


SURVEY PAPER

Open Access



A comprehensive survey of anomaly detection techniques for high dimensional big data

Srikanth Thudumu^{1*} , Philip Branch¹, Jiong Jin¹ and Jugdutt (Jack) Singh²

*Correspondence:
sthudumu@swin.edu.au
¹ School of Software &
Electrical Engineering,
Swinburne University
of Technology, Hawthorn, VIC
3122, Australia
Full list of author information
is available at the end of the
article

Abstract

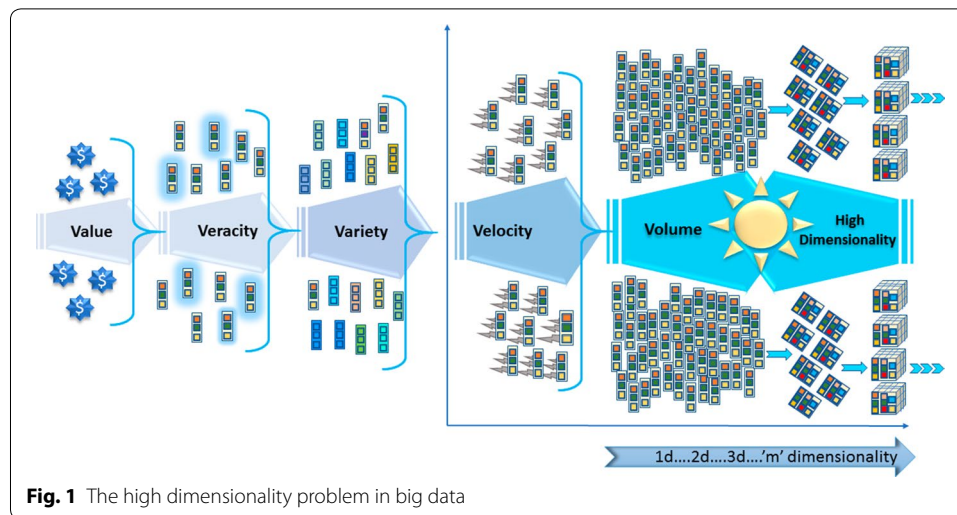
Anomaly detection in high dimensional data is becoming a fundamental research problem that has various applications in the real world. However, many existing anomaly detection techniques fail to retain sufficient accuracy due to so-called “big data” characterised by high-volume, and high-velocity data generated by variety of sources. This phenomenon of having both problems together can be referred to the “curse of big dimensionality,” that affect existing techniques in terms of both performance and accuracy. To address this gap and to understand the core problem, it is necessary to identify the unique challenges brought by the anomaly detection with both high dimensionality and big data problems. Hence, this survey aims to document the state of anomaly detection in high dimensional big data by representing the unique challenges using a triangular model of vertices: the problem (big dimensionality), techniques/algorithms (anomaly detection), and tools (big data applications/frameworks). Authors’ work that fall directly into any of the vertices or closely related to them are taken into consideration for review. Furthermore, the limitations of traditional approaches and current strategies of high dimensional data are discussed along with recent techniques and applications on big data required for the optimization of anomaly detection.

Keywords: Anomaly detection, Big data, Big dimensionality, Big dimensionality tools, High dimensionality, The curse of big dimensionality, The curse of dimensionality

Introduction

Many data sets continuously stream from weblogs, financial transactions, health records, and surveillance logs, as well as from business, telecommunication, and biosciences [1, 2]. Referred to as “big data,” a term that describes the large and distributed nature of the data sets, this area has recently become a focus of scholarship. Gartner [3] defines big data as high-volume, high-velocity, and high-variety data sets that demand cost-effective novel data analytics for decision-making and to infer useful insights. In recent years, the core challenges of big data have been widely established. These are contained within the five Vs of big data—value, veracity, variety, velocity, and volume [4]—as shown in Fig. 1.

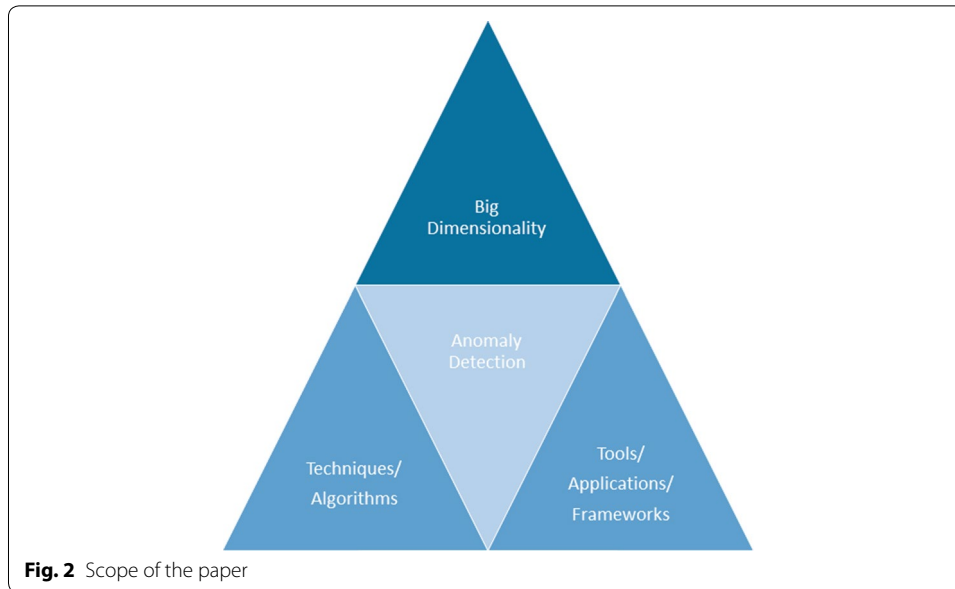
Value refers to the benefits associated with the analysis of data; veracity refers to the accuracy of the data; and variety refers to the many types of data, such as structured,



semi-structured, or unstructured. Volume is the amount of data that is being accumulated (i.e., the size of the data)—the larger the dimensionality of the data, the larger the volume. Dimensionality refers to the number of features or attributes or variables within the data available for analysis. By contrast, velocity refers to the “speed” at which the data are generated and may contain many dimensions. These elements of the current definition of big data address basic challenges. However, such a definition ignores another important aspect: “dimensionality,” which plays a crucial role in real-world data analysis [4, 5]. The increase in dimensions or features or attributes poses significant challenges for anomaly detection in large data sets.

Anomaly detection aims to detect abnormal patterns deviating from the rest of the data, called anomalies or outliers. High dimensionality creates difficulties for anomaly detection because, when the number of attributes or features increase, the amount of data needed to generalize accurately also grows, resulting in data sparsity in which data points are more scattered and isolated. This data sparsity is due to unnecessary variables, or the high noise level of multiple irrelevant attributes, that conceal the true anomalies. This issue is widely acknowledged as the “curse of dimensionality” [6, 7]. It is an obstacle for many anomaly detection techniques addressing high dimensionality that fail to retain the effectiveness of conventional approaches, such as distance-based, density-based, and clustering-based techniques [8].

This survey aims to document the state of anomaly detection in high dimensional big data by identifying the unique challenges using a triangular representation of vertices: the problem (big dimensionality), techniques/algorithms (anomaly detection), and tools (big data applications/frameworks). Authors’ work that falls directly into any of the vertices or closely related to them are taken into consideration for review (see Fig. 2). Furthermore, the limitations of traditional approaches and current strategies of high dimensional data are discussed along with the recent techniques and applications on big data required for the optimization of anomaly detection. Hence, the contributions of this survey are threefold:



1. Review existing literature to highlight the theoretical background and current strategies for managing the big dimensionality problem.
2. Identify unique challenges and review the techniques/algorithms of anomaly detection that address the problems of high dimensionality and big data.
3. Review the big data tools, applications and frameworks for anomaly detection in high dimensional big data.

This paper highlights techniques concerning anomaly detection as well as covering other closely related machine learning fields, such as pattern recognition, outlier detection, spam detection, suspicious detection, fraud detection, deep learning and novelty recognition.

The rest of the paper is structured as follows. “[Related work and scope](#)” section provides an overview of differences between this survey and existing surveys. “[Anomaly detection in big data](#)” section presents the introduction of big dimensionality problem in anomaly detection. “[The high dimensionality problem](#)” section presents the theoretical background including visualization and the curse of dimensionality, and discusses the challenges of traditional models dealing with high dimensionality in anomaly detection. “[Strategies for tackling the problem of high dimensionality](#)” section provides strategies to tackle the high dimensionality problem, such as dimensionality reduction methods to ascertain the benefit of one approach over another. “[Unique challenges brought by anomaly detection in high-dimensional big data](#)” section provides a taxonomy of big data anomaly detection problems with regard to high dimensionality in the form of a survey of closely related work. “[Tools in high dimensional big data](#)” section provides the review of big data tools and frameworks handling big dimensional data. The final section concludes the survey and considers the direction of future research.

Related work and scope

Anomaly detection in high-dimensional data is becoming a fundamental research area that has various applications in the real world. As such, a large body of research has been devoted towards addressing this problem. Nevertheless, most existing surveys focus on the individual aspects of anomaly detection or high dimensionality. For example, Agrawal and Agrawal [9] provide a review of various anomaly detection techniques, with the aim of presenting a basic insight into various approaches for anomaly detection. Akoglu et al. [10] present several real-world applications of graph-based anomaly detection and concluded with open challenges in the field. Chandola et al. [11] present a survey of several anomaly detection techniques for various applications. Hodge and Austin [7] present a survey of outlier detection techniques by comparing techniques' advantages and disadvantages. Patcha and Park [12] have conducted a comprehensive survey of anomaly detection systems and hybrid intrusion detection systems by identifying open problems and challenges. Jiang et al. [13] present a survey of advanced techniques in detecting suspicious behaviour; they also present detection scenarios for various real-world situations. Sorzano et al. [14] categorize dimensionality reduction techniques, along with the underpinning mathematical insights. Various other surveys can also be observed, such as those by Gama et al. [15], Gupta et al. [16], Heydari et al. [17], and Jindal and Liu [18], Pathasarathy [19], Phua et al. [20], Tamboli et al. [21], and Spirin et al. [22], which further highlight the problems either in anomaly detection or in high-dimensional data.

A limited amount of work has been done that emphasizes anomaly detection and high dimensionality problems together, either directly or indirectly. Zimek et al. [23] present a detailed survey of specialized algorithms for anomaly detection in high-dimensional numerical data; they also highlight important aspects of the curse of dimensionality. Parsons et al. [24] present a survey of the various subspace clustering algorithms for high-dimensional data and discuss some potential applications in which the algorithms can be used.

To the best of the authors' knowledge, no surveys directly highlight the problems of anomaly detection and high dimensionality in big data. In this survey, we present an integrated overview of these two problems from the perspective of big data. Table 1 summarises the differences between this survey and other related works.

Anomaly detection in big data

Anomaly detection is an important technique for recognizing fraud activities, suspicious activities, network intrusion, and other abnormal events that may have great significance but are difficult to detect [25]. The significance of anomaly detection is that the

Table 1 Comparison of our survey to other related survey articles

Surveys	Anomaly detection	High dimensionality problem	Big data aspects
[7, 9–11, 14, 20]	✓	✗	✗
[13, 17, 18, 22–24]	✓	✓	✗
[15, 16, 19, 21]	✓	✗	✓
Our survey	✓	✓	✓

process translates data into critical actionable information and indicates useful insights in a variety of application domains [11]. For example, cancer treatment plans need to be formulated based on the readings from an Intensity-modulated Radiation Therapy (IMRT) machine [26]; locating anomalies is critical before recommending the amount of radiation for a patient. Even if part of the treatment is based on anomaly detection, the accuracy of the data plays a crucial role and may have negative consequences if the data are analyzed ineffectively. As mentioned earlier, Chandola et al. [11] provided a survey, taxonomy, and analysis of several anomaly detection techniques for various applications, such as manufacturing defects, sensor networks, intrusion detection, and finding abnormal behavior of the data. Most of the applications were important for high dimensional data sets that contained thousands or even millions of attributes. The traditional techniques that detect anomalies in high-dimensional space are complicated, as anomalies are rare and generally appear in fractional views of subsets of dimensions or subspaces [27]. It has also been suggested by Aggarwal [28] that almost any anomaly detection algorithm based on the concept of proximity degrades qualitatively in high-dimensional space; therefore, redefinition of the algorithm is necessary. Furthermore, traditional methods become less meaningful with increasing dimensionality, as they use strategies that make certain assumptions about the comparatively low dimensionality of the data [29]. Moreover, it is possible that only a fraction of data points are informative for data sets with high dimensionality [28].

