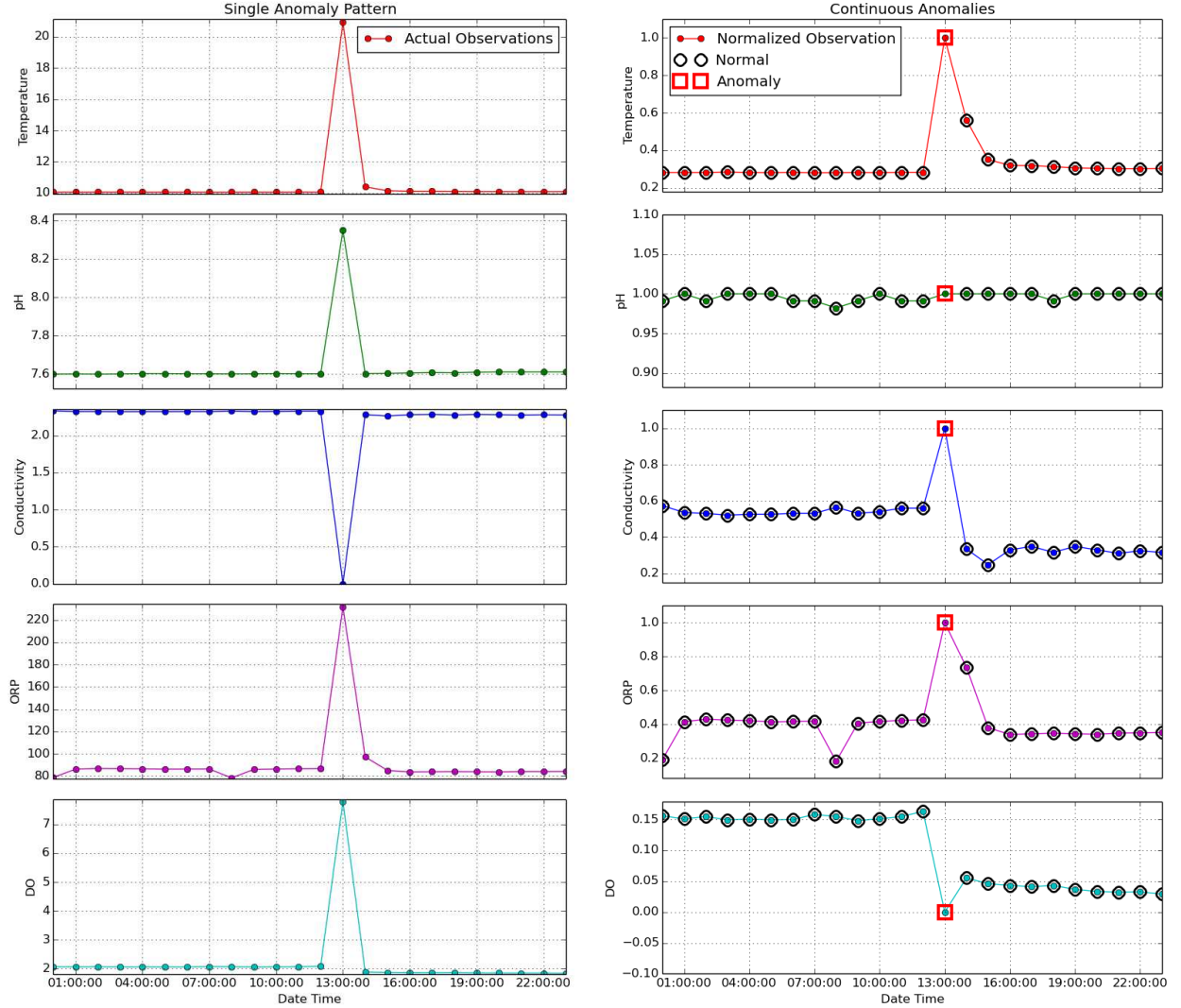
Figure 3.11: *Left:* single-anomaly pattern appears in 24 continuous actual observations; *Right:* the observation states are predicted orderly based upon the normalized training observation and marked with symbols (normal: dark circle, anomaly: red square).

From the two plots, we can see that only state at 13:00:00 is marked as anomaly, which is consistent with the actual observation. All other states are marked with dark circles indicating that these observations are normal.

Third, when baseline change happens among 24 observations, 1-SVM can also give correct indications. The results are shown in Figure 3.12.
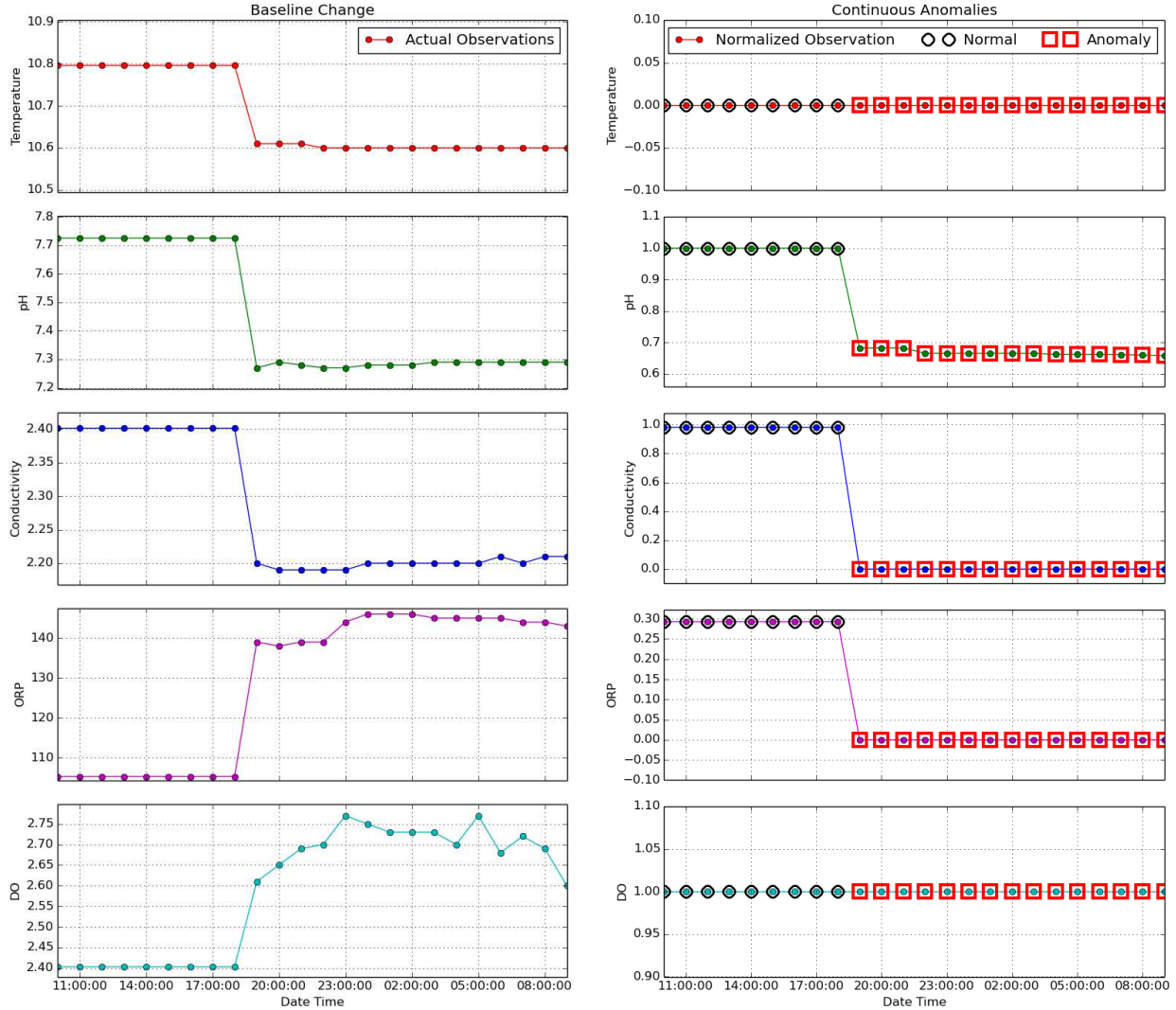

Figure 3.12: *Left:* baseline change pattern appears in 24 continuous actual observations; *Right:* the observation states are predicted orderly based upon the normalized training observation and marked with symbols (normal: dark circle, anomaly: red square).

From the result, we can see that the baseline change happens at 19:00:00. The each surrogate parameter on new baseline are obviously different with the one on the old baseline. This change is detected by 1-SVM, the observations on the new baseline are labeled as anomalies. The left normal observations are marked with dark circles.

