OUTLIER ANALYSIS

OUTLIER ANALYSIS

Authored by
CHARU C. AGGARWAL
IBM T. J. Watson Research Center, Yorktown Heights, NY, USA

Kluwer Academic Publishers
Boston/Dordrecht/London

Contents

Preface

Most of the earliest work on outlier detection was performed by the statistics community. While statistical methods are mathematically more precise, they suffer from several shortcomings, such as simplified assumptions about data representations, poor algorithmic scalability, and a low focus on interpretability. With the increasing advances in hardware technology for *data collection*, and advances in software technology (databases) for data *organization*, computer scientists have increasingly been participating in the latest advancements of this field. Computer scientists approach this field based on their practical experiences in managing large amounts of data, and with far fewer assumptions– the data can be of any type, structured or unstructured, and may be extremely large. Furthermore, issues such as computational efficiency and intuitive analysis of the data are generally considered more important by computer scientists than mathematical precision, though the latter is important as well. This is the approach of professionals from the field of data mining, an area of computer science, which was founded about 20 years ago. This has lead to the formation of multiple academic communities on the subject, which have remained separated, partially because of differences in technical style and opinions about the importance of different problems and approaches to the subject. At this point, data mining professionals (with a computer science background) are much more actively involved in this area, as compared to statisticians. This seems to be a major change in the research landscape. This book presents outlier detection from an integrated perspective, though the focus is towards computer science professionals. Special emphasis was placed on relating the methods from different communities with one another.

The key advantage of writing the book at this point is that the vast amount of work done by computer science professionals in the last two decades has remained largely untouched by a formal book on the subject. The classical books relevant to outlier analysis are as follows:

- P. Rousseeuw and A. Leroy. Robust Regression and Outlier Detection. *Wiley*, 2003.
- V. Barnett and T. Lewis. Outliers in Statistical Data, *Wiley*, 1994.
- D. Hawkins. Identification of Outliers, *Chapman and Hall*, 1980.

We note that these books are quite outdated, and the most recent among them is a decade old. Furthermore, this (most recent) book is really focussed on the relationship between regression and outlier analysis, rather than the latter. Outlier analysis is a much broader area, in which regression analysis is only a small part. The other books are even older, and are between 15 and 25 years old. They are exclusively targeted to the statistics community. This is not surprising, given that the first mainstream computer science conference in data mining (KDD) was organized in 1995. Most of the work in the data mining community was performed after the writing of these books. Therefore, many key topics of interest to the broader data mining community are not covered in these books. Given that outlier analysis has been explored by a much broader community, including databases, data mining, statistics, and machine learning, we feel that our book explores a much broader audience and brings together different points of view.

The chapters of this book have been organized carefully, with a view of covering the area extensively in an order which is natural. Emphasis was placed on simplifying the content, so that students and practitioners can also benefit from the book. While we did not originally intend to create a textbook on the subject, it evolved during the writing process into a work, which can also be used as a teaching aid. Furthermore, it can also be used as a reference book, since each chapter contains extensive bibliographic notes. Therefore, this book can serve a dual purpose, and provide a comprehensive exposition of the topic of outlier detection from multiple points of view.

Chapter 1

AN INTRODUCTION TO OUTLIER ANALYSIS

"Never take the comment that you are different as a condemnation, it might be a compliment. It might mean that you possess unique qualities that, like the most rarest of diamonds is ... one of a kind." – Eugene Nathaniel Butler

1. Introduction

An outlier is a data point which is significantly different from the remaining data. Hawkins formally defined [205] the concept of an outlier as follows:
"An outlier is an observation which deviates so much from the other observations as to arouse suspicions that it was generated by a different mechanism."

Outliers are also referred to as *abnormalities*, *discordants*, *deviants*, or *anomalies* in the data mining and statistics literature. In most applications, the data is created by one or more generating processes, which could either reflect activity in the system or observations collected about entities. When the generating process behaves in an unusual way, it results in the creation of outliers. Therefore, an outlier often contains useful information about abnormal characteristics of the systems and entities, which impact the data generation process. The recognition of such unusual characteristics provides useful application-specific insights. Some examples are as follows:

- **Intrusion Detection Systems:** In many host-based or networked computer systems, different kinds of data are collected about the operating system calls, network traffic, or other activity in the system. This data may show unusual behavior because of malicious

activity. The detection of such activity is referred to as intrusion detection.

- **Credit Card Fraud:** Credit card fraud is quite prevalent, because of the ease with which sensitive information such as a credit card number may be compromised. This typically leads to unauthorized use of the credit card. In many cases, unauthorized use may show different patterns, such as a buying spree from geographically obscure locations. Such patterns can be used to detect outliers in credit card transaction data.

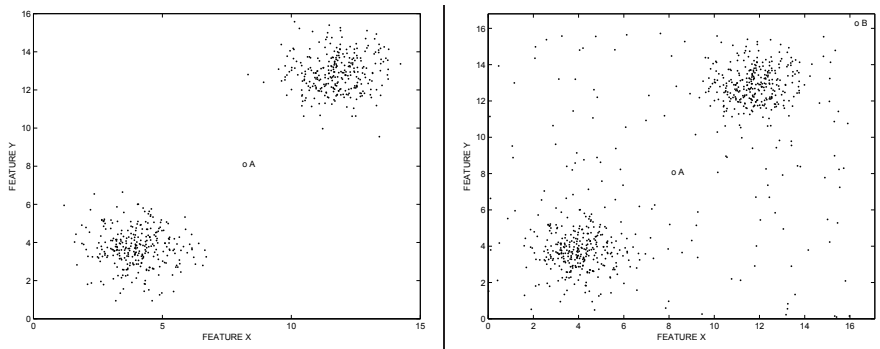
- **Interesting Sensor Events:** Sensors are often used to track various environmental and location parameters in many real applications. The sudden changes in the underlying patterns may represent events of interest. Event detection is one of the primary motivating applications in the field of sensor networks.

- **Medical Diagnosis:** In many medical applications the data is collected from a variety of devices such as MRI scans, PET scans or ECG time-series. Unusual patterns in such data typically reflect disease conditions.

- **Law Enforcement:** Outlier detection finds numerous applications to law enforcement, especially in cases, where unusual patterns can only be discovered over time through multiple actions of an entity. Determining fraud in financial transactions, trading activity, or insurance claims typically requires the determination of unusual patterns in the data generated by the actions of the criminal entity.

- **Earth Science:** A significant amount of spatiotemporal data about weather patterns, climate changes, or land cover patterns is collected through a variety of mechanisms such as satellites or remote sensing. Anomalies in such data provide significant insights about hidden human or environmental trends, which may have caused such anomalies.

In all these applications, the data has a "normal" model, and anomalies are recognized as deviations from this normal model. In many cases such as intrusion or fraud detection, the outliers can only be discovered as a sequence of multiple data points, rather than as an individual data point. For example, a fraud event may often reflect the actions of an individual in a particular sequence. The specificity of the sequence is relevant to identifying the anomalous event. Such anomalies are also referred to as *collective anomalies*, because they can only be inferred collectively from



(a) (b)

Figure 1.1. The difference between noise and anomalies

a *set or sequence* of data points. Such collective anomalies typically represent unusual *events*, which need to be discovered from the data. This book will address these different kinds of anomalies.

The output of an outlier detection algorithm can be one of two types:

- Most outlier detection algorithm output a score about the level of "outlierness" of a data point. This can be used in order to determine a ranking of the data points in terms of their outlier tendency. This is a very general form of output, which retains all the information provided by a particular algorithm, but does not provide a concise summary of the small number of data points which should be considered outliers.

- A second kind of output is a binary label indicating whether a data point is an outlier or not. While some algorithms may directly return binary labels, the outlier scores can also be converted into binary labels. This is typically done by imposing thresholds on outlier scores, based on their statistical distribution. A binary labeling contains less information than a scoring mechanism, but it is the final result which is often needed for decision making in practical applications.

It is often a subjective judgement, as to what constitutes a "sufficient" deviation for a point to be considered an outlier. In real applications, the data may be embedded in a significant amount of noise, and such noise may not be of any interest to the analyst. It is usually the *significantly interesting deviations* which are of interest. In order to illustrate this point, consider the examples illustrated in Figures 1.1(a) and (b). It is evident that the main patterns (or clusters) in the data are identical in both cases, though there are significant differences outside these main

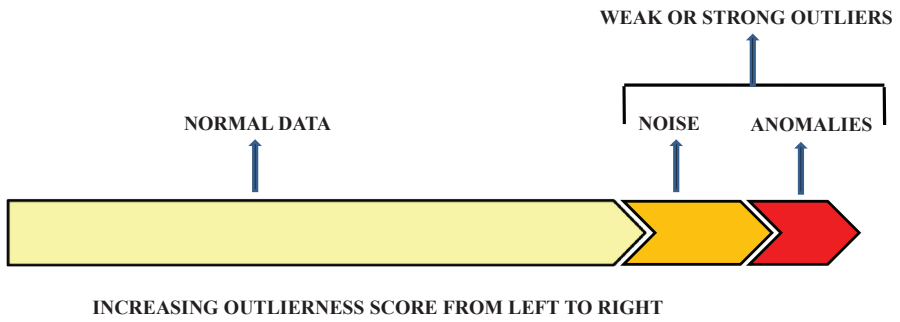


Figure 1.2. The spectrum from normal data to outliers

clusters. In the case of Figure 1.1(a), a single data point (marked by 'A') seems to be very different from the remaining data, and is therefore very obviously an anomaly. The situation in Figure 1.1(b) is much more subjective. While the corresponding data point 'A' in Figure 1.1(b) is also in a sparse region of the data, it is much harder to state confidently that it represents a true deviation from the remaining data set. It is quite likely that this data point represents randomly distributed noise in the data. This is because the point 'A' seems to fit a pattern represented by other randomly distributed points. Therefore, throughout this book the term "outlier" refers to a data point, which could either be considered an abnormality or noise, whereas an "anomaly" refers to a special kind of outlier, which is of interest to an analyst.

In the *unsupervised scenario*, where previous examples of interesting anomalies are not available, the noise represents the semantic boundary between normal data and true anomalies– noise is often modeled as a weak form of outliers, which does not always meet the strong criteria necessary for a data point to be considered interesting or anomalous enough. For example, data points at the boundaries of clusters may often be considered noise. Typically, most outlier detection algorithms use some quantified measure of the *outlierness* of a data point, such as the sparsity of the underlying region, nearest neighbor based distance, or the fit to the underlying data distribution. Every data point lies on a continuous spectrum from normal data to noise, and finally to anomalies, as illustrated in Figure 1.2. The separation of the different regions of this spectrum is often not precisely defined, and is chosen on an ad-hoc basis according to application-specific criteria. Furthermore, the separation between noise and anomalies is not pure, and many data points created by a noisy generative process may be deviant enough to be interpreted

as anomalies on the basis of the outlier score. Thus, anomalies will *typically* have a much higher outlier score than noise, but this is not a distinguishing factor between the two as a matter of *definition*. Rather, it is the interest of the analyst, which regulates the distinction between noise and an anomaly.

Some authors use the terms *weak outliers* and *strong outliers* in order to distinguish between noise and anomalies [4, 262]. The detection of noise in the data has numerous applications of its own. For example, the removal of noise creates a much cleaner data set, which can be utilized for other data mining algorithms. While noise may not be interesting in its own right, its *removal and identification* continues to be an important problem for mining purposes. Therefore, both noise and anomaly detection problems are important enough to be addressed in this book. Throughout this book, methods specifically relevant to *either* anomaly detection or noise removal will be identified. However, the bulk of the outlier detection algorithms could be used for either problem, since the difference between them is really one of semantics.

Since the semantic distinction between noise and anomalies is based on analyst interest, the best way to find such anomalies and distinguish them from noise is to use the feedback from *previously known outlier examples of interest*. This is quite often the case in many applications, such as credit-card fraud detection, where previous examples of interesting anomalies may be available. These may be used in order to learn *a model which distinguishes the normal patterns from the abnormal data.* Supervised outlier detection techniques are typically much more effective in many application-specific scenarios, because the characteristics of the previous examples can be used to sharpen the search process towards more relevant outliers. This is important, because outliers can be defined in numerous ways in a given data set, most of which may not be interesting. For example, in Figures 1.1(a) and (b), previous examples may suggest that only records with unusually high values of both attributes should be considered anomalies. In such a case, the point 'A' in *both* figures should be regarded as noise, and the point 'B' in Figure 1.1(b) should be considered an anomaly instead! The crucial point to understand here is that anomalies need to be *unusual in an interesting way*, and the supervision process re-defines what one might find interesting. Generally, unsupervised methods can be used either for noise removal or anomaly detection, and supervised methods are designed for application-specific anomaly detection.

