IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727
PP 61-64
www.iosrjournals.org

Angle Based Outlier Detection in Big Data Analytics

Ms.R.Divya[1], Mr.S.S.Saravana Kumar[2]
[1](Research Scholar, Department of Computer Science, Kovai kalaimagal College of Arts & Science, India)
[2](Asst. Professor, Department of Information Technology, Kovai kalaimagal College of Arts & Science, India)

Abstract : *In this paper will tackle the challenges of big data analytics in the algorithmic aspects. To handle design and evaluate scalable and efficient algorithms that are able to handle complex data analysis tasks, involving big datasets without extreme use of computational resources. In wide range of application domains, data are represent as high-dimensional vectors in the Euclidean space in organize to benefit from computationally advanced techniques from numerical linear algebra. Our solutions have leveraged simple but fast randomized numerical linear algebra techniques to approximate essential properties of data, such as data norm, pairwise Euclidean distances and dot products. These relevant and useful approximation properties will be used to explain fundamental data analysis tasks in data mining, machine learning and information recovery.*

Keywords: *Big data, Analytics Tools, ABOD, Algorithm, Dataset, Experiments*

I. Introduction

The randomized algorithms are very simple and easy to program. They are also well suited to massively parallel computing environments so that we can exploit distributed parallel architectures for big data. This means that it can trade a small loss of accuracy of results in order to achieve substantial parallel and sequential speedups. Although the found patterns or learned models may have some possibility of being incorrect, if the probability of error is sufficiently small then the dramatic improvement in both CPU and I/O performance may well be worthwhile. In addition, such results can help to speed up interacting with the domain experts to evaluate or adjust new found patterns or learned models.The paper consists of two parts. The first thing is fundamentals of high-dimensional vector in the Euclidean space, and core randomized techniques. The second part contains how advanced randomized techniques, e.g. sampling and sketching, can be applied to solve fundamental data analysis tasks, including outlier detection, classification, and similarity search

.

1.1 Introduction to Big Data

This section present basic definition of high-dimensional vectors in the Euclidean space, and essential concepts widely used in data analysis applications. These concepts, including nearest neighbor search, outlier detection and classification, are taxing problems in data analysis that the thesis aims at solving. To introduce core randomized techniques including random projection, sampling and sketching via hashing mechanism. These randomized techniques are used as powerful algorithmic tools to deal with the data analysis problems in this paper. Big data analytics is the process of probing large and various datasets i.e., big data to uncover hidden patterns, unknown correlations, market trends, customer favorite and other useful information that can help organizations make more-informed business decisions.

1.2 Big Data Analytics Benefits

Driven by focused analytics systems and software, big data analytics can spot the way to various business benefits, including new revenue chances, more effective marketing, improved customer service, enhanced operational efficiency and competitive advantages over rivals.Big data analytics applications permit data scientists, predictive modelers, statisticians and other analytics professionals to analyze increasing volumes of structured transaction data, plus other forms of data that are often left unused by conventional business intelligence (BI) and analytics programs. BI queries answer to basic questions about business operations and performance. Big data analytics is a type of advanced analytics, which involves difficult applications with elements such as predictive models, statistical algorithms and what-if analyses power-driven by high-performance analytics systems.Randomization techniques particularly find out among many other approximation techniques to improve the scalability of first-order methods since we can control their expected behavior.

1.3 Big Data Analytics Technologies And Tools

Unstructured and semi-structured data types usually don't fit well in traditional data warehouses that are based on relational databases oriented to structured datasets. Furthermore, data warehouses may not be capable to handle the processing demands posed by sets of big data that need to be updated frequently -- or even continually, as in the case of real-time data on stock trading, the online activities of website visitors or the performance of portable applications. As a result, many organizations that collect method and evaluate big data turn to NoSQL databases as well as Hadoop and its companion tools, including:

1. **YARN**: A cluster management technology and one of the key features in second-generation Hadoop.
2. **MapReduce**: A software framework that permit developers to write programs that process huge amounts of unstructured data in parallel across a distributed cluster of processors or separate computers.
3. **Spark**: An open-source parallel processing framework that allow users to run large-scale data analytics applications crosswise clustered systems.
4. **HBase**: A column-oriented key/value data store built to lope on top of the Hadoop Distributed File System (HDFS).
5. **Hive**: An open-source data warehouse system for querying and analyzing huge datasets stored in Hadoop files.
6. **Kafka**: A distributed publish-subscribe messaging system designed to replace conventional message brokers.
7. **Pig**: An open-source technology that presents a high-level mechanism for the parallel programming of MapReduce work to be executed on Hadoop clusters.