Forth, the group of observations with continuous anomalies is used to test 1-SVM, generating a series of predicted states. The results are shown in Figure 3.13.
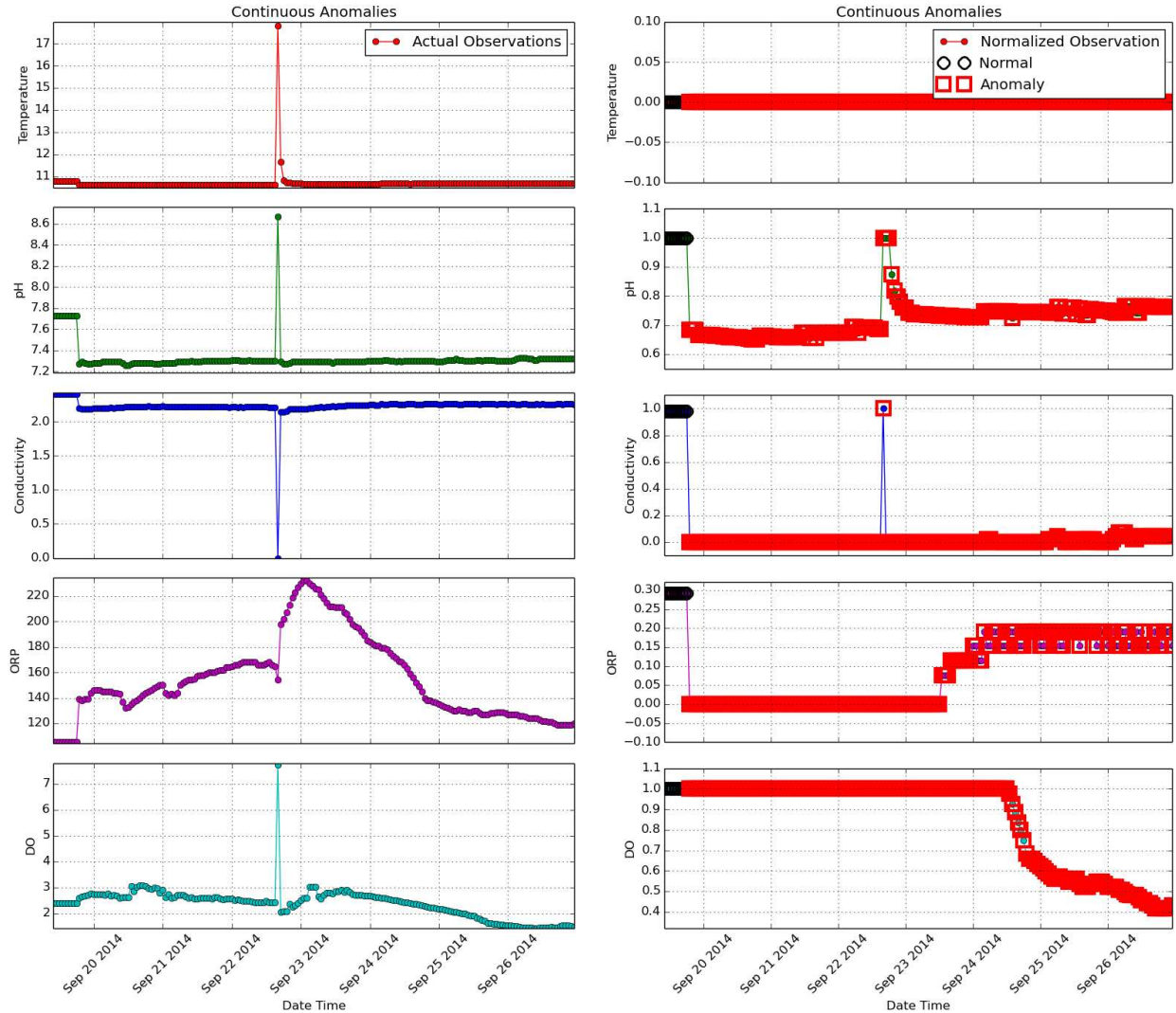
Figure 3.13: *Left:* continuous-anomaly pattern happens among 170 continuous actual observations; *Right:* the observation states are predicted orderly based upon the normalized training observation and marked with symbols (normal: dark circle, anomaly: red square).

1-SVM generates accurate prediction of states, marking normal state with dark circles and anomaly states with red squares. However, unlike MSET, one hardly tell the continuous anomaly happens due to baseline change or some other event, since 1-SVM only gives a binary state indication. Overall, the performance of 1-SVM on various data patterns is good.

3.4 Discussion

In the previous two sections, we implemented two anomaly detection methods: MSET and 1-SVM. Based on the test results, these two methods can adapt to different circumstances and have good capability of real-time groundwater anomaly detection. However, these tests are based on piecewise observations and these data patterns are typical. How can we know their overall performance on routine monitoring work? Due to the complexity of underground environment, the daily monitoring data are always full of noise. Can these methods perform as good as they do above? To answer these questions, we test MSET and 1-SVM based on 1000 continuous observations and get the state estimation result, as shown in Figure 3.14.
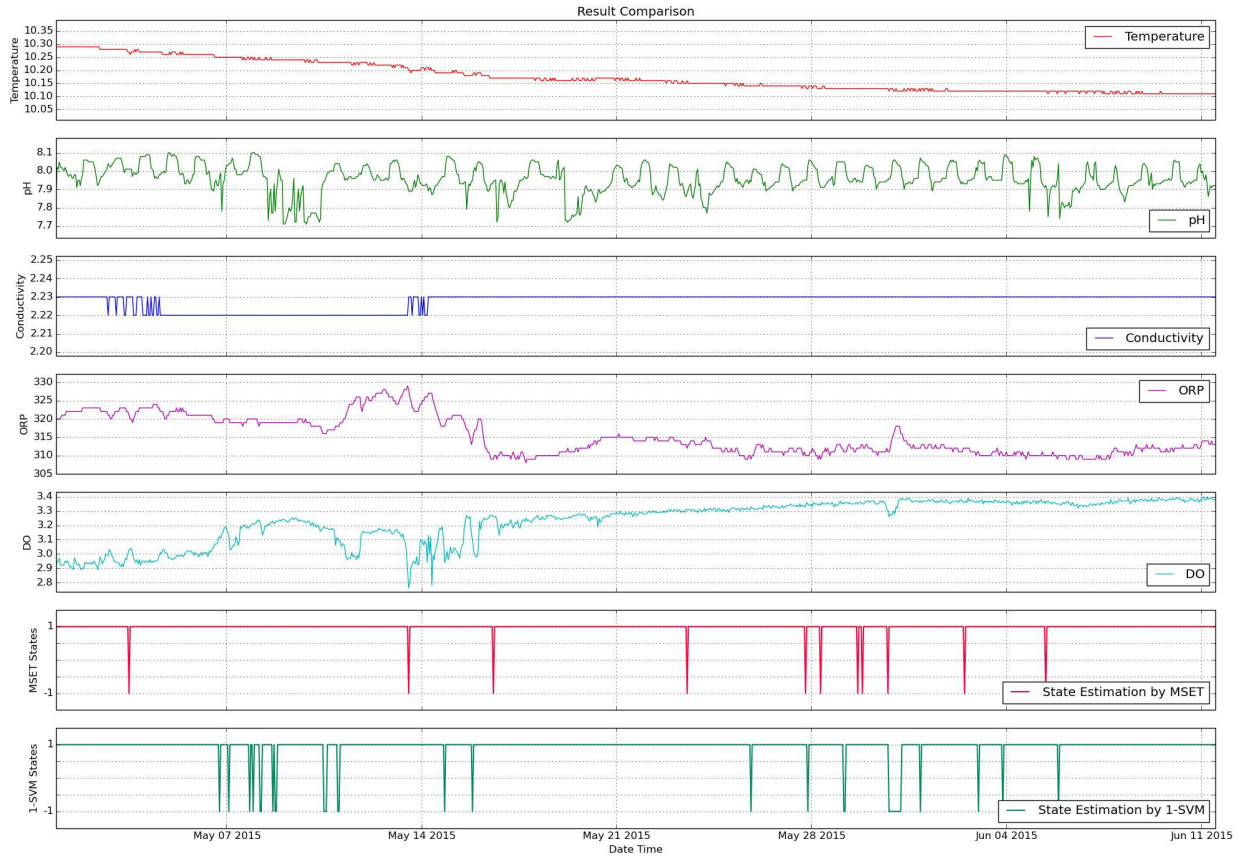
Figure 3.14: From top to bottom, subplot 1-5 shows the actual surrogate parameters; subplot 6 shows the state estimation result generated from MSET; subplot 7 shows the state estimation result generated from 1-SVM. Current state is normal if +1, otherwise anomaly if -1.

This series of observations are full of noisy points, but all of the 1000 observations are normal in fact, i.e. all state should be labeled with +1. However, the noisy observations have great negative effects on the anomaly detection algorithms and cause false alarms. Based on their limited prior knowledge ($n = 720$ historic observations for training), they might wrongly consider the noisy observation as an abnormal observation. That's why the estimated states from MSET and 1-SVM have −1 label (or anomaly state). The statistical results are shown in Table 3.4. MSET performs a little better than 1-SVM, but both have high classification accuracies.

Table 3.4: MSET and 1-SVM state estimation statistical results

Method	Number of Observations	Number of Normal States	Number of Anomaly States	Accuracy
MSET	1000	989	11	98.90%
1-SVM	1000	965	35	96.50%

The inaccurate parts might results from the limitations of themselves. For MSET, it assumes that the training dataset covers the whole healthy range of the surrogate parameters. Otherwise, the estimation will be inaccurate and considered as anomaly if the new observation falls out the range of the training dataset. For 1-SVM, the biggest challenge is to find the optimal parameters ν and γ, a subtle change might lead to totally different estimation result. Grid search K-folds cross validation randomly divide the training dataset into K folds which will be used for training and testing in turn. However, the optimized parameter we get might not be the optimal ones due to the randomness. Leave-one-out cross validation may be a better choice, but it's a time-consuming and resource-consuming method if the training dataset is large. To overcome these limitations, one possible way is to enlarge the training dataset to widen the surrogate parameters' ranges, and to reduce the probability of finding non-optimal parameters, but we need to afford the computational cost.

To sum up, the two methods are suitable for real-time groundwater anomaly detection based on their overall performances over the historic dataset.

CHAPTER 4

GROUNDWATER TRANSPORT SIMULATIONS AND MONITORING

The anomaly detection algorithms studied in the previous chapters are expected to get response when a groundwater contamination event happens. To validate their effectiveness, we simulate the groundwater flow and contamination transport in a rectangular domain through numerical methods. During the process, we monitor the contaminant concentration change along with the groundwater flow over time, and send the collected data into anomaly detection algorithms in real-time. The MSET and/or 1-SVM should be sensitive to such a groundwater event and raise alarm at a certain point if these algorithms are valid. Additionally, Python language and `NumPy` package are employed for the simulations.