Several levels of supervision are possible in practical scenarios. In the fully supervised scenario, examples of both normal and abnormal data are available, and can be clearly distinguished. In some cases, examples

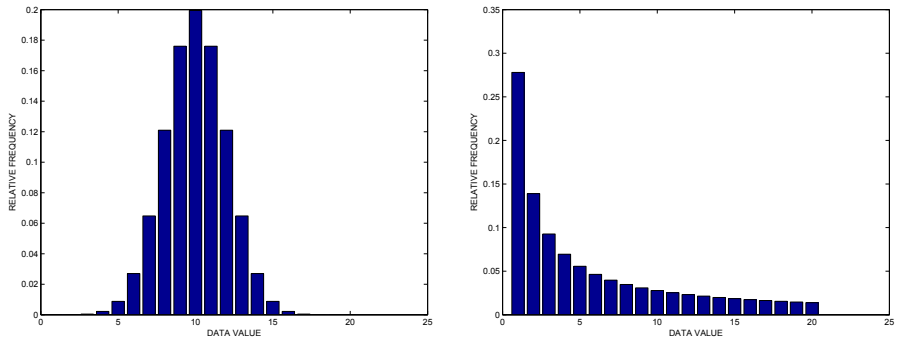
of outliers are available, but the examples of "normal" data may also contain outliers in some (unknown) proportion. This is referred to as classification with positive and unlabeled data. In other semi-supervised scenarios, only examples of normal data or only examples of anomalous data may be available. Thus, the number of variations of the problem are rather large, each of which requires a related but dedicated set of techniques.

Finally, the data representation may vary widely across applications. For example, the data may be purely multidimensional with no relationships among points, or the data may be sequential with temporal ordering, or may be defined in the form of a network with arbitrary relationships among data points. Furthermore, the attributes in the data may be numerical, categorical or may be mixed. Clearly, the outlier detection process needs to be sensitive to the nature of the attributes and relationships in the underlying data. In fact, the relationships themselves may often provide a criterion for outlier detection, in the form of connections between entities which do not usually occur together. Such outliers are referred to as *contextual* outliers. A classical example of this is the concept of *linkage outliers* in social network analysis [15]. In this case, entities (nodes) in the graph, which are normally not connected together may show *anomalous* connections with each other. Thus, the impact of data types on the anomaly detection process is significant, and will be carefully addressed in this book.

This chapter is organized as follows. In section 2, the importance of data modeling in outlier analysis is discussed. In section 3, the basic outlier models for outlier detection are introduced. Meta-algorithms for outlier analysis are addressed in section 4. Section 5 discusses the basic data types used for analysis. Section 6 introduces the concept of supervised modeling of outliers for data analysis. Methods for evaluating outlier detection algorithms are discussed in section 7. The conclusions are presented in section 8.

2. The Data Model is Everything

Virtually all outlier detection algorithms create a model of the normal patterns in the data, and then compute an outlier score of a given data point on the basis of the deviations from these patterns. For example, this data model may be a generative model such as a gaussian mixture model, a regression-based model, or a proximity-based model. All these models make different assumptions about the "normal" behavior of the data. The outlier score of a data point is then computed by evaluating the quality of the fit between the data point and the model. In many



(a) Normal Distribution (b) Zipf Distribution

Figure 1.3. Applying Z-value test on the Normal and Zipf distributions

cases, the model may be algorithmically defined. For example, nearest neighbor-based outlier detection algorithms model the outlier tendency of a data point in terms of the distribution of its k-nearest neighbor distance. Thus, in this case, the assumption is that outliers are located at large distances from most of the data.

Clearly, the choice of the data model is crucial. An incorrect choice of data model may lead to poor results. For example, a fully generative model such as the gaussian mixture model may not work well, if the data does not fit the generative assumptions of the model, or if a sufficient number of data points are not available to learn the parameters of the model. Similarly, a linear regression-based model may work poorly, if the underlying data is clustered arbitrarily. In such cases, data points may be incorrectly reported as outliers *because of poor fit to the erroneous assumptions of the model*. In practice, the choice of the model is often dictated by the analyst's understanding of the kinds of deviations relevant to an application. For example, in a spatial application measuring a behavioral attribute such as the location-specific temperature, it would be reasonable to assume that unusual deviations of the temperature attribute in a spatial locality is a indicator of abnormality. On the other hand, for the case of high-dimensional data, even the definition of data locality may be ill-defined because of data sparsity. Thus, an effective model for a particular data domain may only be constructed after carefully evaluating the relevant modeling properties of that domain.

In order to understand the impact of the model, it is instructive to examine the use of a simple model known as the *Z-value test* for outlier analysis. Consider a set of 1-dimensional quantitative data observations, denoted by $X_1 \ldots X_N$, with mean μ and standard deviation σ. The Z-

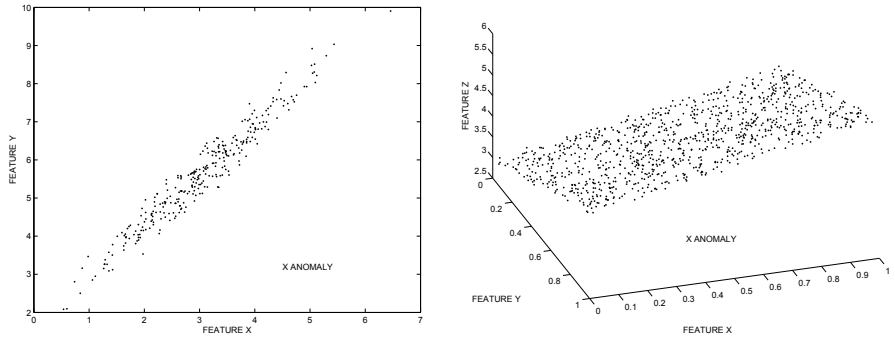
value for the data point X_i is denoted by Z_i, and is defined as follows:

$$Z_i = \frac{|X_i - \mu|}{\sigma} \tag{1.1}$$

The Z-value test computes the number of standard deviations by which the data varies from the mean. This provides a good proxy for the outliers in the data. An implicit assumption is that the data is modeled from a normal distribution. In cases where mean and standard deviation of the distribution can be accurately estimated (or are available from domain knowledge), a good "rule of thumb" is to use $Z_i \geq 3$ as a proxy for the anomaly. However, in many scenarios, where a smaller number of samples are available, the mean and standard deviation of the underlying distribution cannot be estimated accurately. In such cases, the results from the Z-value test need to be interpreted more carefully. This issue will be discussed in Chapter 2.

It is often forgotten by practitioners during outlier modeling, that the test implicitly assumes an approximately normal distribution for the underlying data. When this is not the case, the corresponding Z-values need to be interpreted carefully. For example, consider the two data frequency histograms drawn on values between 1 and 20 in Figure 1.3. In the first case, the histogram is sampled from a normal distribution with $(\mu, \sigma) = (10, 2)$, and in the second case, it is sampled from a Zipf distribution $1/i$. It is evident that most of the data lies in the range $[10 - 2 * 3, 10 + 2 * 3]$ for the normal distribution, and all data points lying outside this range can be truly considered anomalies. Thus, the Z-value test works very well in this case. In the second case with the Zipf distribution, the anomalies are not quite as clear, though the data with very high values (close to 20) can probably be considered anomalies. In this case, the mean and standard deviation of the data are 5.24 and 5.56 respectively. As a result, the Z-value test does not declare *any* of the data points as anomaly (for a threshold of 3), though it does come close. In any case, the significance of the Z-value from the Zipf-distribution is not very meaningful at least from the perspective of distribution of probabilities. This suggests that if mistakes are made at the modeling stage, it can result in an incorrect understanding of the data. While such tests are often used as a *heuristic* to provide a rough idea of the outlier scores even for data sets which are far from normally distributed, it is important to interpret such scores carefully.

An example in which the Z-value test would not work even as a heuristic, would be one in which it was applied to a data point, which was an outlier only because of its relative position, rather than its extreme position. For example, if the Z-value test is applied to an individual di-



(a) 2-d data (b) 3-d data

Figure 1.4. Linearly Correlated Data

mension in Figure 1.1(a), the test would fail miserably, because point A would be considered the most centrally located and normal data point. On the other hand, the test can still be reasonably applied to a set of *extracted* 1-dimensional values corresponding to the k-nearest neighbor distances of each point. Therefore, the effectiveness of a model depends both on the choice of the test used, and *how* it is applied.

The best choice of a model is often data set specific. This requires a good understanding of the data itself before choosing the model. For example, a regression-based model would be most suitable for finding the outliers in the data distributions of Figure 1.4, where most of the data is distributed along linear correlation planes. On the other hand, a clustering model would be more suitable for the cases illustrated in Figure 1.1. An attempt to use the wrong model for a given data set is likely to provide poor results. *Therefore, the core principle of discovering outliers is based on assumptions about the structure of the normal patterns in a given data set. Clearly, the choice of the "normal" model depends highly upon the analyst's understanding of the natural data patterns in that particular domain.*

There is no way around this issue; a highly general model with too many parameters will most likely overfit the data, and will also find a way to fit the outliers. A simple model, which is constructed with a good intuitive understanding of the data (and possibly also an understanding of what the analyst is looking for), is likely to lead to much better results. On the other hand, an oversimplified model, which fits the data poorly is likely to declare normal patterns as outliers. The initial stage of selecting the data model is perhaps the most crucial one in outlier analysis. The theme about the impact of data models will be repeated throughout the book, with specific examples.

3. The Basic Outlier Models

This section will present the broad diversity of the models in the literature, and provide some idea of the impact of using different data models. A detailed discussion of these methods are provided in later chapters. Several factor influence the choice of an outlier model, including the data type, data size, availability of relevant outlier examples, and the need for interpretability in a model. The last of these criteria deserves some further explanation.

The *interpretability* of an outlier detection model is extremely important from the perspective of the analyst. It is often desirable to determine *why* a particular data point is an outlier in terms of its relative behavior with respect to the remaining data. This provides the analyst further hints about the diagnosis required in an application-specific scenario. This is also referred to as the *intensional knowledge* about the outliers [262]. Different models have different levels of interpretability. Typically, models which work with the original attributes, and use fewer transforms on the data such as principal component analysis have higher interpretability. While data transformations can sometimes enhance the contrast between the outliers and normal data points, such transformations do come at the expense of interpretability. Therefore, it is critical to keep these factors in mind, while choosing a specific model for outlier analysis.

3.1 Extreme Value Analysis

The most basic form of outlier detection is extreme value analysis of 1-dimensional data. These are very specific kinds of outliers, in which it is assumed that the values which are either too large or too small are outliers. Such special kinds of outliers are also important in many application-specific scenarios.

The key is to determine the *statistical tails of the underlying distribution*. As illustrated earlier in Figure 1.3, the nature of the tails may vary considerably depending upon the underlying data distribution. The normal distribution is the easiest to analyze, because most statistical tests (such as the Z-value test) can be interpreted directly in terms of probabilities of significance. Nevertheless, even for arbitrary distributions, such tests provide a good heuristic idea of the outlier scores of data points, even when they cannot be interpreted statistically. The problem of determining the tails of distributions has been widely studied in the statistics literature. Details of such methods will be discussed in Chapter 2.

Extreme value statistics [364] is distinct from the traditional definition of outliers. The traditional definition of outliers, as provided by Hawkins, defines such objects by their *generative probabilities* rather than the extremity in their values. For example, in the data set $\{1, 2, 2, 50, 98, 98, 99\}$ of 1-dimensional values, the values 1 and 99, could very mildly, be considered extreme values. On the other hand, the value 50 is the average of the data set, and is most definitely not an extreme value. However, most probabilistic and density-based models would classify the value 50 as the strongest outlier in the data, on the basis of Hawkins' definition of generative probabilities. Confusions between extreme value analysis and outlier analysis are common, especially in the context of multivariate data. This is quite often the case, since many extreme value models also use probabilistic models in order to quantify the probability that a data point is an extreme value.

While extreme value analysis is naturally designed for univariate (one-dimensional) data, it is also possible to generalize it to multivariate data, by determining the points at the multidimensional *outskirts* of the data. It is important to understand that such outlier detection methods are tailored to determining *specific kinds of* outliers even in the multivariate case. For example, the point A in both Figures 1.1(a) and (b) will not be declared as an extreme value by such methods, since it does not lie on the outer boundary of the data, even though it is quite clearly an outlier in Figure 1.1(a). On the other hand, the point B in Figure 1.1(b) can be considered an extreme value, because it lies on the outskirts of the multidimensional data.

Extreme value modeling plays an important role in most outlier detection algorithms as a final step. *This is because most outlier modeling algorithms quantify the deviations of the data points from the normal patterns in the form of a numerical score.* Extreme value analysis is usually required as a final step on these modeled deviations, since they are now represented as univariate values in which extreme values correspond to outliers. In many multi-criteria outlier detection algorithms, a vector of outlier scores may be obtained (such as extreme values of temperature and pressure in a meteorological application). In such cases, multivariate extreme value methods can help *unify* these multiple outlier scores into a single value, and also generate a binary label output. Therefore, even though the original data may not be in a form where extreme value analysis is directly helpful, it remains an integral part of the outlier detection process. Furthermore, many variables are often tracked as statistical aggregates, in which extreme value analysis provides useful insights about outliers.

Extreme value analysis can also be extended to multivariate data with the use of distance-, or depth-based methods [243, 288, 388]. However, these methods are applicable only to certain kinds of specialized scenarios, where outliers are known to be present at the boundaries of the data. Many forms of post-processing on multi-criterion outlier scores may use such methods. On the other hand, such methods have often not found much utility in the literature for *generic* outlier analysis, because of their inability to discover outlier in the sparse *interior* regions of a data set.