Anomaly detection that addresses problems of high dimensionality can be applied in either online or offline modes. In an offline mode, anomalies are detected in historical data sets known as “batch processing.” This relates to the “volume” feature of big data. By contrast, in online mode, new data points, known as “data streams,” are continually introduced while anomalies are being detected. This relates to the “velocity” feature of big data. A number of existing surveys and reviews highlight the problem of high dimensionality for various fields, such as machine learning and data mining. The “size” aspect of the volume [30–34] and “speed” aspect of the velocity [35–40] are frequently addressed in the literature; however, the “dimension” aspect remains largely ignored. Zhai et al. [4] termed this gap “big dimensionality” and we attempt to review the techniques that are related to big dimensionality problem. However, to derive the unique challenges brought by anomaly detection in big dimensionality and to understand the core problem hindering the performance and accuracy of the available techniques, the theoretical background and current strategies are presented in the following sections.

The high dimensionality problem

High dimensionality refers to data sets that have a large number of independent variables, components, features, or attributes within the data available for analysis [41]. The complexity of the data analysis increases with respect to the number of dimensions, requiring more sophisticated methods to process the data. Data sets are growing in terms of sample size n but even more in terms of dimension m . At the same time, in the era of big data, m is commonly misconstrued as high-dimensional, since thousands of dimensions are very common. If a data set has n samples and m dimensions (or features), then the data can be referred to as m -dimensional. In general, a data set can be

called high dimensional when the number of dimensions m causes the effect of ‘curse of dimensionality’.

One of the main sources of high-dimensional data sets are organizations with huge transactions and associated databases. Furthermore, it is common to have multiple dimensions in many application areas such as data mining and machine learning [42]. The advent of cloud-based storage enables organizations to store massive amounts of detailed information easily. A financial organization may monitor its stock price every minute, every hour, or every day, and the stock price may be predicted with linear combinations of thousands of underlying traded portfolios, each of which is a dimension. The other sources might be science or biology related; for example, healthcare data are known for having multiple dimensions, such as treatment plans, radiation therapy, and genetic background. The major difference between these examples (i.e., business and healthcare data) is the number of samples. The number of variables or components can be similar for each (in the thousands); however, the business data set can have millions of records, whereas the health data will rarely involve more than a few thousand samples. A sophisticated approach handles the number of increasing dimensions without affecting accuracy in either situation. High-dimensional data is often referred to as multi-dimensional or multi-aspect or multi-modal data throughout the literature [43, 44]. Highly detailed data from diverse platforms such as the web, social media is typically high-dimensional in nature. A generalised term for multi-dimensional data structure, along with arithmetic operations viability can be referred to as a tensor and appears frequently in many web-related applications.

Visualization in high-dimensional data

Visualization is the graphic representation of data that aims to convert data to a qualitative understanding that supports effective visual analysis. The visualization of a multidimensional dataset is challenging, and numerous research efforts have focused on addressing the increasing dimensionality [45]. The scatter plot matrix is a straightforward technique that employs pairwise scatterplots as matrix elements to represent multidimensional datasets; however, a loss of significant detail limits this technique to two-variable projection. Parallel coordinate plots (PCPs) [46] are proposed as a multivariate visualization technique that overcomes the scatter plot’s two-variable limit; it does this by mapping data points of a multidimensional space to a two-dimensional plane by laying the features as parallel vertical axes at a uniform spacing [47, 48].

Dimensionality reduction that converts a large set of input features into a smaller set of target features is a common method of addressing the issue. It is categorized into “distance” and “neighborhood” preserving methods. The former aims to minimize cost (so-called “aggregated normalized stress”), while the latter maximizes the overlap between the nearest neighborhoods and identifies the patterns [49]. Techniques such as principal component analysis (PCA) [50] and multi dimensional scaling (MDS) [51] are based on dimensionality reduction and aim to preserve as much significant information as possible in the target low-dimensional mapping. PCA is discussed in “Strategies for tackling the problem of high dimensionality” section in detail, MDS is a method for constructing similarity (or dissimilarity) among pairs of generally high-dimensional data as distances among points into a target low-dimensional space. The

standard distances used are Euclidean and cosine, and the algorithm input is an $m \times m$ distance matrix between all paired observations. The advantage of MDS is that it does not require original dimensions; it can generate an output (i.e., a point pattern) by computing a dissimilarity matrix among observations. Conversely, the disadvantage is that it requires analysis and storing of the $m \times m$ distance matrix, making the method computationally expensive when m increases [49, 52]. The Fastmap algorithm that projects the instance space in the directions of the maximum variability by calculating the dissimilarities is significantly faster than MDS [53]. Various other methods have been proposed, such as ISOMAP, Landmarks MDS, and Pivot MDS [54, 55], which can compute the projections in linear time to the number of observations.

In other cases, in which the number of dimensions is very high, the distances between paired observations appears similar and, as such, preserving such distances accurately is ineffective. Instead, preserving the neighborhoods in a projection is advantageous and effective in reasoning the patterns (anomalies) in the high-dimensional data [52]. A most effective technique used in this case is t-stochastic neighbor embedding (t-SNE), which is widely employed in many fields, such as pattern recognition, machine learning, and data mining. t-SNE is designed to capture most of the local structure in the high-dimensional data while simultaneously detailing the global structure, such as number of available clusters [56]. However, when the projection of dimensionality reduction fails, it is impossible to visualize the data in detail and it can only be comprehended mathematically [57]. For example, a linear separator in two-dimensional space is a line:

$$ax_1 + bx_2 = d \quad (1)$$

However, a linear separator in three-dimensional space is a plane:

$$ax_1 + bx_2 + cx_3 = d \quad (2)$$

A linear separator in a high-dimensional space can be mathematically represented as

$$\sum_{i=1}^{\infty} x_i = d \quad (3)$$

The curse of dimensionality

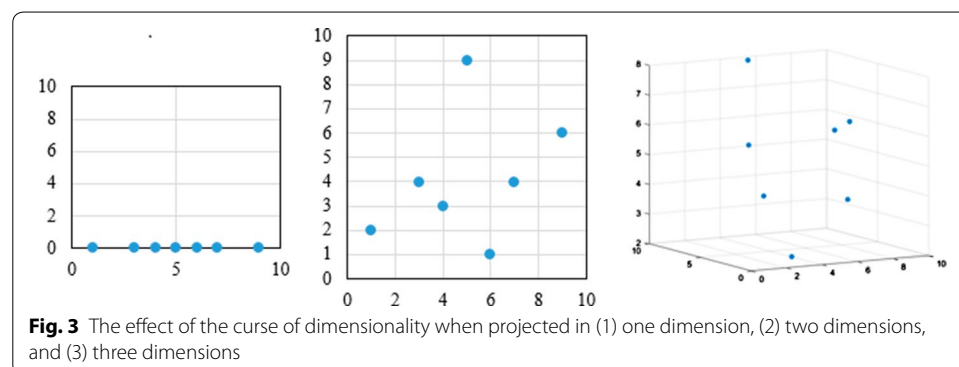
The term “curse of dimensionality” was first introduced by Bellman [58] to describe the problem caused by a rise in the number of dimensions or input variables. When the dimensionality of data increases, data size increases proportionally, leading to data sparsity, and sparse data is difficult to analyze. The curse of dimensionality affects anomaly detection techniques because the level of abnormal nature with respect to the increasing dimensions can be obscured or even concealed by unnecessary attributes [27]. Since outliers are defined as data instances in sparse regions, an inadequate discriminative region is observed in almost equally sparse locations of high-dimensional space [28].

The increase in dimensions makes data points scattered and isolated, and makes it elusive to find the global optima of the data set. The more dimensions that are added to a data set, the more complex it becomes, as each added dimension brings about a substantial number of false positives [42]. Figure 3 illustrates the sparsity of the data when projected in one, two, and three dimensions. The curse of dimensionality refers to a number of occurrences that emerge when analyzing and organizing data in high-dimensional spaces. The very nature of these occurrences is that, whenever there is an increase in dimensionality, the volume of the space increases proportionally, such that all other data points become sparse. This sparsity is challenging for any technique that requires statistical value. Further, it produces numerous complications associated with other noise levels, which may be irrelevant features or unnecessary attributes, that could complicate or even conceal the data instances [29]. This is the main reason why many algorithms struggle with high-dimensional data. As the number of dimensions increases, statistical approaches such as distance measures become less useful, since the points become almost equidistant from each other, due to the curse of dimensionality.

High-dimensional data requires significant computational memory and brings a huge computational burden. For high-dimensional data, recognizing useful insights or patterns becomes complex and challenging. The simplest approach to handle high dimensionality problem is to minimize the features and can be better understood by studying either the intrinsic or embedding dimensionality of data sets. It is essential to understand the subtle difference between intrinsic and embedding dimensionality. Intrinsic dimensionality is the smallest variety of the features that cover the full representation of the data; embedding dimensionality is the representation of the number of features or columns of the whole data space. Moreover, it is also important to understand how statistical models tackle the problem of high dimensionality.

Traditional models addressing the high dimensionality problem

High-dimensional data have special challenges in data mining in general and anomaly detection in particular [23]. Anomaly detection is difficult for data sets with increasing dimensionality and poses serious issues for many traditional data mining techniques. The problems emerging from the curse of dimensionality are not just specific to anomaly detection; they are also applicable to many other data mining techniques. Statistical



models use various methods, such as proximity-based, cluster-based, distance-based, density-based, and classification-based, to tackle the dimensionality problem. However, these methods bring greater computational complexity as the number of dimensions increases, the data instances are scattered and become less dense, affecting data distribution [7]. It is widely known that numerous issues, such as clustering and similarity of search experience, emerge with the increasing range of dimensions. Zimek et al. [23] discussed important aspects of the curse of dimensionality in their study of specialized techniques for anomaly identification. They highlighted common issues related to high-dimensional data and examined the difficulties in anomaly detection. They have concluded that more research is needed to tackle the problem of high dimensionality in anomaly detection. Some of the traditional models that address this problem are discussed below.

Many algorithms build on concepts of proximity to find anomalies that are based on relationships between data points. However, such algorithms suffer from huge computational growth in high-dimensional space and, consequently, fail to retain their effectiveness. This is because computational complexity of the algorithm is directly proportional to both the increased dimensionality of the data m and the number of samples n . The data distribution model interprets sparsity in a completely different way; it implies that every data instance is equidistant to others, which makes it difficult to differentiate close and distant pairs of data instances. The increase in dimensionality scatters the spread of data instances, making them less dense. It has been suggested by Aggarwal [28] that almost any technique that is primarily based on the concept of proximity would degrade qualitatively in high-dimensional space, and would, therefore, need to be redefined in a more meaningful way. Proximity-based techniques, which involve defining a data instance as a point of reference when its locality or proximity is sparsely distributed, are simple to implement. The locality can be defined in various ways, such as cluster-based, distance-based, and density-based, and no prior assumptions about the data distribution model are made. The difference between various proximity techniques lies in how the locality or proximity is defined.

When the data become sparse and methods based on the concept of proximity fail to maintain their viability, interpreting the data becomes difficult. This is because the sparsity of high-dimensional data can be comprehended in several ways that suggest that every data instance is an equally good anomaly with regard to the definition of a distance-based algorithm. Aggarwal and Yu [8] developed a method for outlier detection based on an understanding of the nature of projections. The method focused on lower dimensional projections that are locally sparse and cannot be recognized easily by brute-force techniques due to various possible combinations. They developed the method using a naive brute-force algorithm, which is very slow at identifying the meaningful insights because of its exhaustive search of the entire space, and another faster evolutionary algorithm to quickly discover underpinning patterns of dimensions. Due to the exponentially increasing search space, the brute-force technique is computationally weak for problems of even modest complexity. However, the technique has advantages over simple distance-based outliers that cannot overcome the effects of the curse of dimensionality. Other related studies that address the high dimensionality problem in anomaly detection are discussed below.

