

Anomaly detection 의 이해 및 사례 연구

작성자: 정용민

작성일: 2016 10.13-19

Table of Contents

1. Anomaly Detection 개요
 - 1.1 Anomaly 정의
 - 1.2 Approach on Anomaly Detection: 목적, 입력데이터의 특성, 이상값의 종류
 2. 사례를 통한 Point Anomaly 의 분석 방법 및 이해
 3. 사례를 통한 Contextual Anomaly 의 분석 방법 및 이해
 - 3.1 시계열 데이터의 이해
 - 3.1.1 시계열 데이터 구성요소
 - 3.1.2 분석 접근에 따른 시계열 확률 과정 분류
 - 3.1.3 정상과정 모형과 비정상 과정 모형 비교
 - 3.2. 시계열 데이터에서의 이상탐지
 - 3.2.1 Gaussian Process Regression
 - 3.2.2. Seasonal Hybrid ESD
 - A. 이상탐지 기법 개요
 - B. 관련 통계 이론: MAD, t-distribution, Generalized ESD, STL
 - C. Python 을 이용한 S-H-ESD 알고리즘 구현
 4. 성능평가 및 검증
-

1. Anomaly Detection 개요

1.1 Anomaly 정의:

기존 관측과는 상이하여 다른 매커니즘에 의해 생성되었다고 판단할만한 관측값. (An anomaly is an observation that deviates so much from other observations so as to arouse suspicion that it is was generated by different mechanism.)

1.2 Approach on Anomaly Detection:

Anomaly Detection 은 그 자체가 알고리즘이라기 보다는 '목표하는바/기대하는 결과'에 해당하며, 여러 알고리즘과 분석론을 활용한 '분석 application' 이라고 볼 수 있다. 따라서, 해결하고자 하는 문제의 목적과 컨텍스트에 따라 anomaly detection 하는 방법은 상이 할 수 있음에 유의해야 한다.

이상탐지의 문제 특징(Problem Characteristics) 파악 및 구체적 분석방법론 수립시 고려할 사항들은 아래와 같다. (목적, 데이터 특성, 이상값 종류)

목적(Objectives)

- 기회 탐지 Chance discovery (Positive anomaly)
- 오류 탐지 Fault Discovery (Negative Anomaly)
- 새로움의 탐지 Novelty Detection
- 노이즈 제거 Noise Removal

입력 데이터의 특성(Nature of input data)

- 시계열 Time-Series(sequential) vs Static
- 단변량/다변량 Univariate vs Multivariate
- 데이터 타입 Data Type (Binary /Categorical /Continuous /Hybrid)
- 상호의존적/독립적 Relational vs Independent
- (기존 룰의 적용이 가능할 만큼) 잘 알려진/알려져 있지 않은
Well-known or not (rule existing or not)

이상값의 종류(Type of anomaly)

이상값의 종류는 학계나 업계 각기의 시각에 따라 달라지므로, 엄밀히 규정짓지 못한다. 본 보고서에서는 미네소타 대학 조사(Varun Chandola, <Anomaly Detection: A Survey>, 2009)에서 제시한 anomaly detection taxonomy 에 기반하여 이상값을 분류짓는다.

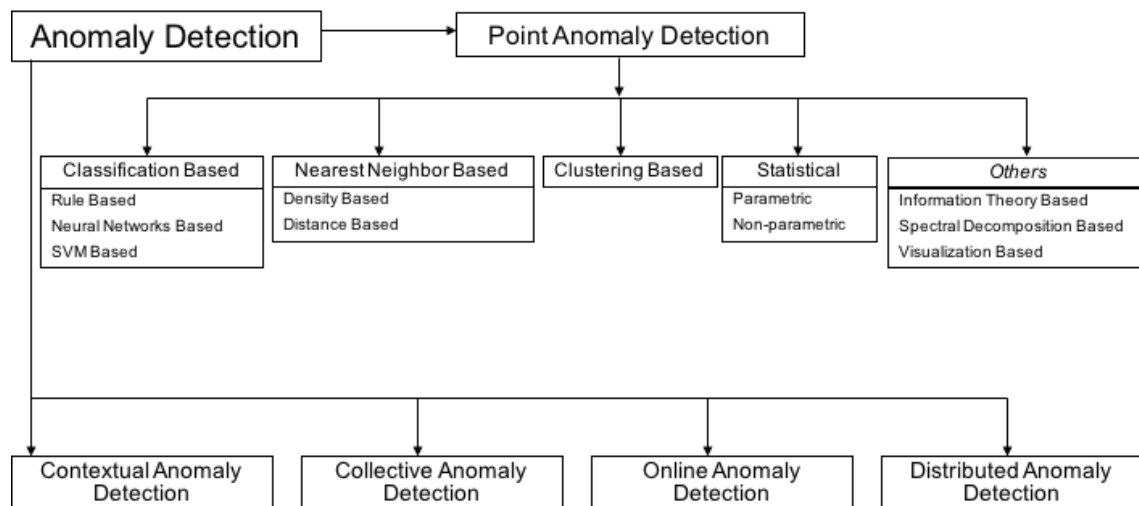


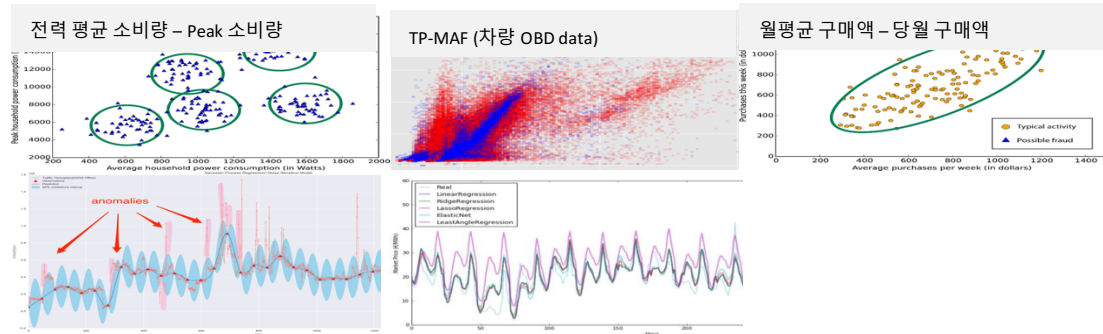
Figure 1: Taxonomy of Anomaly Detection

크게 Point Anomaly, Contextual Anomaly, Collective Anomaly, Online Anomaly, Distributed Anomaly 로 나뉘어 있으나, 본 보고서가 다루는 범위는 Point, Contextual, Collective Anomaly 에 한정 짓는다. 이 또한, Contextual 및 Collective Anomaly 는 시계열적 특성을 주안점으로 파악하여야 하는 유사성이 있으므로 Contextual Anomaly 로 통칭한다. 결론적으로, 본 보고서에서는 이상값을 정적인 데이터의 분포에 주안을 두는 Point Anomaly 와, 데이터의 시계열적 특성을 중점적으로 주안점을 주는 Contextual Anomaly 로 분류한다. 아래 표는 각 anomaly 종류 에 따른 특징 및 예시를 설명한다.