4.1 Flow and Transport in Groundwater

Darcy's law is the fundamental principle which governs how groundwater moves in the subsurface porous medium. In general terms, groundwater movements controls the contaminant transport. So groundwater modeling refers to the mathematical representation of the Darcy flow system with specific hydrogeological conditions. In general, the groundwater flow and contaminant transport in the porous medium domain are three-dimensional [2]. However, groundwater flow in the saturated thickness of an unconfined aquifer is practically horizontal. In this context, all the models are built on the assumption that the aquifer flow is essentially horizontal, i.e. 2-dimensional

To describe the groundwater flow through porous medium, the Darcy problem is used in a bounded polygonal area as below [29]

$$
\begin{cases}
\nabla \cdot (-\mathbf{K}\nabla p) \equiv \nabla \cdot \mathbf{u} = f, \quad \mathbf{x} \in \Omega, \\
p = p_D, \ \mathbf{x} \in \Gamma^D, \qquad \mathbf{u} \cdot \mathbf{n} = u_N, \ \mathbf{x} \in \Gamma^N,
\end{cases}
\tag{4.1}
$$

where $\Omega \subset \mathbb{R}^2$ is the polygonal domain, $\mathbf{K}$ is the horizontal hydraulic conductivity, f is the

specific discharge, p is an unknown pressure, p_D is a Dirichlet boundary condition, and u_N is Neumann boundary condition, $\mathbf{n}$ is a unit outward normal vector on $\partial\Omega = \Gamma^D \cup \Gamma^N$.

The transport process can be prototyped by the convection-diffusion equation,

$$
\begin{cases}
\partial_t c + \nabla \cdot (\mathbf{v}c - D\nabla c) = f(x, y, t), & (x, y) \in \Omega, t \in (0, T), \\
c(x, y, t) = 0, & (x, y) \in \partial\Omega, t \in (0, T), \\
c(x, y, 0) = c_0(x, y), & (x, y) \in \Omega.
\end{cases}
\tag{4.2}
$$

where $c(x, y, t)$ is the unknown contaminant concentration to be solved, $\mathbf{v}$ is the fluid velocity, $D > 0$ is the diffusion coefficient of specified pollutant, f is the source term. Generally speaking, the mechanisms of transport include convection, diffusion and reaction. When there is no reaction, and the diffusion can be safely ignored, we have a pure convection problem which takes the form:

$$
\partial_t c + \nabla \cdot \mathbf{v}c = f(x, y, t).
\tag{4.3}
$$

4.2   Numerical Methods for Simulations of Flow and Transport

Usually, the analytical forms for exact solutions do not exist for flow and transport equations discussed in the previous section. Numerical methods for approximate solutions are expected. There're abundant of numerical methods for the two equations. For rectangular domains, finite difference methods (FDMs) are easier to use. When the geometry of domain becomes complicated, finite element methods (FEMs) work better. In this section, three different numerical methods are described.

4.2.1 Finite Difference Method for Darcy Velocity

To solve the Darcy equation 4.1 and obtain Darcy velocity $\mathbf{u}$, FDM can be applied if the domain is rectangular and discretized. As is known, the Darcy velocity is calculated based

on the equation,

$$\mathbf{u} = -\mathbf{K}\nabla p \tag{4.4}$$

where $\nabla p = (\frac{\Delta h_x}{\Delta l_x}, \frac{\Delta h_y}{\Delta l_y})$ is the gradient of water level in 2-dim, h hydraulic head (unit m) and l_x, l_y flow path along x, y directions, as shown in Figure 4.1
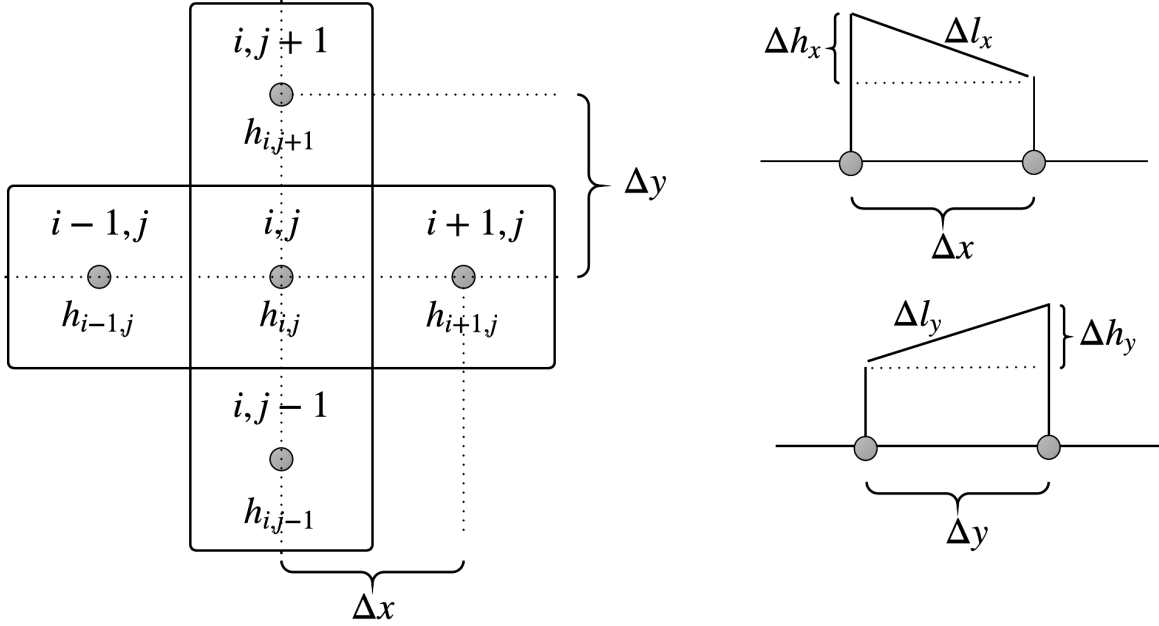
Figure 4.1: A schematic diagram of finite difference scheme for solving Darcy equation

The hydraulic conductivity $\mathbf{K}$ is defined by the following tensor:

$$\mathbf{K} = \begin{bmatrix} K_{xx} & K_{xy} \\ K_{yx} & K_{yy} \end{bmatrix} \tag{4.5}$$

It is a common practice to assume that the principal directions of anisotropy can be aligned with the x, y coordinate axes such that $K_{xy} = K_{yx} = 0$ and $K_x = K_{xx}, K_y = K_{yy}$ [18]. Therefore, Darcy velocity equation 4.4 takes the form

$$\mathbf{u} = \left(-K_x \frac{\partial h_x}{\partial l_x}, -K_y \frac{\partial h_y}{\partial l_y} \right). \tag{4.6}$$

49

If the hydraulic conductivity is homogenous in the grid cell, then $K_x = K_y = K$. The derivative $\frac{\partial h_x}{\partial l_x}$ and $\frac{\partial h_y}{\partial l_y}$ can be obtained in three different ways: forward difference $\frac{\partial h}{\partial l} = \frac{h_{k+1} - h_k}{\Delta l}$, backward difference $\frac{\partial h}{\partial l} = \frac{h_k - h_{k-1}}{\Delta l}$ and central difference $\frac{\partial h}{\partial l} = \frac{h_{k+1} - h_{k-1}}{2\Delta l}$. At the boundary of the domain, forward or backward difference at cells might be the choice, the inner cells can take any of them, central difference is adopted in this thesis.

4.2.2 Upwind Finite Difference Method for Transport Simulations

With the Darcy velocity $\mathbf{u}$, we can solve the convection-diffusion equation for contaminant transport through the discretized area by using FDMs. Based on equation 4.2, we can get

$$\frac{\partial c(x,y)}{\partial t} + v(x,y)\nabla \cdot c(x,y) + c(x,y)\nabla \cdot v(x,y) - D\nabla \cdot \nabla c(x,y) = f(x,y,t) \tag{4.7}$$

We apply the 1st order upwind scheme to the convection problem with flow $v(x,y) = (v_x, v_y)$. If $v_x > 0$ in x direction and $v_y > 0$ in y direction, then the scheme will be:

$$
\begin{aligned}
&\frac{c_{i,j}^{(n+1)} - c_{i,j}^{(n)}}{\Delta t} + \left(v_x \frac{c_{i,j}^{(n)} - c_{i-1,j}^{(n)}}{\Delta x} + v_y \frac{c_{i,j}^{(n)} - c_{i,j-1}^{(n)}}{\Delta y} \right) \\
&+ \left(c_{i,j} \frac{v_{x\,i,j}^{(n)} - v_{x\,i-1,j}^{(n)}}{\Delta x} + c_{i,j} \frac{v_{y\,i,j}^{(n)} - v_{y\,i,j-1}^{(n)}}{\Delta y} \right) \\
&- D \left(\frac{c_{i+1,j}^{(n)} - 2c_{i,j}^{(n)} + c_{i-1,j}^{(n)}}{(\Delta x)^2} + \frac{c_{i,j+1}^{(n)} - 2c_{i,j}^{(n)} + c_{i,j-1}^{(n)}}{(\Delta y)^2} \right) = f_{i,j}^{(n)}
\end{aligned}
\tag{4.8}
$$

The resulting scheme takes the form:

$$
\begin{aligned}
c_{i,j}^{(n+1)} =\; & c_{i,j}^{(n)} - \Delta t \left(v_x \frac{c_{i,j}^{(n)} - c_{i-1,j}^{(n)}}{\Delta x} + v_y \frac{c_{i,j}^{(n)} - c_{i,j-1}^{(n)}}{\Delta y} \right) \\
& - \Delta t \left(c_{i,j} \frac{v_{x\,i,j}^{(n)} - v_{x\,i-1,j}^{(n)}}{\Delta x} + c_{i,j} \frac{v_{y\,i,j}^{(n)} - v_{y\,i,j-1}^{(n)}}{\Delta y} \right) \\
& + D\Delta t \left(\frac{c_{i+1,j}^{(n)} - 2c_{i,j}^{(n)} + c_{i-1,j}^{(n)}}{(\Delta x)^2} + \frac{c_{i,j+1}^{(n)} - 2c_{i,j}^{(n)} + c_{i,j-1}^{(n)}}{(\Delta y)^2} \right) + f_{i,j}^{(n)}
\end{aligned}
\tag{4.9}
$$

