Memory Efficient Incremental Outlier Detection in Streaming Data

Koshal Kumar Garg

Department of Computer Science and Engineering
National Institute of Technology Rourkela

Memory Efficient Incremental Outlier Detection in Streaming Data

Thesis submitted in partial fulfillment

of the requirements for the degree of

Bachelor of Technology

in

Computer Science and Engineering

by

Koshal Kumar Garg

(Roll Number: 114CS0100)

based on research carried out

under the supervision of

Prof. Bidyut Kumar Patra

May, 2018

Department of Computer Science and Engineering

National Institute of Technology Rourkela

Department of Computer Science and Engineering
National Institute of Technology Rourkela

Prof. Bidyut Kumar Patra
Associate Professor

May 10, 2018

Supervisor's Certificate

This is to certify that the work presented in the thesis entitled *Memory Efficient Incremental Outlier Detection in Streaming Data* submitted by *Koshal Kumar Garg*, Roll Number 114CS0100, is a record of original research carried out by him under my supervision and guidance in partial fulfillment of the requirements of the degree of *Bachelor of Technology* in *Computer Science and Engineering*. Neither this thesis nor any part of it has been submitted earlier for any degree or diploma to any institute or university in India or abroad.

_____
Bidyut Kumar Patra

Dedication

You may dedicate your dissertation in this page.
Dedication should be no more than one page.
You may choose your preferred font and size.

Signature

Include this page in the pretext page count,
but do not place a page number on it.

Declaration of Originality

I, *Koshal Kumar Garg*, Roll Number *114CS0100* hereby declare that this thesis entitled *Memory Efficient Incremental Outlier Detection in Streaming Data* presents my original work carried out as a undergraduate student of NIT Rourkela and, to the best of my knowledge, contains no material previously published or written by another person, nor any material presented by me for the award of any degree or diploma of NIT Rourkela or any other institution. Any contribution made to this research by others, with whom I have worked at NIT Rourkela or elsewhere, is explicitly acknowledged in the dissertation. Works of other authors cited in this dissertation have been duly acknowledged under the sections ‘‘Reference’’ or ‘‘Bibliography’’. I have also submitted my original research records to the scrutiny committee for evaluation of my dissertation.

I am fully aware that in case of any non-compliance detected in future, the Senate of NIT Rourkela may withdraw the degree awarded to me on the basis of the present dissertation.

May 10, 2018
NIT Rourkela

Koshal Kumar Garg

Acknowledgment

First of all I would like to express my deepest respect and gratitude towards my supervisor, Dr. Bidyut Kumar Patra, who guided me throughout the past year as I was working on my final year project. I'd like to thank him for introducing me to the field of data science and machine learning, and giving me the opportunity to work under him in this urban computing project. His confidence in me and his nature of being a strict taskmaster has proven to bring out the best version of myself multiple times while working on this project. His focus on clarity of thought and topic is unmatched, which has helped me be very clear about the subject from the get-go. His expertise in his field has proved to be of immense help to me during the course of this thesis work.

I would also like to thank the PhD researchers and my fellow Masters students who made our laboratory so conducive to producing quality research work by setting benchmarks at every stage. They have been very helpful whenever I needed any help with understanding the content matter or implementations. I'd also like to thank my peers who have never hesitated while extending a helping hand whenever I was in need of guidance or support. Along with my thesis supervisor, all my faculties have played an important role in shaping me concepts over the past five years and made this project possible. Hence I'd like to extend my sincere thanks to the Prof. Durga Prasad Mohapatra, Head of the Department of Computer Science and Engineering and all department faculties for their timely co-operation and enthusiasm in giving invaluable feedback for my project work. My project would not be complete had it not been for the researchers and data scientists in working in this field and continue on their path to make the world a better place to live with the help of their research. It gives me great satisfaction that I am able to work in the same field as them, and would like to thank them for their well-documented researches. I apologize, in the event I have overlooked anybody in this section.

May 10, 2018
NIT Rourkela

Koshal Kumar Garg
Roll Number: 114CS0100

Abstract

Write the abstract of the dissertation followed by 3 to 7 keywords or phrases. An abstract is a micro dissertation. Both the dissertation and abstract should answer the following few questions —

- What was done?
- Why was it done?
- How was it done?
- What was found?
- What is the significance of the findings?

In the abstract section, one should answer the above questions in short paragraphs. The total length of the abstract should typically be limited to two pages.

Mention 3 to 7 keywords, phrases, or index terms in ***bold-italics*** separated by semicolons. These words should be carefully chosen in a manner that they convey sufficient information on what the dissertation is all about. These words help other researchers to search and find your work.