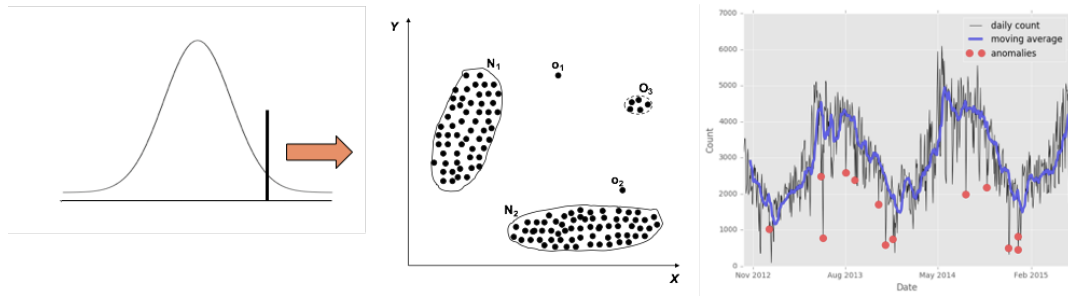
분류	특징	알고리즘		예시
Point Anomaly	<ul style="list-style-type: none"> 비교적 정적인 데이터(Stationary) 값의 분포 (distribution)에 주안 	Statistical	Gaussian distribution, Gaussian Mixture Model	구매이상 패턴탐지
		Clustering	K-means, DBSCAN, +w/ PCA	전력소비이상 패턴탐지
		Classification	SVM, Random Forest, Neural Net	차량이상 상태분석
Contextual Anomaly	<ul style="list-style-type: none"> 비교적 동적인 데이터 (Dynamic) 값의 변화 (Context)에 주안 	Statistical 실시간성 및 단순 탐지 적합	Gaussian Process Regression, Hidden Markov Model	NW Traffic 이상탐지
		Deep Learning Probabilistic forecasting 에 적합	RNN/LSTM	전력가격 이상탐지

Table 1: Anomaly 종류 및 관련 분석 알고리즘

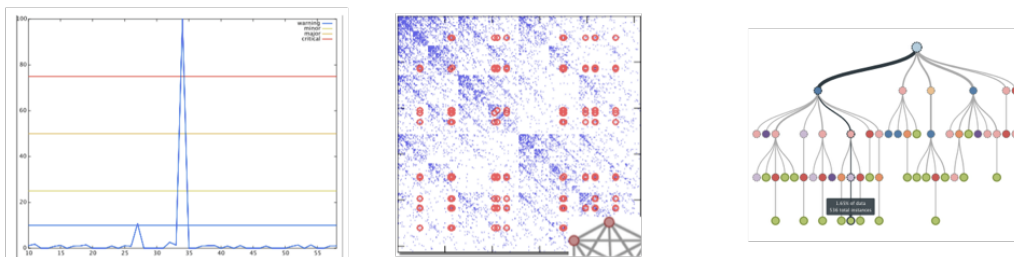
Table 1 적용 예시 (순서대로):



위와 같이, 이상탐지는 분석목적, 데이터의 특성, 이상값의 종류에 따라 그 분석 방법 및 해결책이 상이 할 수 있다. 그러나 이상탐지 분석을 위해서는 위 고려사항들을 기반으로 '정상상태의 정의'는 필수적으로 선행되어야 한다는 공통점이 있다. 아래는 각 도메인별 이상탐지 결과물들의 일면들을 통해 분석방법론이 상이 할 수 있음을 나타낸다.



NO SINGLE SCHEME !!



2. 사례를 통한 Point Anomaly의 분석 방법 및 이해:

Point Anomaly의 구체적 사례 및 분석 방법은, 소내 세미나에서도 공유되었던 KT Convergence 연구소 진행 프로젝트인 'InnoCAR 서비스 OBD 데이터 기반의 차량 오류 상태의 진단'의 세부사항을 주피터(Jupyter)기반으로 작성한 보고서/튜토리얼로 같음한다. 아래는 해당 보고서의 사례연구 결과물 요약이다.



Figure 2: Point Anomaly Detection 사례분석의 결과물 요약

3. 사례를 통한 Contextual Anomaly 의 분석 방법 및 이해:

본 절에서는 시계열적 특성을 중심으로 이상탐지를 하는 Contextual Anomaly 의 분석 방법론 및 구현방법, 결과물등을 간략히 소개한다. KT Convergence 연구소내 진행한 'GiGA Office 분야 트래픽 이상 탐지' 분석 사례를 통해 해당 주제들을 설명한다.

3.1 시계열 데이터의 이해

3.2 에서 다룰 시계열 데이터에서의 이상탐지기법은, 시계열 패턴의 3 요소 (trend, seasonal, Cyclic)와, 정상/비정상 과정 모형(stationary/non-stationary)의 선행적 이해가 요구되므로, 본 절에서 요약 전달한다.

3.1.1. 시계열 데이터 구성 요소:

시계열 데이터는 아래와 같은 요소들로 구성되어있다고 정의되며, 이를 요소들을 통해 시계열간의 특성차이가 발생한다.

- 시계열 데이터 특성
 - Magnitude
 - Width
 - Frequency
 - Direction
- 패턴요소 (figure 3. 참조)
 - 추세(Trend): 장기적으로 나타나는 변동 패턴
 - 계절성(Seasonal): 주,월,분기,반기 단위 등 이미 알려진 시간의 주기로 나타나는 패턴
 - 주기(Cyclic): 최소 2 년 단위로 나타나는 고정된 기간이 아닌 장기적인 변동
- 랜덤요소

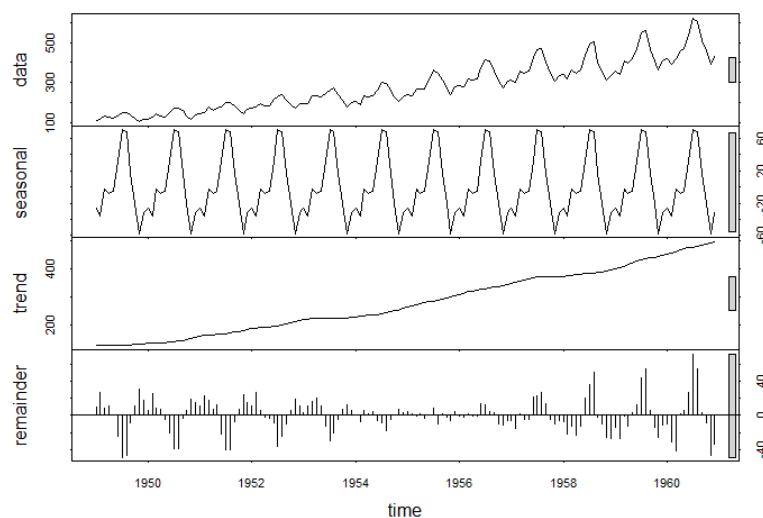


Figure 3: 특정 시계열 데이터 구성 요소 예; seasonal, trend, remainder (출처: AnalyticsVidhya.com)