Distance-based techniques

As mentioned in the above section, distances between pairs of data instances are similar for a wide variety of data distributions and distance functions in high-dimensional space [59]. Distance concentration is one of the problems associated with sparsity and is related to the hubness phenomenon. Hubness is the tendency of some data instances to occur more frequently than others within the data set. It is dependent on feature representation, normalization, and similarity. Hence, hubs can sometime be interpreted as noise effects. Angiulli and Pizzuti [30] proposed a distance-based anomaly detection technique called “HilOut” to detect the top n outliers of large and high-dimensional data sets. HilOut uses the notion of the space-filling curve to linearize the data set; it provides an approximate solution before calculating the exact solution by examining the candidate outliers that remain after the first phase. Angiulli and Pizzuti presented both in-memory and disk-based versions of their proposed HilOut algorithm, as well as a thorough analysis of the method, which scaled well.

One of the more effective ways to handle sparse data in high-dimensional space is to employ functions based on dissimilarities of the datapoints. An assessment of small portions of the larger data set could help to identify anomalies that would otherwise be obscured by other anomalies if one were to examine the entire data set as a whole. Measuring the similarity of one data instance to other within a data set is a critical part of low-dimensional anomaly detection procedures. This is because an uncommon data point possesses insufficient data instances that are alike in the data set. Several anomaly detection methods practice Manhattan or Euclidian distance to estimate similarity among data instances [35]; however, when Euclidian distance is used to measure the similarity, the results are often treated as meaningless due to the unwanted nearest neighbors from multiple dimensions. This is due to the distance between two similar data instances and the distance between two non-similar data instances can be almost equal; hence, methods such as k -nearest neighbor with $O(n^2m)$ runtime are not viable for high dimensionality data sets unless the runtime is improved. Nevertheless, Euclidean distance is the most common distance metric used to calculate the similarity of low-dimensional data sets. Though Euclidean distance is suitable for low-dimensional data sets, it does not work as effectively in high-dimensional data [60].

To address the issue, Koufakou and Georgiopoulos [61] proposed an anomaly detection strategy where the speedup is achieved by its distributed version which is very close to linear. They called the approach as “fast distributed” and intended for mixed-attribute data sets that deal with sparse high-dimensional data. Their method, which takes the sparseness of the data set into consideration, is extremely scalable, as it has several points and many attributes in the data set.

Clustering-based techniques

Clustering is an important technique of data analysis. Ertoz et al. [63] introduced a shared nearest neighbor clustering algorithm to identify clusters of fluctuating shapes, sizes, and densities, along with the anomalies. Their method, which deals with multidimensionality and changing densities, automatically calculates the number of clusters. Initially, the algorithm recognizes the nearest neighbors of each data instance; it then redefines the similarity among pairs of data instances in terms of the number of nearest neighbors the data

instances share. The method recognizes core points and constructs clusters around these using the similarity. The shared nearest neighbor definition of similarity diminishes issues with differing densities and increasing dimensions. The authors identified a number of optimizations that allow their technique to process large data sets efficiently.

Density-based techniques

Density-based techniques deal with the dense localities of the data space, identified by various regions of lower object density. These are not effective when the dimensionality of the data increases because, as data points are scattered through a large volume, their density decreases due to the curse of dimensionality, making the relevant set of data points harder to find [7]. Chen et al. [62] introduced a density estimator for estimating measures in high-dimensional data and applied this to the problem of recognizing changes in data distribution. They approximated the probability of data μ , aiming to bypass the curse of dimensionality by utilizing the assumption that μ is lying around a low-dimensional subset embedded in a high-dimensional space. However, the estimators they proposed for μ are based on a geometric multiscale decomposition of the given data while controlling the overall model complexity. Chen et al. [62] proved strong finite sample performance bounds for various models and target probability measures that are based only on the intrinsic complexity of the data. The techniques implementing this construction are fast and parallelizable, and showed high accuracy in recognizing the outliers.

Classification-based techniques

Regarding classification and high dimensionality, a common problem occurs when the dimensionality m of the feature vector is much larger than the available training sample size n . According to Fan and Fan [64], high dimensionality in classification is intrinsic and due to the presence of irrelevant noise effects that do not help in minimizing the classification error. High dimensionality also affects classification accuracy, which is the predictive power and model interpretability, meaning the model can interpret the kind of connection between input and output. When the number of features is high, some of the traditional classification methods yield poor accuracy, with increasing dimensionality resulting in false classifications. The relationship between variables often builds a model that better understands the data; however, too many dimensions can complicate the interpretation model [64]. The most effective strategies focus on the relevant features and can process large numbers of dimensions within a manageable time span. However, many statistical techniques are vulnerable to increasing dimensionality. See Table 2 for the summary of statistical methods. Some machine learning techniques are based on the

Table 2 Summary of statistical methods and author's work

	Distance-based	Density-based	Clustering-based	Classification-based
[30]	✓	✗	✗	✗
[59]	✓	✗	✗	✗
[62]	✗	✓	✗	✗
[63]	✗	✓	✓	✗
[64]	✗	✗	✗	✓

assumption that dimensionality reduction can be achieved by projecting the data onto a lower dimensional space where learning might be easier after such lower dimensional projection [65, 66]. Techniques that are based on dimensionality reduction strategy projects data onto a lower dimensional subspace [8] or PCA [7, 67, 68], as discussed in the following section.

This section covered the theoretical background of the high dimensionality problem including ‘visualization’ and ‘the curse of dimensionality’, and also discusses the challenges of traditional models addressing the problem of high dimensionality in anomaly detection.

Strategies for tackling the problem of high dimensionality

One way to address the problem of high dimensionality is to reduce the dimensionality which projects the whole data set into a lower dimensional space, either by combining dimensions into linear combinations of attributes [69] or selecting the subsets of locally relevant and low-dimensional attributes called subspaces. As discussed above, many dimensionality reduction methods, such as PCA, MDS, Karhunen–Loeve Transform, local linear embedding, Laplacian Eigenmaps, and diffusion maps, have been proposed to achieve dimensionality reduction [51, 70–74]. In many application areas, data representations consists of dimensions that are dependent on each other, and correlation and redundancy can be noted. The intrinsic dimensionality of those data representation is smaller than the available relevant dimensions. Many techniques have been proposed to measure the intrinsic dimensionality [75–80].

Principal component analysis

PCA was first described by Pearson [50]. One of the oldest and most popular methods to address multidimensionality, it is used in many scientific disciplines [81]. Synonyms for PCA, such as empirical orthogonal functions, correspondence analysis, factor analysis, multifactor analysis, Eigenvector analysis, and latent vector analysis, can be found throughout the literature [81]. PCAs are generally based on the assumption that data are extracted from a single low-dimensional subspace of a high-dimensional space; however, in reality, data may derive from multiple subspaces, including unknown ones [82]. The aim of PCA is to derive all the significant attributes from the data set and form new orthogonal attributes called principal components. This provides an estimation of a data set which is, a data matrix x in terms of the product of two small matrices T and P . These (T and P) identify the important data patterns of X [83] to reduce the dimensionality by merging all the relevant attributes, called principal components, before eliminating all other remaining attributes [84, 85]. In general, PCA can be used on any data set by estimating the correlation structure of the features of the data set. It has three main goals: (1) project the most important dimensions of the data set, (2) combine the selected dimensions and so reduce the size of the data set, and (3) simplify the data set by analyzing the structure of the observations and associated dimensions.

Wang et al. [86] presented a PCA, as well as separable compression sensing, to identify different matrices. Compressive sensing (or compressed sampling [CSG]) theory was proposed by Candes and Wakin [87], that uses a random measurement matrix to convert a high-dimensional signal to low-dimensional signal until the signal is compressible after

which the original signal is restructured from the data of the low-dimensional signal. Moreover, the low-dimensional setting contains the main features of the high-dimensional signal, which means CSG can provide an effective method for anomaly detection in high-dimensional data sets. In the model of Wang et al. [86], abnormalities are more noticeable in a matrix of uncompression compared to a matrix of compression. Hence, their model could attain equal performance in volume anomaly detection, as it used the original uncompressed data and minimized the computational cost significantly.

The major drawback of using dimensionality reduction approaches based on PCA is that they may lead to a significant loss of information. This is because PCA is generally aimed at detecting orthogonal projections of the data set that contain the highest variance possible to identify hidden linear correlations among attributes [73]. If the attributes of the data set are non-linear or unrelated, PCA may result in complicated, and possibly uninterpretable, false positives.

Subspace approach

The subspace approach is an extension of feature selection that aims to identify local, relevant subspaces instead of the entire data space. Subspaces are the subsets of dimensions of a dataset that use less than the full dimensional space. For example, in applying the subspace approach to clustering, each cluster is a set of data instances recognized by a subset of dimensions, and different subsets of dimensions are represented in different clusters. The difference between traditional clustering and subspace clustering is the simultaneous discovery of cluster memberships of objects and the subspace of each cluster [88]. Cluster memberships describe similarities in objects and can be referred to as local relevant spaces in a normal subspace approach. Clusters are generally embedded in the subspaces of high-dimensional data. Aggarwal [89] concluded that by evaluating the nature of data, it is practical to suggest more meaningful clusters that are specific to a particular subspace. This is because anomalies may only be identified in low-dimensional subspaces of the data or in data sets with missing attributes [8]. Every subspace can be identified with its own patterns. This is due to different regions of the data being dense with respect to different subsets of dimensions. These clusters are referred to as projected clusters or subspace clusters [24, 90, 91].

Anomaly detection techniques that are based on subspace approaches are complex; they assume that anomalies are rare and can be found only in some subsets (locally relevant subsets) of dimensions. For this reason, statistical aggregates on individual dimensions in a specific region often provide very weak hints in subspace searches, resulting in the omission of useful dimensions, especially when locally relevant subspaces provide only a small view of the total dimensionality of the data. The challenge is identifying relevant subspaces, as the number of possible subspaces is directly proportional to the increasing dimensionality of the data. To build a robust anomaly detection model based on this assumption, it is necessary to simultaneously find relevant subspaces and low-dimensional spaces. Simultaneous discovery plays a crucial role because different subsets of dimensions are relevant to different anomalies. The discovery of multiple subspaces is important because selection of a single or only a few relevant subspaces may cause unpredictable results [28]. Hence, subspace anomaly detection is an ensemble-centric problem [89]. Lazervic and Kumar [92] proposed a model that detects anomalies using a scoring system called “feature bagging” that

randomly selects subspaces. However, this results in the rise of irrelevant dimensions due to random subspace selection. To address this problem, Kriegel et al. [23] adopted a technique to select informative or relevant dimensions. Müller et al. [93] proposed a subspace method called “OUTERS,” an outrank approach that ranks anomalies in heterogeneous high-dimensional data by introducing a novel scoring algorithm using subspace clustering analysis to detect anomalies in any number of dimensions. Zhang et al. [94] proposed an angle-based subspace outlier detection approach that selects meaningful features of subspace and executes anomaly detection in the selected subspace projection in order to retain the accuracy of detecting anomalies in high-dimensional data sets. They also included subspace detection approaches and illustrated the steps of relevant subspace selection suitable for analysis. Thudumu et al. [41] proposed a method to detect outliers, by bifurcating a high-dimensional space into locally relevant and low-dimensional subspaces using Pearson correlation coefficient (PCC) and PCA. They derived candidate subspaces where anomalies may possibly be hidden and developed an adaptive clustering approach to filter the anomalies [5].

Another algorithm, called Outlier Detection for Mixed-Attribute Data sets (ODMAD) proposed by Koufakou and Georgiopoulos [61], detects anomalies based on their categorical attributes in the first instance, before focusing on subsets of data in the continuous space by utilizing information about the subsets based on those categorical attributes. Their results demonstrated that the speed of the distributed version of ODMAD is close to linear speedup with the number of nodes used in computation. Mixed attributes are addressed by Ye et al. [95], who proposed an outlier detection algorithm to detect anomalies in high-dimensional mixed-attribute datasets by calculating the anomaly subspace, combined with information entropy. They used a bottom-up method to detect interesting anomaly subspaces and compute the outlying degree of anomalies in high-dimensional mixed-attribute data sets.