3.2 Probabilistic and Statistical Models

In probabilistic and statistical models, the data is modeled in the form of a closed form probability distribution, and the parameters of this model are learned. Thus, the key assumption here is about the choice of the data distribution with which the modeling is performed. For example, a gaussian mixture model is a generative model, which characterizes the data in the form of a generative process containing a mixture of gaussian clusters. The parameters of these gaussian distributions are learned with the use of an *Expectation-Maximization (EM)* algorithm on the data set. A key output of this method is the membership probability of the data points to the different clusters, as well as the density-based fit to the modeled distribution. This provides a natural way to model the outliers, because data points which have very low fit values may be considered outliers. As discussed earlier, an extreme value test may be applied to these probability values in order to determine the outliers.

A major advantage of probabilistic models is that they can be easily applied to virtually any data type (or mixed data type), as long as an appropriate generative model is available for each mixture component. For example, if the data is categorical, then a discrete bernoulli distribution may be used to model each component of the mixture. For a mixture of different types of attributes, a product of the attribute-specific generative components may be used. Since such models work with probabilities, the issues of data normalization are already accounted for by the generative assumptions. Thus, probabilistic models provide a generic EM-based framework, which is relatively easy to apply to any specific data type. This is not necessarily the case for many other kinds of models.

A downside of probabilistic models is that they try to fit the data to a particular kind of distribution, which may often not be appropriate for the underlying data. Furthermore, as the number of model parameters increases, over-fitting becomes more common. In such cases, the outliers

may fit the underlying model of normal data. Many parametric models are also harder to interpret in terms of intensional knowledge, especially when the parameters of the model cannot be intuitively presented to an analyst in terms of underlying attributes. This can defeat one of the important purposes of anomaly detection, which is to provide diagnostic understanding of the abnormal data generative process. A detailed discussion of probabilistic methods, including the EM algorithm is provided in Chapter 2.

3.3 Linear Models

These methods model the data into lower dimensional embedded subspaces with the use of linear correlations [387]. For example, in the case of Figure 1.4, the data is aligned along a 1-dimensional line in a 2-dimensional space. The optimal line which passes through these points is determined with the use of regression analysis. Typically, a least squares fit is used to determine the optimal lower dimensional subspace. The distances of the data points from this plane are determined. Extreme values analysis can be applied on these deviations in order to determine the outliers. For example, in the 2-dimensional example of Figure 1.4, a linear model of the data points $\{(x_i, y_i), i \in \{1 \ldots N\}$ in terms of two coefficients a and b may be created as follows:

$$y_i = a \cdot x_i + b + \epsilon_i \quad \forall i \in \{1 \ldots N\} \tag{1.2}$$

Here ϵ_i represents the *residual*, which is essentially the error of the modeling. The coefficients a and b need to be *learned from the data* in order to minimize the least squares error denoted by $\sum_{i=1}^{N} \epsilon_i^2$. This is a convex non-linear programming problem, whose solution can be obtained either in closed form through either matrix operations (principal component analysis), or by gradient descent. The derived residuals can then be used in conjunction with extreme value analysis in order to determine the underlying outliers.

The concept of dimensionality reduction and principal component analysis (PCA) is quite similar [244], except that it uses a non-parametric approach in order to model the data correlations. PCA can be derived through multivariate regression analysis, by determining the plane which optimizes the least squares error of representation in terms of the normal distance to the plane. In other words, it provides the subspaces, such that by projecting the data into these subspaces, the aggregate least square errors of the residuals are minimized. The absolute sizes of these residuals can be analyzed in order to determine the outliers. Data points, which have large residuals, are more likely to be outliers, because their behavior does not conform to the natural subspace patterns in the

data. In addition, Principal Component Analysis techniques can be used for noise *correction* [18], where the attributes of data points are modified in order to reduce the noise in the data. Clearly, outlier data points are likely to be corrected more significantly than other data points.

Dimensionality reduction and regression modeling are particularly hard to interpret in terms of original attributes, when the underlying data dimensionality is high. This is because the subspace embedding is defined as a linear combination of attributes with positive or negative coefficients. This cannot easily be intuitively interpreted in terms specific properties of the data attributes. Dimensionality reduction and regression analysis methods for outlier detection are discussed in Chapter 3.

3.3.1 Spectral Models. Many of the matrix decomposition methods such as PCA are also used in the context of graphs and networks. The main difference is in how the matrix is created for decomposition. Such methods are also referred to as spectral models, when applied to certain kinds of data such as graphs and networks. Spectral methods are used commonly for clustering graph data sets, and are also used in order to identify anomalous changes in temporal sequences of graphs [229]. Such spectral models will be discussed in Chapter 11.

3.4 Proximity-based Models

The idea in proximity-based methods is to model outliers as points which are isolated from the remaining data. This modeling may be performed in one of three ways. Specifically, the three methods are cluster analysis, density-based analysis or nearest neighbor analysis. In clustering and other density-based methods, the dense regions in the data are found directly, and outliers are defined as those points, which do not lie in these dense regions. The main difference between clustering and density-based methods is that clustering methods segment the *points*, whereas the density-based methods segment the *space*.

In nearest neighbor methods [261, 381], the distance of each data point to its kth nearest neighbor is determined. By picking a value of $k > 1$, small groups of points, which are close together, but far away from the remaining data set are also treated as outliers. It is reasonable to treat such sets of data points as outliers, because small related sets of points can often be generated by an anomalous process. For example, consider the case illustrated in Figure 1.5, which contains a large cluster containing 4000 data points, and a small set of isolated but three closely spaced and related anomalies. Such situations are quite common, because anomalies which are caused by the same (rare) process, may result

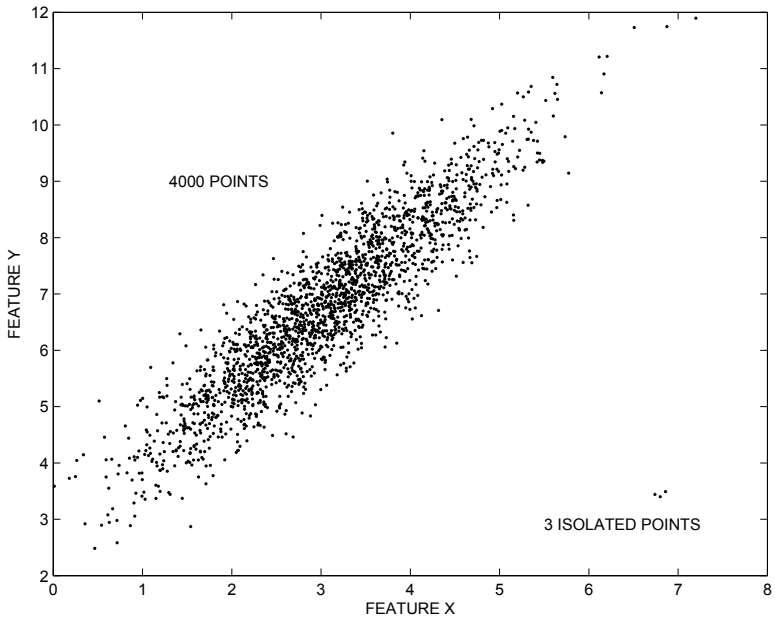


Figure 1.5. Small groups of anomalies can be a challenge to density-based methods

in small sets of data points which are similar to one another. In this case, the points within an anomaly set are close to one another, and cannot be distinguished on the basis of the 1-nearest neighbor distance. Such anomalies are often hard to distinguish from noise by using certain kinds of clustering and density-based algorithms, which are not sensitive to the global behavior of the data. On the other hand, the k-nearest neighbor approach can sometimes be effective. In the case of Figure 1.5, such sets of related anomalies may be identified by using $k \geq 3$. The kth nearest neighbor score provides an outlier score of the data set. This method can typically be computationally expensive, because it is required to determine the kth nearest neighbor of every point in the data set. Unless efficient indexing methods are available, this can require $O(N^2)$ time for a data set containing N points.

In the case of clustering methods, the first step is to use a clustering algorithm in order to determine the dense regions of the data set. In the second step, some measure of the fit of the data points to the different clusters is used in order to compute an outlier score for the data point. For example, in the case of a k-means clustering algorithm, the distance of the data point to the nearest centroid may be used as a measure of its anomalous behavior. One challenge with the use of many clustering algorithms is that they implicitly assume specific kinds of cluster shapes

depending upon the specific algorithm or distance function used within the clustering algorithm. Therefore, methods which divide the data into small regions in which the density can be estimated are very useful for scoring the sparsity of different regions in the data.

Density-based methods provide a high level of interpretability, when the sparse regions in the data can be presented in terms of combinations of the original attributes. For example, combinations of constraints on the original attributes can be presented as the specific criteria for particular data points being interpreted as outliers. Proximity-based methods for outlier detection are discussed in Chapter 4.

3.5 Information Theoretic Models

Many of the aforementioned models for outlier analysis use some form of data summarization method in terms of either generative probabilistic model parameters, clusters, or lower dimensional hyper-planes of projections. This provides a small summary of the data, the deviations from which are flagged as outliers. Information theoretic measures are broadly based on this principle. The idea is that outliers increase the minimum code length required to describe a data set. For example, consider the following two strings:

```
ABABABABABABABABABABABABABABAB
ABABACABABABABABABABABABABABAB
```

The second string is of the same length as the first, and is different at only a single position containing the unique symbol C. The first string can be described concisely as "AB 17 times". However, the second string has a single position corresponding to the alphabet "C". Therefore, the second string can no longer be described as concisely. In other words, the presence of the symbol C somewhere in the string increases its minimum description length. It is also easy to see that this symbol corresponds to an outlier. Information theoretic models are closely related to conventional models, because both use a concise representation of the data set as a baseline for comparison. For example, in the case of multidimensional data sets, both kinds of models use the following different kinds of concise descriptions.

- A probabilistic model describes a data set in terms of generative model parameters, such as a mixture of gaussian distributions or a mixture of exponential power distributions [74].
- A clustering or density-based summarization model describes a data set in terms of cluster descriptions, histograms or other summarized representations, along with maximum error tolerances [233].

- A PCA model or spectral model describes the data in terms of lower dimensional subspaces of projection of multi-dimensional data or a condensed representation of a network [429].

- A frequent pattern mining method describes the data in terms of an underlying code book of frequent patterns. These are among the most common methods used for information-theoretic anomaly detection [34, 123, 410].

All these models represent the data approximately in terms of individual condensed components representing aggregate trends. In general, outliers increase the length of the description in terms of these condensed components to achieve the same level of approximation. For example, a data set with outliers will require a larger number of mixture parameters, clusters, PCA-based subspace dimensionality, or frequent patterns in order to achieve *the same level of approximation*. Correspondingly, in information theoretic methods, the key idea is to construct a *code book* in which to represent the data, and outliers are defined as points which removal results in the *largest decrease* in description length [123], or the most accurate summary representation in the same description length after removal [233]. The term "code book" is rather loosely defined in outlier analysis and refers to the condensed aggregate components of the data in terms of which the data is described. The actual construction of the coding is often heuristic, and an effective choice is key to the success of the approach. In general, the determination of the minimum length coding is a computationally intractable problem for a given data set, and therefore a variety of heuristic models (or code books) may be used for representation purposes [34, 123, 233, 410]. In many cases, these techniques can be related to conventional data summarization models for outlier analysis. In some cases, the coding is not explicitly constructed, but measures such as the entropy or Kolmogorov complexity are used as a surrogate in order to estimate the level of unevenness of a specific segment of the data [297, 259]. Segments with greater unevenness may be selectively explored to determine the outliers.

Conventional models look at this problem in a complementary way, by defining outliers as points which are expressed in the *least precise way* by (or deviations from) from a *fixed* model with a particular length. On the other hand, information theoretic models examine the *differential impact* of *removing* an outlier point from the data set on the tradeoff between coding length and representation accuracy. The two are clearly closely related. Since information theoretic methods largely differ from conventional models in terms of how the measure is defined, they often use similar methods as conventional techniques (eg. frequent pattern

mining [34, 410], histograms [233] or spectral methods [429]) in order to create the coding representation. Therefore, information theoretic methods will be discussed at various places in this book along with the chapter containing similar techniques or data types. Information theoretic methods can also be used for change detection in temporal data [96], by examining specific temporal segments of the data, and measuring the description length of these segments. The segments with the greatest change will typically have a larger description length.

3.6 High-Dimensional Outlier Detection

The high-dimensional case is particularly challenging for outlier detection. This is because, in high dimensionality, the data becomes sparse, and all pairs of data points become almost equidistant from one another [22, 215]. From a density perspective, all regions become almost equally sparse in full dimensionality. Therefore, it is no longer meaningful to talk in terms of extreme value deviations based on the distances in full dimensionality. The reason for this behavior is that many dimensions may be very noisy, and they may show similar pairwise behavior in terms of the addition of the dimension-specific distances. The sparsity behavior in high dimensionality makes all points look very similar to one another.

A salient observation is that the true outliers may only be discovered by examining the distribution of the data in a lower dimensional *local* subspace [4]. In such cases, *outliers are often hidden in the unusual local behavior of lower dimensional subspaces, and this deviant behavior is masked by full dimensional analysis.* Therefore, it may often be fruitful to explicitly search for the appropriate subspaces, where the outliers may be found. This approach is a generalization of both (full-dimensional) clustering and (full data) regression analysis. It combines local data pattern analysis with subspace analysis in order to mine the significant outliers. This can be a huge challenge, because the simultaneous discovery of relevant data localities and subspaces in high dimensionality can be computationally very difficult. Typically evolutionary heuristics such as genetic algorithms can be very useful in exploring the large number of underlying subspaces [4].