3.1.2. 분석 접근에 따른 시계열 확률과정 분류:

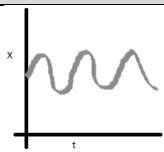
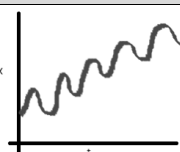
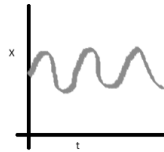
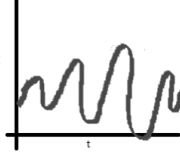
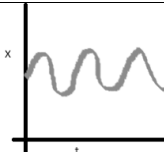

자체추정 self-projecting	
정상과정 모형 stationary process	정상과정 모형은 시간이 지나도 신호의 확률적 특성이 그대로 유지되는 확률 과정. 달리말해, 시간에 확률과정이 의존하지 않고 시간차이(lag)에 의존하는 모형. 대부분의 시계열 분석은 정상 과정 분석 방법에 토대를 둬.
	대표 모형: ARMA(Auto-regressive moving average), white noise
비정상과정 모형 non-stationary process	
	시간이 지나면서 기대값의 수준이나 분산이 커지는 등 시계열의 특성이 변화하는 과정
	대표 모형: ARIMA(Auto-regressive integrated moving average)
원인결과 추정 cause-and-effect	
	은닉 마르코브 모형
	기타

3.1.3. 정상과정 모형과 비정상 과정 모형 비교

일반적으로 시계열 분석의 용이성을 위해 아래와 같이 비정상과정 모형(y_t)에 따르는 시계열 데이터 또한 추정 가능한 결정론적 추세함수 ($f(t)$, trend) 와 확률 정상과정(x_t)의 합으로 가정하고 분석한다.

$$Y_t \sim f(t) + X_t$$

따라서 시계열 데이터 분석에서 정상과정 모형의 특성 및 분석방법들을 이해하는 것이 우선적으로 요구된다. 다음은 정상 시계열 모형과 비정상 시계열 모형의 특징 비교이다.

	정상 과정 모형 Stationary Series	비정상 과정 모형 non-Stationary series
시간추이에 따른 평균의 불변여부		
시간추이에 따른 분산의 불변여부		
두 시점간의 공분산		

3.2 시계열 데이터에서의 이상탐지 Contextual Anomaly Detection

3.2.1. Gaussian Process Regression (Gaussian Process 기반 회귀/확률적 추정)

가우시안 확률과정을 통한 회귀 분석은 가우시안 프로세스(mean=0)를 통해 생성된 Random 함수들이 확률적으로 존재 가능한 영역 추정하는 방식으로, 얻어진 회귀식/신뢰구간에서 확률적 추정이나 값을 예측하는 회귀분석에 활용할 수 있는데, 이를 기반으로 희귀한 관측을 탐지하는 이상탐지으로도 활용 가능하다. 가우시안 프로세스

에 대한 조금 더 이론적으로 구체적인 설명은 아래와 같다. (출처: C. E. Rasmussen & C. K. I. Williams, Gaussian Processes for Machine Learning, the MIT Press, 2006)

Gaussian Process 개요 및 적용 예:

가우시안 프로세스는 아래와 같은 평균(함수)과 공분산(함수)로 규정할 수 있는 확률과정을 따르는 확률변수들의 집합이다.(a collection of random variables)

$$f(x) \sim \mathcal{GP}(m(x), k(x, x'))$$
$$\text{where } m(x) = \mathbb{E}[f(x)]$$
$$\text{and } k(x, x') = \mathbb{E}[(f(x) - m(x))(f(x') - m(x'))]$$

본 확률과정에서 확률변수는 x 에 대응하는 함수값, $f(x)$ 라고 할 수 있다. 이 가우시안 프로세스를 함수공간관점(function-space view)에서 해석 하면, 가능한 입력값 x 가 가질 수 있는 함수 값/확률 사건이 $f(x)$ 일 확률을 대응시키는 확률 변수들로 이루어진 확률과정이다.

베이지안 선형회귀 모델을 통해 가우시안 확률 과정의 가장 간단한 예를 만들 수 있는데, 함수 $f(x)$ 를 $f(x) = \phi(x)^T \mathbf{w}$, where prior $\mathbf{w} \sim \mathcal{N}(0, \Sigma_p)$ 라고 가정할 때(\mathbf{w} 는 정규분포를 따르는 사전 prior 확률), 확률변수들의 기대값이 0, 공분산 함수의 기대값이 $\phi(x)^T \Sigma_p \phi(x')$ 의 형태를 띄게 되는 확률 과정이라고 할 수 있다. 이때, 관측이 많이 일어나더라도 분포를 바꾸지 못하는 성질 때문에,,)

함수값 $f(x_1), f(x_2), f(x_3), \dots, f(x_n)$ 들은 실제 모수보다 작은 n 에 대하여 늘 정규분포를 따른다.

이러한 관측값을 통해서, 기존 사전 확률을 업데이트한 사후확률 Prior 을 규정한다. 이 사후확률을 달리 말해, 주어진 관측값에 기반해서 주어진 x 에서 $f(x)$ 가 존재할 확률이라고 할 수 있다.

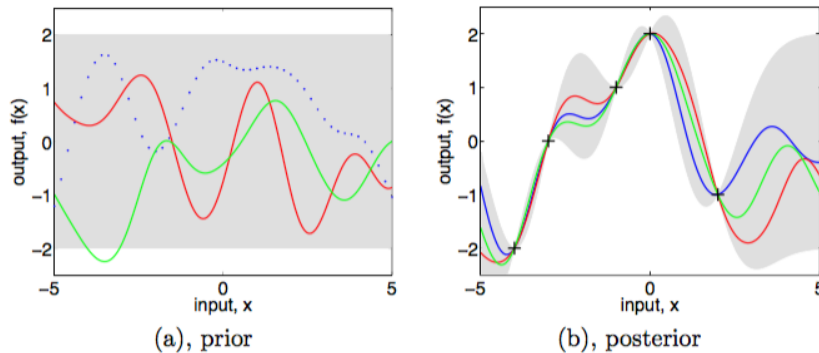


Figure 4: 음영은 input x 에 대해서 95% 신뢰도의 분포가능한 영역 ($m \pm 2\sigma$). 각각 prior, posterior 확률에 대응.