This section provided strategies to tackle the high dimensionality problem, such as subspace methods and principal component analysis to ascertain the benefit of one approach over another.

Unique challenges brought by anomaly detection in high-dimensional big data

As discussed in “[Introduction](#)”, the two features of big data that have the greatest effect on the problem of high dimensionality are “volume” and “velocity.” The high dimensionality problem not only leads to difficulties in detecting anomalies, but also brings additional challenges when data are increasing and arriving at speed as unbounded data streams. The most common strategies to address the problem of high dimensionality are to locate the most important dimensions (i.e., subset of features), known as the variable selection method, or to combine dimensions into a smaller set of new variables, known as the dimensionality reduction method [73]. However, these strategies become highly complex and ineffective in detecting anomalies due to the challenges associated with large data sets and data streams. This is because anomaly detection techniques are generally based on statistical hypothesis tests, such as the Grubb’s test and the Dixon test, which work on an assumed distribution of some underlying mechanism within the data [96]. However, the underlying distributions are unclear and most of the techniques

are generalized to work with few dimensions. When the data size is large, accumulating, and generated with speed, anomaly detection techniques bring further challenges. Many researchers have addressed the challenges of anomaly detection techniques with regard to high dimensionality and big data problems individually, but not jointly or comprehensively.

Moreover, each problem has their individual challenges, for example, ‘velocity’ aspect of big data has many challenges [35] such as transience, arrival rate, and the notion of infinity, but they are insignificant while tackling high dimensionality or anomaly detection problems. Furthermore, big data is still evolving, and there is no specific technique or algorithm to address the increasing ‘dimensionality’ as it varies from one application to other. To set future directions in this area, we review the works of various authors that are closely related (either directly or indirectly) to the above three problems, thus by, deriving the unique challenges. Here, we present a taxonomy of unique challenges (see Fig. 4) brought by anomaly detection in high-dimensional big data. We have described each one of them in the following subsections. Table 3 presents the description of challenges of anomaly detection from the perspective of high dimensionality problem.

Many anomaly detection techniques assume that data sets have only a few features, and most are aimed at identifying anomalies in low-dimensional data. Techniques that address the high dimensionality problem when data size is increasing face challenges in anomaly detection due to a range of factors, as outlined in Table 3. Anomalies are masked in high-dimensional space and are concealed in multiple high-contrast subspace projections. The selection of subspace is both vital and complex, especially when data is increasing. Fabien and Kelloer [100] proposed to estimate the contrast of subspaces by enhancing the quality of traditional anomaly rankings by calculating the scores of high-contrast projections as evaluated on real data sets. The factor that affects the nature of distance in high-dimensional space, hindering the anomaly detection process, is distance

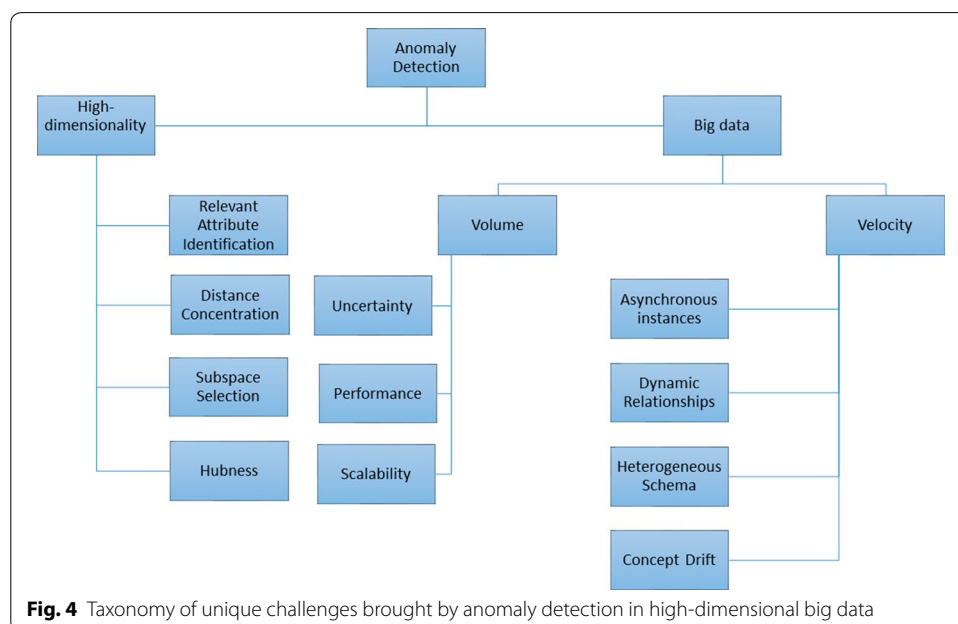


Table 3 Challenges of anomaly detection in context of high dimensionality problem

Characteristic Features	Description
1. Relevant attribute identification	This refers to the difficulty of describing the relevant quantitative locality of data instances in the high-dimensional space
2. Distance concentration	Due to the sparsity of data, the datapoints become nearly equidistant in high dimensional space depending on the distance measure used [59, 97–99]
3. Subspace selection	The potential features of subspace increase exponentially in line with the increasing dimensionality of the input data, which results in an exponential search space
4. Hubness	The behavior of high-dimensional data containing data instances that are frequently appearing in nearest neighbors known as hubs

concentration: all data points become essentially equidistant [101]. Tomasev et al. [102] addressed the problem of clustering in high-dimensional data by evaluating frequently occurring points known as hubs. They developed an algorithm that proved that hubs are the best way of defining the point centrality within a high-dimensional data cluster. Radovanović et al. [103] provided useful insights using data points, known as anti-hubs; although they appear very infrequently, they clearly distinguish the connection between outliers. The authors evaluated several methods, such as angle-based techniques, the classic k-NN method, density-based outlier detection, and anti-hub-based methods.

Challenges in the context of volume aspect of big data

With the advent of big data, the processing efficiency of anomaly detection techniques becomes increasingly complex. When the underlying probability distribution is unknown, and the data set size is huge, computational requirements increase. The “volume” feature of big data stresses the storage, memory, and computing capacity of the system to handle the increasing data size [104]. Performance is the major challenge of anomaly detection techniques when dealing with large data sets. When the data size is large, anomaly detection techniques may falter, due to limited computational capacity and associated factors. To overcome this issue, researchers have proposed the use of a parallel and distributed computing model. This section focuses on the challenges of anomaly detection in parallel, distributed environments and ensemble strategies. The unique challenges brought by anomaly detection from the ‘volume’ aspect of big data when data are high-dimensional is listed in Table 4.

Managing computational power and disk input/output (I/O) communication can result in improving the efficiency of the technique. Shin et al. [106] proposed a technique called D-cube, which is a disk-based detection technique to find fraudulent lockstep behaviour in largescale multi-aspect data and runs in a distributed manner across multiple machines. They compared D-cube technique across state-of-the-art methods and proved that D-cube is memory efficient, they have also proved it accurate by successfully

Table 4 Challenges of anomaly detection in context of big data problem (volume aspect)

Characteristic features	Description
1. Uncertainty	Data can be uncertain due to external events from vulnerable sources, such as failing to calculate the measure of attributes, imprecision, vagueness, inconsistency, and ambiguity. Data that cannot be depended upon with complete certainty are known as uncertain [105]
2. Performance	The performance of the technique in terms of time and the amount of memory is vital while detecting anomalies in high-dimensional data
3. Scalability	It is the ability of the technique to cater the increasing dimensions and data size

spotting network attacks from TCP dumps and synchronised behaviour in rating data with highest accuracy. Hung and Cheung [107] introduced an efficient and parallel version of the nested loop (NL) algorithm, based on the NL algorithm proposed by Knox and Ng [108], that reduces both computation and disk I/O costs. The NL algorithm is a straightforward method to mine outliers in a database. Ramaswamy et al. [109] proposed a construction model for distance-based anomalies, which is built on the distance of a point from its neighbor, and also developed an effective partition-based algorithm to pull out abnormal patterns in huge data sets. The former segregates input data into separate subsets and prunes partitions that do not contain anomalies, resulting in considerable savings in computation. Angiulli and Fassetti [110] presented a distance-based anomaly detection approach called “DOLPHIN” (for Detecting OutLIers PusHing data into an INdex)—that works on disk-resident data sets in huge datasets.

Ensemble strategies

Some studies aim to improve the efficiency of anomaly detection techniques by using ensemble strategies. To solve the sequential exception problem in large databases, Arning et al. [111] proposed a linear algorithm using a dissimilarity function to capture the similarity rate of a data point. More and Hall [112] proposed an algorithm to group large-scale datasets without clustering entire data in a single instance. Erfani et al. [6] introduced an unsupervised technique for high-dimensional large-scale unlabeled data sets to detect anomalies that are a combination of a deep belief network (DBN) and one-class support vector machines (SVM). One-class SVMs (1SVMs) are used for detecting outliers through unsupervised learning and aim to model the underlying distribution of data while not considering irrelevant attributes or outliers in the training records. Features derived from training samples are taken as input to train 1SVMs. Conversely, a DBN is a multiclass semi-supervised approach and dimensionality reduction tool. It uses multi-layer generative models (non-linear manifold) that learn one layer of features at a time from unlabeled data. Camacho et al. [113] presented an interesting outline for anomaly detection, identifying and interpreting unusual events in a forensics system network by

addressing the 4Vs of big data—volume, variety, veracity, and velocity. Their framework is based on input from several heterogeneous data sources. First, the system pre-processes incoming data; second, extracted features are calculated from the streaming data; third, all features representing diverse sources are combined and called a parameterization step; and, fourth, these new features are utilized in updating a set of intermediate data structures representing the present stage of the network. The updated strategy follows an exponentially weighted moving average method and uses a PCA model.

Challenges in the context of velocity aspect of big data

Velocity refers to the challenges associated with the continuous generation of data at high speed. A data stream is an infinite set of data instances in which each instance is a set of values with an explicit or implicit time stamp [35]. Data streams are unbounded sequences and the entry rate is continuously high, as the respective variations repeatedly change over time. Data streams reflect the important features of big data in that the aspects of both volume and velocity are temporally ordered, evolving, and potentially infinite. The data may comprise irrelevant attributes that raise problems for anomaly detection, and there are several other factors involved, such as whether the data are from a single source or multiple sources. Multiple data streams are made up of a set of data streams, and every data stream comprises an infinite sequence of data instances accompanied by an explicit or implicit time stamp history. In a single data stream, anomaly detection compares the history of data instances to determine whether an instance is an outlier or anomaly. By contrast, in multiple data streams, data instances are recognized as anomalies by comparing them to the history of data instances from the same stream or other streams. The unique challenges [35, 105, 114–117] of anomaly detection for data streams are listed in Table 5.

Most anomaly detection strategies assume that there is a finite volume of data generated by an unknown, stationary probability distribution, which can be stored and analyzed in various steps using a batch-mode algorithm [119]. Anomaly detection on streaming data is a highly complex process because the volume of data is unbounded and cannot be stored indefinitely [35]. Data streams are produced at a high rate of

Table 5 Challenges of anomaly detection in context of big data problem (velocity aspect)

Characteristic features	Description
1. Asynchronous instances	Multiple asynchronous data streams arrive at different times and are independent of one another. The data instances from any source may be missing at any point of time, or delay in arrival is possible; therefore, to detect anomalies, the specific temporal context should be determined on the data instances of both the streams. In multiple data streams, there are many sources from which data points are generated and these arrive at distinctive times. Such data points are described as asynchronous [35, 118]
2. Dynamic relationship	The correlation of data points is continuously monitored from multiple data streams that differ due to the asynchronous behavior of data [35]
3. Heterogeneous schema	Data instances arriving from various data sources may have different schemas. Compiling various multiple data instances over different schemas is a complex task, as is detecting an anomaly [35, 118]
4. Concept drift	The data distribution changes over time, which means that the properties of the target variable that are being predicted by the model also changes over time. This is called concept drift [116]

generation, which makes the task of detecting anomalies and extracting useful insights challenging. This is a main disadvantage that arises in several application areas [37].