***Keywords*: KW_1; KW_2; KW_3; KW_4; KW_5.**

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Outlier detection is used in variety of applications, such as fraud detection for credit cards, insurance, or health care, intrusion detection for cyber-security, fault detection in safety critical systems, and military surveillance for enemy activities. However, outlier detection on streaming data is particularly challenging, since the volume of data to be analyzed is effectively unbounded and cannot be stored indefinitely in memory for processing [4]. Data streams are generated at a high data rate and hence the computation speed and efficiency of algorithm has to be high. An outlier detection system in wireless sensor networks must work with the limited memory in each sensor node in order to detect rare events in near real time. In the case of data streams, where the number of data points is unbounded and can arrive at a high rate, keeping all data points is impossible. Simply deleting some of the points does not help because it may affect the accuracy and detection efficiency of upcoming points. Deleting previous points can cause two problems: i) Deleting the previous data points decreases the detection accuracy of local outlier factor for new data points, ii)We can not differentiate between past events and new events.

Anomaly detection refers to the problem of finding patterns in data that do not conform to expected behavior. These nonconforming patterns are often referred to as anomalies, outliers, discordant observations, exceptions, aberrations, surprises, peculiarities, or contaminants in different application domains. [1]. Anomalies are patterns in data that do not conform to a well defined notion of normal behavior.

1.1 Thesis Organization

In this project we are using relative density based anomaly detection technique, LOF to calculate the anomaly score of data points. The objective of this project is to use LOF technique to find outliers in streaming data. e work done in this project can be summed up as follows:

Chapter 2

Literature Survey

Outside the domain of physical formatting and layout, this document does not intend to foray into the contents of the dissertation to be written. That would influence contents and stubble creativity. The following unstructured guidelines, however, may sometime prove useful to the students and their guides, who are looking for a good model.

Following papers were studied

2.1 Anomaly Detection: A Survey

2.1.1 Aspects of Anomaly Detection

1. *Nature of Input Data* : Input is generally a collection of data instances. Each data instance contains some attribute values. Attributes can be of different type such as binary, categorical, or continuous.For nearest-neighbor-based techniques, the pairwise distance between instances might be provided in the form of a distance or similarity matrix. In such cases, techniques that require original data instances are not applicable.

2. *Type of Anomaly* : Anomalies are of three types

   (a) *Point Anomalies* : If a single data instance is considerd outlier to rest of the data points then it is called point anomaly. For examle in credit card fraud detection the data instance the amount spent is very high compared to the normal range of expenditure for that person will be a point anomaly.

   (b) *Contextual Anomalies* : If a data point is considered anomaly in certain context but not otherwise then it is called contextual or conditional anomaly. For example 1000 dollar expense per week may be considered anomaly in a normal week but not in Christmas week.

   (c) *Collective Anomalies* : If a collection of related data points are anomalous with respect to entire data set then they are termed as collective anomalies. Individually these data points may not be anomalous but taking together they are outliers.

3. *Data Labels* Data labels for each instance shows whether it is a normal point or anomalous. Labeling is often done manually by a human expert which is clearly very expensive and time taking. Getting a training data set that contains the representative of all possible outliers is very difficult. Based on the extend of availability of labeled data, anomaly detection technique can operate in three modes

   (a) *Supervised Anomaly Detection* : Training data set that has labeled instances for both normal and anomalous class. Any new data instances is classified in one among these two classes. Major problem with this approach is to get labeled data and anomalous instances are very few as compared to normal instances. This imbalanced class distribution reduces the accuracy.

   (b) *Semisupervised Anomaly Detection.* : Training data set has labeled instances for either normal or anomalous class only. It is very difficult to put representative instances for all possible outliers that can occur in the data.

   (c) *Unsupervised Anomaly Detection* : In unsupervised anomaly detection there is no training data. This works with the assumption that normal instances are far more than anomalies in test data.

4. *Output of Anomaly Detection*

   (a) *Scores* : Score is assigned to each instance depending on the degree to which that instance is considered anomaly. Selecting the cutoff for anomaly is decided based on the number of top anomalous points.

   (b) *Label* : Binary label is assigned to each instance whether it is normal or anomalous.

2.1.2 Anomaly Detection Techniques

1. *Nearest Neighbor Based* : Nearest neighbor based anomaly detection techniques works on the assumption that normal data points occur in dense clusters and oultliers lie far from their neighbors. It requires a distance or similarity measure defined between two data instances. Nearest neighbor based technique is broadly classified in two types

   (a) *Using Distance to K^{th} Nearest Neighbor* The anomaly score of an instance is the distance between the point and it's K^{th} nearest neighbor. A different approach to compute anomaly score is to count no of instances in a radius of d.