이러한 가우시안 프로세스 방법으로 주어진 관찰 값에 대한 함수가 분포할 수 있는 영역을 추정할 수 있는데, 관찰 대상이 노이즈를 포함하고 있는지, 100 프로 신뢰할 수 있는 (노이즈 zero) 값인지에 따라 파라미터 설정 및 결과값이 다르다. 아래는 실제 GiGA Office 도메인에서 throughput 값에 대한 두가지 mode 를 (Standard Model & Noise Robust Model) 적용해 본 결과이다.

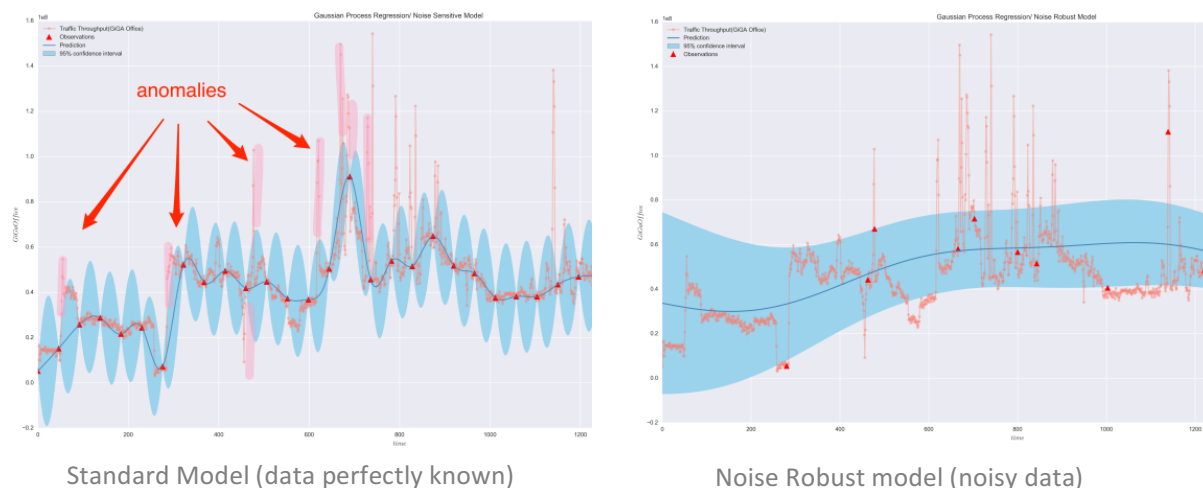


Figure 5: Gaussian Process Regression 기법을 활용한 Anomaly Detection 분석 적용 예. (좌) 데이터에 noise 가 없는 데이터를 가정했을 경우. (우) 분석 대상 데이터에 noise 가 섞여 있다고 판단 되었을 경우.

이와 같이 Gaussian Process Regression 은 엄밀한 확률과정을 통해 탄탄한 이론에 기반을 둔 성능 높은 분석 방법이지만, 이 방법을 통한 시계열적 이상값 탐지 방법은 몇몇 단점이 있다. 첫째로, 개념 자체의 난이도가 있어 충분한 이해가 선행되지 못한 상태에서 분석개발을 할 수 있다. 둘째로, hyper-parameter 설정 문제가 있다. Gaussian Process 적용에서 성능 튜닝, 특히 noise-robust 모델을 사용하거나 Ridge 와 같은 kernel 기법을 적용하는 경우, 설정해야 할 hyper-parameter 들이 존재한다. 여느 머신러닝 문제와 같이 이러한 파라미터 설정은 단번에 최적값을 찾기 어렵다. 마지막으로, 현재 본 보고서에서 Gaussian Process Regression 구현을 위해 사용한 라이브러리는 python 외부 패키지인 scikit-learn 인데, 다음 절에서 소개할 S-H-ESD 알고리즘보다 구동 속도가 느리다. 이는 첫번째 두번째 이유와 맥을 같이 하는데, 알고리즘 자체가 다소 고차원적이고 복잡하기 때문이라고 할 수 있다.

3.2.2. Seasonal Hybrid ESD (twitter anomaly detection 제안 기법)

본 절에서는 시계열 데이터에서 널리 활용되고 있는 Seasonal Hybrid ESD(S-H-ESD) 알고리즘의 이론적 이해, Python 구현 방법등에 대해서 다룬다. S-H-ESD 는 앞서 소개한 Gaussian Process 보다 상대적으로 기초 통계에 기반 하였으며, 글로벌 기업 Twitter 에서 실제 상용적용 가능한 수준의 구현을 통해 적용성 feasibility 가 검증되었다는 장점이 있다.

A. 이상탐지 기법 개요

우선, Twitter 사에서 제안한 S-H-ESD 기법은 간략히 요약하면, 기존의 outlier 를 찾는 방법 (평균 μ 과 표준편차 σ 를 이용해 $\mu \pm 2\sigma$ 또는 $\mu \pm 3\sigma$ 에서 벗어난 값을 찾는 방식) 은 다음과 같은 문제점이 있다.

기존 이상탐지 방법의 한계점
<ol style="list-style-type: none"> 1. 잘못된 계산 지표 using wrong metrics: 기존의 단순 평균μ 표준편차σ이용하는 방식 자체가 outlier 값을 같이 계산에 포함하기 때문에 이상값에 취약함 2. Multi-modality 에 취약: 평균과 표준편차가 seasonality 등에 의해 변화되어 outlier 를 놓치게 되는 경우 발생

이러한 문제점으로 outlier 들을 놓치게 되는 문제를 개선하기 위해 Twitter 가 제안한 S-H-ESD 방법은 아래와 같은 접근을 취한다.

S-H-ESD 기법의 이상탐지 방법
<ol style="list-style-type: none"> 1. Use Robust statistics/Metric <ol style="list-style-type: none"> a. Median Absolute Deviation(MAD) b. Grubb's Test & Generalized Extreme Studentized Deviate (ESD) 2. Remove impact of seasonality and trend <ol style="list-style-type: none"> a. Addressing & determining seasonality b. Seasonal Trend decomposition using Loess(STL)

위 표들에 대한 요약 설명을 하면, 기존 이상탐지 방법에서 단순 평균 μ 표준편차 σ 를 이용하는 방식에는 계산에 이상값을 포함하므로 내재적으로 이상탐지에 문제가 있고, 앞서 설명한 시계열 데이터의 multi-modality 의 특성을 가지는 경우 또한 이상탐지에 취약점을 가지고 있다. 이러한 문제점을 개선하기 위해 제안된 S-H-ESD 는, (1) 각각에 대응하여 좀더 outlier 에 덜 취약하다고 알려진 중위수 절대 편차(MAD)와 generalized ESD 으로 이상값을 탐지한다. 하지만 이방법들 또한 단일 이상값을 탐지하는데 적합하고, 정규분포를 가정한다는점, 그리고 여전히 시계열의 multi-modality 를 고려하지 못한다는 점에서 단점이 있다. 이를 보완하고자 (2) 계절성과 추세를 시계열 데이터에서 분리하여 잔차 Residual 만을 분석하는 방법(STL)을 추가적으로 결합한다. 이를 통해 Multi-modality 를 제거한 uni-modal 한 시계열 데이터 분석이 가능하며, 분석 방법에도 좀더 이상값이 덜 취약한 방법(Robust Statistics: MAD, ESD)으로 탐지할 수 있게 된다.

