B-spline 회귀 시계열 요소 분해를 이용한 효율적인 이상 탐지

김 호 현
한국방송통신대학교 대학원
hohkim@koreanair.com

Efficient Anomaly Detection using Time Series Decomposition with B-spline Regression

Hohyun Kim
Korea National Open University Graduate School

요 약

항공 회사들은 대리점의 급작스런 부도로 인한 매표대금의 손실을 방지하기 위하여 대리점의 비정상적인 판매 급증을 조기에 탐지하여 경보하는 시스템을 운용할 필요가 있다. 그러나 입력 데이터로 주어지는 일별 판매액은 *Time Series Data*의 일반적 특성인 seasonality와 trend 속성을 가지고 있어 단순히 과거 시점과 비교하는 방식의 시스템은 높은 false alarm rate를 보인다.

본 논문에서는 이동평균 값에 *B-spline Regression*을 적용한여 trend 패턴을 분리해 내는 방법과, 이상여부 판정 시 seasonality 주기 내의 구성 요소 간 분산이 다른 경우, 구성 요소별로 다른 threshold를 적용하여 탐지 신뢰성을 높이는 독특한 방법을 제시하였다. 실험을 통해 제안 알고리즘의 우수성을 입증해 보였다.

1. 서론

회사는 대리점의 급작스런 부도로 인한 판매대금 손실을 예방하기 위해 대리점의 판매 이상 급증을 탐지하여 지점에 통보하는 경보 시스템을 운용하고 있으나, 현 시스템은 단순히 과거와 비교하여 일정 수준 이상 증가 시 이상 건으로 판별하고 있어 높은 false alarm rate를 보이고 있다.

정확한 탐지가 어려운 이유는 판매 실적이 *Time Series Data*의 특징인 seasonality와 trend 속성을 가지고 있기 때문이다.

*Twitter*에서는 *Long-Term Time-Series Data Anomaly Detection* 알고리즘을 개발하여 open source로 공개하였는데 탐지 성능이 상당히 좋다고 한다.[1] 그러나 이는 데이터를 여러 구간으로 나누어 각각 처리한 후 결과를 합치는 piecewise 방법을 사용하고 있어, 일별 대리점 판매 실적과 같이 데이터 량이 많지 않은 경우는 적용하기 어려운 문제가 있었다.

*Time Series Data*에 대한 *Anomaly Detection*은 데이터가 가지고 있는 trend 속성을 어떻게 분리해 내는가에 따라 큰 영향을 받는다. 본 논문에서는 *Time Series Data*로부터 *Moving Average+ B-spline Regression*을 이용하여 trend component를 추출하는 새로운 방법을 제안한다.

2절은 동 논문의 기여점을 밝히고, 3절은 기존의 *Anomaly Detection* 방법론과 *Time Series Data*의 일반적인 특성을 기술한다. 4절에서는 *Time Series Decomposition*, 5절에서는 *Anomaly Detection* 탐지 제안 알고리즘을 설명하고, 6절에서는 비교 실험을 통해 제안 알고리즘의 우수성을 입증하며, 7절에서는 향후 연구방향 등으로 결론을 맺는다.

2. 논문의 기여

본 논문에서는 다음과 같은 점에서 *Time Series Decomposition*의 알고리즘 개선에 기여하고자 하였다.

- *Time Series Decomposition*의 전통적인 방법은 trend 요소 추출 시 이동평균법, Loess smoothing (*STL*)[1]을 많이

[1] Loess: LOcal regrESSion, STL: Seasonal Trend Decomposition using Loess

사용한다. 본 논문은 일반적 추세 흐름을 보다 잘 추적할 수 있도록 *Moving Average*에 *B-spline Regression*을 결합한 방법을 시도하였다.

- Seasonality 주기 내의구성 요소 간 – 예를 들어, 주간 seasonality에서 각 요일 간 – 를 갖는 경우, 각 요일 간 – 의 분산이 서로 다른 경우, 전체에 대해 동일한 threshold를 적용하여 판정을 하면 오류율이 증가하게 된다. 본 논문에서는 seasonality의 하위 구성요소(요일) 별로 서로 다른 threshold를 적용하여 신뢰성을 높이는 방법을 제시하였다.