High-dimensional methods provide an interesting direction for intensional understanding of outlier analysis, when the subspaces are described in terms of the original attributes. In such cases, the output of the algorithms provide *specific combinations of attributes along with data locality*, which resulted in such data points being declared as outliers. This kind of interpretability is very useful, when a small number of interesting attributes need to be selected from a large number of possibil-

ities for outlier analysis. Methods for high dimensional outlier detection are discussed in Chapter 5.

4. Meta-Algorithms for Outlier Analysis

In many data mining problems such as clustering and classification, a variety of meta-algorithms are used in order to improve the robustness of the underlying solutions. For example, in the case of the classification problem, a variety of ensemble methods such as bagging, boosting and stacking are used in order to improve the robustness of the classification [146]. Similarly, in the case of clustering, ensemble methods are often used in order to improve the quality of the clustering [20]. Therefore, it is natural to ask whether such meta-algorithms also exist for the outlier detection problem. The answer is in the affirmative, though the work on meta-algorithms for outlier detection is often quite scattered in the literature, and in comparison to other problems such as classification, not as well formalized. In some cases such as sequential ensembles, the corresponding techniques are often repeatedly used in the context of *specific* techniques, though are not formally recognized as general purpose meta-algorithms which can be used in order to improve outlier detection algorithms. The different meta-algorithms for outlier detection will be discussed in the following subsections.

There are two primary kinds of ensembles, which can be used in order to improve the quality of outlier detection algorithms:

- In *sequential ensembles*, a given algorithm or set of algorithms are applied sequentially, so that future applications of the algorithms are impacted by previous applications, in terms of either modifications of the base data for analysis or in terms of the specific choices of the algorithms. The final result is either a weighted combination of, or the final result of the last application of an outlier analysis algorithm. For example, in the context of the classification problem, boosting methods may be considered examples of sequential ensembles.

- In *independent ensembles*, different algorithms, or different instantiations of the same algorithm are applied to either the complete data or portions of the data. The choices made about the data and algorithms applied are independent of the results obtained from these different algorithmic executions. The results from the different algorithm executions are combined together in order to obtain more robust outliers.

Algorithm SequentialEnsemble(Data Set: $\mathcal{D}$
Base Algorithms: $\mathcal{A}_1 \ldots \mathcal{A}_r$)
begin
$j = 1$;
repeat
Pick an algorithm $\mathcal{A}_j$ based on results from past executions;
Create a new data set $f_j(\mathcal{D})$ from $\mathcal{D}$ based on results from past executions;
Apply $\mathcal{A}_j$ to $\mathcal{D}_j$;
$j = j + 1$;
until(termination);
report outliers based on combinations of results from previous executions;
end

Figure 1.6. Sequential Ensemble Framework

4.1 Sequential Ensembles

In sequential-ensembles, one or more outlier detection algorithms are applied sequentially to either all or portions of the data. The core principle of the approach is that each application of the algorithms provides a better understanding of the data, so as to enable a more refined execution with either a modified algorithm or data set. Thus, depending upon the approach, either the data set or the algorithm may be changed in sequential executions. If desired, this approach can either be applied for a fixed number of times, or be used in order to converge to a more robust solution. The broad framework of a sequential-ensemble algorithm is provided in Figure 1.6.

In each iteration, a successively refined algorithm may be used on a refined data, based on the results from previous executions. The function $f_j(\cdot)$ is used to create a refinement of the data, which could correspond to data subset selection, attribute-subset selection, or generic data transformation methods. The description above is provided in a very general form, and many special cases can be possibly instantiated from this general framework. For example, in practice, only a single algorithm may be used on successive modifications of the data, as data is refined over time. Furthermore, the sequential ensemble may be applied in only a small number of constant passes, rather than a generic

convergence-based approach, as presented above. The broad principle of sequential ensembles is that a greater knowledge of data with successive algorithmic execution helps focus on techniques and portions of the data which can provide fresh insights.

Sequential ensembles have not been sufficiently explored in the outlier analysis literature as general purpose meta-algorithms. However, many *specific* techniques in the outlier literature use methods, which can be recognized as special cases of sequential ensembles. A classic example of this is the use of two-phase algorithms for building a model of the normal data. In the first-phase, an outlier detection algorithm is used in order to remove the obvious outliers. In the second phase, *a more robust* normal model is constructed after removing these obvious outliers. Thus, the outlier analysis in the second stage is much more refined and accurate. Such approaches are commonly used for cluster-based outlier analysis (for constructing more robust clusters in later stages) [55], or for more robust histogram construction and density estimation (see Chapter 4). However, most of these methods are presented in the outlier analysis literature as specific optimizations of *particular* algorithms, rather than as general meta-algorithms which can improve the effectiveness of an *arbitrary* outlier detection algorithm. There is significant scope for further research in the outlier analysis literature, by recognizing these methods as general-purpose ensembles, and using them to improve the effectiveness of outlier detection.

4.2 Independent Ensembles

In independent ensembles, different instantiations of the algorithm or different portions of the data are used for outlier analysis. Alternatively, the same algorithm may be applied, but with either a different initialization, parameter set or even random seed in the case of a randomized algorithms. The results from these different algorithm executions can be combined in order to obtain a more robust outlier score. A general purpose description of independent ensemble algorithms is provided in the pseudo-code description of Figure 1.7.

The broad principle of independent ensembles is that different ways of looking at the same problem provides more robust results which are not dependent on specific artifacts of a particular algorithm or data set. Independent ensembles have been explored much more widely and formally in the outlier analysis literature, as compared to sequential ensembles. Independent ensembles are particularly popular for outlier analysis in high-dimensional data sets, because they enable the exploration of dif-

Algorithm IndependentEnsemble(Data Set: $\mathcal{D}$
Base Algorithms: $\mathcal{A}_1 \ldots \mathcal{A}_r$)
begin
$j = 1$;
repeat
Pick an algorithm $\mathcal{A}_j$;
Create a new data set $f_j(\mathcal{D})$ from $\mathcal{D}$;
Apply $\mathcal{A}_j$ to $f_j(\mathcal{D})$;
$j = j + 1$;
until(termination);
report outliers based on combinations of results
from previous executions;
end

Figure 1.7. Independent Ensemble Framework

ferent subspaces of the data in which different kinds of deviants may be found. These methods will be discussed in detail in Chapter 5.

Examples exist of both picking different algorithms and data sets, in order to combine the results from different executions. For example, the methods in [289, 310] sample subspaces from the underlying data in order to determine outliers from each of these executions independently. Then, the results from these different executions are combined in order to determine the outliers. The idea in these methods is that results from different subsets of sampled features may be bagged in order to provide more robust results. Some of the recent methods for subspace outlier ranking and outlier evaluation can be considered independent ensembles which combine the outliers discovered in different subspaces in order to provide more robust insights. These methods will be discussed in detail in Chapter 5.

5. The Basic Data Types for Analysis

Most of our aforementioned discussion in the previous sections was focussed on multidimensional numerical data. Furthermore, it was assumed that the data records are independent of one another. However, in practice, the underlying data may be much more complex, both in terms of the kinds of attributes, and the relationships between different data records. Some examples of the different kinds of data, which may be encountered in real applications are discussed in this section.

5.1 Categorical, Text and Mixed Attributes

Many data sets in real applications may contain categorical attributes, which take on *discrete unordered* values. For example, demographic data may contain attributes such as race, gender, or the zip-code. Such attribute values are not ordered, and therefore require different data analysis techniques. Furthermore, the different kinds of attributes (numerical and categorical) may be mixed with one another. Many of the techniques for nearest neighbor and density-based classification can be extended to the case of such attributes, because the concept of proximity can be extended to such cases. The major challenge is to construct a distance function, which remains semantically meaningful for the case of discrete data.

Regression-based models can also be used in a limited way over discrete attribute values, when the number of possible values of an attribute is not too large. The typical methodology is to convert the discrete data to binary data by creating one attribute for each categorical value. Regression models such as principal component analysis may then be applied to this binary data set. Such methods can be more easily extended to text, where there is an inherent ordering among the frequencies of the words. In such cases, the correlations among occurrence of text words can be used in order to create linear-regression based models. In fact, some of the most successful models for text de-noising are based on latent semantic indexing (LSI), which is a form of linear regression analysis [133]. Other common methods for text and categorical data include clustering [26], proximity-based methods [515], probabilistic models [478], and methods based on frequent pattern mining [34, 208, 410]. Methods for outlier detection in categorical and mixed attribute data sets are discussed in Chapter 7.

5.2 When the Data Values have Dependencies

Most of the aforementioned discussion in this chapter is about the common multidimensional scenario, where it is assumed that the data records can be treated independently of one another. In practice, the different data values may be related to each other temporally, spatially, or through explicit network relationship links between the data items. The presence of such dependencies greatly changes the anomaly detection problem, because the nature of the dependencies plays a critical role in the anomaly detection process. In such cases, the *expected values* of data items are influenced by their contextual dependencies, and therefore outliers are defined on the basis of such contextually modeled deviations. When a single data item (eg. value from a time series) is

declared as an anomaly because of its *relationship* to its related data items, it is referred to as a *contextual* outlier or anomaly. Such outliers are also sometimes referred to as *conditional anomalies* [416]. For example, a sudden spike in a time series is a contextual anomaly, because it is very different from its expected value based on the values of its adjacent items. When a set of data items are declared anomalous *as a group of points*, it is referred to as a *collective* anomaly or outlier. For example, an unusual and rapid oscillation over time for a stock ticker value may be considered a collective anomaly, and it includes all the data items in the oscillation. Virtually, all anomalies in dependency-oriented data are *contextual* or *collective* anomalies, because they compute *expected* values based on relationships with adjacent data points in order to determine unexpected patterns. Furthermore, in such data sets, there are usually *multiple ways* to model anomalies, depending upon what an analyst might be looking for. Some examples of such data domains are presented in this section.

5.2.1 Times Series Data and Data Streams. Time-series contains a set of values which are typically generated by continuous measurement over time. Therefore, the values in consecutive time-stamps do not change very significantly, or change in a smooth way. In such cases, *sudden changes* in the underlying data records, can be considered *anomalous events*. Therefore the discovery of anomalous points in time series, is usually closely related to the problem of anomalous *event detection*, in the form of either *contextual or collective anomalies over related time stamps* [9, 16, 260]. Typically such events are created by a sudden change in the underlying system, and may be of considerable interest to an analyst. For example, let us consider the following time-series of values, along with the corresponding time-stamps implicitly defined by the index of the data point.

3, 2, 3, 2, 3, 87, 86, 85 87, 89, 86, 3, 84, 91, 86, 91, 88

The time-series is illustrated in Figure 1.8. It is evident that there is a sudden change in the data value at time-stamp 6 from 3 to 87. This corresponds to an outlier. Subsequently, the data stabilizes at this value, *and this becomes the new normal*. At time-stamp 12, the data value again dips to 3. *Even though this data value was encountered before*, it is still considered an outlier because of the sudden change in the consecutive data values. Thus, it is critical to understand that in this case, treating the data values independent of one another is not helpful for anomaly detection, because the data values are highly influenced by the adjacent values of the data points. Thus, the problem of outlier detection in time

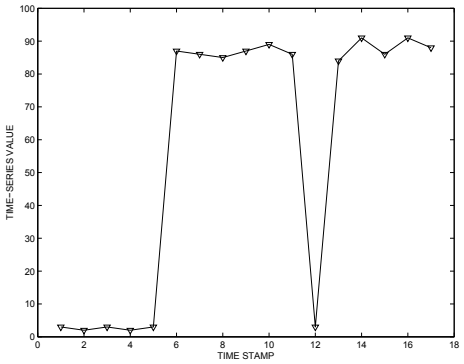


Figure 1.8. Example of Time Series

series data is highly related to the problem of change detection, because the normal models of data values are highly governed by adjacency in temporal ordering. When completely new data values are encountered, they are referred to as *novelties* [328, 329, 325], though outlier detection is relevant to any form of abrupt change, rather than only novelties, which are a specific kind of outliers.

It should be emphasized that change analysis and outlier detection (in temporal data) are very closely related areas, but not necessarily identical. The change in a temporal data set could happen in one of two possible ways:

- The values and trends in the data stream change slowly over time, a phenomenon which is referred to as *concept drift* [327, 10]. In such cases, the concept drift can only be detected by detailed analysis over a long period of time, and is not immediately obvious in many circumstances.

- The values and trends in the data stream change *abruptly*, so as to *immediately arouse suspicion that the underlying data generation mechanism has somehow changed fundamentally.*

The first scenario does not necessarily correspond to outliers, though the second scenario is more relevant to outlier detection. It is easy to see the parallels between the second scenario and the definition of outliers due to Hawkins [205], which was introduced at the very beginning of this chapter.

A common challenge in such scenarios is to perform the outlier detection *in real time*, as new data values are encountered. Many scenarios of change analysis and anomaly detection in temporal data are too tightly integrated to be treated separately. In such cases, solutions for one can

be used for the other, and vice-versa. On the other hand, the formulations of anomaly detection in temporal data are very diverse, not all of which are directly related to change detection. Usually online analysis is suited to change detection, whereas offline analysis may explore other unusual aspects of the data. Some examples are as follows:

- When the data is in the form of a time-series (eg, sensor data) large changes in *trends* may correspond to anomalies. These can be discovered as deviations from forecasted values using window-based analysis. In some cases, it may be desired to determine time-series subsequences of unusual shapes rather than change points in the data.
- For multidimensional data streams, changes in the aggregate distribution of the streaming data may correspond to unusual events. For example, network intrusion events may cause *aggregate* change points in a network stream. On the other hand, *individual* point novelties may or may not correspond to aggregate change points. The latter case is similar to multidimensional anomaly detection with an efficiency constraint for the streaming scenario.