B. 관련 통계 이론

위에서 골자를 밝힌 S-H-ESD 기법을 좀더 자세히 이해하려면, 다음과 같은 통계학적 개념들의 이해가 요구된다. (출처: <https://warrenmar.wordpress.com/tag/seasonal-hybrid-esd/>)

- **Median Absolute Deviation(MAD)**
- **Student t-distribution**
- **Extreme Studentized Deviate (ESD) test**

- Generalized ESD
- Seasonal Trend decomposition using Loess(STL)

각 개념들에 대한 간략한 설명은 아래와 같다.

Median Absolute Deviation(MAD); 중위수(중앙값) 절대 편차 :

중위수 절대 편차는 표본그룹의 중앙값(\bar{x})과 각 표본(x_i)의 차이값의 절대값을 취해서 중앙값을 추출하는 방법이다. 양적 자료의 퍼짐을 알고 싶을 때, 표본분산과 표준편차 보다 이상치에 덜 영향을 받는 강건성 *robustness* 있는 분산 측정 방법이다.

$$MAD = \text{median}_i |X_i - \bar{X}|$$

$$\sigma_{MAD} = K \cdot MAD$$

where $K = 1.4826$ for normal distributed data

Student t-distribution:

t-분포는 정규분포 (μ, σ^2)에서 n 개의 표본들을 확률변수로 정의($\bar{X} = X_1, \dots, X_n$)한 확률 분포이다. 이 확률 분포 또한 정규분포($\mu, \frac{\sigma^2}{n}$)이며, 이를 수식으로 나타내면 아래와 같다.

$$\bar{X} \sim N\left(\mu, \frac{\sigma^2}{n}\right)$$

$$T = \frac{\bar{X} - \mu}{S/\sqrt{n}}$$

Grubb's test(=ESD test):

단일 이상치를 테스트 하는데 ESD test 방법은 널리 알려진 기법이다. ESD 검증 방법의 상세한 설명은 본 보고서에서 생략하며, 주요 수식표현은 아래와 같다.

(https://en.wikipedia.org/wiki/Grubbs%27_test_for_outliers 참고)

ESD 검정은 아래와 같은 귀무가설/대립가설을 통해 검정한다.

H_0 : 데이터 셋에 이상치가 없다

H_a : 데이터 셋에 최소한 하나의 이상치가 존재한다.

아래 정의와 같은 G 값을 통해 outlier 인지 판별한다

$$G = \max |Y_i - \bar{Y}|/s$$

\bar{Y} : sample mean, s : standard deviation

최대값과 최소값을 둘 다 검정하는 two-sided test 에서, 이상치가 없다는 귀무가설은 significance level(α) 가 아래를 만족할시 기각된다.

$$G > \frac{N-1}{\sqrt{N}} \sqrt{\frac{t_{\alpha(2N),N-2}^2}{N-2 + t_{\alpha(2N),N-2}^2}}$$

$t_{\alpha(2N),N-2}^2$: upper critical value of the t -test with

$N-2$ degrees of freedom and significance level of $\alpha/(2N)$

ESD 테스트는 정상성 normality 를 가정하고, 단일 이상치를 탐지하는데 적합하다는 한계 때문에, 시계열과 같은 연속적 데이터에서 지속적으로 이상탐지를 해야하는 경우, 아래와 같은 Generalized ESD 의 사용이 권장된다.

Generalized Extreme Studentized Deviate (Generalized ESD):

Generalized ESD 는 Grubb's test 와 달리 여러개의 outlier 를 가정한 검정방법이다. 가장 높은 G 밸류를 제거해나가면서 지속적으로 순회 iterate 하여 평균과 표준편차를 업데이트해 나가는 방식이다.

$$R_i = \max |x_i - \bar{x}|/s$$

$$\lambda_i > \frac{(N-1)t_{p,n-i-1}}{\sqrt{(N-i-1 + t_{p,n-i-1}^2)(n-i+1)}}$$

위에 제시된 Critical Value(λ_i) 또한 지속적으로 업데이트 되며, $R_i > \lambda_i$ 를 만족하는 i 가 이상값의 개수를 결정하게 된다. 이 검정방법은 앞선 Grubb's test(ESD) 보다 여러개의 outlier 들이 검출 가능하다는 장점이 있으나, 여전히 데이터의 정규성을 가정하고 있으므로 정규성 테스트가 선행되어야 하고, 계절성을 고려하지 않는 seasonality unaware 단점이 있다.

Seasonal Trend decomposition using Loess(STL):

STL 은 시계열 데이터에서 계절성, 추세, 잔차 세가지 패턴요소로 분해하는 기법이다. 3.1.1. 절 "시계열 데이터 구성요소"와 Figure3 에서 이 주제에 관해 다루었다. STL 을 통해 seasonality 와 trend 를 제거하면, 이상탐지에 적합한 normal distribution 과 유사한 형태의 residual 만 남게 된다.

C. Python 을 이용한 S-H-ESD 알고리즘 구현

앞서 소개하였듯이, 이상탐지를 위한 S-H-ESD 알고리즘은 Twitter 사가 처음 제안하였으며, 이를 실제 구현한 R 코드 또한 Twitter 공식 깃헙(Github) 페이지에 오픈소스로 공개되어 있다. (GNU Public License) (링크: <https://github.com/twitter/AnomalyDetection>) 본 보고서에서는 이를 좀더 시스템 친화적인 프로그래밍 언어인 Python 으로 변환 구현해 GiGA Office 트래픽 데이터에 이상검출한 유즈케이스를 소개한다.

Python 으로 S-H-ESD 기반의 Anomaly Detection 기능을 변환 porting 한 오픈 소스 패키지 또한 존재하므로, (pycularity <https://github.com/nicolasmiller/pycularity>) 본 사례연구/구현에서는 해당 패키지를 기반으로 구현되어 있다. 다만 해당 패키지에서는 순수 Python 라이브러리 외에 R의 인스톨을 요구하는 rpy2 라는 다소 불편하고 시스템 이식성이 떨어지는 라이브러리를 중심으로 활용한다. 이에 본 사례적용 케이스에서는 Python 친화적이고 기계학습/통계에서 major 한 패키지(statsmodel) 를 활용하여 성능을 좀더 업그레이드 하였다. 아래는 패키지 구성과 기능 상세와 구현 방법에 대한 설명이다.