If $v_x < 0$ in x direction and $v_y < 0$ in y direction, then the resulting scheme will be:

$$
\begin{aligned}
c_{i,j}^{(n+1)} = & c_{i,j}^{(n)} - \Delta t \left(v_x \frac{c_{i+1,j}^{(n)} - c_{i,j}^{(n)}}{\Delta x} + v_y \frac{c_{i,j+1}^{(n)} - c_{i,j}^{(n)}}{\Delta y} \right) \\
& - \Delta t \left(c_{i,j} \frac{v_{xi+1,j}^{(n)} - v_{xi,j}^{(n)}}{\Delta x} + c_{i,j} \frac{v_{yi,j+1}^{(n)} - v_{yi,j}^{(n)}}{\Delta y} \right) \\
& + D\Delta t \left(\frac{c_{i+1,j}^{(n)} - 2c_{i,j}^{(n)} + c_{i-1,j}^{(n)}}{(\Delta x)^2} + \frac{c_{i,j+1}^{(n)} - 2c_{i,j}^{(n)} + c_{i,j-1}^{(n)}}{(\Delta y)^2} \right) + f_{i,j}^{(n)}
\end{aligned}
\tag{4.10}
$$

Similarly, we can get the combination equations when $v_x > 0, v_y < 0$ and $v_x < 0, v_y > 0$. For solving this partial differential equation, the Courant-Friedrichs-Lewy (CFL) condition is necessary for stability, that is,

$$
\Delta t \leq \frac{\min(\Delta x, \Delta y)}{||v||_{max}}
\tag{4.11}
$$

4.2.3 Upwind Finite Volume Method for Transport Simulations

The explicit Euler (EE) and Upwind Finite Volume Method (UFVM) can be used for solving a pure convection problem [25, 28, 29]. Let c be the concentration of the transport substance. In the simplest case, the transport equation takes the following form:

$$
\partial_t c + \nabla \cdot c\mathbf{u} = s, \quad \mathbf{x} \in \Omega, \quad t \in (0, T],
\tag{4.12}
$$

along with some boundary conditions and an initial condition

$$
c(\mathbf{x}, 0) = c_0(\mathbf{x}), \quad \mathbf{x} \in \Omega.
\tag{4.13}
$$

We discretize the time interval $[0, T]$ as $0 = t_0 < t_1 < \cdots < t_{n-1} < t_n < \cdots < t_N = T$, where $\Delta t_n := t_n - t_{n-1}$. Let $\mathcal{E}_h$ be a mesh for Ω, and assume a numerical velocity $\mathbf{u}_h$ on $\mathcal{E}_h$ is already obtained with continuous normal fluxes and mass conservative property. The

51

contaminant concentration could be approximated by element-wise constant $C_h^{(n)}$ on $\mathcal{E}_h$ at time t_n. For a typical element $E \in \mathcal{E}_h$, we have the equation:

$$\frac{C^{(n)} - C^{(n-1)}}{\Delta t_n} + \int_{\partial E} C_E^{(n-1)} \mathbf{u}_h \cdot \mathbf{n} = \int_E s$$

and hence a time-marching scheme

$$C^{(n)} = C^{(n-1)} - \Delta t_n\, C^{(n-1)} \int_{\partial E} \mathbf{u}_h \cdot \mathbf{n} + \Delta t_n \int_E s(\cdot, t_n). \tag{4.14}$$

On the right-hand side of the scheme, the 2nd term should be handled by UFVM according to the in-out fluxes related to the element E. For a rectangular element E with each edge e among the four sides (bottom, right, top, left), the sign of

$$\text{flux} = \int_e \mathbf{u}_h \cdot \mathbf{n} \tag{4.15}$$

will be checked. For instance, let e be the left edge. If the sign is negative, then the element has an in-flux and $C^{(n-1)}$ is calculated based on the value associated with the left-neighbor element. If the sign is non-negative, then the element has an out-flux and $C^{(n-1)}$ is calculated based on the value associated with the element itself.

To compute the contaminant concentration at a new time step, the numerical scheme uses the physical information of upwinding (convection direction). Therefore, the computation is mass-conservative and maintains the positivity of concentration.

4.2.4 Weak Galerkin Finite Element Method for Transport Simulations

The newly developed weak Galerkin (WG) FEMs [27, 29, 50] have advantages in solving partial differential equations, compared to traditional numerical methods. To solve numerically the unknown pressure in the Darcy equation, we discretize the domain Ω into rectangular meshes $\mathcal{E}_h$ and set pressure unknowns both in element interiors and the edges (known as constants in the lowest WG (Q_0, P_0, RT_0)), and specify their discrete weak gradients in the

Ravariat-Thomas $RT_{[0]}$ space. For each rectangular element E, we have $\dim(RT_{[0]}(E)) = 4$. The following four vector-valued functions

$$\mathbf{w}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \mathbf{w}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad \mathbf{w}_3 = \begin{bmatrix} X \\ 0 \end{bmatrix}, \quad \mathbf{w}_4 = \begin{bmatrix} 0 \\ Y \end{bmatrix},$$

span $RT_{[0]}(E)$, where $X = x - x_c, Y = y - y_c$. The Gram matrix of this basis is diagonal matrix, as below:

$$GM = \text{diag}\left(|E|, |E|, \frac{1}{12}(x_2 - x_1)^2 |E|, \frac{1}{12}(y_2 - y_1)^2 |E|\right).$$

Then we solve a linear system with Gram matrix as the coefficients, and get their discrete weak gradients in the above $RT_{[0]}(E)$ basis. The linear system can be represented as the following form:

$$\begin{bmatrix} \nabla_{w,d}\phi_0 \\ \nabla_{w,d}\phi_1 \\ \nabla_{w,d}\phi_2 \\ \nabla_{w,d}\phi_3 \\ \nabla_{w,d}\phi_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & \frac{-12}{(x_2-x_1)^2} & \frac{-12}{(y_2-y_1)^2} \\ 0 & \frac{-1}{y_2-y_1} & 0 & \frac{6}{(y_2-y_1)^2} \\ \frac{1}{x_2-x_1} & 0 & \frac{6}{(x_2-x_1)^2} & 0 \\ 0 & \frac{1}{y_2-y_1} & 0 & \frac{6}{(y_2-y_1)^2} \\ \frac{-1}{x_2-x_1} & 0 & \frac{6}{(x_2-x_1)^2} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \\ \mathbf{w}_3 \\ \mathbf{w}_4 \end{bmatrix}.$$

The discrete weak gradients are used for the approximation of classical gradients in the Darcy equation. We use the FEM scheme to seek $p_h = \{p_h^\circ, p_h^\partial\} \in S_h(l, m)$ such that $p_h^\partial|_{\Gamma^D} = Q_h^\partial p_D$, and

$$\mathcal{A}_h(p_h, q) = \mathcal{F}(q), \qquad \forall q = \{q^\circ, q^\partial\} \in S_h^0(l, m), \tag{4.16}$$

$$\mathcal{A}_h(p_h, q) := \sum_{E \in \mathcal{E}_h} \int_E \mathbf{K} \nabla_{w,n} p_h \cdot \nabla_{w,n} q, \qquad \mathcal{F}(q) := \sum_{E \in \mathcal{E}_h} \int_E f q^\circ - \sum_{\gamma \in \Gamma_h^N} \int_\gamma u_N q^\circ. \tag{4.17}$$

The Darcy velocity could be computed based on the solved pressure p_h and the following framework:

$$\mathbf{u}_h = R_h(-\mathbf{K}\nabla_{w,n}p_h),\qquad(4.18)$$

where R_h is the local L^2-projection onto $V(E,n)$.

Based on the assumption that Ω is rectangular with rectangular mesh $\mathcal{E}_h$, we can solve the transport equation by using a similar idea. Let $C_h^{(n)}(n \geq 1)$ be an approximation at discrete time t_m, then there holds for $n \geq 1$,

$$
\begin{aligned}
&\sum_{E\in\mathcal{E}_h}(C_h^{(n)},w)_E - \Delta t \sum_{E\in\mathcal{E}_h}(\mathbf{u}_h C_h^{(n)},\nabla_{w,d}w)_E + \Delta t\, D \sum_{E\in\mathcal{E}_h}(\nabla_{w,d}C_h^{(n)},\nabla_{w,d}w)_E \\
&= \sum_{E\in\mathcal{E}_h}(C_h^{(n-1)},w)_E + \Delta t \sum_{E\in\mathcal{E}_h}(f,w)_E.
\end{aligned}
\qquad(4.19)
$$

On the left side of the scheme, the 2nd term represents the interaction of the flow (Darcy velocity) and the concentration (discrete weak) gradient. On the right side, the 1st term represents the mass at previous time moment, and 2nd term depicts the source/sink contribution during the time period $[t_{n-1}, t_n]$. Then we can obtain an initial approximantion $C_h^{(0)}$ through the local L_2-projection of $c_0(x,y)$ into the WG finite element space. This scheme has two nice properties: i) On each element E, $C_h^{(n)}|_{E^\circ}$ represents the cell average of concentration; ii) It is locally and hence globally conservative.