   (b) *Using Relative Density* : An instance that lies in a neighborhood of less density is considered outlier while an instance that lies in a dense neighborhood is considered normal. But this technique fails for dataset with varying densities.

To address this problem the density of instances relative to their neighborhood is computed.

To assign a relative density score, Local Outlier Factor was proposed. For any given data instance, the LOF score is equal to ratio of average local density of the k nearest neighbors of the instance and the local density of the data instance itself[2].

2. *Clustering Based Anomaly Detection Techniques* :

   Clustering is grouping similar data points into clusters. Clustering is unsupervised. These techniques work on the assumption that

   - Normal points lie inside clusters and anomalous points does not belong to any cluster.
   - Normal points lie closer to the cluster center while outliers lie far away from the cluster centers.
   - Normal data points belongs to large and dense clusters while outliers belong to small and parse clusters.

2.2 LOF: Identifying Density-Based Local Outliers

the degree depends on how isolated the object is with respect to the surrounding neighborhood. The outlier factor is local in the sense that only a restricted neighborhood of each object is taken into account.We show that for most objects in a cluster their LOF are approximately equal to 1.

Local density based outlier detection is preferred because global density based techniques performs poorly for datasets with varying densities. Consider the 2-D data set shown in Figure 1. Both points p1 and p2 should be detected as outliers. But if we consider global density, p2 will not be detected as outlier because of low density of cluster C1. For a point q in cluster C1, the distance between q and its nearest neighbor will always be greater than distance between p2 and its nearest neighbor.

- **K-distance** :The distance between a data point p and its K^{th} nearest neighbor (K-NN).
- **Reachability distance (reach-dist)** of a data point p with respect to another data point o

$$reach - dist_K(p, o) = max\{k - distance(o), d(p, o)\}$$

where d(p,o) is the euclidean distance between p and o.

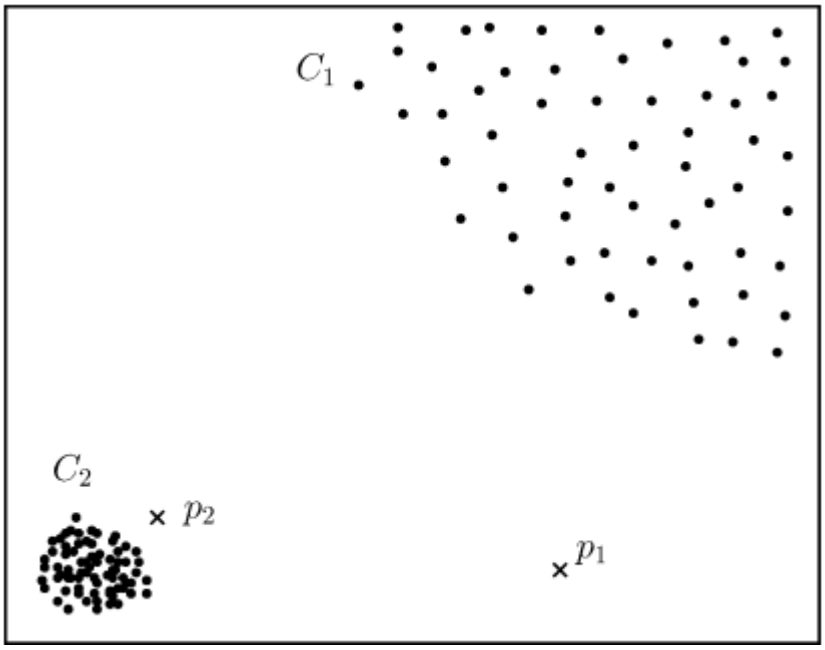

Figure 2.1: Result

- **Local Reachability Density(lrd)** of a data point p

$$lrd_k(p) = \left(\frac{1}{K} \sum_{o \in N_{(p,k)}} reach - dist(p, o) \right)^{-1}$$

where $N_{(p,k)}$ is the set of k nearest neighbors of p.

- **Local Outlier Factor(LOF)** of a data point p

$$LOF_K(p) = \frac{1}{K} \sum_{o \in N_{(p,k)}} \frac{lrd_K(o)}{lrd_K(p)}$$

LOF is a score assigned to each instance that shows the degree of anomalous behavior. Instances inside a cluster have LOF value nearly equal to 1.

2.3 Incremental Local Outlier Detection for Data Streams

Incremental LOF for data streams refer to computation LOF score of newly added data points and updating the same for older data points. Outlier detection techniques are categorized into 4 types

1. *Statistical Approach* : the data points are typically modeled using a stochastic distribution, and points are labeled as outliers depending on their relationship with this model.