패키지 및 기능 구성:

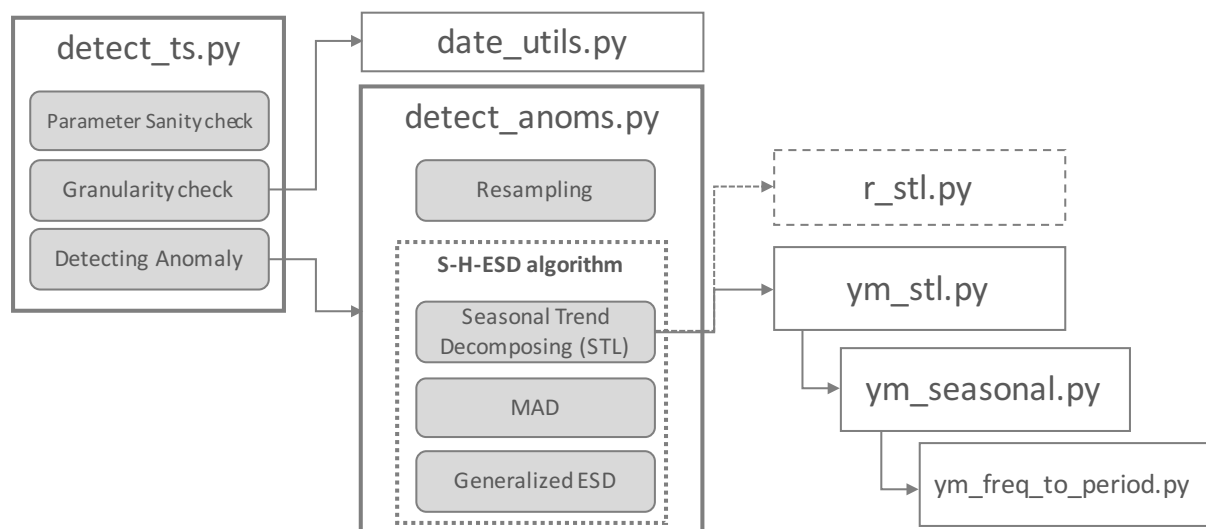


Figure 6: anomaly_detection 패키지 주요 라이브러리 구성도

"detect_ts.py" 는 anomaly detection 분석의 최상위 모듈로, 후에 resampling을 위해 시계열 데이터 granularity check을 (분/시간/일/월) 하고 기타 여러 옵션기능들에 대한 파라미터 값들의 에러핸들링(parameter sanity check)을 기본적으로 수행한다. 그리고 "detect_anoms.py"를 호출해 이상검출을 실시한다. 실제로 이상탐지에 가장 중요한 기능들은 이 "detect_anoms.py"에서 수행되는데, 다양한 주기/단위 (e.g. 30초, 1분, 3분, 1일)로 수집된 시계열 데이터를 모두 수용할 수 있기 위해 데이터 시계열 단위를 지정하여 새로 변환하는 resampling 기능을 우선적으로 수행한다. 그리고 S-H-ESD 알고리즘의 핵심적 세가지 기능을 수행하는데, 첫째로 시계열 데이터의 multi-modality로 인한 오차를 줄이기 위해 seasonal, trend, residual로 분해한다. 이 분해된 decomposed data에서 residual을 추출하여 이 데이터의 MAD 값을 얻어낸 후, Generalized ESD를 통해 이상치를 검출한다. (STL, MAD, Generalized ESD의 이론적 이해는 3.2.2.B 참조.)

STL를 통해 decomposing을 하는 방법에서, 앞서 언급하였듯이 기존 Twitter를 그대로 Porting한 pycularity 패키지에서는 R함수를 Python을 위해 포팅 해주는 rpy2라는 외부 패키지를 활용해 구현하였다. 하지만 rpy2는 R의 설치를 요구하고 R이 실행된 상태를 가정하기 때문에, 추가적인 종속성과 시스템 자원을 요구한다. 때문에 시스템 호환성/이식성/ 성능 등에 영향을 필연적으로 미치게 되어있다. 이러한 문제 때문에, 본 프로젝트 및 사례연구를 위해서 rpy2의 패키지를 대체하는

라이브러리를 statsmodel을 기반으로 새로 작성하였다.(ym_stl.py, ym_seasonal.py, ym_freq_to_period.py)

Anomaly Detection 기능을 이용하기 위해서는 detect_ts.py를 호출하는 스크립트를 실행하면 된다. 아래는 detect_ts.py를 호출하여 anomaly를 검출하는 스크립트 예시이다. 단순히 anomaly를 검출하는데 벗어나, 검출값들을 시각화 하고자 하면, test_output.py 스크립트 참조.

```
from pyculianity import detect_ts
import pandas as pd

n_file = 'traffic_xxUniv_throughput_0514_0613'
timeS_DF = pd.read_csv('./data/%s.csv'% n_file, usecols = ['Time',
'InOctets'])

results = detect_ts(timeS_DF, max_anoms=0.02, direction='pos',
only_last=None)

""" Deliverables """
print '>>> the number of anomaly: ', len(results['anoms'])
print results['anoms']
```

아래는 MAD와 generalized ESD으로 S-H-ESD 기능을 구현한 부분이다.

```
for i in range(1, max_outliers + 1):
    if one_tail:
        if upper_tail:
            ares = data.value - data.value.median()
        else:
            ares = data.value.median() - data.value
    else:
        ares = (data.value - data.value.median()).abs()

    data_sigma = mad(data.value)

    if data_sigma == 0:
        break

    ares = ares/float(data_sigma)
    R = ares.max()
    temp_max_idx = ares[ares == R].index.tolist()[0]
    R_idx[i - 1] = temp_max_idx
    data = data[data.index != R_idx[i - 1]]
    if one_tail:
        p = 1 - alpha / float(n - i + 1)
    else:
        p = 1 - alpha / float(2 * (n - i + 1))
    t = student_t.ppf(p, (n - i - 1))
    lam = t * (n - i) / float(sqrt((n - i - 1 + t**2) * (n - i + 1)))

if R > lam:
    num_anoms = i
```

4. 성능평가 및 검증

현재 Twitter 사에서 개방한 R 기반 S-H-ESD 패키지에 대해서 성능평가는 다수 진행되었고, 대체로 공개 솔루션 중 가장 좋은 평가를 받는다.

Twitter 의 Anomaly Detection 은 우선 아래와 같은 주요 특징적 기능들이 있다.

Key Features
<ul style="list-style-type: none">• 이상치의 방향 direction (Positive & Negative)• 전역적 & 지역적 이상치 global & local anomaly• 최근 하루/한시간 last day/last hour• 기대값(~회귀) expected value• 장기적 추세에 따른 이상탐지 long term

아래는 twitter 사의 이상탐지 패키지의 성능 벤치마크를 진행한 내용에 대한 링크이다.