Silva et al. [119] highlighted the following restrictions for the successful development of algorithms in data streams: (1) the data instances arrive continuously; (2) there is no control over the order of processing or handling of data instances; (3) the size of a data stream can be unbounded; (4) data instances are deleted after processing; and (5) the data generation process can be nonstationary; hence, its probability distribution may evolve over the period.

A hypothesis behind all data stream models is that the most recent data instances are more than historical data. The simplest model uses sliding windows of fixed size and works on the basis of first in, first out [38]. A sequence-based sliding window is one in which the size of the window is defined as the number of observations in regard to fixed size and specific time [75]. There is another type of sliding window called a time stamp window in which the size of the window is representative of the duration of the data [38].

Angiulli and Fasseti [120] introduced techniques for recognizing distance-based outliers in data streams using a sliding window model in which anomaly queries are executed for the purpose of identifying anomalies in the current window. Their algorithms execute anomaly queries and return an approximate answer based on accurate estimations with a statistical guarantee. These algorithms are based on a method known as stream outlier miner, or STORM, which is used to find outliers on distance-based windowed data streams. A sliding window is used to continuously inspect the object until it expires. Later, Angiulli and Fasseti [121] presented an additional algorithm for identifying distance-based anomalies in data streams using the sliding window model; although based on their previous approximate algorithm, this algorithm emphasized fixed memory requirements.

An algorithm based on the sliding window model for mining constrained frequent item sets on uncertain data streams was introduced by Yu et al. [39]. Known as CUSF-growth (constrained uncertain data stream frequent item sets growth), the algorithm determines the order of items in transactions and analyzes the properties of constraints. According to the order of items determined by the properties of constraints a CUSF-tree is created; later, after the frequent item sets are satisfied, the constraints are mined from the CUSF-tree.

Kontaki et al. [122] also studied the problem of continuous anomaly detection in data streams using sliding windows. They proposed four algorithms that sought effective anomaly monitoring with lesser memory requirements. Apart from assuming that the data are in metric space, their model did not make any assumptions about the behavior of input data. Their techniques have considerable flexibility with regard to parameter values, enabling the execution of multiple distance-based outlier detection tasks with different values, and they reduced the number of distance computations using micro-clusters. Their primary concerns were to improve efficiency and reduce storage consumption. Their methods incorporate an event-based framework that bypasses unneeded computations benefiting from the expiration time of objects.

A continuous outlier detection algorithm (CODA) is an event-based technique that estimates the expiration time of objects to circumvent undesirable computations. This

technique quickly determines the nature of elements by probabilistic pruning, thereby improving efficiency and consuming significantly less storage. Since different users may have different views of outliers, an advanced CODA that can handle numerous values by enabling the concurrent execution of various monitoring strategies has also been proposed. Another algorithm to decrease the quantity of distance computations—micro-cluster CODA—was inspired by the work of Zhang et al. [123].

The two popular ensemble learning techniques are Bagging and Boosting. Bagging is used for variance reduction and was first introduced by Breiman [124]. Boosting was first proposed by Schapire [125] to strengthen the ability of weak learners to achieve arbitrarily high accuracy. Oza and Russell [126] proposed two online variations of bagging and boosting for data streams. They showed how the training data could be simulated from the data stream perspective. They achieved the simulation of training data from the replication of the sampling bootstrap process. Bifet et al. [127] developed a model for examining concept drift in data streams with two new adaptations of bagging: ADWIN bagging and adaptive-size Hoeffding tree bagging. The Hoeffding tree is a technique that can learn from data streams that does not change over time. It is an incremental algorithm based on decision tree induction and the hypothesis of distribution generating examples [40]. The framework and approach of Bifet et al. [127] was similar to that proposed by Narasimhamurthy and Kuncheva [128], which accommodates STAGGER and moving hyperplane generation strategies.

ADWIN is a change detector and an estimator that interprets the issue of tracking the average of a stream of bits or real-valued numbers in a well-specified way [129]. The drift detection method introduced by Gama et al. [130] controls the errors generated by the learning model during the stage of prediction and then evaluates two windows; the first window comprises all the data, the second contains the data until the number of errors increases. These windows are not stored in-memory; only statistics and a window of recent errors using the binomial distribution are kept. Faria et al. [131] proposed a method for multiclass novelty detection in data streams that associate with novelty patterns, which are recognized by the algorithm through unsupervised learning using a confusion matrix that increases over time. Chu and Zaniolo [36] proposed a fast and light boosting for adaptive mining of data streams. The technique supports concept drift in data streams via change detection, which is built on a dynamic sample-weight assignment scheme

The problems that occur when data streams have multidimensional features are caused by the curse of dimensionality and drifting of data streams. To simultaneously address these two challenges, Zhang et al. [27] presented an unsupervised, online subspace learning approach to anomaly detection from nonstationary high-dimensional data streams. In order to find low-dimensional subspace faults from high-dimensional data sets, an angle-based subspace anomaly detection technique is designed by selecting fault-relevant subspaces and calculating vectorial angles, which compute the local anomaly score of the data instance in its subspace projection. The technique is extended to an online mode to continuously monitor systems and to detect anomalies from data streams based on the sliding window strategy. This section provided a taxonomy of big data anomaly detection problems with regard to high dimensionality in the form of a survey of closely related work. The aim of the review is to provide an understanding of

the underlying relations and patterns among variables by evaluating anomaly detection techniques dealing high dimensionality and big data problems.

Tools in high dimensional big data

Parallel or distributed computing is one of the most important techniques to handle and process big data [132]. In general, big data requires sophisticated methodologies to handle and process the large volume of data within limited run times efficiently. It distributes intensive computations over numerous computer processors and accelerates the overall run time by running parallelizable parts of the computation concurrently. Therefore, scalability is the major issue to many existing state-of-the-art techniques with big dimensionality problem. Luengo et al. [133] analysed this issue using two popular data repositories that ranged from 256 to 20 million dimensions and suggested that the future systems are to equip with necessary techniques to handle the big dimensionality issue. Zhai et al. [4] observed that the dimensionality issue handled by state-of-the-art techniques are not satisfactory and suggested that majority of them might fail to process any data with more than 10,000 dimensions. Hence, techniques based on distributed computing that supports the scalability are needed to handle the complex requirements of big dimensionality enabling quick processing and storage handling. MapReduce is one of the first distributed programming paradigms to handle big data storing and processing. Henceforth, many frameworks for distributed computing such as Apache Hadoop [134], Apache Storm [135], Apache Spark [136], Apache Flink [137] and MXNet [138] were developed addressing the increasing demands of big data. Most of these frameworks have in-house machine learning (ML) libraries, and Apache Spark has a powerful ML library than any of the other frameworks [139]. In this section, we review anomaly detection techniques implemented on these frameworks. Other parallel and distributed models that are focused on the scalability of the algorithms/techniques rather than frameworks are discussed in the following subsection “Other parallel and distributed models” (see “Other models” column in Table 6).

MapReduce, a programming paradigm for processing vast amounts of data, provides high-level parallelization that runs on clusters or grids of nodes (i.e., computers) to handle big data [156, 157]. MapReduce splits processing into “map” and “reduce” phases and each phase is based on key-value pairs used as input and output [158]. All operations of each map function are independent of each other and are fully parallelizable as the map function takes a single record. However, the reduce function

Table 6 Review of big data tools in context of anomaly detection

Works	MapReduce	Spark	Storm	MXNet	Flink	Other models
[32, 36, 140, 141]	✓	✗	✗	✗	✗	✗
[139, 142, 143]	✗	✓	✗	✗	✗	✗
[144]	✗	✗	✓	✗	✗	✗
[145]	✗	✗	✗	✓	✗	✗
[139, 146]	✗	✗	✗	✗	✓	✗
[44, 147–151]	✗	✗	✗	✗	✗	✓
[33, 34, 152–155]	✗	✗	✗	✗	✗	✓

is processed in parallel based on the intermediate pairs with the same key. Apache Hadoop is one of the most popular large-scale opensource frameworks that is based on the MapReduce [156]. Hence, we combine both MapReduce and Hadoop into one subsection and present the relevant work related to both as seen in Table 6. Koufakou et al. [140] proposed a fast parallel anomaly detection approach that is dependent on the attribute value frequency approach, a scalable, high-speed outlier detection process for categorical data that is effortless to parallelize and intended to recognize anomalies in large data mining issues. They used MapReduce because it offers load balancing and fault tolerance, and is extremely scalable concerning a number of nodes. Leung and Jiang [36] reported a solution which utilizes MapReduce to mine uncertain big data for frequent patterns satisfying user-specified anti-monotonic restrictions. Extant big data mining algorithms mainly focus on association analysis, which amounts to mining interesting patterns from particular databases. Jiang et al. [141] reported on a tree-based technique, BigSAM, that permits operators to express the patterns to be mined according to their interests via the use of constraints, as MapReduce enables uncertain big data to be mined for common patterns. He et al. [32] presented a parallel application of a KD-Tree-based anomaly identification technique to manage large data sets, implemented as a parallel KD-Tree-based anomaly detection algorithm. Their experimental results showed that the algorithm not only processed large data sets on commodity hardware efficiently, but also scaled well.

Apache Spark(Spark) [136] is another distributed framework based on MapReduce for processing large volumes of data on a distributed system, however, has a feature called in-memory computation [159, 160]. In contrast to MapReduce two-phase paradigm, Spark's in-memory computing model aims at speed and extensibility to handle both batch and real-time workloads. In Spark, implicit data parallelism is achieved on a cluster of computers that offer fault-tolerance, locality-aware scheduling and automatic load balancing. Jiang et al. [142] reported a scalable, parallel sequential pattern identification algorithm to identify probable motifs (i.e., non-exact contiguous sequences) from large DNA sequences using the Spark in-memory parallel resilient distributed data set approach. As DNA sequences are detailed using a set of four letters, their algorithm restricts the search space, decreases mining time, and integrates user-specified features in identifying sequential patterns from big uncertain DNA in the Spark framework. Terzi et al. [143] proposed an unsupervised approach using Apache Spark as their distributed framework. They reported 96% accuracy in the identification of anomalies in a public big network data.

For real-time processing, Apache Storm [135] was developed by Twitter an open-source distributed framework with guaranteed features such as scalability, fault-tolerant, and resilience. Zhang et al. [144] developed a framework built on Apache Storm to support the distributed learning of large-scale Convolutional Neural Networks (CNN) using two datasets, that is suitable for real-time stream processing. Their experiment demonstrates that more reasonable parallelisms can significantly improve the performance speed of their framework. Veen et al. [161] focused on the elasticity of a streaming analysis platform using virtual machines from a public cloud based on Apache Storm. They chose the framework because of the possibility of adding and removing processing nodes is easier.

MXNet is an open-source, scalable, memory-efficient, high-performance modular deep learning framework, that offers a range of application programming interfaces (APIs) for programming languages such as C++, Python, Matlab, and R. It runs on heterogeneous systems, starting from mobile devices to distributed GPU clusters [162]. Abeyrathna et al. [145] proposed an anomaly proposal-based approach that can establish an efficient architecture for real-time fire detection using MXNet framework. Their architecture is built together with a Convolutional Autoencoder (CAE) module for picking doubtful regions and a Convolution Neural Network (CNN) classifier that can recognize fire as an outlier. For both stream processing and batch processing, Apache Flink [137] another opensource framework was proposed combining the scalability and programming flexibility of other distributed paradigms such as MapReduce [163, 164]. Toliopoulos et al. [146] conducted their work on distance-based outlier detection and examined in a massively parallel setting using three real-world and one synthetic dataset. They have considered three main parallel streaming platforms such as Apache Storm, Apache Spark and Apache Flink. However, they mainly targeted to investigate the challenges of customizing state-of-the-art techniques to Apache Flink. Their experiments showed the speed-ups of up to 117 times over a non-parallel solution implemented in Flink. The methods they used are publicly available as open-source software as they intend to offer solutions in the large-scale streaming big data analytics. García-Gil et al. [139] have performed a comparative study on the scalability of two frameworks Apache Spark and Apache Flink. Their experimental results showed that Spark has better performance and overall runtimes than Flink.