2. *Distance Based Approach* : detect outliers by computing distances among points.

3. *Profiling Methods* : Profiles of normal behavior is built and deviation from that are considered outliers.

4. *Model-based Approach* : First characterization of normal points using predictive models like neural networks and SVM. Deviation from these models are considered outliers.

Static LOF algorithms can be applied to data stream is three ways. However thwese are computationally inefficient.

1. *Periodic LOF* : Static LOF is applied periodically on entire data set every time new data blocks are entered. The problem with such approach is that a data point which may be anomalous while added may not be detected as outlier later as many other points are added. they may create a cluster of their own and outliers will be missed. If LOF is computed each time a data point is added, the change is behavior can be detected at that moment.

2. *Supervised LOF* : K-distance, lrd and LOF of the data points are precomputed. Any new data point that arrives all these parameters are computed without updating for previous points. As a result the LOF accuracy decreases.

3. *Iterated LOF* : Aplpy static LOF every time a new data point enters. This gives high accuracy at a cost of very high computational time.

In Incremental approach, LOF score is computed as soon as a data enters. It is determined whether it is outlier or not. LOF of other data points are also updated.

2.3.1 Insertion

K-distance, lrd, LOF are computed for each inserted data point and updated for affected points.

2.4 Memory Efficient Incremental Local Outlier Factor (MiLOF)

Local Outliers detection in data streams when limited memory is available. It is impractical to store all the instances of data streams. Some of the points has to be removed and summarized. Memory available is just sufficient to store K-Distance, lrd and LOF of m data points only.

Algorithm 1 iLOF Insertion

INPUT: A data point p_t at time t
OUTPUT: LOF value LOF(p_t)

Compute KNN and K-distance of p_t
for all o ∈KNN(p_t) **do**
  Compute `reach-dist(`p_t`,o)`
end for
S_{update} ←Reverse KNNs of p_t
for all o ∈S_{update} and q ∈$N_{(o,k)}$ **do**
  Update K-distance(o) and reach-dist(q,o)
  if o ∈$N_{(q,k)}$ **then**
    S_{update} ←S_{update} ∪q
  end if
end for
for all o ∈S_{update} **do**
  Update lrd(o) and LOF(R-KNN(o,k))
end for
Compute lrd(p_t) and LOF(p_t)
return LOF

The available m memory is divided in two parts of size b and c to store original points and summarized points respectively. When memory limit is reached, first $\frac{b}{2}$ data points are summarized to c clusters and deleted to free memory. If there already exists summarized data, the old and new cluster centers are merged. Hence at any time maximum memory used is b=m+c. Three steps of MiLOF are

2.4.1 Summarization

Whenever memory reaches limit b, summarization phase is invoked. This phase summarizes first $\frac{b}{2}$ points, their K-distance, lrd and LOF and deletes these points from memory. Recent points are retained because data points might have evolved with time and recent points are most important. As the width of the summarization window decreases, it resembles iLOF. So MiLOF is direct generalization of iLOF.

Summarization is explained in algorithm 2. In summarization, if memory is reached first $\frac{b}{2}$ points are summarized to c clusters. K-distance, lrd and LOF for these cluster centers are computed by the following formulas.

- K-Distance of cluster center $v_j^i \in V^i$

$$K - Distance(v_j^i) = \frac{\sum_{p \in C_j^i} K - Distance(p)}{|C^i|}$$

- lrd of cluster center $v_j^i \in V^i$

$$lrd_k(v_j^i) = \frac{\sum_{p \in C_j^i} lrd_k(p)}{|C^i|}$$

- LOF of cluster center $v_j^i \in V^i$

$$LOF_k(v_j^i) = \frac{\sum_{p \in C_j^i} LOF_k(p)}{|C^i|}$$

Algorithm 2 MiLOF

INPUT: A data point p
OUTPUT: LOF value LOF(p)

Compute LOF of p
if No of points=b **then**
  $C^i \leftarrow$ First $\frac{b}{2}$ points
  (V^i,N^i) $\leftarrow$ C-means(C^i)
  for all $v^i \in V^i$ **do**
    Compute avereage K-distance, lrd and LOF
  end for
  Delete C^i
  if i>0 **then**
    (Z;W) $\leftarrow$ Weighted c-means(V^i U V^{i-1} ,N^i U N^{i-1})
    for all z $\in$Z **do**
      Compute avereage K-distance, lrd and LOF
    end for
    $V^{i-1} \leftarrow$Z
    Delete V^{i-1},Z
  end if
end if
i $\leftarrow$ i+1
return LOF

2.4.2 Merging

Summarization is performed every time new $\frac{b}{2}$ new data points arrives. Clustering these points gives c cluster centers V^i. These clusters are to be merged with old c cluster centers $Vi-1$ so that finally only c centers are available. Cluster centers X=V^i U V^{i-1} are merged using a weighted clustering algorithm where weight of each center is no of objects in that cluster.