- Anomali.io: <https://anomaly.io/anomaly-detection-twitter-r/>
- NUMENTA: Evaluating Real-time Anomaly Detection Algorithms – the Numenta Anomaly Benchmark (논문)

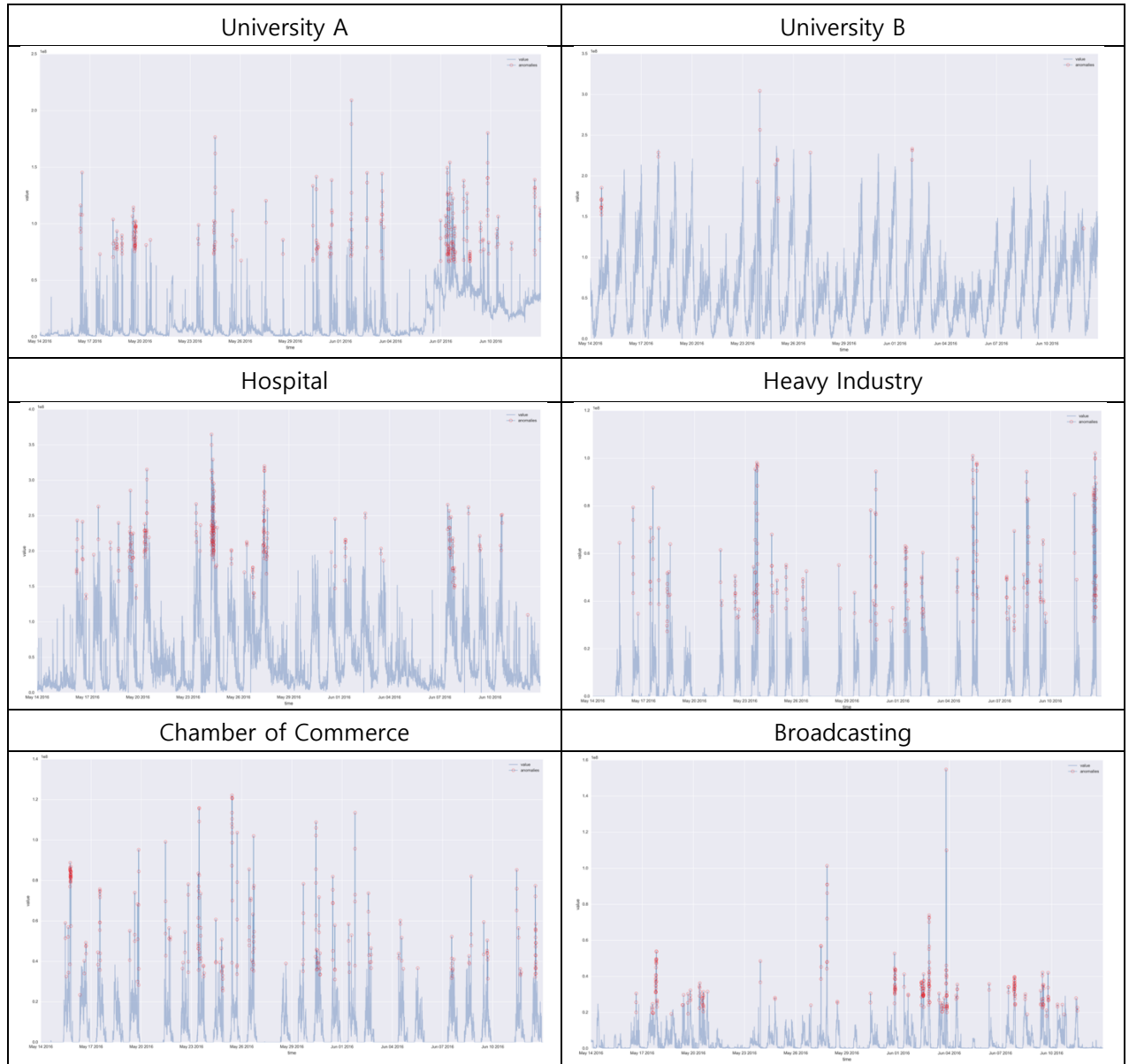
위 벤치마크에 따르면, 이상치를 잘 탐지하거나 그렇지 못한 경우는 아래와 같다.

Detected	Not Detected
<ul style="list-style-type: none">• 노이즈의 증가 More noise• 급작스런 상승,급등점 Sudden grow; spike• 하강 Break down• 희귀 값 Activity when usually none	<ul style="list-style-type: none">• 점진적 증가하는 신호 Seasonal grow• 평면적 신호 Flat signal• 점진적 증가하는 신호에서의 음의 방향 이상치 Negative seasonal anomaly

아래 4.1 ~ 4.3 은 본 사례연구에서 기능 수정된(Python Porting, Different Packages) 프로그램의 기능 검증 및 성능 평가이다.