4.3   Coupling of Groundwater Monitoring and Transport Simulations

We will simulate a groundwater contamination event resulting from the fracking fluid leakage. Chloride (Cl) is abundant inorganic chemical that transports along with the groundwater flow and passes through the monitor station. Installed sensors collect the groundwater quality information in real-time and send them into a system to perform anomaly detection. MSET and 1-SVM should get response to the new coming data, estimate their states and finally raise alarm if anomaly appears. This scenario gets further explanation by Figure 4.2. The purpose of this coupling is to validate the effectiveness of anomaly detection methods.
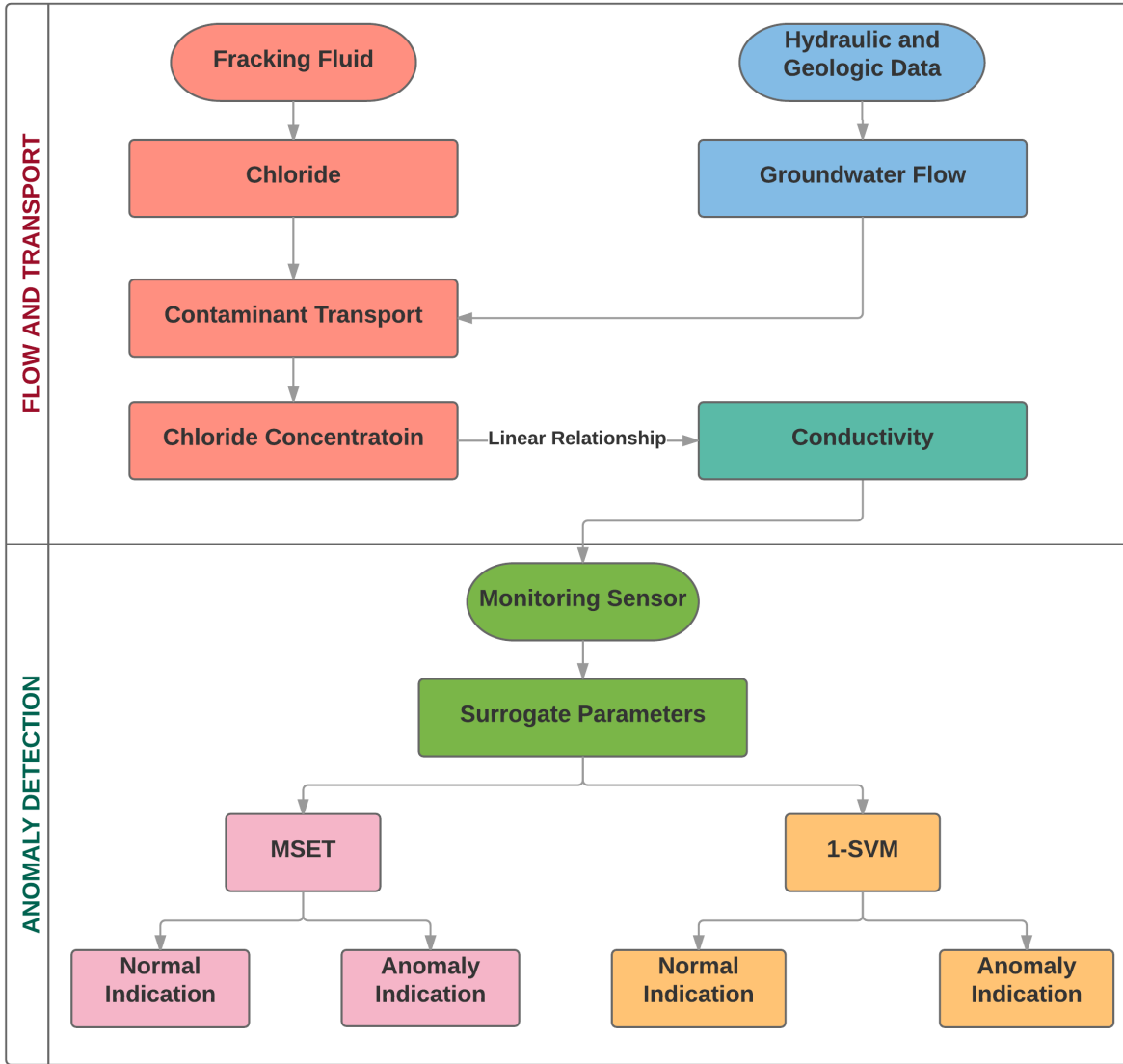
Figure 4.2: Coupling of transport simulation and groundwater monitoring

Normally, the concentration of chloride (Cl) in groundwater is around 150 mg/L, but the concentration in fracking fluid is around $20,000$ mg/L. The concentration directly influences the groundwater conductivity $Cond$. A linear relationship exists between them [36], i.e., $Cl = k \cdot Cond$, where k is a scalar coefficient. The diffusion coefficient of chloride in groundwater is very small, around $2.03\mathrm{e}{-11}$ m^2/s [42]). Therefore, we ignore the diffusion here and just consider a convection problem.

4.4   Gilcrest/LaSalle Project

4.4.1 Dataset Discription

To simulate the contaminant transport in groundwater, we select a study area with sufficient hydraulic and geologic data, see Figure 4.3, located in the south rural area of Greeley city, CO. This area is one part of the Wattenberg filed with activate oil and gas production operations and covered by the groundwater monitoring system - CWW. Therefore, we can couple the transport simulation with groundwater monitoring process.
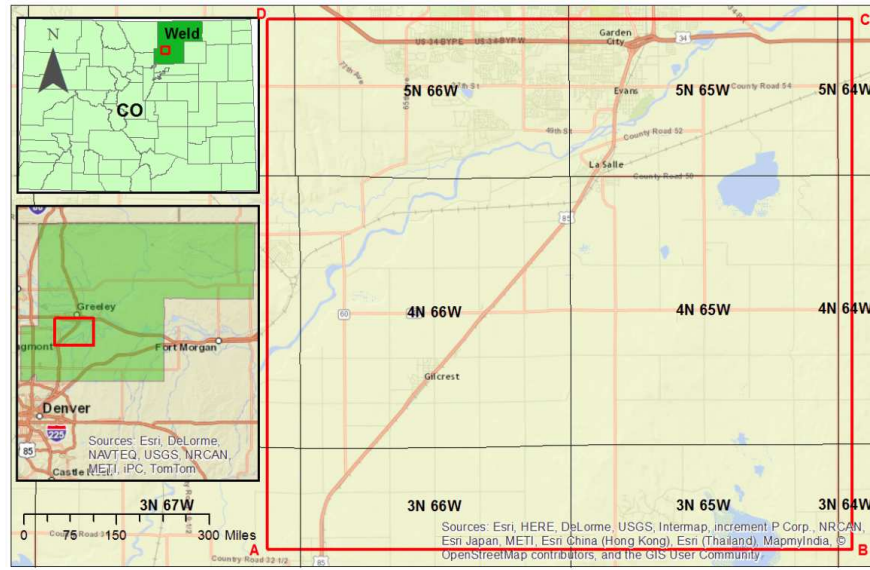
Figure 4.3: The rectangular area for study, 21,000 meters (13.048 miles) from A to B and 19,000 meters (11.806 miles) from A to D. The coordinates of A, B, C and D are listed below.

Table 4.1: The corner coordinates of the study area

Corner	Longitude	Latitude	X (meters)	Y (meters)
A	-104.844	40.228	513296.815	4453116.455
B	-104.597	40.228	534296.815	4453116.455
C	-104.597	40.399	534296.815	4472116.455
D	-104.844	40.399	513296.815	4472116.455

Projected Coordinate System: NAD_1983_UTM_Zone_13N

56

According to Colorado Oil & Gas Conservation Commission (COGCC), nearly 3500 producing wells were built in the study area by 2014. Two monitoring stations named Gilcrest and Platteville were built among the thousands of producing wells, see Figure 4.4.
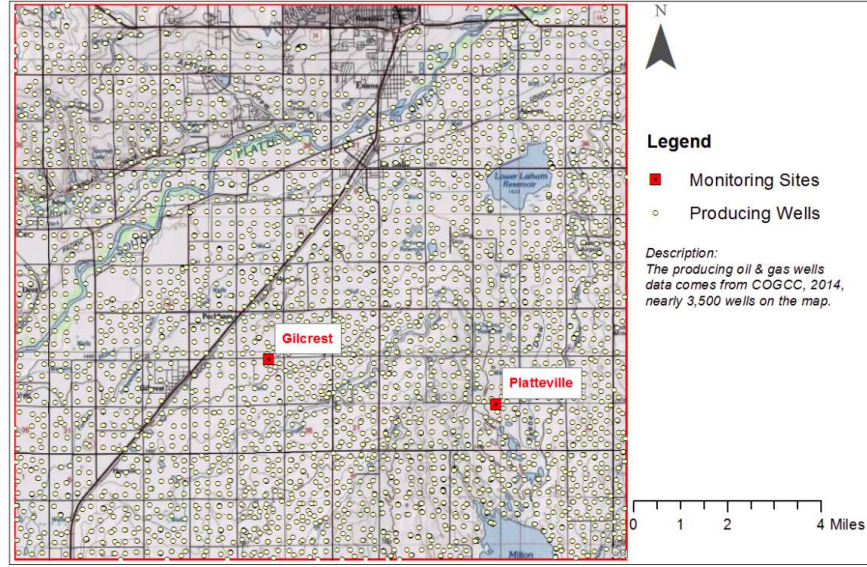
Figure 4.4: The oil-gas producing wells and two monitoring stations in the study area

To calculate the Darcy velocity, the groundwater elevation from Colorado Geological Survey[1] and the horizontal hydraulic conductivity from the SPDSS [11] are discretized by 19×21 cells, each cell is $1000m \times 1000m$. as shown in Figure 4.5.
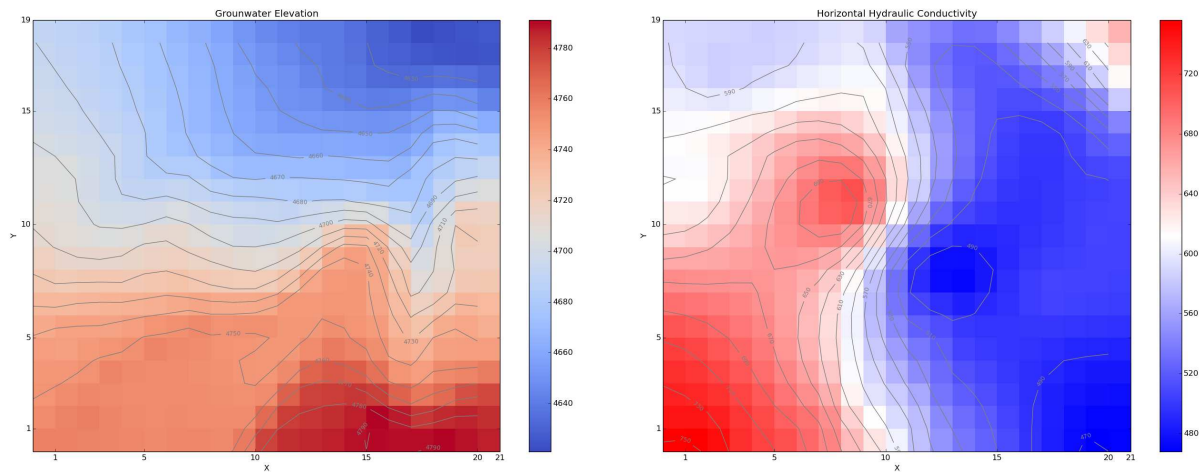


Figure 4.5: *Left:* Groundwater elevation, unit: ft, $1m = 3.281ft$; *Right:* Horizontal hydraulic conductivity, unit: ft/day. Cells with high values are rendered with red, and cells with low values are rendered with blue.