3. 관련연구

3.1 Anomaly Detection

*Anomaly Detection*은 데이터 가운데서 기대되는 정상적인 행동을 따르지 않는 패턴을 찾아내는 것이다.[5] 이러한 이상한 패턴은 anomaly 또는 Outlier 라고 불리운다. *Anomaly Detection*은 fraud detection, intrusion Detection, fault detection 및 military surveillance 등의 여러 응용분야에서 광범위하게 사용되고 있다.

[5]에서는 *Anomaly Detection*기법을 1. Classification Based 2. Clustering Based 3. Nearest Neighbor Based 4. Statistical Based 5. 기타 (Information Theoretic, Spectral) 와 같이 분류하고 있는데, 어떠한 기법을 선택할 것인가는 1. 입력 데이터의 성격 (Sequence, Spartial, Graph) 2. Anomaly의 종류 (Point, Contextual) 3. Data Label의 유무 4. Output 형태 (Score, Label) 등을 고려하여 선택하여야 한다.

3.2 Time Series Data

Time Series Data(시계열 자료)는 일정한 시간 간격을 두고 발생하는 일련의 값으로, 1. Seasonality (일정 주기로 반복되는 패턴) 2. Trend (시간이 경과하면서 나타나는 값이 전반적인 증감 추세) 3. Random (무작위적인 증감) 요소를 포함하고 있어, *Time Series Decomposition*을 통한 *Anomaly Detection*방법에 대한 연구가 활발하게 이루어져 왔다.

[8]에서는 *Time Series Decomposition*에 *Generalized ESD(Extreme Studentized Deviation) Test*를 결합한 Seasonal Hybrid Extreme Studentized Deviates *Anomaly Detection*기법을 소개하였고, 동 알고리즘을 Open Source화 하여 공개하였는데 탐지 성능이 매우 우수하다.[2]

*Generalized ESD(Extreme Studentized Deviation) Test*는 n개의 데이터에서 k개의 anomaly를 탐지해 내는 통계적 기반의 *Anomaly Detection*기법이다.[7]

[9]에서는 *Time Series Decomposition* 기반의 신뢰구간 추정을 통해 선박의 이상을 감지하는 시스템을 연구하였고, [11]는 트위터 메세지에 많이 나타난 복수의 토픽들에 대해 *Time Series Decomposition*을 수행 후 토픽 상호 간의 상관관계를 분석하여 비정상 이벤트의 발생을 탐지하는 과정을 시각적으로 분석할 수 있도록 하는 연구를 진행하였다.

또한 시계열 데이터의 예측에 활발히 사용되고 있는 ARIMA (*Auto Regression Integrated Moving Average)* 모델을 이용한 Anomaly Detection 연구가 많이 이루어지고 있는데 *[4]*과 *[6]*이 좋은 예이다. 국내에서도 *ARIMA* 모델을 이용한 안드로이드 악성코드 탐지 기법을 *[10]*에서 소개하고 있다.

이러한 통계적 기반의 Anomaly Detection 기법은 데이터에 *Label* 정보가 없는 경우 비지도학습 (Unsupervised Learning)에도 적용 가능하며, 신뢰구간과 연결지어 anomaly에 score를 주어 의사결정에 도움을 줄 수 있다는 장점이 있다. 그러나 데이터가 어떤 분포를 따른다는 가정에 의존하는데 그 가정이 사실과 다를 수 있다는 것이 단점이다.[5]

4. Time Series Decomposition

*Time Series Decomposition*은 시계열 데이터를 몇 가지 패턴 유형을 나타내는 요소들로 분해하는 통계적인 방법을 말한다.[3]

Time Series Data y_t는 다음과 같이 정의된다.[2]

$$y_t = S_t + T_t + E_t$$

여기서 y_t, S_t, T_t는 각각 t 시점에서의 데이터, seasonal 요소, trend 요소이며, E_t는 t 시점에서의 remainder 이다.