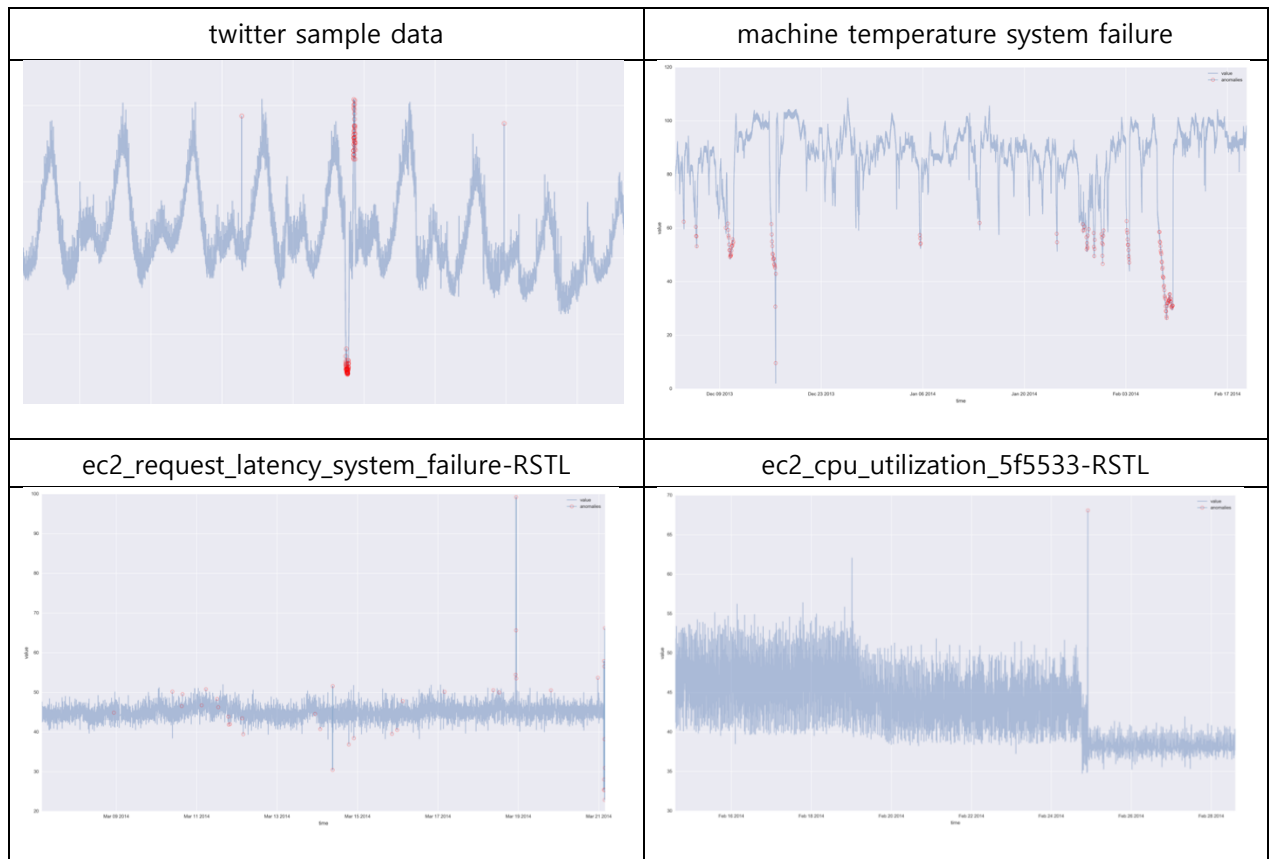
4.1 GiGA OFFICE Traffic Throughput

다음은 약 2 달간 수집된(2016.5-6) GiGA Office Traffic throughput(InOctet) 데이터를 활용해서 급등하는 지점 spike 들을 검출하는 기능을 검증한 결과물이다. 전달받은 18 개 site 중 대표적인 6 개 site 에 대한 결과물 이다. 실제 장애를 일으키는 수준을 명명한 critical points 에 대한 정보가 없으므로, 검출율에 관한 성능 metric 은 적용할 수 없으나, 육안상 급등점들을 성공적으로 검출해 낼 수 있다.



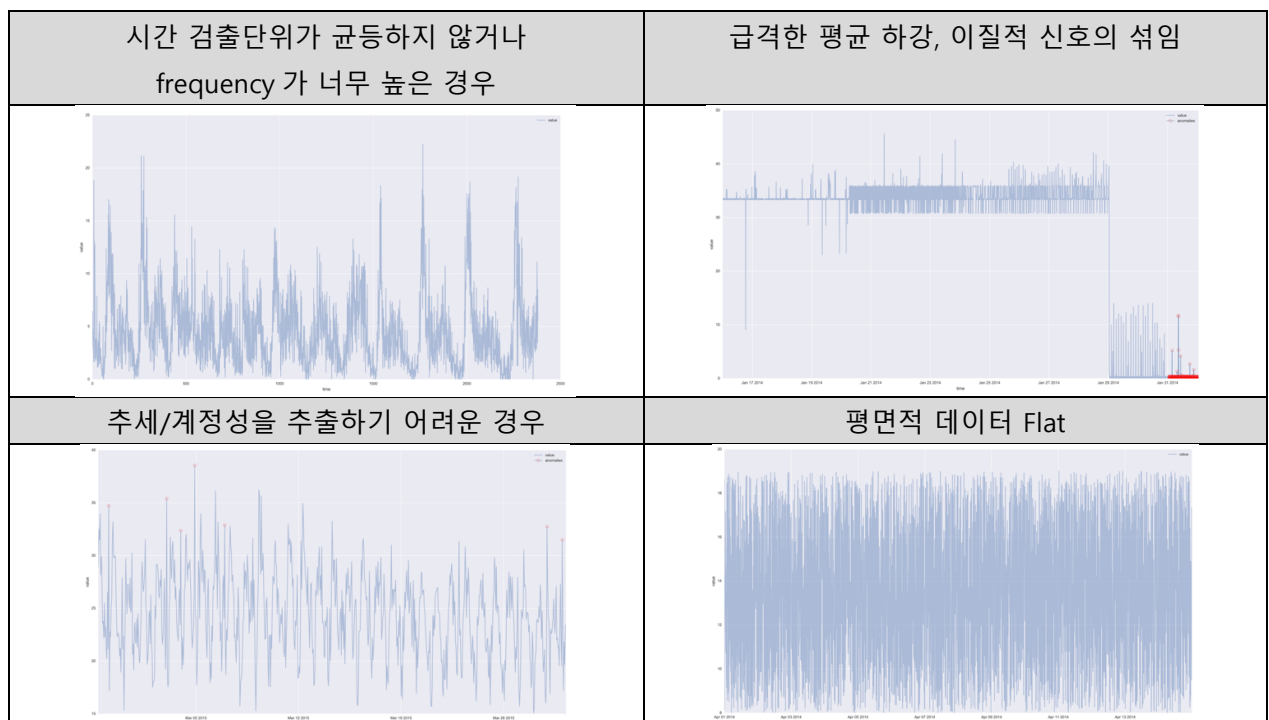
4.2 Other Domain

다음은 GiGA Office Traffic 데이터 외에 다양한 양상을 가진 데이터에 대하여 성능검증을 한 결과이다. 아래 결과에서 볼 수 있듯이, 추세/계절성이 변하더라도 양/음의 방향을 가지는 급등 들을 대체적으로 잘 검출함을 확인 할 수 있었다.



4.3 Not working cases

그러나 모든 경우에 대해서 잘 동작하지는 않았는데, 이는 위 벤치마킹 연구에서 알려진 사실이기도 하다. 아래는 잘 동작하지 않았던 경우에 대한 예이다.



5. References

- 1) Varun Chandola, 2009, <Anomaly Detection: A Survey>, ACM Computing Survey 09 2009 p1-72
- 2) Arindam Banerjee, <Anomaly Detection: A Tutorial>, United Technology Research Center
- 3) 이기천 한양대 교수, 2013, <시계열 데이터의 통계적 분석방법>, 강의자료
- 4) A Complete Tutorial on Time Series Modeling,
<https://www.analyticsvidhya.com/blog/2015/12/complete-tutorial-time-series-modeling/>
- 5) C. E. Rasmussen & C. K. I. Williams, Gaussian Processes for Machine Learning, the MIT Press, 2006
- 6) Problem of the Month: Anomaly Detection,
<https://warrenmar.wordpress.com/tag/seasonal-hybrid-esd/>
- 7) Arun Kejariwal, Statistical Learning Based Anomaly Detection @ Twitter, Nov 2014