Other parallel and distributed models

Many other distributed frameworks have been proposed to support efficient parallel processing of anomaly detection techniques in big data. Angiulli et al. [33] presented a distributed framework for identifying distance-based anomalies in massive data sets based on the perception of an anomaly detection resolving set, which is a small subset of the data set. The authors also presented a modified method—the “lazy distributed solving set”—that reduced the volume of data to be swapped from the nodes on the distributed solving set by implementing a strategy for the transmission of a condensed number of distances, which, to some extent, simultaneously increased the number of communications. Later, Angiulli et al. [152] proposed a set of parallel and distributed algorithms for GPUs resulting in two distance-based anomaly detection algorithms: BruteForce and SolvingSet. The difference between them is the way they utilize the architecture and memory hierarchy of GPUs and provide improvements with respect to CPU versions, both regarding scalability and exploitation of parallelism. The authors detailed the algorithms’ computational properties, measured their performance with extensive experimentation, and compared several implementations showing significant increases in speed. Matsumoto et al. [153] published a parallel algorithm for anomaly detection on uncertain data using density sampling, establishing implementation on both graphic processing units (GPUs) and multi-core central processing units (CPUs) through the OpenCL framework.

Lozano and Acufia [154] designed two parallel algorithms to identify distance-based anomalies using randomization and a pruning rule to recognize density-based local

anomalies. They also constructed parallel versions of Bay and Local Outlier Factor (LOF) procedures, which exhibited good performance in anomaly detection and run time. Bai et al. [34] focused on the issue of distributed density-based anomaly detection for large data. They proposed a grid-based partition algorithm as a data pre-processing technique that splits the data set into grids before distributing these to data nodes in a distributed environment. A distributed LOF computing method was presented for discovering density-based outliers in parallel by utilizing a few network communications. Reilly et al. [155] proposed a PCA-based outlier identification approach that works in a distributed environment, demonstrating robustness in its extraction of the principal components of a training set comprising outliers. Minimum volume elliptical PCA can determine principal components more vigorously in the presence of outliers by building a soft-margin, tiniest volume, ellipse around the data that lessens the effects of outliers in the training set. Local and centralized approaches to outlier detection were also studied. The projected outlier detection technique was reformulated using distributed convex optimization, which splits the issue across a number of nodes. Gunter et al. [147] explored various techniques for identifying outliers in large distributed systems and argued for a lightweight approach to enable real time analysis. No single optimal method was found; therefore, they concluded that combinations of various methods are needed due to the change in effectiveness that depends on the definition of the anomaly.

It is important to understand whether the anomaly detection technique fits the model and is scalable to the size of the data and dimensionality [7]. Maruhashi et al. [148] present a technique to find patterns and anomalies in heterogeneous networks with millions of edges and proved empirically that the technique is scalable to high-dimensional datasets. Shin et al. [149] develop a novel flexible framework that can identify dense blocks in a large-scale high-order tensor and showed that their method is scalable in real data by spotting network attacks from a TCP dump with near perfect accuracy. Oh et al. [44] provide a scalable approach that can handle high-dimensional data. In particular, they also prove that their approach is better than many baseline methods in terms of dimensionality. Hooi et al. [150] propose a scalable densest-subgraph based anomaly detection method called FRAUDAR that can not only detect various fraud attacks in real world graphs but also can detect a large number of previously undetected behaviour in large data. Jiang et al. [151] propose a scalable algorithm called CROSSSPOT that scores and ranks the level of suspiciousness to find dense, suspicious blocks in real-world large multi-modal data.

Conclusions

With the world becoming increasingly data driven and with no generic approach for big data anomaly detection, the problem of high dimensionality is inevitable in many application areas. Moreover, the loss of accuracy is greater and computationally more complex as the volume of data increases. Identifying anomalous data points across large data sets with high dimensionality issues is a research challenge. This survey has provided a comprehensive overview of anomaly detection techniques related to the big data features of volume and velocity, and has; examined strategies for addressing the problem of high dimensionality. It is evident that further study and evaluation of big data

anomaly detection strategies that address the high dimensionality problem, are needed. To address this research problem, we propose a future research direction of building a novel framework that enables anomalous data points to be identified across large volumes of data with high dimensionality issues. The main contribution will be improving the balance between performance and accuracy of anomaly detection in big data with high dimensionality problems.

Abbreviations

1SVM: One-class support vector machine; BD: Big data; COD: Curse of dimensionality; CODA: Continuous outlier detection algorithm; CPU: Central processing units; CSG: Compressive sensing or compressed sensing; CUSF: Constrained uncertain data stream frequent item; DBM: Deep belief network; GPU: Graphic processing units; IMRT: Intensity-modulated Radiation Therapy; I/O: Input/output; LOF: Local Outlier Factor; NL: Nested loop; ODMAD: Outlier Detection for Mixed-Attribute Dataset; PCA: Principal component analysis; PCC: Pearson correlation coefficient; PCP: Parallel coordinate plots; MDS: Multi dimensional scaling; SVM: Support vector machine; tSNE: t-stochastic neighbor embedding.

Acknowledgements

The article processing charge is funded by Swinburne University of Technology, Australia.

Authors' contributions

ST conducted the systematic literature review and examined various techniques related to the problems of anomaly detection in high-dimensional big data. ST wrote the first draft of the manuscript. PB, JJ, and JS have made significant contributions to the design and structure of the review. PB, JJ, and JS took on a supervisory role and oversaw the completion of the work by reviewing and editing the manuscript. All authors read and approved the final manuscript.

Funding

Not applicable.

Availability of data and materials

All papers studied in this systematic review are available in ACM Digital Library, IEEE Xplore and ScienceDirect. Please see the references below.

Competing interests

The authors declare that they have no competing interests.

Author details

¹ School of Software & Electrical Engineering, Swinburne University of Technology, Hawthorn, VIC 3122, Australia.

² Sarawak State Government, Sarawak, Malaysia.

Received: 21 February 2020 Accepted: 21 June 2020

Published online: 02 July 2020

References

- Aggarwal CC. Managing and mining sensor data. Berlin: Springer Science & Business Media; 2013.
- Jiang F, Leung CK, Pazdor AG. Big data mining of social networks for friend recommendation. In: Advances in social networks analysis and mining (ASONAM), 2016 IEEE/ACM international conference on. IEEE. 2016. pp. 921–2.
- Gartner I. Big data definition. <https://www.gartner.com/it-glossary/big-data/>. Accessed 14 Feb 2020.
- Zhai Y, Ong Y-S, Tsang IW. The emerging "big dimensionality". IEEE Comput Intell Mag. 2014;9(3):14–26.
- Thudumu S, Branch P, Jin J, Singh JJ. Adaptive clustering for outlier identification in high-dimensional data. In: International conference on algorithms and architectures for parallel processing. Springer. 2019. pp. 215–28.
- Erfani SM, Rajasegarar S, Karunasekera S, Leckie C. High-dimensional and large-scale anomaly detection using a linear one-class svm with deep learning. Pattern Recogn. 2016;58:121–34.
- Hodge V, Austin J. A survey of outlier detection methodologies. Artif Intell Rev. 2004;22(2):85–126.
- Aggarwal CC, Philip SY. An effective and efficient algorithm for high-dimensional outlier detection. VLDB J. 2005;14(2):211–21.
- Agrawal S, Agrawal J. Survey on anomaly detection using data mining techniques. Procedia Comput Sci. 2015;60:708–13.
- Akoglu L, Tong H, Koutra D. Graph based anomaly detection and description: a survey. Data Mining Knowl Discov. 2015;29(3):626–88.
- Chandola V, Banerjee A, Kumar V. Anomaly detection: a survey. ACM Comput Surv. 2009;41(3):15.
- Patcha A, Park J-M. An overview of anomaly detection techniques: existing solutions and latest technological trends. Comput Netw. 2007;51(12):3448–70.
- Jiang M, Cui P, Faloutsos C. Suspicious behavior detection: current trends and future directions. IEEE Intell Syst. 2016;31(1):31–9.
- Sorzano COS, Vargas J, Montano AP. A survey of dimensionality reduction techniques. arXiv preprint [arXiv:1403.2877](https://arxiv.org/abs/1403.2877). 2014.
- Gama J. Knowledge discovery from data streams. London: Chapman and Hall/CRC; 2010.

16. Gupta M, Gao J, Aggarwal CC, Han J. Outlier detection for temporal data: a survey. *IEEE Trans Knowl Data Eng.* 2014;26(9):2250–67.
17. Heydari A, ali Tavakoli M, Salim N, Heydari Z. Detection of review spam: a survey. *Expert Syst Appl.* 2015;42(7):3634–42.
18. Jindal N, Liu, B. Review spam detection. In: *Proceedings of the 16th international conference on world wide web.* ACM. 2007. pp. 1189–90.
19. Parthasarathy S, Ghoting A, Otey ME. A survey of distributed mining of data streams. In: *Data streams.* Springer; 2007. pp. 289–307.
20. Phua C, Lee V, Smith K, Gayler R. A comprehensive survey of data mining-based fraud detection research. *arXiv preprint arXiv:1009.6119.* 2010.
21. Tamboli J, Shukla M. A survey of outlier detection algorithms for data streams. In: *Computing for sustainable global development (INDIACom), 2016 3rd international conference on.* IEEE. 2016. pp. 3535–40.
22. Spirin N, Han J. Survey on web spam detection: principles and algorithms. *ACM SIGKDD Explor Newsl.* 2012;13(2):50–64.
23. Zimek A, Schubert E, Kriegel H-P. A survey on unsupervised outlier detection in high-dimensional numerical data. *Stat Anal Data Mining ASA Data Sci J.* 2012;5(5):363–87.
24. Parsons L, Haque E, Liu H. Subspace clustering for high dimensional data: a review. *ACM SIGKDD Explor Newsl.* 2004;6(1):90–105.
25. Goldstein M, Uchida S. A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data. *PLoS ONE.* 2016;11(4):0152173.
26. Varian I. IMRT (Intensity Modulated Radiation Therapy). <https://patient.varian.com/en/treatments/radiation-therapy/treatment-techniques>. Accessed 26 June 2020.
27. Zhang L, Lin J, Karim R. Sliding window-based fault detection from high-dimensional data streams. *IEEE Trans Syst Man Cybern Syst.* 2017;47(2):289–303.
28. Aggarwal CC. High-dimensional outlier detection: the subspace method. In: *Outlier analysis.* Springer; 2017. pp. 149–84.
29. Donoho DL, et al. High-dimensional data analysis: the curses and blessings of dimensionality. *AMS Math Chall Lect.* 2000;1:32.
30. Angiulli F, Pizzuti C. Outlier mining in large high-dimensional data sets. *IEEE Trans Knowl Data Eng.* 2005;17(2):203–15.
31. Koufakou A. Scalable and efficient outlier detection in large distributed data sets with mixed-type attributes. Florida: University of Central Florida; 2009.
32. He Q, Ma Y, Wang Q, Zhuang F, Shi Z, Parallel outlier detection using kd-tree based on mapreduce. In: *Cloud computing technology and science (CloudCom), 2011 IEEE third international conference on.* IEEE. 2011. pp. 75–80.
33. Angiulli F, Basta S, Lodi S, Sartori C. Distributed strategies for mining outliers in large data sets. *IEEE Trans Knowl Data Eng.* 2013;25(7):1520–32.
34. Bai M, Wang X, Xin J, Wang G. An efficient algorithm for distributed density-based outlier detection on big data. *Neurocomputing.* 2016;181:19–28.
35. Sadik S, Gruenwald L. Research issues in outlier detection for data streams. *ACM SIGKDD Explor Newsl.* 2014;15(1):33–40.
36. Chu F, Zaniolo C, Fast and light boosting for adaptive mining of data streams. In: *Pacific-Asia conference on knowledge discovery and data mining.* Springer. 2004. pp. 282–92.
37. Salehi M, Leckie C, Bezdek JC, Vaithianathan T, Zhang X. Fast memory efficient local outlier detection in data streams. *IEEE Trans Knowl Data Eng.* 2016;28(12):3246–60.
38. Gama J. A survey on learning from data streams: current and future trends. *Progr Artif Intell.* 2012;1(1):45–55.
39. Yu Q, Tang K-M, Tang S-X, Lv X. Uncertain frequent itemsets mining algorithm on data streams with constraints. In: *International conference on intelligent data engineering and automated learning.* Springer. 2016. pp. 192–201.
40. Domingos P, Hulten G. Mining high-speed data streams. In: *Proceedings of the sixth ACM SIGKDD international conference on knowledge discovery and data mining.* ACM. 2000. pp. 71–80.
41. Thudumu S, Branch P, Jin J, Singh J. Elicitation of candidate subspaces in high-dimensional data. In: *2019 IEEE 21st international conference on high performance computing and communications; IEEE 17th international conference on smart city; IEEE 5th international conference on data science and systems (HPCC/SmartCity/DSS), IEEE.* 2019. pp. 1995–2000.
42. Thudumu S, Branch P, Jin J, Singh J. Estimation of locally relevant subspace in high-dimensional data. In: *Proceedings of the Australasian computer science week multiconference.* 2020. pp. 1–6.
43. Shin K, Hooi B, Kim J, Faloutsos C. Densealert: Incremental dense-subtensor detection in tensor streams. In: *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining.* ACM. 2017. pp. 1057–66.
44. Oh J, Shin K, Papalexakis EE, Faloutsos C, Yu H. S-hot: Scalable high-order tucker decomposition. In: *Proceedings of the Tenth ACM international conference on web search and data mining.* ACM. 2017. pp. 761–70.
45. Tatu A, Maaß F, Färber I, Bertini E, Schreck T, Seidl T, Keim D. Subspace search and visualization to make sense of alternative clusterings in high-dimensional data. In: *Visual analytics science and technology (VAST), 2012 IEEE conference on.* IEEE. 2012. pp. 63–72.
46. Inselberg A. The plane with parallel coordinates. *Vis Comput.* 1985;1(2):69–91.
47. Roberts R, Laramee RS, Smith GA, Brookes P, D’Cruze T. Smart brushing for parallel coordinates. *IEEE Trans Vis Comput Graph.* 2018;25:1575–90.
48. Johansson J, Forsell C. Evaluation of parallel coordinates: overview, categorization and guidelines for future research. *IEEE Trans Vis Comput Graph.* 2016;22(1):579–88.
49. Krüger JF, Rauber PE, Martins RM, Kerren A, Kobourov S, Telea AC. Graph layouts by t-sne. In: *Computer graphics forum, vol. 36.* Wiley Online Library; 2017. pp. 283–94.