5. 제안 *Anomaly Detection* 알고리즘

5.1 이동평균과 *B-spline* 회귀를 결합한 시계열 분해

Seasonality 요소 분해 일정 주기로 반복되는 패턴을 찾는 것이므로 주기만 주어지면 반복되는 주기의 각 값을 평균하여 쉽게 산출할 수 있으며, *Time Series Anomaly Detection*의 성능은 trend를 얼마나 잘 분리해 내느냐에 크게 영향을 받는다. 본 논문에서는 *Moving Average*에 *B-spline Regression*를 결합하여 trend를 분해하는 방법을 새롭게 제안한다.

[2]이것은 Additive Model의 경우이다. 만일 시간이 경과함에 따라 Seasonal Pattern이나 Treand Pattern의 variation의 정도가 비례하여 변화하는 경우는 Multiplicative Model이 적합하며 이 경우는 $y_t = S_t * T_t + E_t$ 이다.

	알고리즘	비고
1	$X \leftarrow$ input data	시계열 데이터
	$v = [] \leftarrow$ output	
	$\alpha \leftarrow 1.96$	95% 신뢰수준
	$\beta \leftarrow 0.05$	95% 신뢰수준
	$w \leftarrow$ const	이동평균 window
	$k \leftarrow$ const	*B-spline* knot 갯수
	$\theta \leftarrow$ null	threshold
2	$(S_x, T_x) =$ decompose(X)	요소 분리
	- $S_x =$ avg(X_{wday})	요일별 평균
	- $T_x =$ **ma**_w(X)	**moving average**
3	$T_x =$ **B-spline**(T_x, k)	**B-spline 회귀 이용 trend smoothing**
4	$R_x = X - S_x - T_x$	random 요소 분리
5	$\sigma =$ sd(R_x)	전체의 표준편차
	$\sigma[\,] =$ sd(Rx_s)	각 요일별 표준편차
6	$p =$ **anova**(abs(R_x, wday))	**요일별 분산 차이 존재 여부 확인**,
	if $p < \beta$	차이가 있으면
	then $\theta = \sigma_s[] * \alpha$	**요일별 표준편차를 사용, 판정값 산정**
	else $\theta = \sigma * \alpha$	
7	for all X_t do	Random 요소 값이
	if $R_x > \theta$	임계치보다 크면
	then $v = v + R_t$	이상값에 추가
	end for	
8	return v	anomaly 반환

표 1: 제안 *Anomaly Detection* 알고리즘

1. Seasonality component: 반복되는 주기 내의 각 값을 평균하여 구한다.

2. Trend component

   a) *Moving Average*: window 크기를 정한 후(예 7일), window를 이동하면서 평균값을 계산한다.

   b) *B-spline Regression*: 위에서 구한 이동평균 값을 그대로 사용하면 trend가 너무 급격한 변화를 보이는 경향이 있다. 따라서 보다 일반화된 Trend 패턴을 얻기 위하여, 위의 *Moving Average*값에 *B-spline Regression* 를 적용하여 trend를 구한다.

5.2 분산의 차이를 고려한 Multi-threshold 판정

Source 데이터에서 seasonal 및 trend 요소를 차감하여 remainder(E_t)를 얻은 후, 이 E_t가 threshold(예, $2\sigma, 3\sigma$)를 초과한 경우 anomaly로 판정하는 것이 일반적인 방법이다.

그러나 seasonality의 구성요소 – 예를 들어, 일주일 주기를 갖는 경우, 월, 화, 수, 목, 금, 토, 일 – 간의 분산이 서로 다르다면, 동일한 threshold를 적용 시 오류율이 높아지는 문제가 있다. 이 문제를 해결하기 위하여 요일별 분산의 차이가 있는지를 *anova* 분석을 통해 확인하고, 차이가 있다면 요일별로 다른 threshold 값을 계산하여 판정에 사용하는 방법을 사용하였다.