Methods for anomaly detection in time series data and multidimensional data streams are discussed in detail in Chapter 8.

5.2.2 Discrete Sequences. Many discrete sequence-based applications such as intrusion-detection and fraud-detection are clearly temporal in nature. This scenario can be considered a categorical or discrete analogue of time series data. Discrete sequences may not necessarily be temporal in nature, but may be based on their relative *placement* with respect to one another. An example is the case of biological data, where the sequences are defined on the basis of their relative placement.

As in the case of autoregressive models of continuous data, it is possible to use (typically markovian) *prediction-based* techniques in order to forecast the value of a single *position* in the sequence. Deviations from forecasted values correspond to *contextual* outliers. It is often desirable to perform the prediction in real time. In other cases, anomalous events can be identified only by variations from the normal *patterns* exhibited by the *subsequences* over multiple time stamps. This is analogous to the problem of unusual *shape detection* in time series data, and it represents a set of *collective* outliers.

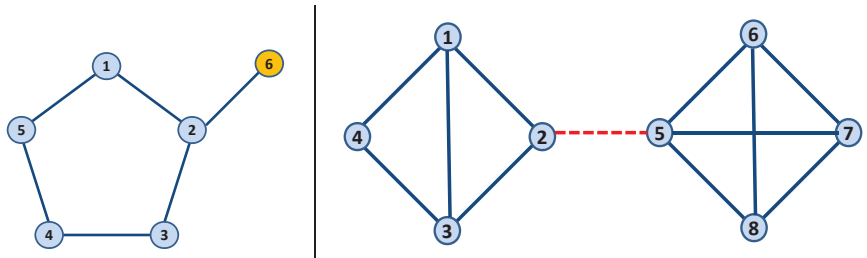
Therefore, discrete sequences are analogous to continuous sequences, except that the different data representation typically requires different similarity functions, representation data structures, and more complex predictive techniques such as markovian models as opposed to autore-

gressive forecasting techniques. The problem formulations for the two cases are also similar at the high level. On the other hand, the specific techniques used for the two cases are very different. This is quite simply because numerical time series values are *ordered*, and therefore the values can be meaningfully compared across a continuous spectrum. However, two different discrete values cannot be meaningfully compared in a similar way. Value-continuity is lost in discrete data. Therefore, in order to maintain a coherent presentation, the case of discrete sequences has been addressed in a different chapter.

Discrete data is common in many real applications. Most biological sequences are discrete in nature, where each value in the sequence is drawn from a discrete set of possibilities. Similarly, host-based intrusion applications typically lead to discrete data, because numerous diagnostic events are drawn from a discrete set of instances [108]. Methods for anomaly detection in discrete sequences are discussed in Chapter 9.

5.2.3 Spatial Data. In spatial data, many non-spatial attributes (eg. temperature, pressure, image pixel color intensity) are measured at spatial locations. Unusual local changes in such values are reported as outliers. It should be pointed out that outlier detection in temporal data shares some resemblance to that in spatial data [433]. Both typically require the attribute of interest to exhibit a certain level of continuity. For example, consider the measurement of the temperature, where the measurement could be associated with a time-stamp and spatial coordinates. Just as it is expected that temperatures at consecutive time-stamps do not vary too much (temporal continuity), it is also expected that temperatures at spatially close locations do not vary too much (spatial continuity). In fact, such unusual spatial variations in sea surface temperatures and pressures [433] are used in order to identify significant and anomalous spatiotemporal events in the underlying data (eg. formation of cyclones). Spatiotemporal data is a generalization of both spatial and temporal data, and the methods used in either domain can often be generalized to such scenarios. Methods for finding outliers in spatial data are discussed in Chapter 10.

5.2.4 Network and Graph Data. In network or graph data, the data values may correspond to nodes in the network, whereas the relationships among the data values may correspond to the edges in the network. In such cases, outliers may be modeled in different ways depending upon the irregularity of *either* the nodes in terms of their relationships to other nodes, or the edges themselves. For example, a node which shows irregularity in its structure within its locality may be



(a) Node Outlier (b) Edge Outlier

Figure 1.9. Examples of Node and Edge Outliers

considered an outlier [33]. Similarly, an edge which connects disparate communities of nodes may be considered a *relationship* or *community outlier* [15, 180]. In Figure 1.9, two examples of outliers in networks are illustrated. The left example in Figure 1.9(a) contains an example of a node outlier, because the node 6 has an unusual locality structure which is significantly different from the other nodes. On the other hand, the edge (2, 5) in Figure 1.9(b) may be considered a relationship outlier or community outlier, because it connects two communities which are usually not connected to one another. Thus, in graph data, there is significantly more complexity and flexibility in terms of how outliers may be defined or modeled. In general, *the more complex the data, the more the analyst has to make prior inferences of what is considered normal for modeling purposes.*

It is also possible to combine different kinds of dependencies in a given data set. For example, graphs may be temporal in nature. In such a case, the data may have both structural and temporal dependencies, which change and also influence each other over time [15]. Therefore, outliers may be defined in terms of significant changes in the underlying network community or distance structure. Such models combine network analysis and change detection in order to detect structural and temporal outliers from the underlying data. A detailed discussion of methods for temporal and non-temporal outlier detection in graphs is provided in Chapter 11.

6. Supervised Outlier Detection

In many scenarios, previous examples of outliers may be available. A subset of the data may be labeled as anomalies, whereas the remaining data may be considered normal. The corresponding methods are referred to as *supervised outlier detection*, because the labels are used in order to train a model which can determine *specific kinds* of anomalies.

Supervised methods are *generally* designed for anomaly detection, rather than noise removal, because they are based on the assumption that the labels represent what an analyst might specifically be looking for, rather than examples of what one might want to remove for data cleaning. Supervised models may often provide *very different* results from the unsupervised case, because they reflect an understanding of the underlying data. For example, let us consider the following time-series data:

3, 2, 3, 2, 3, 87, 2, 2, 3, 3, 3, 84, 91, 86, 91, 81

In this case, sudden changes in the data values (at 87 and 84) may be considered anomalies in the unsupervised scenario. However, in an application such as credit-card transaction levels, previous labeled examples of time-series may suggest that high consecutive values of the data should be considered anomalous. In such cases, the first occurrence of 87 should not be considered anomalous, whereas the occurrence of 84 along with its following values should be considered (collectively) anomalous.

Supervised anomaly detection finds numerous applications in fraud detection, intrusion detection, fault and disease diagnosis. In all these cases, the class of interest is very rare. It is this rarity that makes these instances outliers. Furthermore, it is usually much more important to correctly identify all the outliers, rather than the normal instances.

Supervised outlier detection is a (difficult) special case of the classification problem. The main characteristic of this problem is that the labels are extremely unbalanced in terms of relative presence [102]. The normal data is usually easy to collect and is therefore copiously available. On the other hand, outlier examples are very sparsely available in the data. In the classical machine learning literature, this problem is also referred to as the *rare class detection* problem. The imbalance in the class labels may often make the problem rather difficult to solve, because very few instances of the rare class may be available for modeling purposes. This may also make standard classification models prone to over-training, since the actual data *distinguishing* the rare class from the normal data is quite small. Furthermore, several variations of the classification problem also correspond to different levels of supervision:

- A limited number of instances of the positive (outlier) class may be available, whereas the "normal" examples may contain an unknown proportion of outliers [152]. This is referred to as the Positive-Unlabeled Classification (PUC) problem in machine learning. This variation is still quite similar to the fully supervised rare-class scenario, except that the classification model needs to be more cognizant of the contaminants in the negative (unlabeled) class. In cases, where the unlabeled training instances do not ac-

curately reflect the test instances, it may be desirable to discard the training instances for the unlabeled class, and treat it as a one-class problem, where only positive instances are available.

- Only instances of a subset of the normal and anomalous classes may be available, but some of the anomalous classes may be missing from the training data [325, 326, 445]. Such outliers are also referred to as *semi-supervised novelties*. This is distinct from *unsupervised novelties*, which tracks the formation of new clusters and trends in the data [26, 503, 515]. For example, in a bio-terrorist attack modeling scenario, no examples of the attack may be available, whereas copious examples of normal behavior and other kinds of more common anomalies may be available. This variation is also a semi-supervised scenario for learning, though it is quite similar to the unsupervised version of the problem. A more interesting case is one in which labeled examples of all normal and some anomalous classes are available, though the labels for the anomalous classes are not exhaustive. Such situations are quite common in scenarios such as intrusion detection, where some intrusions may be known, but other intrusions are continually created over time.

- Supervised outlier detection is closely related to active learning, in which human feedback is utilized in order to identify relevant outlier examples. This is because such methods do create models distinguishing between positive and negative examples of outliers, even when the example identification process is executed in parallel with the classification [360]. This process is also referred to as *Active Learning*.

All these different variants require careful design of the underlying classification algorithms. For example, cost-sensitive variations of standard machine learning algorithms can be used in order to make accurate predictions of anomalies in the data [151]. In such variations, the classifier is tuned, so that errors in classification of the anomalous class are penalized more heavily than the errors in classification of the majority class. The idea is that it is better to predict a negative class as an anomaly (false positive), rather than miss a true outlier (false negative). The different choices on costs may lead to different tradeoffs between false positives and false negatives. This tradeoff is characterized by either a *Precision-Recall (PR)* curve, or a *Receiver Operating Characteristics (ROC)* curve. These two kinds of curves are intimately related to one another. The issue of outlier evaluation will be discussed in the next section. Supervised methods for anomaly detection are discussed in greater detail in Chapter 6.

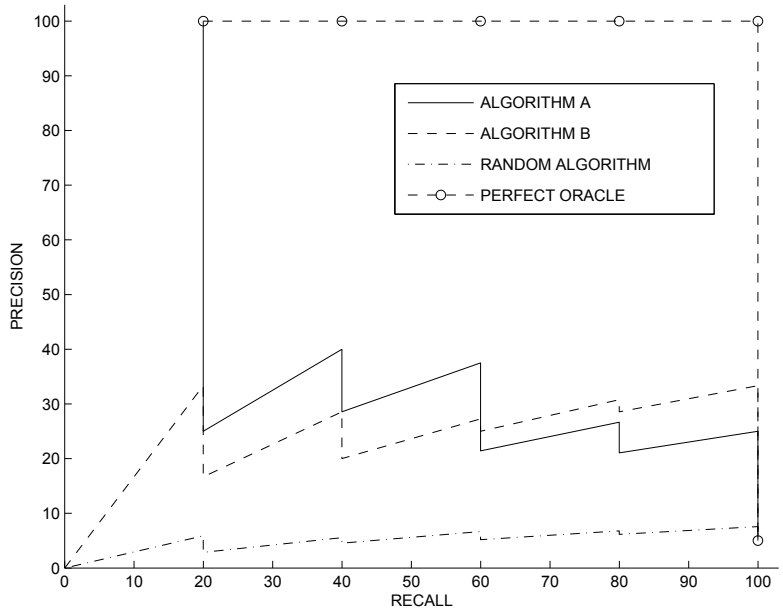


Figure 1.10. Precision-Recall Curves

Algorithm	Rank of Ground-truth Outliers
Algorithm A	1, 5, 8, 15, 20
Algorithm B	3, 7, 11, 13, 15
Random Algorithm	17, 36, 45, 59, 66
Perfect Oracle	1, 2, 3, 4, 5

Table 1.1. Rank of ground-truth outliers can be used to construct Precision-Recall curves

7. Outlier Evaluation Techniques

A key question arises as to how the effectiveness of an outlier detection algorithm should be evaluated. Unfortunately, this is often a difficult task, because outliers, by definition, are rare. This means that the ground-truth about which data points are outliers, is often not available. This is especially true for unsupervised algorithms, because if the ground-truth were indeed available, it could have been used to create a more effective supervised algorithm. In the unsupervised scenario (without ground-truth), it is often the case, that no realistic quantitative methods can be used in order to judge the effectiveness of the underlying algorithms in a rigorous way. Therefore, much of the research literature uses case studies in order to provide an intuitive and qualita-

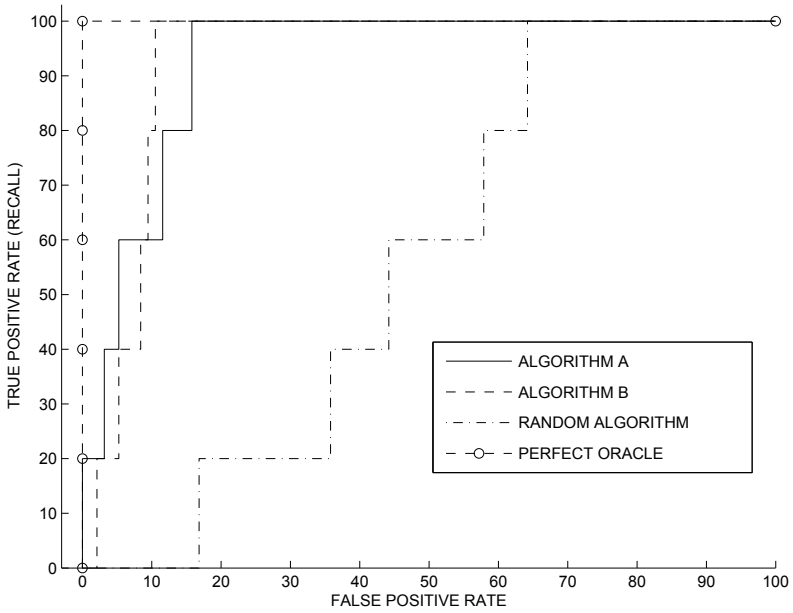


Figure 1.11. Receiver Operating Characteristic Curves

tive evaluation of the underlying outliers in unsupervised scenarios. In some cases, the data sets may be adapted from imbalanced classification problems, and the rare labels may be used as surrogates for the ground truth outliers.