4.4.2 Flow and Transport

Based on groundwater elevation and horizontal hydraulic conductivity data, we can calculate Darcy velocity $\mathbf{u} = (-K\frac{\partial h_x}{\partial l_x}, -K\frac{\partial h_y}{\partial l_y})$ which has a close relationship with the fluid velocity $\mathbf{v}$, i.e.,

$$\mathbf{v} = \frac{\mathbf{u}}{\rho} \tag{4.20}$$

where ρ is the effective porosity of the medium. In this area, the medium (named the Laramie-Fox Hills aquifer) has porosity range $[0.21, 0.44]$ and mean 0.32 according to the USGS report [41]. Here, we assume that the porosity in this relatively small area is homogenous and use the mean value $\rho = 0.32$. The results of Darcy velocity and fluid velocity are shown in Figure 4.6.
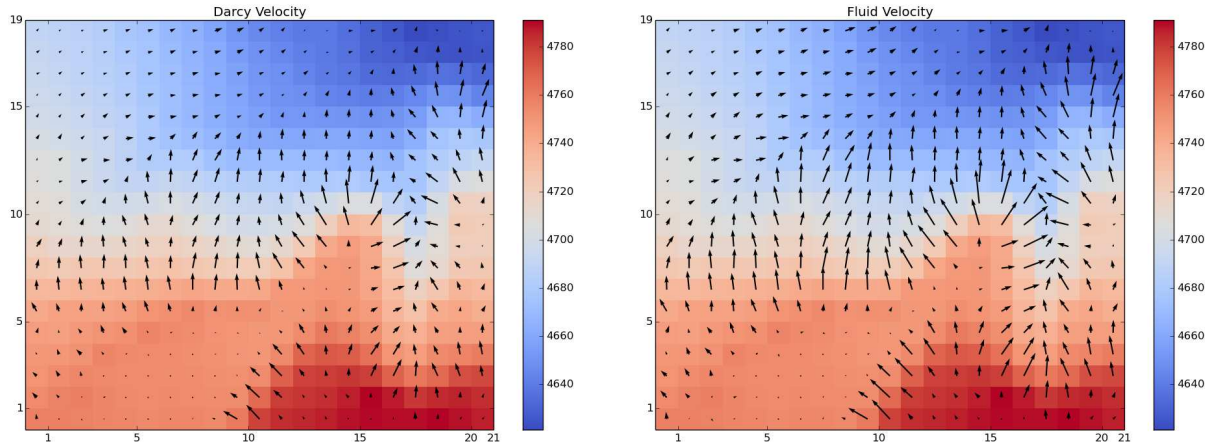
Figure 4.6: *Left:* Darcy velocity; *Right:* Fluid velocity. The background color map is rendered based on the groundwater elevation, where red color represents high levels and the blue represents low levels. The arrow goes along the terrain from high to low.

Then the solved fluid velocity can be input into the convection equation to simulate the chloride transport. Given two contamination sources at time t_0, and the concentrations of chloride respectively are $c_0^1(7,7) = 20,000 \ mg/L, \quad c_0^2(18,4) = 20,000 \ mg/L$. The diffusion effect of chloride is ignored, i.e. $D = 0$. And, set $f = 0$, and time step $\Delta t = 3600$ s (1 hour). Based on the initial conditions, we simulate the chloride's transport process within a period of time. The results are shown in Figure 4.7.

(a) initial conditions

(b) after 1 year

(c) after 2 years

(d) after 3 years

(e) after 4 years

(f) after 5 years
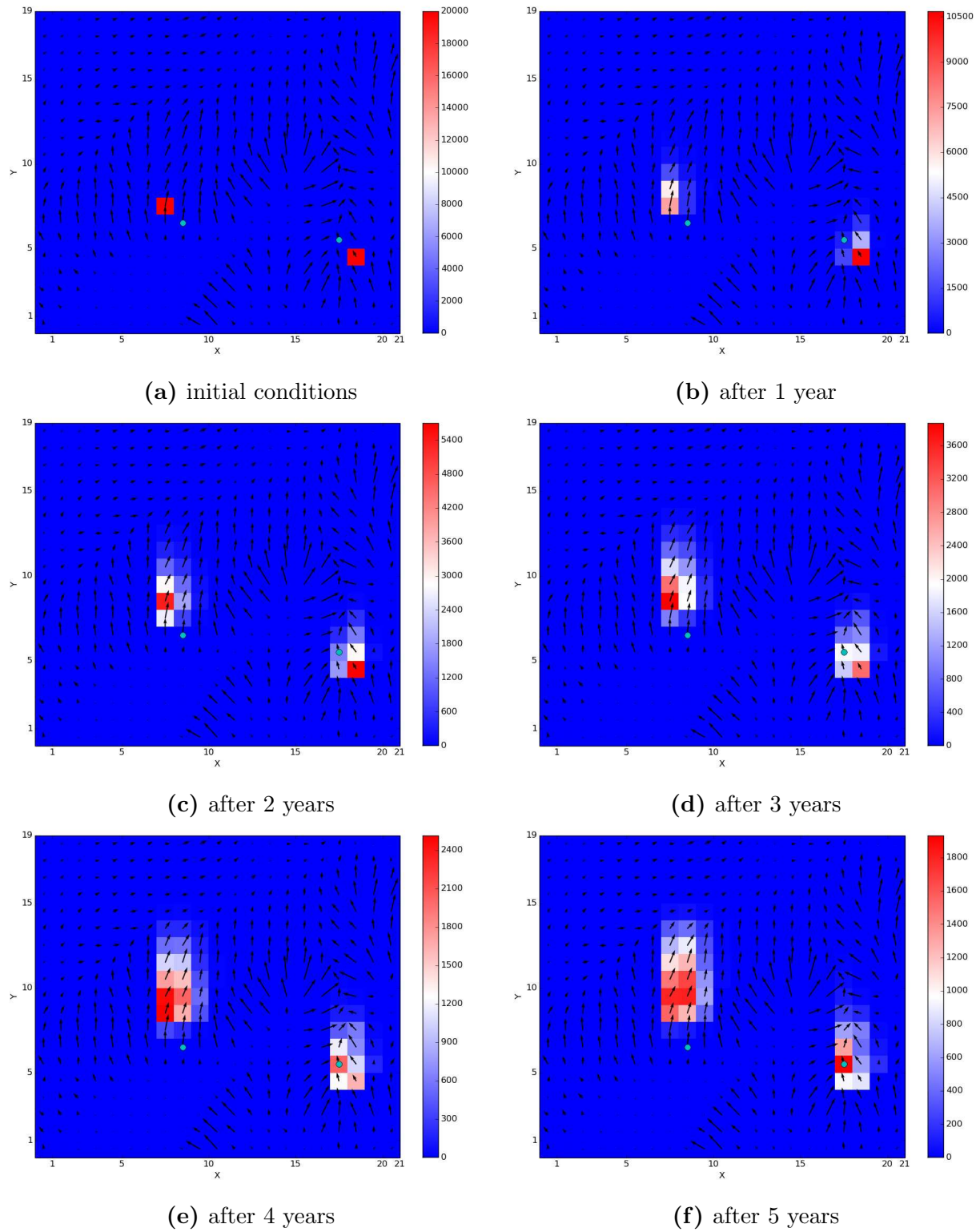
Figure 4.7: The change of contaminant concentration with time. Red color represents high concentration, and blue color represents low concentration. Two green points represent the location of monitor stations.

4.4.3 Groundwater Monitoring

From the transport simulation results, we can see that the contaminant chloride transport with groundwater flow, the contaminant concentration reduces with time going. After 5 years, the concentration drops to $1,900$ mg/L from $20,000$ mg/L. To extract a series of chloride concentration over time at where the monitor station - Platteville locates, we get the result shown in Figure 4.8. The concentration increases at the first 4 years (48 months) and reaches a peak, then it will decrease in the following time, also see Table 4.2.
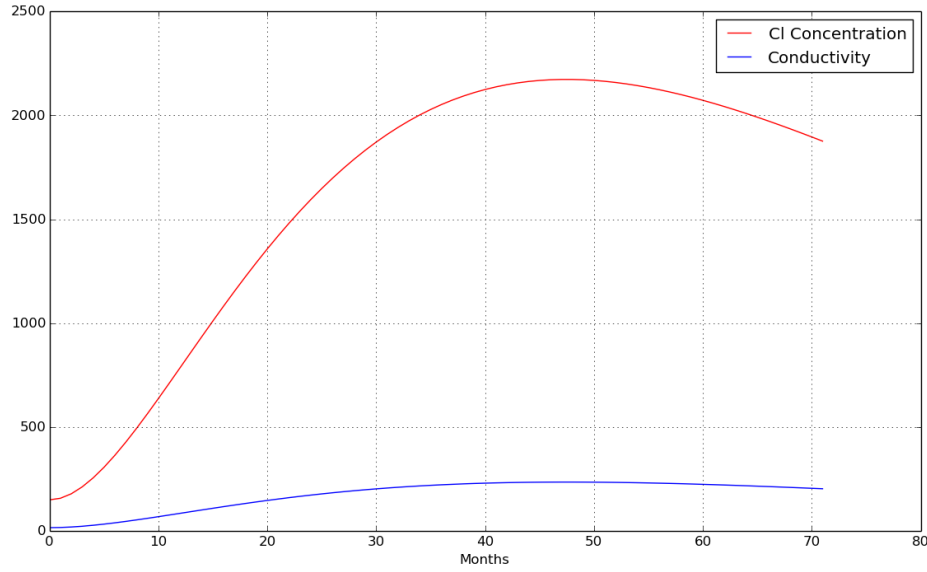
Figure 4.8: Chloride concentration and groundwater conductivity change at Platteville over time.