1. Remainder 데이터 set의 표준편차(대표 σ)를 구한다.

2. Seasonality 주기 내의 하위 구성요소 (Weekly 내의 Day, Monthly 내의 Week 등) 간의 분산에 차이가 있는지를 *anova* 분석을 실시하여 확인한다.

3. Threshold 값을 결정한다. 3σ 로 정했다면,

   - 분산의 차이가 없다면, Threshold = $3 * \sigma$
   - 분산의 차이가 있다면, 요일별 표준편차 σ[i]를 구하여, Threshold = $3 * \sigma$[i]

4. 각 데이터에 대해 Anomaly 여부를 판정한다.

   if E_t > Threshold
   then Anomaly
   else Normal

표 1에 제안 알고리즘을 *psudo code*로 나타내었다.

6. 실험

6.1 실험 데이터

2010.1.1일부터 2016.5.18일 까지의 실제 일별 판매자료를 바탕으로 실험용 데이터를 생성하였다. 원래 data를 decompose하여 seasonal, trend, remainder 요소를 분리한 후, remainder의 요일별 표준편차와 평균을 이용하여 정규분포를 갖는 난수를 발생시켜 noise 값을 만들었다. 그 후 seasonal, trend, noise를 합성하여 실험에 사용할 new sales data를 생성하고, data가 anomaly인지를 나타내는 label을 추가하였다.

- new sales data $\leftarrow$ seasonal + trend + noise
- label $\leftarrow$ true, if noise $> 2\sigma$

실험 데이터는 그림 1과 같이 요일별 분산의 차이가 있다.

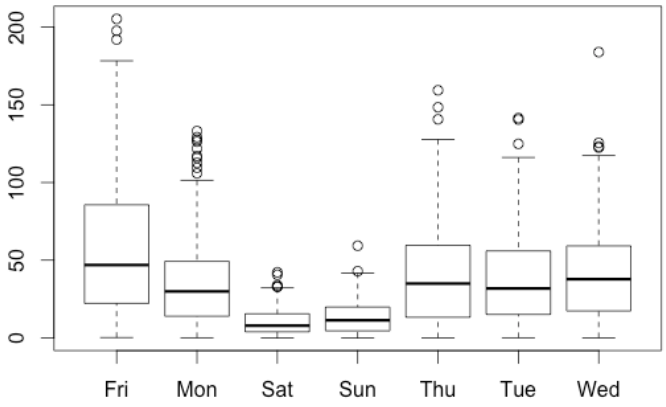

그림 1: 요일별 Noise의 분포

6.2 비교: Uni-threshold vs Multi-threshold

STL(Loess smoothing)을 이용하여 trend를 추출 후, 전체에 대해 동일한 threshold를 적용하여 이상값을 판정한 경우와 요일별로 다른 threshold를 적용한 경우를 비교한 결과는 표 2와 같다. 요일별로 다른 값을 적용한 경우의 F-score가 약 14% 나 우수한 성능을 보였다.

- Uni-threshold: $\theta = \sigma * 1.96$
- Multi-threshold: $\theta = \sigma_s[] * \alpha$

t.window	구분	Precision	Recall	F-score
15일	Uni-θ	44.4%	56.7%	49.8%
	Multi-θ	61.9%	67.7%	64.7%
60일	Uni-θ	49.4%	59.8%	54.1%
	Multi-θ	76.7%	78.0%	77.3%

표 2: 알고리즘 비교: Uni-θ vs Multi-θ

6.3 비교: MA vs MA + B-spline vs STL

Trend 성분 추출 알고리즘의 비교를 실시하였다. 비교 대상은 *Moving Average*, *Moving Average + B-spline Regression*, *STL*(Loess) 로 하였다. 이상 판정은 Multi-θ 방법을 공통적으로 적용하였으며, 결과는 표 3과 같다.