50. Pearson K. Liii. on lines and planes of closest fit to systems of points in space. *Lond Edinb Dublin Philos Mag J Sci*. 1901;2(11):559–72.
51. Kruskal JB. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*. 1964;29(1):1–27.
52. da Silva RR, Rauber PE, Telea AC. Beyond the third dimension: visualizing high-dimensional data with projections. *Comput Sci Eng*. 2016;18(5):98–107.
53. Faloutsos C, Lin K-I. FastMap: a fast algorithm for indexing, data-mining and visualization of traditional and multimedia datasets, vol. 24. New York: ACM; 1995.
54. Tenenbaum JB, De Silva V, Langford JC. A global geometric framework for nonlinear dimensionality reduction. *Science*. 2000;290(5500):2319–23.
55. Cavallo M, Demiralp Ç. A visual interaction framework for dimensionality reduction based data exploration. In: Proceedings of the 2018 chi conference on human factors in computing systems. ACM. 2018. p. 635.
56. Maaten Lvd, Hinton G. Visualizing data using t-sne. *J Mach Learn Res*. 2008;9:2579–605.
57. Verleysen M, François D. The curse of dimensionality in data mining and time series prediction. In: International work-conference on artificial neural networks. Springer. 2005. pp. 758–70.
58. Bellman R. Dynamic programming. Chelmsford: Courier Corporation; 2013.
59. Beyer K, Goldstein J, Ramakrishnan R, Shaft U. When is “nearest neighbor” meaningful? In: International conference on database theory. Springer. 1999. pp. 217–35.
60. Shen Y, Bo J, Li K, Chen S, Qiao L, Li J. High-dimensional data anomaly detection framework based on feature extraction of elastic network. In: International conference on machine learning and intelligent communications. Springer. 2019. pp. 3–17.
61. Koufakou A, Georgiopoulos M. A fast outlier detection strategy for distributed high-dimensional data sets with mixed attributes. *Data Mining Knowl Discov*. 2010;20(2):259–89.
62. Chen G, Iwen M, Chin S, Maggioni M. A fast multiscale framework for data in high-dimensions: measure estimation, anomaly detection, and compressive measurements. In: Visual communications and image processing (VCIP), 2012 IEEE. 2012. pp. 1–6.
63. Ertöz L, Steinbach M, Kumar V. Finding clusters of different sizes, shapes, and densities in noisy, high dimensional data. In: Proceedings of the 2003 SIAM international conference on data mining. SIAM. 2003. pp. 47–58.
64. Fan J, Fan Y. High dimensional classification using features annealed independence rules. *Ann Stat*. 2008;36(6):2605.
65. Talwalkar A, Kumar S, Rowley H. Large-scale manifold learning. In: Computer vision and pattern recognition, 2008. CVPR 2008. IEEE conference on. IEEE. 2008. pp. 1–8.
66. Zhang L, Chen S, Qiao L. Graph optimization for dimensionality reduction with sparsity constraints. *Pattern Recogn*. 2012;45(3):1205–10.
67. Parra L, Deco G, Miesbach S. Statistical independence and novelty detection with information preserving nonlinear maps. *Neural Comput*. 1996;8(2):260–9.
68. Korn F, Labrinidis A, Kotidis Y, Faloutsos C, Kaplunovich A, Perkovic D. Quantifiable data mining using principal component analysis. Technical report. 1998.
69. Agrawal R, Gehrke J, Gunopulos D, Raghavan P. Automatic subspace clustering of high dimensional data for data mining applications, vol. 27. London: ACM; 1998.
70. Ross, I. Nonlinear dimensionality reduction methods in climate data analysis. arXiv preprint [arXiv:0901.0537](https://arxiv.org/abs/0901.0537). 2009.
71. Fukunaga K, Olsen DR. An algorithm for finding intrinsic dimensionality of data. *IEEE Trans Comput*. 1971;100(2):176–83.
72. Kirby M. Geometric data analysis: an empirical approach to dimensionality reduction and the study of patterns. Hoboken: Wiley; 2000.
73. Van Der Maaten L, Postma E, Van den Herik J. Dimensionality reduction: a comparative. *J Mach Learn Res*. 2009;10:66–71.
74. Ham J, Lee DD, Mika S, Schölkopf B. A kernel view of the dimensionality reduction of manifolds. In: Proceedings of the twenty-first international conference on machine learning. ACM. 2004. p. 47.
75. Pettis KW, Bailey TA, Jain AK, Dubes RC. An intrinsic dimensionality estimator from near-neighbor information. *IEEE Trans Pattern Anal Mach Intell*. 1979;1:25–37.
76. Szepesvári C, Audibert J-Y, et al. Manifold-adaptive dimension estimation. In: Proceedings of the 24th international conference on machine learning. ACM. 2007. pp. 265–72.
77. Carter KM, Raich R, Hero AO III. On local intrinsic dimension estimation and its applications. *IEEE Trans Signal Process*. 2010;58(2):650–63.
78. Ceruti C, Bassis S, Rozza A, Lombardi G, Casiraghi E, Campadelli P. Danco: an intrinsic dimensionality estimator exploiting angle and norm concentration. *Pattern Recogn*. 2014;47(8):2569–81.
79. Camastra F. Data dimensionality estimation methods: a survey. *Pattern Recogn*. 2003;36(12):2945–54.
80. Gupta MD, Huang TS. Regularized maximum likelihood for intrinsic dimension estimation. arXiv preprint [arXiv:1203.3483](https://arxiv.org/abs/1203.3483). 2012.
81. Abdi H, Williams LJ. Principal component analysis. *Wiley Interdiscip Rev Comput Stat*. 2010;2(4):433–59.
82. Vidal R, Ma Y, Sastry S. Generalized principal component analysis (GPCA). *IEEE Trans Pattern Anal Mach Intell*. 2005;27(12):1945–59.
83. Wold S, Esbensen K, Geladi P. Principal component analysis. *Chemom Intell Lab Syst*. 1987;2(1–3):37–52.
84. Shlens J. A tutorial on principal component analysis. arXiv preprint [arXiv:1404.1100](https://arxiv.org/abs/1404.1100). 2014.
85. Chakrabarti K, Mehrotra S. Local dimensionality reduction: a new approach to indexing high dimensional spaces. In: VLDB. Citeseer. 2000. pp. 89–100.
86. Wang W, Wang D, Jiang S, Qin S, Xue L. Anomaly detection in big data with separable compressive sensing. In: Proceedings of the 2015 international conference on communications, signal processing, and systems. Springer. 2016. pp. 589–94.
87. Candès EJ, Wakin MB. An introduction to compressive sampling. *IEEE Signal Process Mag*. 2008;25(2):21–30.