Nevertheless, many scenarios do exist, in which ground-truth is available. In most supervised algorithms, ground-truth is available, a part of which can be used in order to perform the supervision, and the remaining can be used for evaluation. Even in unsupervised scenarios, the ground-truth often becomes available after a period of time, even though it may not have been available at the time of outlier analysis. Therefore, a natural question arises as to how the ground-truth can be used to evaluate effectiveness. Most outlier detection algorithms output an outlier score, and a threshold on this score is used in order to declare data points as outliers. If the threshold is picked too restrictively in order to minimize the number of declared outliers, then the algorithm will miss true outlier points (false negatives). On the other hand, if the algorithm declares too many data points as outliers, then it will lead to too many false positives. This tradeoff can be measured in terms of *precision* and *recall*, which is commonly used for measuring the effectiveness of set-based retrieval.

For any given threshold t on the outlier score, the declared outlier set is denoted by $S(t)$. As t changes, the size of $S(t)$ changes as well. G represent the true set (ground-truth set) of outliers in the data set. Then, for any given threshold t, the *precision* is defined as the percentage of *reported* outliers, which truly turn out to be outliers.

$$Precision(t) = 100 * \frac{|S(t) \cap G|}{|S(t)|}$$

The value of $Precision(t)$ is *not* necessarily monotonic in t, because both the numerator and denominator may change with t differently. The *recall* is correspondingly defined as the percentage of *ground-truth* outliers, which have been reported as outliers at threshold t.

$$Recall(t) = 100 * \frac{|S(t) \cap G|}{|G|}$$

By varying the parameter t, it is possible to plot a curve between the precision and the recall. This is referred to as the *Precision-Recall* curve. This curve is *not necessarily* monotonic. On the other hand, for more effective algorithms, high values of precision may often correspond to low values of recall and vice-versa. The precision-recall (PR) curve can also be generated by using thresholds on the *rank* of the data points, when sorted by outlier score. In the absence of ties in the outlier scores, a rank-based and score-based PR curve would be identical.

A *Receiver Operating Characteristics Curve (ROC)* is closely related to a Precision-Recall curve, but is sometimes visually more intuitive. In this case, the *True Positive Rate* is graphed against the *False Positive Rate*. The true positive rate $TPR(t)$ is defined in the same way as the recall. The false positive rate $FPR(t)$ is the percentage of the falsely reported positives out of the ground-truth negatives. Therefore, for a data set D with ground truth positives G, these definitions are as follows:

$$TPR(t) = Recall(t)$$

$$FPR(t) = 100 * \frac{|S(t) - G|}{|D - G|}$$

Note that the end points of the ROC curve are always at $(0, 0)$ and $(100, 100)$, and a random method is expected to exhibit performance along the diagonal line connecting these points. The *lift* obtained above this diagonal line provides an idea of the accuracy of the approach. The ROC curve is simply a different way to characterize the tradeoffs than the precision-recall curve, though the two can be derived from one another. The ROC curve has the advantage of being monotonic, and more easily

interpretable in terms of its lift characteristics. On the other hand, the tradeoffs are sometimes more clearly understood at a detailed level with the use of a PR curve.

In order to illustrate the insights gained from these different graphical representations, consider an example of a data set with 100 points, from which five points are outliers. Two algorithms A and B are applied to this data set, which rank all data points from 1 to 100, with lower rank representing greater propensity to be an outlier. Thus, the precision and recall values can be generated by determining the ranks of the 5 ground truth outlier points. In Table 1.1, some hypothetical ranks for the 5 ground truth outliers have been illustrated for the different algorithms. In addition, the ground truth ranks for a random algorithm have been indicated. The random algorithm which outputs a random score for outlier detection of a given data point. Similarly, the ranks for a "perfect oracle" algorithm which ranks the correct top 5 points as outlier have also been illustrated in the table. The corresponding PR curve for this hypothetical output of outlier scores are illustrated in Figure 1.10. Other than the oracle algorithm, all the tradeoff curves are non-monotonic. This is because the discovery of a new outlier at any particular relaxation in rank threshold results in a spike in the precision, which becomes less pronounced at higher values of the recall. The corresponding ROC curve is illustrated in Figure 1.11. Unlike the PR curve, this curve is clearly monotonic.

What do these curves really tell us? For cases in which one curve strictly dominates another, it is clear that the algorithm for the former curve is superior. For example, it is immediately evident that the oracle algorithm is superior to all algorithms, and the random algorithm is inferior to all the other algorithms. On the other hand, the algorithms A and B show domination at different parts of the ROC curve. In such cases, it is hard to say that one algorithm is strictly superior. From Table 1.1, it is clear that Algorithm A, ranks three of the correct ground truth outliers very highly, but but the remaining two outliers are ranked poorly. In the case of Algorithm B, the highest ranked outliers are not as well ranked as the case of Algorithm A, though all five outliers are determined much earlier in terms of rank threshold. Correspondingly, Algorithm A dominates on the earlier part of the PR curve, whereas Algorithm B dominates on the later part. Some practitioners use the area under the ROC curve as a proxy for the overall effectiveness of the algorithm, though such a measure should be used very carefully, because all parts of the ROC curve may not be equally important for different applications.

8. Conclusions and Summary

The problem of outlier detection finds applications in numerous domains, where it is desirable to determine interesting and unusual events in the activity which generates such data. The core of all outlier detection methods is the creation of a probabilistic, statistical or algorithmic model which characterizes the normal behavior of the data. The deviations from this model are used to determine the outliers. A good domain-specific knowledge of the underlying data is often crucial in order to design simple and accurate models which do not overfit the underlying data. The problem of outlier detection becomes especially challenging, when significant relationships exist among the different data points. This is the case for time-series and network data in which the patterns in the relationships among the data points (whether temporal or structural) play the key role in defining the outliers. Outlier analysis has tremendous scope for research, especially in the area of structural and temporal analysis.

9. Bibliographic Survey

A number of books and surveys have been written on the problem of outlier analysis. The classic books [58, 205, 387] in this area have mostly been written from the perspective of the statistics community. Most of these books were written before the wider adoption of database technology, and are therefore not written from a computational perspective. More recently, this problem has been studied quite extensively by the computer science community. These works consider practical aspects of outlier detection, corresponding to the cases, where the data may be very large, or may have very high dimensionality. Numerous surveys have also been written, which discuss the concept of outliers from different points of view, methodologies or data types [30, 62, 107, 108, 325, 326]. Among these, the survey by Chandola et al [107], is the most recent, and arguably the most comprehensive. It is an excellent review, which covers the work on outlier detection quite broadly from the perspective of multiple communities.

The issue of distinguishing between spurious abnormalities (or noise) and true outliers has also been discussed in [9], where the challenges of finding true anomalies in time series have been discussed. The Z-value test discussed in section 2 is used commonly in the statistical literature, and many variants for limited sample sizes such as the Grubb's test [188] and t-value test are also available. While this test makes the normal distribution assumption for large data sets, it has been used

fairly extensively as a good heuristic even for data distributions which do not satisfy the normal distribution assumption.

The basic models discussed in this chapter have also been researched extensively, and have been studied widely in the literature. Details of these methods (along with the corresponding bibliographic notes) will be provided in later chapters. Here only the most important works in each area are covered. The key statistical techniques on regression-based modeling are covered in [387]. The basic EM-algorithm for unsupervised modeling of data sets was first proposed in [135]. The non-parametric technique of principal component analysis (PCA) discussed in section 2 is described well in [244]. The core technique of PCA was extended to text (with some minor variations) as Latent Semantic Indexing [133]. A variety of distance-based methods for outlier detection are proposed in [261, 381, 441], and density-based methods for outlier detection were proposed in [78]. Methods for interpreting distance-based outliers were first proposed in [262]. A variety of information theoretic methods for outlier detection are discussed in [34, 45, 74, 96, 123, 211, 212, 297, 410].

The issues of poor behavior of high dimensional applications such as clustering and nearest neighbor search have been observed in several prior works in the literature [5, 7, 8, 22, 215]. The problem of high-dimensional outlier detection was first proposed in [4]. Subspace approaches for outlier detection were proposed in this paper, and a number of other recent methods have followed a similar line of work [256, 273, 337–339, 341, 498–501, 513].

Outliers have been studied extensively in the context of different data domains. While numeric data is the most commonly studied case, numerous methods have also been proposed for categorical and mixed data [30, 478]. Methods for unsupervised outlier detection in text corpora are proposed in [197]. The problem of detecting outliers with dependencies has also been studied extensively in the literature. Methods for detecting outliers and changes in time series and streams were proposed in [9, 15, 16, 26, 257–260]. Novelty detection [325] is an area which is closely related to outlier analysis, and it is often studied in the context of supervised models, where novel classes from a data stream are detected in real time [328, 329], with the use of learning methods. However, novelty detection is also studied often in the unsupervised scenario, particularly in the context of *first story detection* in topic detection and tracking in text streams [515]. Spatial outliers [3, 268, 317, 401–404] are closely related to the problem of finding outliers in temporal data, since such data also shows spatial continuity, just as temporal data shows temporal continuity. Some forms of spatial data also have a temporal component

to them, which requires the determination of spatiotemporal outliers [113, 114].

Outlier detection in discrete sequences is related to the problem of temporal outlier detection in continuous sequences. For discrete sequences, an excellent survey may be found in [108]. Methods for finding node outliers with unusual neighborhood behavior in graphs were proposed in [33], and techniques for finding relationship outliers, subgraph outliers and community outliers were proposed in [15, 180, 349, 378]. The primary ideas in all these methods is that outlier regions in a network are caused by unusual relationships in the form of edges, subgraphs, and communities. The temporal analysis of graph streams in the context of significant community evolution was studied in [17, 192, 429]. The problem of discovering significant structural change in temporal networks in the form of distance changes was studied in [193].

Recently, some *meta-algorithms for outlier detection* have been designed. The core-idea of this approach is that multiple methods for outlier detection will provide different results, and these results can be combined in order to provide more robust results. This approach lies at the core of *ensemble-based methods* [289, 310, 271]. In the case of sequential ensembles, most of the currently available techniques are adhoc, and apply to specific algorithms. These techniques are often not recognized as general-purpose meta-algorithms, which can be used in order to improve the effectiveness of *any* arbitrary outlier detection algorithm, though the interests in this area have increased recently. Independent ensemble algorithms are based on the idea that multiple ways of solving the same problem are likely to lead to more robust results. For example, if two different methods find the same data point as an outlier, this is a more robust indicate of outlierness, since it does not result from a particular overfitting of the specific algorithm. The work in [289] designs methods for using different subsets of features in outlier detection methods, and combining them in order to provide more effective results. The work in [337–339] shows how to combine the scores from different subspaces found by outlier detection algorithms in order to provide a unified and more robust result. The work in [271] also shows how outlier scores of different algorithms can be best interpreted and unified into more robust outputs.

The supervised version of the outlier detection problem has been studied extensively in the form of *rare class detection*. For the supervised case, readers are referred to a general book on classification [146], since this problem is essentially a cost-sensitive variation [102, 151] on the standard classification problem, in which the class distributions are very imbalanced. In particular, the readers are referred to [102, 151] for a

thorough discussion on the foundations of cost-sensitive learning from imbalanced data sets. A number of methods for classification from positive and unlabeled data are discussed in [152], and a good review of the previous work in this area may also be found from the references in this paper. The work in [360, 512, 513] first showed how human supervision could be used to significantly improve the effectiveness of outlier detection. Finally, the *semi-supervised* scenario of novelty detection has been discussed extensively in [325, 326, 445].

Evaluation methods for outlier analysis are essentially identical to the techniques used in information retrieval for understanding precision-recall tradeoffs, or in classification for ROC curve analysis. A detailed discussion of the proper construction of such curves may be found in [159]. While the ROC and PR curves are the traditional methods for outlier evaluation, it has recently been noted [337] that these methods may not necessarily provide all the insights needed for different kinds of analysis. Therefore, the work in [337] has proposed a coefficient, which was based on the Spearman correlation between the best possible ranking and the ranking determined by the algorithm. The work in [395] provides further ways of examining the ranks of outlier scores, which also provides insights into the effectiveness of outlier ensembles. Other visual methods of evaluating outliers include the LOCI plot [356] (discussed in Chapter 4), and the ELKI [2] software, which shows the contrasts in outlier scores in the form of histograms and bubble plots.