구분	파라미터	Pr.	Rc.	F-score
이동평균	window: 7일	55.7%	54.4%	55.1%
이동평균 +	knots: 매월말	76.5%	70.4%	73.3%
B-spline	knots: 분기말	77.2%	76.0%	76.6%
STL	t.window: 31일	68.4%	74.4%	71.3%
(Loess)	t.window: 91일	78.6%	79.2%	78.9%

표 3: Trend decompose 알고리즘 비교

단순 이동평균 보다 *B-spline Regression*, *STL*방법이 훨씬 우수하였다. *B-spline*과 *STL*간의 성능 차이는 크지 않았는데, smoothing factor – spline 내부 knots 값과 *STL*의 trend window size 값– 을 1개월로 한 경우는 *B-spline*이 더 나은 성능을 보였지만, 3개월로 지정한 경우는 *STL*이 보다 조금 더 나은 결과를 보였다. 이는 데이터의 특성에 맞게 파라미터를 지정하는 것이 성능에 영향을 크게 미친다는 것을 의미한다.

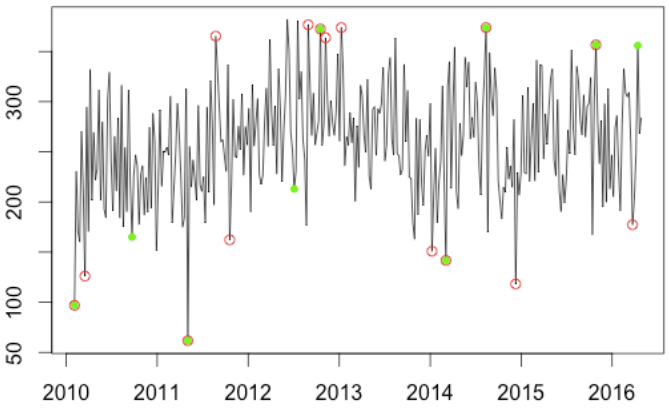

그림 2: 탐지 결과: Moving Average (수요일)

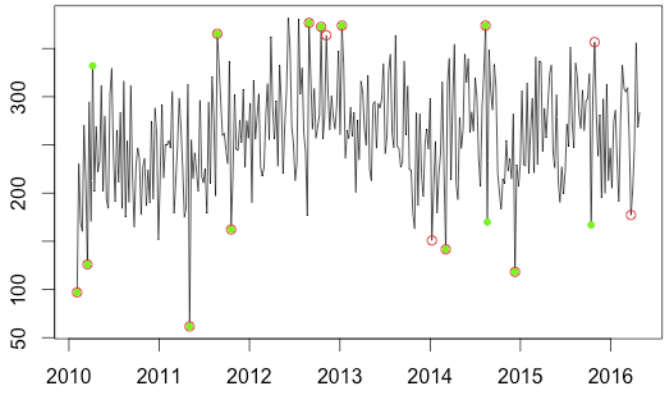

그림 3: 탐지 결과: MA + B-spline regression (수요일)

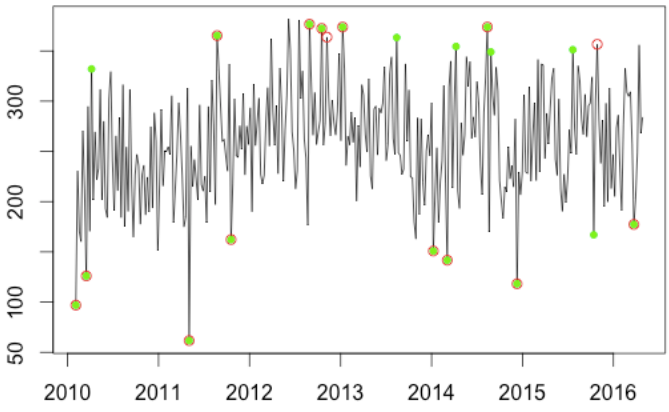

그림 4: 탐지 결과: *STL* (Loess) (수요)

그림 2, 3, 4는 *Anomaly Detection* 결과 중 수요일에 해당하는 건을 그래프로 표시한 것이다. 붉은색 동그라미가 실제

anomaly 이며, 녹색 포인트가 동그라미가 알고리즘에 의해 탐지된 anomaly를 나타낸다.