88. Jing L, Ng MK, Huang JZ. An entropy weighting k-means algorithm for subspace clustering of high-dimensional sparse data. *IEEE Trans Knowl Data Eng.* 2007;19(8):1026–41.
89. Aggarwal CC. Outlier analysis. In: *Data mining*. Springer. 2015. pp. 237–63.
90. Patrikainen A, Meila M. Comparing subspace clusterings. *IEEE Trans Knowl Data Eng.* 2006;18(7):902–16.
91. Kriegel H-P, Kröger P, Zimek A. Clustering high-dimensional data: a survey on subspace clustering, pattern-based clustering, and correlation clustering. *ACM Trans Knowl Discov Data.* 2009;3(1):1.
92. Lazarevic A, Kumar V. Feature bagging for outlier detection. In: *Proceedings of the eleventh ACM SIGKDD international conference on knowledge discovery in data mining*. ACM. 2005. pp. 157–66.
93. Müller E, Günemann S, Assent I, Seidl T. Evaluating clustering in subspace projections of high dimensional data. *Proc VLDB Endow.* 2009;2(1):1270–81.
94. Zhang L, Lin J, Karim R. An angle-based subspace anomaly detection approach to high-dimensional data: with an application to industrial fault detection. *Reliab Eng Syst Saf.* 2015;142:482–97.
95. Ye M, Li X, Orłowska ME. Projected outlier detection in high-dimensional mixed-attributes data set. *Expert Syst Appl.* 2009;36(3):7104–13.
96. Júnior B, Bezerra A, Pires PSdM. An approach to outlier detection and smoothing applied to a trajectory radar data. *J Aerosp Technol Manage.* 2014;6(3):237–48.
97. Hall P, Marron JS, Neeman A. Geometric representation of high dimension, low sample size data. *J R Stat Soc Ser B Stat Methodol.* 2005;67(3):427–44.
98. Ahn J, Marron J, Muller KM, Chi Y-Y. The high-dimension, low-sample-size geometric representation holds under mild conditions. *Biometrika.* 2007;94(3):760–6.
99. Aggarwal CC, Hinneburg A, Keim DA. On the surprising behavior of distance metrics in high dimensional space. In: *International conference on database theory*. Springer. 2001. pp. 420–34.
100. Keller F, Müller E, Böhm K. Hics: high contrast subspaces for density-based outlier ranking. In: *Data engineering (ICDE), 2012 IEEE 28th international conference on*. IEEE. 2012. pp. 1037–48.
101. Francois D, Wertz V, Verleysen M. The concentration of fractional distances. *IEEE Trans Knowl Data Eng.* 2007;19(7):873–86.
102. Tomasev N, Radovanovic M, Mladenic D, Ivanovic M. The role of hubness in clustering high-dimensional data. *IEEE Trans Knowl Data Eng.* 2014;26(3):739–51.
103. Radovanović M, Nanopoulos A, Ivanović M. Reverse nearest neighbors in unsupervised distance-based outlier detection. *IEEE Trans Knowl Data Eng.* 2015;27(5):1369–82.
104. Gadepally V, Kepner J. Big data dimensional analysis. In: *High performance extreme computing conference (HPEC), 2014 IEEE*. 2014. pp. 1–6.
105. Tatbul N. Streaming data integration: challenges and opportunities. 2010.
106. Shin K, Hooi B, Kim J, Faloutsos C. D-cube: Dense-block detection in terabyte-scale tensors. In: *Proceedings of the tenth ACM international conference on web search and data mining*. ACM. 2017. pp. 681–9.
107. Hung E, Cheung DW. Parallel mining of outliers in large database. *Distrib Parallel Database.* 2002;12(1):5–26.
108. Knox EM, Ng RT. Algorithms for mining distancebased outliers in large datasets. In: *Proceedings of the international conference on very large data bases*. Citeseer. 1998. pp. 392–403.
109. Ramaswamy S, Rastogi R, Shim K. Efficient algorithms for mining outliers from large data sets. In: *ACM Sigmod record*, vol. 29. ACM. 2000. pp. 427–38.
110. Angiulli F, Fassetto F. Very efficient mining of distance-based outliers. In: *Proceedings of the sixteenth ACM conference on conference on information and knowledge management*. 2007. pp. 791–800.
111. Arning A, Agrawal R, Raghavan P. A linear method for deviation detection in large databases. In: *KDD*. 1996. pp. 164–9.
112. More P, Hall LO. Scalable clustering: a distributed approach. In: *Fuzzy systems, 2004. Proceedings. 2004 IEEE international conference on*. IEEE. vol. 1. 2004. pp. 143–8.
113. Camacho J, Macia-Fernandez G, Diaz-Verdejo J, Garcia-Teodoro P. Tackling the big data 4 vs for anomaly detection. In: *Computer communications workshops (INFOCOM WKSHPs), 2014 IEEE conference on*. IEEE. 2014. pp. 500–5.
114. Carney D, Çetintemel U, Cherniack M, Conway C, Lee S, Seidman G, Stonebraker M, Tatbul N, Zdonik S. Monitoring streams: a new class of data management applications. In: *Proceedings of the 28th international conference on very large data bases. VLDB endowment*. 2002. pp. 215–26.
115. Babcock B, Babu S, Datar M, Motwani R, Widom J. Models and issues in data stream systems. In: *Proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART symposium on principles of database systems*. ACM. 2002. pp. 1–16.
116. Jiang N, Gruenwald L. Research issues in data stream association rule mining. *ACM Sigmod Rec.* 2006;35(1):14–9.
117. Stonebraker M, Çetintemel U, Zdonik S. The 8 requirements of real-time stream processing. *ACM Sigmod Rec.* 2005;34(4):42–7.
118. Wu W, Gruenwald L. Research issues in mining multiple data streams. In: *Proceedings of the first international workshop on novel data stream pattern mining techniques*. ACM. 2010. pp. 56–60.
119. Silva JA, Faria ER, Barros RC, Hruschka ER, De Carvalho AC, Gama J. Data stream clustering: a survey. *ACM Comput Surv.* 2013;46(1):13.
120. Angiulli F, Fassetto F. Detecting distance-based outliers in streams of data. In: *Proceedings of the sixteenth ACM conference on conference on information and knowledge management*. ACM. 2007. pp. 811–20.
121. Angiulli F, Fassetto F, Palopoli L. Detecting outlying properties of exceptional objects. *ACM Trans Database Syst.* 2009;34(1):7.
122. Kontaki M, Gounaris A, Papadopoulos AN, Tsiachas K, Manolopoulos Y. Continuous monitoring of distance-based outliers over data streams. In: *Data engineering (ICDE), 2011 IEEE 27th international conference on*. IEEE. 2011. pp. 135–46.
123. Zhang T, Ramakrishnan R, Livny M. Birch: an efficient data clustering method for very large databases. In: *ACM Sigmod record*, vol. 25. ACM. 1996. pp. 103–14.
124. Breiman L. Bias, variance, and arcing classifiers. 1996.
125. Schapire RE. The strength of weak learnability. *Mach Learn.* 1990;5(2):197–227.

126. Oza Nikunj C, Russell Stuart J. Online bagging and boosting. Jaakkola Tommi and Richardson Thomas, editors. In: Eighth international workshop on artificial intelligence and statistics. 2001. pp. 105–12.
127. Bifet A, Holmes G, Pfahringer B, Kirkby R, Gavaldà R. New ensemble methods for evolving data streams. In: Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining. ACM. 2009. pp. 139–48.
128. Narasimhamurthy AM, Kuncheva LI. A framework for generating data to simulate changing environments. In: Artificial intelligence and applications. 2007. pp. 415–20.
129. Bifet A, Gavaldà R. Learning from time-changing data with adaptive windowing. In: Proceedings of the 2007 SIAM international conference on data mining. SIAM. 2007. pp. 443–8.
130. Gama J, Medas P, Castillo G, Rodrigues P. Learning with drift detection. In: Brazilian symposium on artificial intelligence. Springer. 2004. pp. 286–95.
131. de Faria ER, Goncalves IR, Gama J, de Leon Ferreira ACP, et al. Evaluation of multiclass novelty detection algorithms for data streams. *IEEE Trans Knowl Data Eng.* 2015;27(11):2961–73.
132. Schadt EE, Linderman MD, Sorenson J, Lee L, Nolan GP. Computational solutions to large-scale data management and analysis. *Nat Rev Genet.* 2010;11(9):647.
133. Luengo J, García-Gil D, Ramírez-Gallego S, García S, Herrera F. Big data preprocessing.
134. Apache Hadoop. <https://hadoop.apache.org/>. Accessed 14 Feb 2020.
135. Apache Storm. <https://storm.apache.org/>. Accessed 14 Feb 2020.
136. Apache Spark. <https://spark.apache.org/>. Accessed 14 Feb 2020.
137. Apache Flink. <https://flink.apache.org/>. Accessed 14 Feb 2020.
138. Apache MXNet. <https://mxnet.apache.org/>. Accessed 14 Feb 2020.
139. García-Gil D, Ramírez-Gallego S, García S, Herrera F. A comparison on scalability for batch big data processing on apache spark and apache flink. *Big Data Anal.* 2017;2(1):1.
140. Koufakou A, Secretan J, Reeder J, Cardona K, Georgiopoulos M. Fast parallel outlier detection for categorical datasets using mapreduce. In: 2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence), IEEE. 2008. pp. 3298–304.
141. Jiang F, Leung CK-S, MacKinnon RK. Bigsam: mining interesting patterns from probabilistic databases of uncertain big data. In: Pacific-Asia conference on knowledge discovery and data mining. Springer. 2014. pp. 780–92.
142. Jiang F, Leung CK, Sarumi OA, Zhang CY. Mining sequential patterns from uncertain big DNA in the spark framework. In: Bioinformatics and biomedicine (BIBM), 2016 IEEE international conference on. IEEE. 2016. pp. 874–81.
143. Terzi, D.S., Terzi, R., Sagiroglu, S.: Big data analytics for network anomaly detection from netflow data. In: 2017 International conference on computer science and engineering (UBMK), IEEE. 2017. pp. 592–7.
144. Zhang W, Lu Y, Li Y, Qiao H. Convolutional neural networks on apache storm. In: 2019 Chinese automation congress (CAC), IEEE. 2019. pp. 2399–404.
145. Abeyrathna D, Huang P-C, Zhong X. Anomaly proposal-based fire detection for cyber-physical systems. In: 2019 International conference on computational science and computational intelligence (CSCI). IEEE. 2019. pp. 1203–7.
146. Toliopoulos T, Gounaris A, Tsihlas K, Papadopoulos A, Sampaio S. Continuous outlier mining of streaming data in flink. arXiv preprint [arXiv:1902.07901](https://arxiv.org/abs/1902.07901). 2019.
147. Gunter D, Tierney BL, Brown A, Swamy M, Bresnahan J, Schopf JM. Log summarization and anomaly detection for troubleshooting distributed systems. In: Grid computing, 2007 8th IEEE/ACM international conference on. IEEE. 2007. pp. 226–34.
148. Maruhashi K, Guo F, Faloutsos C. Multiaspectforensics: mining large heterogeneous networks using tensor. *Int J Web Eng Technol.* 2012;7(4):302–22.
149. Shin K, Hooi B, Faloutsos C. M-zoom: fast dense-block detection in tensors with quality guarantees. In: Joint European conference on machine learning and knowledge discovery in databases. Springer. 2016. pp. 264–80.
150. Hooi B, Song HA, Beutel A, Shah N, Shin K, Faloutsos C. Fraudar: Bounding graph fraud in the face of camouflage. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. ACM. 2016. pp. 895–904.
151. Jiang M, Beutel A, Cui P, Hooi B, Yang S, Faloutsos C. Spotting suspicious behaviors in multimodal data: a general metric and algorithms. *IEEE Trans Knowl Data Eng.* 2016;28(8):2187–200.
152. Angiulli F, Basta S, Lodi S, Sartori C. Gpu strategies for distance-based outlier detection. *IEEE Trans Parallel Distrib Syst.* 2016;27(11):3256–68.
153. Matsumoto T, Hung E, Yiu ML. Parallel outlier detection on uncertain data for gpus. *Distrib Parallel Databases.* 2015;33(3):417–47.
154. Lozano E, Acufia E. Parallel algorithms for distance-based and density-based outliers. In: Data mining, fifth IEEE international conference on. IEEE. 2005. p. 4.
155. O'Reilly C, Gluhak A, Imran MA. Distributed anomaly detection using minimum volume elliptical principal component analysis. *IEEE Trans Knowl Data Eng.* 2016;28(9):2320–33.
156. Dean J, Ghemawat S. Mapreduce: simplified data processing on large clusters. *Commun ACM.* 2008;51(1):107–13.
157. Ferreira Cordeiro RL, Traina Junior C, Machado Traina AJ, López J, Kang U, Faloutsos C. Clustering very large multi-dimensional datasets with mapreduce. In: Proceedings of the 17th ACM SIGKDD international conference on knowledge discovery and data mining. ACM. 2011. pp. 690–8.
158. Dolev S, Florissi P, Gudes E, Sharma S, Singer I. A survey on geographically distributed big-data processing using mapreduce. arXiv preprint [arXiv:1707.01869](https://arxiv.org/abs/1707.01869). 2017.
159. Zaharia M, Xin RS, Wendell P, Das T, Armbrust M, Dave A, Meng X, Rosen J, Venkataraman S, Franklin MJ, et al. Apache spark: a unified engine for big data processing. *Commun ACM.* 2016;59(11):56–65.
160. Meng X, Bradley J, Yavuz B, Sparks E, Venkataraman S, Liu D, Freeman J, Tsai D, Amde M, Owen S, et al. Mllib: machine learning in apache spark. *J Mach Learn Res.* 2016;17(1):1235–41.
161. van der Veen JS, van der Waaij B, Lazovik E, Wijbrandi W, Meijer RJ. Dynamically scaling apache storm for the analysis of streaming data. In: 2015 IEEE first international conference on big data computing service and applications. IEEE. 2015. pp. 154–61.

162. Chen T, Li M, Li Y, Lin M, Wang N, Wang M, Xiao T, Xu B, Zhang C, Zhang Z. Mxnet: A flexible and efficient machine learning library for heterogeneous distributed systems. arXiv preprint [arXiv:1512.01274](https://arxiv.org/abs/1512.01274). 2015.
163. Katsifodimos A, Schelter S. Apache flink: stream analytics at scale. In: 2016 IEEE international conference on cloud engineering workshop (IC2EW). IEEE. 2016. p. 193.
164. Carbone P, Katsifodimos A, Ewen S, Markl V, Haridi S, Tzoumas K. Apache flink: stream and batch processing in a single engine. *Bull IEEE Comput Soc Tech Comm Data Eng*. 2015;36(4).

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

Submit your next manuscript at ▶ [springeropen.com](https://www.springeropen.com)