10. Exercises

1. Which of the following points from each of the following sets of points below is an outlier? Why?

- (1-dimensional) { 1, 3, 2, 1, 3, 2, 75, 1, 3, 2, 2, 1, 2, 3, 2, 1 }
- (1-dimensional) { 1, 2, 3, 4, 2, 19, 9, 21, 20, 22 }
- (2-dimensional) { (1, 9), (2, 9), (3, 9), (10, 10), (10, 3), (9, 1), (10, 2) }

2. Use MATLAB or any other mathematical software to create a histogram of the data distribution along each of the dimensions in the different cases of Exercise 1. Can you see the outliers visually? Which ones? In which case are the outliers not clear and why?

3. For the 2-dimensional case of Exercise 1, plot the data points on a 2-dimensional plane. Can you see the outliers visually? Which ones?

4. Apply the Z-value test to each of the cases in Exercise 1. For the 2-dimensional case, apply the Z-value test to the individual dimensions. Do you discover the correct outliers?

5. For the 2-dimensional case in Exercise 1, construct the function $f(x_1, x_2) = |x_1 - x_2|$. Apply the Z-value test to $f(x_1, x_2)$ over each of the data points. Do you obtain the correct outliers, as suggested by your visual analysis in Exercise 3? Why?

6. Determine the nearest neighbor of each data point for the cases in Exercise 1. Which data points have the largest value of the nearest neighbor distance? Are they the correct outliers?

7. Apply a k-means clustering algorithm to each of the cases in Exercise 1, while setting $k = 2$. Which data points lie furthest from the two means thus found? Are these the correct outliers?

8 Consider the following time-series:

- 1, 2, 3, 3, 2, 1, 73, 1, 2, 3, 5
- 1, 2, 3, 4, 3, 2, 1, 3, 73, 72, 74, 73, 74, 1, 2, 3, 4, 2
- 1, 2, 3, 5, 6, 19, 11, 15, 17, 2, 17, 19 , 17, 18

Which data points would you consider outliers? How does the temporal component influence your choice of outliers? Now examine the points at which the time series changes significantly? How do these points relate to the outliers?

9. Consider the undirected network $G = (N, A)$ of 8 nodes in N indexed from 1 through 8. Let the edge set A be { (1, 2), (1, 3), (1, 4), (1, 5), (1, 6), (1, 7), (1, 8) }. Draw the network on paper to visualize it. Is there any node, which you would consider an outlier? Why?

- Now delete the edge $(1, 7)$. Does this change the set of nodes you would consider outliers? Why?

10. Consider the undirected network $G = (N, A)$ of 8 nodes in N indexed from 1 through 8. Let the edge set A be { (1, 2), (1, 3), (1, 4), (2, 3), (2, 4), (5, 7), (4, 7), (5, 6), (6, 8), (5, 8), (6, 7) }. Draw the network on paper to visualize it. Is there any edge, which you would consider an outlier? Why?

11. Consider three algorithms A, B and C, which are run on a data set with 100 points and 5 outliers. The rank of the outliers by score for the three algorithms are as follows:

A: 1, 3, 5, 8, 11
B: 2, 5, 6, 7, 9
C: 2, 4, 6, 10, 13
Draw the PR curves for each of the algorithms. Would you consider any of the algorithms strictly superior to any of the others? Why?

References

[1] N. Abe, B. Zadrozny, and J. Langford. Outlier Detection by Active Learning, *ACM KDD Conference*, 2006.

[2] E. Achtert, A. Hettab, H.-P. Kriegel, E. Schubert, and A. Zimek. Spatial Outlier Detection: Data, Algorithms, Visualizations. *SSTD Conference*, 2011.

[3] N. R. Adam, V. P. Janeja, and V. Atluri. Neighborhood based detection of anomalies in high dimensional spatio-temporal sensor datasets. *ACM SAC Conference*, 2004.

[4] C. C. Aggarwal, and P. S. Yu. Outlier Detection in High Dimensional Data, *ACM SIGMOD Conference*, 2001.

[5] C. C. Aggarwal, C. Procopiuc, J. Wolf, P. Yu, and J. Park. Fast Algorithms for Projected Clustering, *ACM SIGMOD Conference*, 1999.

[6] C. Aggarwal, J. Han, J. Wang, and P. Yu. A Framework for Projected Clustering of High Dimensional Data Streams. In *VLDB Conference*, 2004.

[7] C. C. Aggarwal, and P. S. Yu. Finding Generalized Projected Clusters in High Dimensional Spaces, *ACM SIGMOD Conference*, 2000.

[8] C. C. Aggarwal. Re-designing Distance Functions and Distance-based Applications for High Dimensional Data, *ACM SIGMOD Record*, 2001.

[9] C. C. Aggarwal. On Abnormality Detection in Spuriously Populated Data Streams, *SIAM Conference on Data Mining*, 2005.

[10] C. C. Aggarwal. Data Streams: Models and Algorithms, *Springer*, 2007.

[11] C. C. Aggarwal. Social Network Data Analytics, *Springer*, 2011.

[12] C. C. Aggarwal. On Effective Classification of Strings with Wavelets, *ACM KDD Conference*, 2002.

[13] C. C. Aggarwal, N. Ta, J. Wang, J. Feng, and M. J. Zaki. Xproj: A Framework for Projected Structural Clustering of XML Documents. *ACM KDD Conference*, 2007.

[14] C. C. Aggarwal, and P. S. Yu. On String Classification in Data Streams, *ACM KDD Conference*, 2007.

[15] C. C. Aggarwal, Y. Zhao, and P. S. Yu. Outlier Detection in Graph Streams, *ICDE Conference*, 2011.

[16] C. C. Aggarwal. A Framework for Diagnosing Changes in Evolving Data Streams, *ACM SIGMOD Conference*, 2003.

[17] C. C. Aggarwal, and P. S. Yu. Online Analysis of Community Evolution in Data Streams, *SDM Conference*, 2005.

[18] C. C. Aggarwal. On the Effects of Dimensionality Reduction on High Dimensional Similarity Search, *ACM PODS Conference*, 2001.

[19] C. C. Aggarwal. Managing and Mining Sensor Data, *Springer*, 2013.

[20] C. C. Aggarwal, and C. K. Reddy. Data Clustering: Algorithms and Applications, *CRC Press*, 2013.

[21] C. Aggarwal, and C. Zhai. Managing and Mining Text Data, *Springer*, 2012.

[22] C. C. Aggarwal, A. Hinneburg, and D. Keim. On the Surprising Behavior of Distance Metrics in High Dimensional Space, *ICDT Conference*, 2001.

[23] C. C. Aggarwal, and P. S. Yu. Outlier Detection with Uncertain Data, *SDM Conference*, 2008.

[24] C. C. Aggarwal, Y. Xie, and P. S. Yu. On Dynamic Data-Driven Selection of Sensor Streams, *ACM KDD Conference*, 2011.

[25] C. C. Aggarwal, J. Han. J. Wang, and P. Yu. A Framework for Clustering Evolving Data Streams, *VLDB Conference*, 2003.

[26] C. C. Aggarwal, and P. Yu. On Clustering Massive Text and Categorical Data Streams, *Knowledge and Information Systems*, 24(2), pp. 171–196, 2010.

[27] C. C. Aggarwal, and K. Subbian. Event Detection in Social Streams, *SDM Conference*, 2012.

[28] R. Agrawal, T. Imielinski, and A. Swami. Mining Association Rules Between Sets of Items in Large Databases. *ACM SIGMOD Conference*, 1993.

[29] R. Agrawal, and R. Srikant. Fast algorithms for finding Association Rules in Large Databases, *VLDB Conference*, 1994.

[30] M. Agyemang, K. Barker, and R. Alhajj. A Comprehensive Survey of Numeric and Symbolic Outlier Mining Techniques, *Intelligent Data Analysis*, 10(6). pp. 521–538, 2006.

[31] A. Ahmad and L. Dey. A Method to Compute Distance between two Categorical Values of Same Attribute in Unsupervised Learning for Categorical Data Set. *Pattern Recognition Letters*, 28(1), pp. 110–118, 2007.

[32] R. Ahuja, J. Orlin, and T. Magnanti. Network Flows: Theory, Algorithms and Applications, *Prentice Hall*, 1993.

[33] L. Akoglu, M. McGlohon, and C. Faloutsos. Oddball: Spotting Anomalies in Weighted Graphs, *PAKDD Conference*, 2010.

[34] L. Akoglu, H. Tong, J. Vreeken, and C. Faloutsos. Fast and Reliable Anomaly Detection in Categorical Data, *CIKM Conference*, 2012.

[35] E. Aleskerov, B. Freisleben, and B. Rao. CARDWATCH: A Neural Network based Database Mining System for Credit Card Fraud Detection. *IEEE Computational Intelligence for Financial Engineering*, pp. 220–226, 1997.

[36] T. Al-Khateeb, M. Masud, L. Khan, C. Aggarwal, J. Han, and B. Thuraisingham. Recurring and Novel Class Detection using Class-based Ensemble, *ICDM Conference*, 2012.

[37] J. Allan, R. Papka, and V. Lavrenko. On-line new event detection and tracking. *ACM SIGIR Conference*, 1998.

[38] J. Allan, V. Lavrenko, and H. Jin. First story detection in TDT is hard. *ACM CIKM Conference*, 2000.

[39] J. Allan, J. Carbonell, G. Doddington, J. Yamron, and Y. Yang. Topic Detecting and Tracking Pilot Study Final Report, *CMU Technical Report, Paper 341*, 1998.

[40] F. Alonso, J. Caraca-Valente, A. Gonzalez, and C. Montes. Combining Expert Knowledge and Data Mining in a Medical Diagnosis Domain. *Expert Systems with Applications*, 23(4), pp. 367–375, 2002.

[41] Y. Altun, I. Tsochantaridis, and T. Hofmann. Hidden Narkov Support Vector Machines. *ICML Conference*, 2003.

[42] M. R. Anderberg. Cluster Analysis for Applications, *Academic Press*, New York, 1973.

[43] D. Anderson, T. Lunt, H. Javitz, A. Tamaru, and A. Valdes. Detecting Unusual Program Nehavior using the Statistical Components of NIDES, *Techical Report, SRI–CSL–95–06, Computer Science Laboratory*, SRI International, 1995.

[44] D. Anderson, T. Frivold, A. Tamaru, and A. Valdes. Next-generation Intrusion Detection Expert System (nides), Software Users Manual, Beta-update Release. *Technical Report SRI–CSL–95–07, Computer Science Laboratory*, SRI International, 1994.

[45] S. Ando. Clustering Needles in a Haystack: An Information Theoretic Analysis of Minority and Outlier Detection. *ICDM Conference*, 2007.

[46] F. Angiulli, and C. Pizzuti. Fast Outlier Detection in High Dimensional Spaces. *European Conferece on Principles of Knowledge Discovery and Data Mining*, 2002.

[47] F. Angiulli, F. Fassetti, amd L. Palopoli. Finding Outlying Properties of Exceptional Objects, *ACM Transactions on Database Systems*, 34(1), 2009.

[48] F. Angiulli and F. Fassetti. Detecting Distance-based Outliers in Streams of Data, *ACM CIKM Conference*, 2007.

[49] A. Arning, R. Agrawal, and P. Raghavan. A Linear Method for Deviation Detection in Large Databases. *ACM KDD Conference*, 1996.

[50] I. Assent, P. Kranen, C. Beldauf, and T. Seidl. AnyOut: Anytime Outlier Detection in Streaming Data, *DASFAA Conference*, 2012.

[51] I. Assent, R. Krieger, E. Muller, and T. Seidl. Subspace Outlier Mining in Large Multimedia Databases, *Parallel Universes and Local Patterns*, 2007.

[52] A. Auer Jr. Correlation of Land Use and Cover with Meteorological Anomalies. *Journal of Applied Meteorology*, 17(5) pp. 636–643, 1978.

[53] L. Backstrom, D. Huttenlocher, J. Kleinberg, and X. Lan. Group formation in large social networks: membership, growth, and evolution. *KDD Conference*, 2006.

[54] D. Ballard. Generalizing the Hough Transform to Detect Arbitrary Shapes, *Pattern Recognition*, 11(2), pp. 111–122, 1981.

[55] D. Barbara, Y. Li, J. Couto, J.-L. Lin, and S. Jajodia. Bootstrapping a Data Mining Intrusion Detection System. *Symposium on Applied Computing*, 2003.

[56] D. Barbara, J. Couto, S. Jajodia, and N. Wu. ADAM: A Testbed for Exploring the Use of Data Mining in Intrusion Detection. *ACM SIGMOD Record*, 30(4), pp. 15–24, 2001.

[57] D. Barbara, J. Couto, S. Jajodia, and N. Wu. Detecting Movel Network Intrusions using Bayes Estimators. *SIAM Conference on Data Mining*, 2001.

[58] V. Barnett and T. Lewis. Outliers in Statistical Data, *Wiley*, 1994.

[59] R. Baragona and F. Battaglia. Outlier Detection in Multivariate Time Series by Independent Component Analysis. *Neural Computation*, 19(1), pp. 1962–1984, 2007.

[60] S. Bay, and M. Schwabacher. Mining distance-based outliers in near linear time with randomization and a simple pruning rule. *ACM KDD Conference*, 2003.

[61] S. Bay, K. Saito, N. Ueda, and P. Langley. A Framework for Discovering Anomalous Regimes in Multivariate Time-series Data with Local Models. *Technical report, Center for the Study of Language and Information*, Stanford University, 2004.