- 붉은색 동그라미 내의 녹색 포인트: True Positive
- 동그라미가 없는 녹색 포인트: False Positive

7. 결론

Time Series data는 seasonal, trend 패턴을 포함하고 있는 특성으로 인해, *Time Series Decomposition*을 이용한 *Anomaly Detection*이 활발히 이용되고 있다. 시계열 분해 시 trend 패턴을 얼마나 실제에 근접하게 분리해 내는가가 탐지 성능에 크게 영향을 미치는데, 본 논문에서는 Moving average을 산출한 후, 이에 대해 다시 *B-spline Regression*을 적용하여 smoothing 함으로써 보다 일반화된 trend 패턴을 추출하는 방안을 고안하였다.

원 data로부터 seasonal, trend component를 제거하고 난 후 얻어진 remainder는 정규분포를 따른다는 가정 하에, 데이터가 평균으로부터 일정 수준 (Threshold: 보통 표준편차 σ의 배수로 결정) 이상 떨어진 경우 anomaly로 판정할 수 있다. 그런데 seasonality 주기 내의 데이터들 간의 분산 정도가 다르다면, uni-threshold를 적용시 탐지 오류율이 높아지므로, 주기 구성 요소별로 각각 다르게 multi-treshold를 적용하여 탐지 신뢰성을 향상시킬 수 있었다.

실험을 통해 동 제안 방법의 우수성을 입증하였다. Loess smoothing을 사용하는 *STL*알고리즘과 비교해서는 *B-spline* 내부 Knots를 어떻게 정하느냐에 따라 결과가 달랐으나, 전체적으로 큰 성능의 차이는 보이지 않았다.

B-spline 은 내부 Knost의 갯수를 어떻게 정하느냐에 따라 Smoothing 정도가 달라진다. 향후 최적의 *B-spline* 내부 Knots 결정에 대한 연구와 동 알고리즘을 적용한 "판매 급증 조기 경보 시스템"을 실제 구현하여 운영해 보고 그 결과를 feedback 하여 알고리즘을 개선하는 추가 연구를 진행할 예정이다.

참고 문헌

[1] URL https://anomaly.io/anomaly-detection-twitter-r/.

[2] URL https://github.com/twitter/AnomalyDetection.

[3] URL https://en.wikipedia.org/wiki/Decomposition_of_time_series.

[4] Ana Maria Bianco, M Garcia Ben, EJ Martinez, and Víctor J Yohai. Outlier detection in regression models with arima errors using robust estimates. *Journal of Forecasting*, 20(8):565–579, 2001.

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

[6] Annick M Leroy and Peter J Rousseeuw. Robust regression and outlier detection. *Wiley Series in Probability and Mathematical Statistics, New York: Wiley, 1987*, 1, 1987.

[7] Bernard Rosner. Percentage points for a generalized esd many-outlier procedure. *Technometrics*, 25(2):165–172, 1983.

[8] Owen Vallis, Jordan Hochenbaum, and Arun Kejariwal. A novel technique for long-term anomaly detection in the cloud. In *6th USENIX Workshop on Hot Topics in Cloud Computing (HotCloud 14)*, Philadelphia, PA, June 2014. USENIX Association.

[9] 박진관 정민아 김영주, 허유경. 시계열 분석 기반 신뢰구간 추정을 통한 효율적인 이상감지. 한국통신학회논문지, 39(8):708–715, 8 2014.

[10] 김환희. 자기회귀 이동평균 모델을 이용한 안드로이드 악성코드 탐지 기법. 한국통신학회논문지, 40(8):1551–1559, 8 2015.

[11] 장윤 연한별. Seasonal-trend decomposition과 시계열 상관관계 분석을 통한 비정상 이벤트 탐지 시각적 분석 시스템. 정보과학회논문지, 41(12):1066–1074, 12 2014.