Table 4.2: Concentration of chloride and corresponding groundwater conductivity over time

Months	$t=0$	$t=1$	$t=2$	$t=4$	$t=6$	$t=12$	$t=24$	$t=48$	$t=60$	$t=72$
Cl	21	28	50	126	235	732	1518	2041	1928	1705
$Cond$	2.3	3.1	5.4	13.7	25.5	79.4	164.7	221.5	209.1	184.9

Suppose the relationship between groundwater conductivity and the concentration of chloride is linear, $Cl = k*Cond$ with $k = 9.222$ [36]. The change of groundwater conductivity is also shown in Figure 4.8. During the transport, anomaly detection system will take the simulated conductivity and other surrogates as input to validate the performance of MSET

and 1-SVM. Assume that the other surrogates do not affected by this event and keep as healthy constants. The baseline data is used to estimate the state of simulated observations. Baseline data describes the groundwater quality in specific area and used for water quality control. 720 samples are selected from the baseline dataset which cover the whole healthy ranges. MSET and 1-SVM learn experience from the sample instances and get response to the change after a period of time. The results of coupling of transport simulations and groundwater monitoring are shown in 4.9.
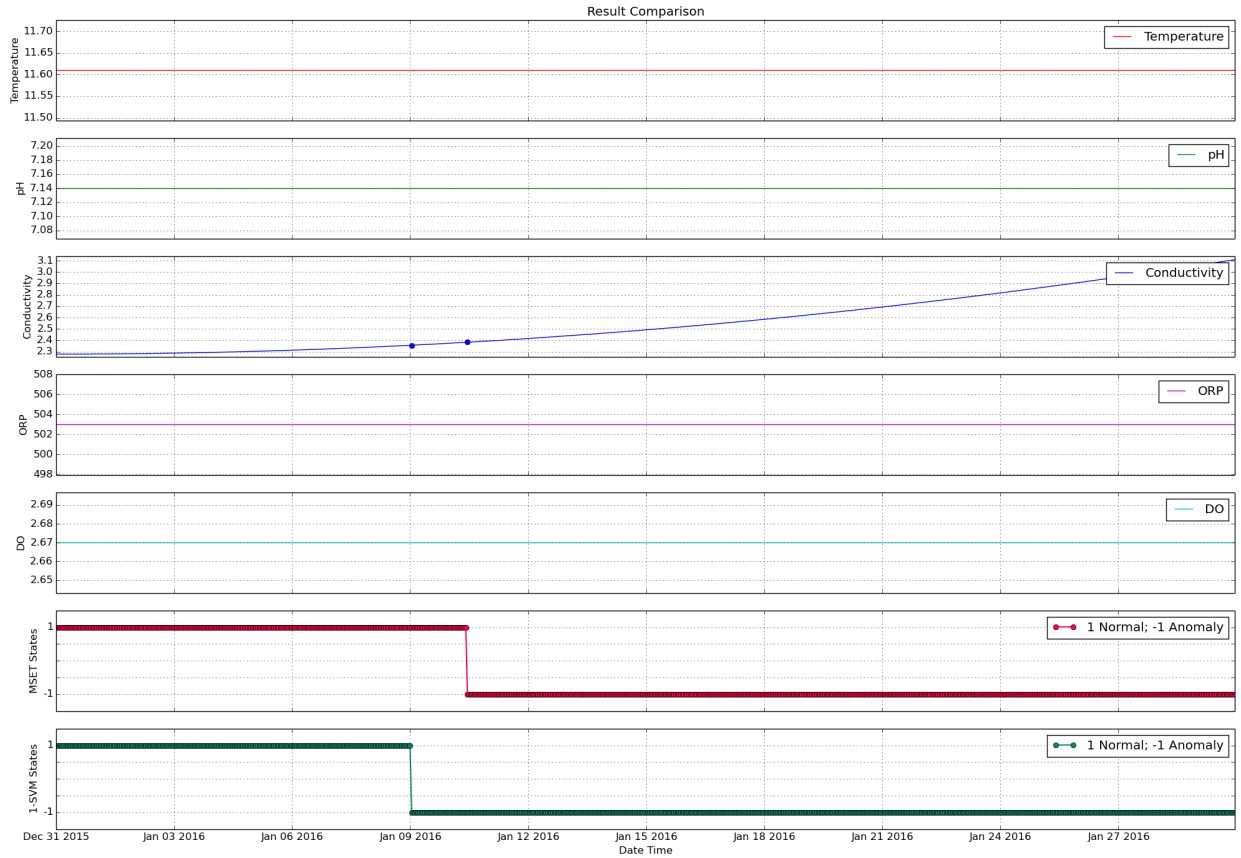
Figure 4.9: From top to bottom, subplot 1-5 shows the simulated surrogate parameters; subplot 6-7 show the estimated states of MSET and 1-SVM respectively.

We suppose the event happens at 2015-12-31 00:00:00. The monitoring result shows that MSET get response to the raising level of conductivity after 11 days, when conductivity is higher than 3.82 S/m. For 1-SVM, it raises alarm after 9 days, when conductivity is higher than 3.56 S/m. 1-SVM takes a little shorter time to respond the change of groundwater

quality, but the difference is not obvious. The distance between the contaminant source and monitor station predominantes the time when the system gets response. What's more, the conductivity level at which the anomaly detection algorithms get response highly depends on the range of baseline data. Mathematically, these two methods are sensitive to the change of groundwater conductivity, and they can capture the contaminant event when the chloride concentration reaches up to some level.

4.5 Discussion

In this section, we simulated an groundwater event with contaminant transport by using FDM and used it to test the real-time responsiveness of MSET and 1-SVM to groundwater contamination. Compared with FEM, FDM are easy to implement on rectangular domain, and also generates reliable Darcy velocity and transport result in this application. The coupling of transport simulation and groundwater monitoring provide sound evidence that MSET and 1-SVM have the ability of detecting groundwater event in real-time.

But the simulated model is still idealistic at some extent, and need to be improved further. For instance, we assume that the porosity in this area is homogenous, but it's not in the real environment. Thus, more detailed hydraulic data should be collected to improve this model. And, the cell size ($1000m \times 1000m$) is relatively large, the groundwater elevation and horizontal hydraulic conductivity are not homogenous, too, but varies a lot within it, especially for the shallow groundwater. To improve the accuracy of this model, discretized study area with smaller cells should be tried.

However, even though there are some shortcomings in this model, the trial of this experiment still generates much useful experience when promoting the mathematical model into reality life, the sketched model could get improve with further studies. For example, WGFEMs might be used to enhance the performance.

CHAPTER 5

CONCLUDING REMARKS

5.1 Summary

In this study, we presented the background of groundwater anomaly detection in the oil and gas fields, the problems and challenges encountered, and the potential methods for exploration in the literature. Then we focused on two anomaly detection methods: MSET and 1-SVM. Their mathematical principles behind the equation were explained in detail at first, and then they were applied into the real world project Colorado Water Watch (CWW) and implemented through Python. CWW uses surrogate technique and gathers groundwater quality information hourly. To adapt this characteristic, we designed real-time anomaly detection frameworks based on MSET and 1-SVM respectively. The test results showed that the performance of the two methods were good, especially for MSET. First, both of them had good responses to various typical data patterns and generated anomaly indications in real-time. Second, they all had good performance when encountering noisy data, their classification accuracies were above 96%. They satisfied our initial requirements. We can safely conclude that they are suitable for groundwater monitoring.

To ensure the solidness of the two methods, we then simulated a groundwater event, i.e. fracking fluid contamination, and couples it with CWW groundwater monitoring. The simulation was based on the Darcy equation and the convection-diffusion equation. We presented three numerical methods: finite difference method (FDM), finite volume method (FVM) and finite element method (FEM), for solving these two partial differential equations. We adopted FDM for the rectangular study area here and implemented by using Python, too. On the one hand, Darcy velocity on this area were calculated based on field geologic and hydraulic datasets, which was consistent with the real world terrain. On the other hand, contaminant chloride transport in groundwater were explored, the numerical results showed the change of chloride concentration along with groundwater flow over time. Based

63

on the linear relationship between the concentration of chloride and surrogate parameter conductivity, we then combined the anomaly detection methods developed previously with the transport simulation here. The result indicated the sensitivity of MSET and 1-SVM to this simulated groundwater contaminant event.

This research bring MSET and 1-SVM into this area. The complexity and uncertainty of underground environment make the groundwater anomaly detection problem very complicated, which leads to the comprehensiveness of this research. Diverse modeling techniques from mathematics, statistics and machine learning are explored to resolve this issue. The originality of this study is ensured by its application to the groundwater monitoring in oil and gas fields. Further, the implementation of the methods in Python makes our work accessible. The code for MSET, 1-SVM and transport simulations can be easily transfered and extended to engineering practices.

5.2   Further Work

Still, we can go further in the area of groundwater anomaly detection. Based on the work presented in this thesis, some interesting directions for further study are outlined.

- To apply MSET and 1-SVM based anomaly detection methods to multiple monitoring stations and test their robustness through various data obtained from various environment, which can ensure the engineering quality in practical implementation.