II. Angle-Based Outlier Detection

Outlier mining in d-dimensional point sets is a fundamental and well studied data mining task due to its variety of applications. Most such applications arise in high-dimensional domains. A bottleneck of existing approaches is that implicit or explicit assessments on concepts of distance or nearest neighbour are deteriorated in high-dimensional data. Following up on the work of Kriegel et al. (KDD'08), we investigate the use of angle-based outlier factor in mining high-dimensional outliers.While their algorithm runs in cubic time (with a quadratic time heuristic), we propose a novel random projection-based technique that is able to estimate the angle-based outlier factor for all data points in time near-linear in the size of the data. Also, our approach is suitable to be performed in parallel environment to achieve a parallel speedup. Introduce a theoretical analysis of the quality of approximation to guarantee the reliability of our estimation algorithm. The empirical experiments on synthetic and real world datasets demonstrate that our approach is efficient and scalable to very large high-dimensional datasetsFor example, consider the problem of fraud detection for credit cards and the dataset containing the card owners' transactions. The transaction records consist of usage profiles of each customer corresponding the purchasing behavior. The purchasing behavior of customer usually changes when the credit card is stolen. The abnormal purchasing patterns may be reflected in transaction records that contain high payments, high rate of purchase or the orders comprising large numbers of duplicate items, etc.The smaller the angle spectrum of an object to other pairs of objects is, the more likely it is an outlier. Because angles are more stable than distances in high-dimensional space [4], this approach does not substantially deteriorate in high-dimensional data.

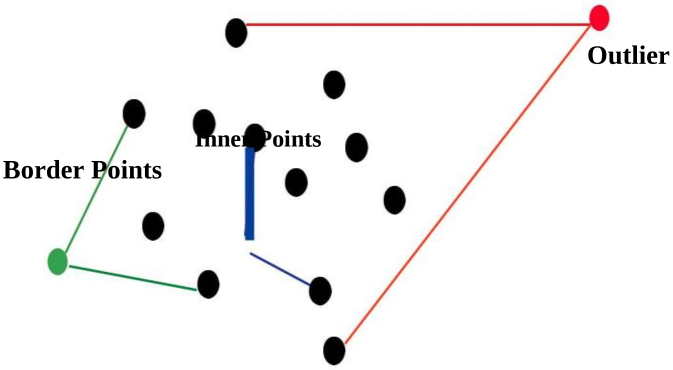


Fig.1: An Intuition of Angle-based Outlier Detection

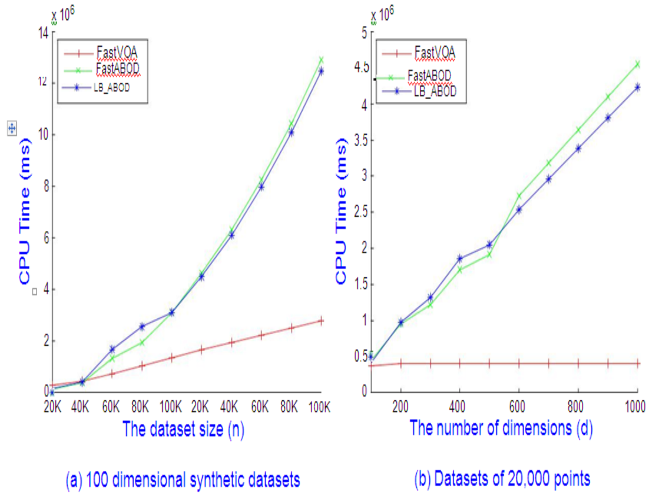
The main technical insight is the combination between random hyperplane projections [2, 6] and AMS Sketches on product domains [1, 7], which enables us to reduce the computational complexity from cubic time complexity in the nave approach to near-linear time complexity in the approximation solution. Another advantage of our algorithm is the suitability for parallel processing.

III. Algorithm Overview and Preliminaries

The general idea is to efficiently compute an unbiased estimator of the variance of the angles for each point of the dataset. In other words, the expected value of estimation is equal to the variance of angles, and it is concentrated around its expected value. These estimated values are then used to rank the points. The top m points having the smallest variances of angles are retrieved as top m outliers of the dataset.In order to estimate the variance of angles between a point and all other pairs of points, That first dataset on the hyperplanes orthogonal to random vectors whose coordinates are chosen from the standard normal distribution N(0:1). Based on the partitions of the dataset after projection is able to estimate the unbiased mean of angles for each point. An approximate the second moment and derive its variance by applying the AMS Sketches to summarize the frequency moments of the points projected on the random hyperplanes. The combination between random hyperplane projections and AMS Sketches on product domains enables us to reduce the computational complexity to O(n log n(d + log n)) time with some basic notions of random hyperplane projection and AMS Sketch, then the algorithm used to estimate the variance of angles for each point of the dataset.