2.4.3 Revised Insertion

Whenever a new data points arrives, LOF value for it is computed similar to iLOF with only difference that iLOF uses only data points to compute LOF where as in revised insertion we use both the data points and the cluster centers. While calculating the KNN, if any of the point is cluster center then it is assumed that all other nearest neighbor belongs to the same cluster. Hence distance from the point and that cluster center is taken as the K-distance for that point.

Chapter 3

Implementation and Results

3.1 LOF

We have used a synthetic 2-D dataset with 3 clusters of varying density to implement and verify LOF. Dataset description
No of attributes : 2
No of instances : 1500
No of clusters : 3
Cluster 1 : 750 points in range [600,1000]
Cluster 2 : 500 points in range [200,450]
Cluster 3 : 200 points in range [0,100]
Noisy points : 50

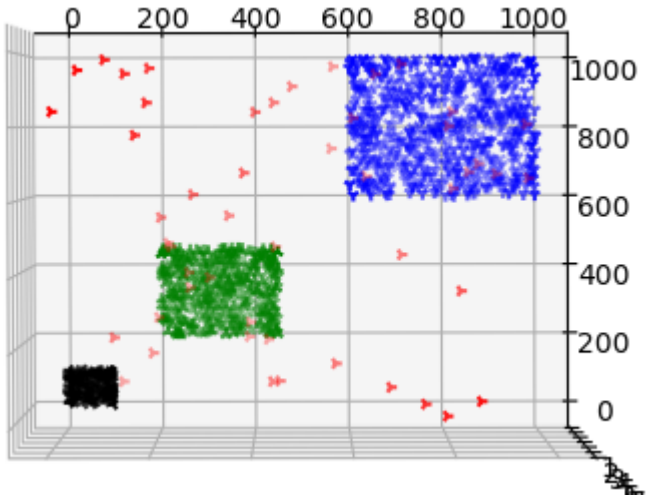

Figure 3.1: 2-D Dataset used for LOF

It can be seen from the plot that points inside a cluster has LOF values nearly equal to 1 where as noisy points have LOF upto 7. More the LOF score more is the degree of anomaly.

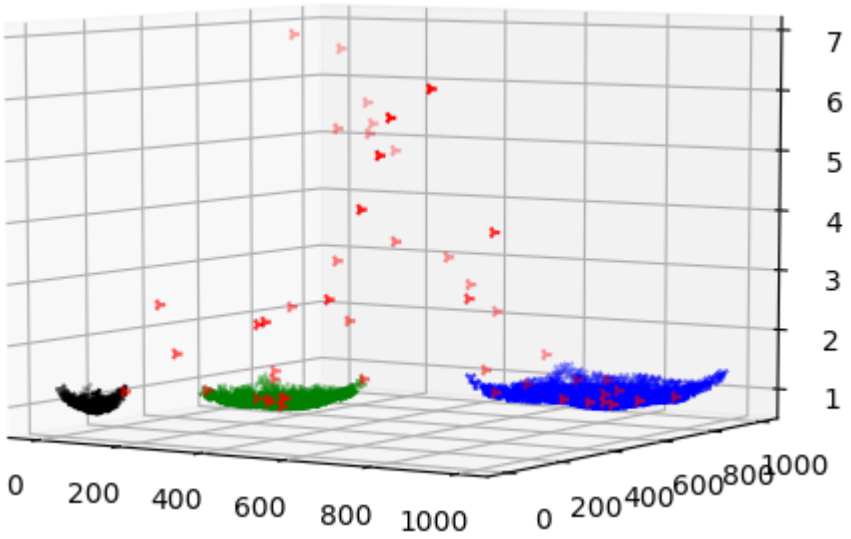

Figure 3.2: LOF values for synthetic 2-D dataset

References

[0]This reference format follows ASME style. You are advised to follow one reference format of any dominant journal of your field.

Dissemination

Internationally indexed journals (*Web of Science, SCI, Scopus, etc.*)[1]

1.

2.

Other journals and Book chapters [1]

1.

2.

Conferences [1]

1.

2.

Article under preparation [2]

1.

2.

[1] Articles already published, in press, or formally accepted for publication.
[2] Articles under review, communicated, or to be communicated.