- To develop predictive mathematical models, which can enable the monitoring system to forecast the trend of groundwater quality, and provide guidance for human being activities in advance. Right now, MSET and 1-SVM can only watch the current state, having no capability of prediction.

- To further explore FEMs for groundwater event simulations. FEMs are capable of incorporating intrinsic geometrical properties into computational fluid dynamics, since they are more flexible for complicated domains and boundary conditions.

All further studies can make the groundwater anomaly detection system more powerful.

BIBLIOGRAPHY

[1] P. E. Barkmann, A. Horn, A. Moore, J. Pike, and W. Curtiss. *Gilcrest/LaSalle Pilot Project Hydrogeologic Characterization Report.* Colorado Geological Survey, Golden, Colorado, Sept 2014.

[2] J. Bear, M. S. Beljin, and R. R. Ross. *Fundamentals of ground-water modeling.* Superfund Technology Support Center for Ground Water, Robert S. Kerr Environmental Research Laboratory, 1992.

[3] A. Ben-Israel and T. N. Greville. *Generalized inverses: theory and applications*, volume 15. Springer Science & Business Media, 2003.

[4] C. L. Black, R. E. Uhrig, and J. W. Hines. System modeling and instrument calibration verification with a nonlinear state estimation technique. In *Proceedings of the Maintenance and Reliability Conference.* Knoxville, TN, May, 1998.

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

[6] C.-C. Chang and C.-J. Lin. Libsvm: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

[7] X. Chen and Y. Zhan. Multi-scale anomaly detection algorithm based on infrequent pattern of time series. *Journal of Computational and Applied Mathematics*, 214(1):227–237, 2008.

[8] S. Cheng and M. Pecht. Multivariate state estimation technique for remaining useful life prediction of electronic products. *Parameters*, 1:x2, 2007.

[9] Y. Chetouani. A sequential probability ratio test (sprt) to detect changes and process safety monitoring. *Process Safety and Environmental Protection*, 92(3):206 – 214, 2014.

[10] `http://waterwatch.colostate.edu/`. Accessed: 2016-08-12.

[11] `http://cdss.state.co.us/GIS/Pages/Division1SouthPlatte.aspx`. Accessed: 2016-08-10.

[12] V. N. Dao and V. Vemuri. A performance comparison of different back propagation neural networks methods in computer network intrusion detection. *Differential equations and dynamical systems*, 10(1&2):201–214, 2002.

[13] D. C. DiGiulio, R. T. Wilkin, C. Miller, and G. Oberley. *Investigation of ground water contamination near Pavillion, Wyoming.* Office of Research and Development, National Risk Management Research Laboratory, 2011.

[14] J. E. Drew, H. D. Leslie, P. N. Armstrong, G. Michard, et al. Automated microseismic event detection and location by continuous spatial mapping. In *SPE Annual Technical Conference and Exhibition.* Society of Petroleum Engineers, 2005.

[15] E. Eskin, A. Arnold, M. Prerau, L. Portnoy, and S. Stolfo. A geometric framework for unsupervised anomaly detection. In *Applications of data mining in computer security*, pages 77–101. Springer, 2002.

[16] H. Foroughi, A. Rezvanian, and A. Paziraee. Robust fall detection using human shape and multi-class support vector machine. In *Computer Vision, Graphics & Image Processing, 2008. ICVGIP'08. Sixth Indian Conference on*, pages 413–420. IEEE, 2008.

[17] R. Fujimaki, T. Yairi, and K. Machida. An approach to spacecraft anomaly detection problem using kernel feature space. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 401–410. ACM, 2005.

[18] B. J. Galluzzo. A finite-difference based approach to solving the subsurface fluid flow equation in heterogeneous media. 2011.

[19] K. Gross et al. Proactive fault monitoring in enterprise servers. In *IEEE international multiconference in computer science & computer engineering*, 2005.

[20] K. Gross, R. Singer, S. Wegerich, J. Herzog, R. VanAlstine, and F. Bockhorst. Application of a model-based fault detection system to nuclear plant signals. In *Proceedings of 9th International Conference on Intelligent Systems Application to Power System, Seoul, Korea*, volume 610, page 212218, 1997.

[21] K. C. Gross and W. Lu. Early detection of signal and process anomalies in enterprise computing systems. In *Proceedings of the 2002 International Conference on Machine Learning and Applications*, pages 204 –210, 2002.

[22] S.-J. Han and S.-B. Cho. Evolutionary neural networks for anomaly detection based on the behavior of a program. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 36(3):559–570, 2005.

[23] 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*, volume 32, page 503. Citeseer, 2007.

[24] J. W. Hines and A. Usynin. Mset performance optimization through regularization. *Nuclear Engineering and Technology*, 37(2):177, 2005.

[25] D. Kuzmin. A guide to numerical methods for transport equations. *University Erlangen-Nuremberg*, 2010.

[26] H. Li, J.-H. Son, and K. H. Carlson. Concurrence of aqueous and gas phase contamination of groundwater in the wattenberg oil and gas field of northern colorado. *Water research*, 88:458–466, 2016.

[27] G. Lin, J. Liu, and F. Sadre-Marandi. A comparative study on the weak galerkin,

discontinuous galerkin, and mixed finite element methods. *Journal of Computational and Applied Mathematics*, 273:346–362, 2015.

[28] J. Liu. DarcyLite: A matlab toolbox for simulations of flow and transport in porous media. `http://www.math.colostate.edu/~liu/code.html`. Accessed: 2016-08-31.

[29] J. Liu, F. Sadre-Marandi, and Z. Wang. DarcyLite: A matlab toolbox for darcy flow computation. *Procedia Computer Science*, 80:1301–1312, 2016.

[30] L. Lopez. Advanced electronic prognostics through system telemetry and pattern recognition methods. *Microelectronics Reliability*, 47(12):1865–1873, 2007.

[31] M. Mehrjoo, N. Khaji, H. Moharrami, and A. Bahreininejad. Damage detection of truss bridge joints using artificial neural networks. *Expert Systems with Applications*, 35(3):1122–1131, 2008.

[32] A. Miron. *A wavelet approach for development and application of a stochastic parameter simulation system.* PhD thesis, University of Cincinnati, 2001.

[33] A. P. Muniyandi, R. Rajeswari, and R. Rajaram. Network anomaly detection by cascading k-means clustering and c4. 5 decision tree algorithm. *Procedia Engineering*, 30:174–182, 2012.

[34] E. Ngai, Y. Hu, Y. Wong, Y. Chen, and X. Sun. The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature. *Decision Support Systems*, 50(3):559–569, 2011.

[35] A. Patcha and J.-M. Park. An overview of anomaly detection techniques: Existing solutions and latest technological trends. *Computer networks*, 51(12):3448–3470, 2007.

[36] H. Peinado-Guevara, C. Green-Ruíz, J. Herrera-Barrientos, O. Escolero-Fuentes, O. Delgado-Rodríguez, S. Belmonte-Jiménez, and M. Ladrón de Guevara. Relationship between chloride concentration and electrical conductivity in groundwater and its

estimation from vertical electrical soundings (vess) in guasave, sinaloa, mexico. *Ciencia e investigación agraria*, 39(1):229–239, 2012.

[37] J. Peng, L. Fan, W. Xiao, and J. Tang. Anomaly monitoring method for key components of satellite. *The Scientific World Journal*, 2014.

[38] A. D. Pietersma. Feature space learning in support vector machines through dual objective optimization. *Order*, 501:3295, 2010.

[39] Y. Rajabzadeh, A. H. Rezaie, and H. Amindavar. A dynamic modeling approach for anomaly detection using stochastic differential equations. *Digital Signal Processing*, 54:1–11, 2016.

[40] R. Rastogi, Z. Khan, and M. Khan. Network anomalies detection using statistical technique: A chi-square approach. *nternational Journal of Computer Science Issues*, 9:515–522, 2012.

[41] S. G. Robson. Bedrock aquifers in the denver basin, colorado; a quantitative water-resources appraisal. Technical report, US Geological Survey, 1987.

[42] E. Samson, J. Marchand, and K. Snyder. Calculation of ionic diffusion coefficients on the basis of migration test results. *Materials and Structures*, 36(3):156–165, 2003.

[43] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson. Estimating the support of a high-dimensional distribution. *Neural computation*, 13(7):1443–1471, 2001.

[44] B. Schölkopf, R. C. Williamson, A. J. Smola, J. Shawe-Taylor, J. C. Platt, et al. Support vector method for novelty detection. In *NIPS*, volume 12, pages 582–588. Citeseer, 1999.

[45] R. M. Singer, K. C. Gross, J. P. Herzog, R. W. King, and S. Wegerich. Model-based nuclear power plant monitoring and fault detection: Theoretical foundations. Technical report, Argonne National Lab., IL (United States), 1997.

[46] J.-H. Son and K. H. Carlson. Real-time surrogate analysis for potential oil and gas contamination of drinking water resources. *Applied Water Science*, 5(3):283–289, 2015.

[47] A. Vengosh, N. Warner, R. Jackson, and T. Darrah. The effects of shale gas exploration and hydraulic fracturing on the quality of water resources in the united states. *Procedia Earth and Planetary Science*, 7:863–866, 2013.

[48] R. D. Vidic, S. L. Brantley, J. M. Vandenbossche, D. Yoxtheimer, and J. D. Abad. Impact of shale gas development on regional water quality. *Science*, 340(6134):1235009, 2013.

[49] A. Wald. *Sequential Analysis*. Dover, New York, 1947.

[50] J. Wang and X. Ye. A weak galerkin finite element method for second-order elliptic problems. *Journal of Computational and Applied Mathematics*, 241:103–115, 2013.

[51] K. Wang, J. Thompson, C. Peterson, and M. Kirby. Identity maps and their extensions on parameter spaces: applications to anomaly detection in video. In *Science and Information Conference (SAI), 2015*, pages 345–351. IEEE, 2015.

[52] M. Zhao and V. Saligrama. Anomaly detection with score functions based on nearest neighbor graphs. In *Advances in neural information processing systems*, pages 2250–2258, 2009.