3.1 Experiments

In this paper implemented algorithms in C++ and conducted experiments in a 2.67 GHz core i7 Windows platform with 3GB of RAM on both synthetic and real world datasets. All results are over 5 runs of the algorithms.For the sake of fair comparison, we made use of the same synthetic data generation process as the ABOD approach [8]. The Gaussian mixture including 5 equally weighted clusters having random means and variances as normal points and employed a uniform distribution as the outliers. All points were generated in full-dimensional space. For each synthetic dataset and generated 10 outliers which are independent on the Gaussian mixture to evaluate the performance of all algorithms on synthetic datasets with varying sizes and dimensions.For the real world high-dimensional datasets, used to pick three datasets (Isolet, Multiple Features and Optical Digits) designed for classification and machine learning tasks from UCI machine learning repository [5]. Isolet contains the pronunciation data of 26 letters of the alphabet while Multiple Features and Optical Digits consist of the data of handwritten numerals (`0' - `9'). For each dataset all data points from some classes having common behaviors as normal points and 10 data points from another class as outliers.



(a) 100 dimensional synthetic datasets

(b) Datasets of 20,000 points

Fig.2: Comparison of CPU time of FastVOA, FastABOD and LB_ABOD

This section compares the running time of algorithms, namely FastVOA, LB_ABOD [3] and FastABOD [8] on the large high-dimensional datasets. In fact, there are very few large real world datasets where the outliers are identified exactly in advance to evaluate the efficiency of these 3 approaches on synthetic datasets. This experiment measured the CPU time of each approach on datasets with varying both size and dimensions in ranges 10,000 - 100,000 points and 100 - 1000 respectively.

IV. Conclusion

This paper analyzed a random projection-based algorithm to approximate the variance of angles between pairs of points of the dataset, a robust outlier score to detect high-dimensional outlier patterns. By combining random projections and AMS Sketches on product domains, our approximation algorithm runs in near-linear time in the size of dataset and is suited for parallel processing. The empirical experiments on synthetic and real world datasets demonstrate the scalability, effectiveness and efficiency on detecting outliers in very large high-dimensional datasets.

Bibliography

[1]. M. Charikar. Similarity estimation techniques from rounding algorithms. In Proceedings of STOC'02, pages 380 388, 2002.

[2]. A.Das, M. Datar, A. Garg, and S. Rajaram. Google news personalization: scalable online collaborative filtering. In Proceedings of WWW'07, pages 271 280, 2007.A

[3]. Hinneburg, C. C. Aggarwal, and D. A. Keim. What is the nearest neighbor in high dimensional spaces? In Proceedings of VLDB'00, pages 506 515, 2000.

[4]. P. Indyk and A. McGregor. Declaring independence via the sketching of sketches. In Proceedings of SODA'08, pages 737 745, 2008.

[5]. P. Li and A. C. Konig. b-bit minwise hashing. In Proceedings of WWW'10, pages 671 680, 2010.

[6]. V. Mahadevan, W. Li, V. Bhalodia, and N. Vasconcelos. Anomaly detection in crowded scenes. In Proceedings of CVPR'10, pages 1975 1981, 2010.

[7]. K. I. Williams and M. Seeger. Using the Nystrom method to speed up kernel machines. In Advances in NIPS'01, pages 682 688, 2001.

[8]. Statnikov A, Aliferis C, Tsamardinos I, Hardin D, Levy S (2005)," A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis", Bioinformatics 21: 631–643.

[9]. C. Aggarwal and P. S. Yu. Outlier detection for high dimensional data. In Proceedings of SIGMOD'01, pages 37 46, 2001.

[10]. N. Ailon and B. Chazelle. Approximate nearest neighbors and the fast johnson-lindenstrauss transform. In Proceedings of STOC'06, pages 557 563, 2006.

[11]. N. Alon, Y. Matias, and M. Szegedy. The space complexity of approximating the frequency moments. In Proceedings of STOC'96, pages 20 29, 1996.

[12]. S. D. Bay and M. Schwabacher. Mining distance-based outliers in near lin-ear time with randomization and a simple pruning rule. In Proceedings of KDD'03, pages 29 38, 2003.

[13]. Hinneburg, C. C. Aggarwal, and D. A. Keim. What is the nearest neighbor in high dimensional spaces? In Proceedings of VLDB'00, pages 506 515, 2000.

[14]. M. Knorr and R. T. Ng. Algorithms for mining distance-based outliers in large datasets. In Proceedings of VLDB'98, pages 392 403, 1998.