[62] R. Beckman, and R. Cook. Outliers, *Technometrics*, 25(2), pp. 119–149, 1983.

[63] N. Beckmann, H.-P. Kriegel, R. Schneider, and B. Seeger. The R*-tree: An Efficient and Robust Access method for Points and Rectangles. *ACM SIGMOD Conference*, 1990.

[64] M. Berlingerio, F. Bonchi, B. Bringmann, and A. Gionis. Mining Graph Evolution Rules. *ECML/PKDD Conference*, 2009.

[65] K. Beyer, J. Goldstein, R. Ramakrishnan, and U. Shaft. When is "nearest neighbor" meaningful? *International Conference on Database Theory*, 1999.

[66] K. Bhaduri, B. Matthews, and C. Giannella. Algorithms for Speeding up Distance-based Outlier Detection. *ACM KDD Conference*, 2011.

[67] E. Blanzieri and A. Bryl. A Survey of Learning-based Techniques of Email Spam Filtering. *Artificial Intelligence Review*, 29(1), pp. 63–92, 2008.

[68] M. Bosc, F. Heitz, J.-P. Armspach, I. Namer, D. Gounot, and L. Rumbach. Automatic Change Detection in Multimodal Serial MRI: application to multiple sclerosis lesion evolution, *NeuroImage*, 20(2), 2003, Pages 643–656

[69] Y. Bilberman. A Context Similarity Measure. *ECML Conference*, 1994.

[70] P. Billingsley. Probability and Measure, Second Edition, *Wiley*, 1986.

[71] D. Birant, and A. Kut. Detecting Spatio-temporal Outliers in Large Databases, *Journal of Computing and Information Technology*, 14(4), pp. 291–297, 2006.

[72] D. Blei, and J. Lafferty. Dynamic topic models. *ICML Conference*, 2006.

[73] D. Blei, A. Ng, and M. Jordan. Latent Dirichlet allocation, *Journal of Machine Learning Research*, 3: pp. 993–1022, 2003.

[74] C. Bohm, K. Haegler, N. Muller, and C. Plant. Coco: Coding Cost for Parameter Free Outlier Detection, *ACM KDD Conference*, 2009.

[75] S. Boriah, V. Chandola, and V. Kumar. Similarity Measures for Categorical Data: A Comparative Evaluation, *SIAM Conference on Data Mining*, 2008.

[76] J. Branch, B. Szymanski, C. Giannella, R. Wolff, and H. Kargupta. In-Network Outlier Detection in Wireless Sensor Networks. *ICDCS Conference*, 2006.

[77] T. Brants, F. Chen, and A. Farahat. A System for New Event Detection. *ACM SIGIR Conference*, 2003.

[78] M. Breunig, H.-P. Kriegel, R. Ng, and J. Sander. LOF: Identifying Density-based Local Outliers, *ACM SIGMOD Conference*, 2000.

[79] M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander. OPTICS-OF: identifying local outliers. *PKDD Conference*, 1999.

[80] L. Brieman. Bagging Predictors, *Machine Learning*, 24: pp. 123–140, 1996.

[81] M. R. Brito, E. L. Chavez, A. J. Quiroz, and J. E. Yukich. Connectivity of the Mutual k-Nearest Neighbor Graph in Clustering and Outlier Detection. *Statistics and Probability Letters*, 35(1), pp. 33–42, 1997.

[82] R. G. Brown, and P. Hwang. Introduction to Random Signals and Applied Kalman Filtering, *John Wiley and Sons*, 1997.

[83] Y. Bu, L. Chen, A. W.-C. Fu, and D. Liu. Efficient Anomaly Monitoring over Moving Object Trajectory Streams. *ACM KDD Conference*, 2009.

[84] S. Budalakoti, A. Srivastava, R. Akella, and E. Turkov. Anomaly Detection in Large Sets of High-dimensional Symbol Sequences, *NASA Ames Research Center*, Technical Report NASA TM-2006-214553, 2006.

[85] S. Budalakoti, A. Srivastava, and M. Otey. Anomaly Detection and Diagnosis Algorithms for Discrete Symbol Sequences with Applications to Airline Safety, *IEEE International Conference on Systems, Man, and Cybernetics*, 37(6), 2007.

[86] S Burdakis, A Deligiannakis. Detecting Outliers in Sensor Networks using the Geometric Approach, *ICDE Conference*, 2012.

[87] T. Burnaby. On a Method for Character Weighting a Similarity Coefficient, Employing the Concept of Information. *Mathematical Geology*, 2(1), 25–38, 1970.

[88] S. Byers, and A. Raftery. Nearest-Neighbor Clutter Removal for Estimating Features in Spatial Point Processes, *JASIS*, 93, pp. 577–584, June 1998.

[89] I. Cadez, D. Heckerman, C. Meek, P. Smyth, and S. White. Visualization of Navigation Patterns on a Web Site using Model-based Clustering, *ACM SIGMOD Conference*, 2000.

[90] P. H. Calamai. Projected gradient methods for linearly constrained problems. *Mathematical Programming*, 39: pp, 93–116, 1987.

[91] C. Campbell, and K. P. Bennett. A Linear-Programming Approach to Novel Class Detection, *NIPS Conference*, 2000.

[92] M. J. Canty. Image Analysis, Classification and Change Detection in Remote Sensing: with Algorithms for ENVI/IDL, *CRC Press*, 2006.

[93] C. Caroni. Outlier Detection by Robust Principal Component Analysis. *Communications in Statistics – Simulation and Computation*, 29: pp. 129–151, 2000.

[94] L. E. Carr and R. L. Elsberry. Monsoonal interactions leading to sudden tropical cyclone track changes, *Monthly Weather Review*, 123(2), pp. 265–290, Feb. 1995.

[95] K. Chakrabarti, S. Mehrotra. Local Dimensionality Reduction: A New Approach to Indexing High Dimensional Spaces. *VLDB Conference Proceedings*, 2000.

[96] S. Chakrabarti, S. Sarawagi, and B. Dom. Mining Surprising Patterns using Temporal Description Length. *VLDB Conference*, 1998.

[97] V. Chandola, V. Mithal, and V. Kumar. A Comparative Evaluation of Anomaly Detection Techniques for Sequence Data, *International Conference on Data Mining*, 2008.

[98] A. Chaudhary, A. S. Szalay, and A. W. Moore. Very Fast Outlier Detection in Large Multidimensional Data Sets. *DMKD Workshop*, 2002.

[99] D. Chakrabarti, and C. Faloutsos. Evolutionary Clustering, *ACM KDD Conference*, 2006.

[100] D. Chakrabarti. AutoPart: Parameter-Free Graph Partitioning and Outlier Detection. *PKDD Conference*, 2004.

[101] C.-H. Chan and G. Pang. Fabric Defect Detection by Fourier Analysis, *IEEE Transactions on Industry Applications*, 36(5), 2000.

[102] N. V. Chawla, N. Japkowicz, and A. Kotcz. Editorial: Special Issue on Learning from Imbalanced Data Sets, *ACM SIGKDD Explorations Newsletter*, 6(1), 1–6, 2004.

[103] N. V. Chawla, K. W. Bower, L. O. Hall, and W. P. Kegelmeyer. SMOTE: synthetic minority over-sampling technique, *Journal of Artificial Intelligence Research (JAIR)*, 16, pp. 321–356, 2002.

[104] N. Chawla, A. Lazarevic, L. Hall, and K. Bowyer. SMOTEBoost: Improving prediction of the minority class in boosting, *PKDD*, pp. 107–119, 2003.

[105] N. V. Chawla, D. A. Cieslak, L. O. Hall, and A. Joshi. Automatically Cuntering Imbalance and its Empirical Relationship to Cost. *Data Mining and Knowledge Discovery*, 17(2), pp. 225–252, 2008.

[106] P. K. Chan and S. J. Stolfo. Toward Scalable Learning with Non-uniform Class and Cost Distributions: A Case Study in Credit Card Fraud Detection. *KDD Conference*, pp. 164–168, 1998.

[107] V. Chandola, A. Banerjee, and V. Kumar. Anomaly Detection: A Survey. *ACM Computing Surveys*, 41(3), 2009.

[108] V. Chandola, A. Banerjee, and V. Kumar. Anomaly Detection for Discrete Sequences: A Survey. *IEEE Transactions on Knowledge and Data Engineering*, 24(5): pp. 823–839, 2012.

[109] I. Chang, G. C. Tiao, and C. Chen. Estimation of Time Series Parameters in the Presence of Outliers. *Technometrics*, 30(2), pp. 193–204, 1988.

[110] C. Chen and L.-M. Liu. Joint Estimation of Model Parameters and Outlier Effects in Time Series. *Journal of the American Statistical Association*, 88(421), pp. 284–297, March 1993.

[111] Y. Chen, and L. Tu. Density-based Clustering for Real Time Stream Sata, *ACM KDD Conference*, 2007.

[112] D. Chen, C.-T. Lu, Y. Chen, and D. Kou. On Detecting Spatial Outliers, *Geoinformatica*, 12: pp. 455–475, 2008.

[113] T. Cheng and Z. Li. A Hybrid Approach to Detect Spatialtemporal Outliers. *International Conference on Geoinformatics*, 2004.

[114] T. Cheng and Z. Li. A Multiscale Approach for Spatio-temporal Outlier Detection, *Transactions in GIS*, 10(2), pp. 253–263, March 2006.

[115] H. Cheng, P.-N. Tan, C. Potter, and S. Klooster. Detection and Characterization of Anomalies in Multivariate Time Series, *SIAM Conference on Data Mining*, 2009.

[116] Y. Chi, X. Song, D. Zhou, K. Hino, and B. Tseng. Evolutionary Spectral Clustering by Incorporating Temporal Smoothness. *ACM KDD Conference*, 2007.

[117] A. Chiu, and A. Fu. Enhancements on Local Outlier Detection. *Database Engineering and Applications Symposium*, 2003.

[118] M. Chow, R. Sharpe, and J. Hung. On the Application and Design of Artificial Neural Networks for Motor Fault Detection. *IEEE Transactions on Industrial Electronics*, 40(2), 1993.

[119] C. Chow, sand D. Yeung. Parzen-Window Network Intrusion Detectors. *International Conference on Pattern Recognition*, 4, 2002.

[120] W. Cohen. Fast Effective Rule Induction. *ICML Conference*, 1995.

[121] D. Cohn, R. Atlas, and N. Ladner. Improving Generalization with Active Learning, *Machine Learning*, 15, pp. 201–221.

[122] R. Cooley, B. Mobashar, and J. Srivastava. Data Preparation for Mining World Wide Web Browsing Patterns, *Knowledge and Information Systems*, 1, pp, 5–32, 1999.

[123] D. Cook, and L. Holder. Graph-Based Data Mining. *IEEE Intelligent Systems*, 15(2), pp. 32–41, 2000.

[124] G. Cormack. Email Spam Filtering: A Systematic Review, *Foundations and Trends in Information Retrieval*, 1(4), pp. 335–455, 2007.

[125] C. Darwin. The Origin of the Species by Natural Selection, 1859. Manuscript now publicly hosted at: `http://www.literature.org/authors/darwin-charles/the-origin-of-species/`

[126] G. Das and H. Mannila. Context-based Similarity Measures for Categorical Databases. *PKDD Conference*, 2000.

[127] K. Das, J. Schneider, and D. Neill. Anomaly Pattern Detection in Categorical Data Sets, *ACM KDD Conference*, 2008.

[128] S. Das, B. Matthews, A. Srivastava, and N. Oza. Multiple kernel learning for heterogeneous anomaly detection: algorithm and aviation safety case study. *ACM KDD Conference*, 2010.

[129] D. Dasgupta and S. Forrest. Novelty Detection in Time Series using Ideas from Immunology, *International Conference on Intelligent Systems*, 1996.

[130] D. Dasgupta, and F. Nino. A comparison of negative and positive selection algorithms in novel pattern detection. *IEEE Conference on Systems, Man, and Cybernetics*, 1, pp. 125–130, 2000.

[131] T. Dasu, S. Krishnan, S. Venkatasubramanian, and K. Yi. An Information-Theoretic Approach to Detecting Change in Multi-dimensional Data Streams, *Symposium on the Interface of Computer Science, Statistics, and Applications*, 2006.

[132] N. Delannay, C. Archambeau, and M. Verleysen. Improving the Robustness to Outliers of Mixtures of Probabilistic PCAs. *PAKDD Conference*, 2008.

[133] S. T. Deerwester, S. T. Dumais, G. Furnas, and R. Harshman. Indexing by Latent Semantic Analysis, *JASIS*, 1990.

[134] K. A. De Jong. Analysis of the behaviour of a class of Genetic Adaptive Systems. *Ph.D. Dissertation, University of Michigan*, Ann Arbor, MI, 1975.

[135] A. P. Dempster, N. M. Laird and D. B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm, *Journal of the Royal Statistical Society*, B, vol. 39(1), pp. 1–38, 1977.

[136] D. Denning. An Intrusion Detection Model. *IEEE Transactions of Software Engineering*, 13(2), pp. 222–232.

[137] R. Derrig. Insurance Fraud. *Journal of Risk and Insurance*, 69(3), pp. 271–287, 2002.

[138] M. Desforges, P. Jacob, and J. Cooper. Applications of Probability Density Estimation to the Detection of Abnormal Conditions in Engineering. *Proceedings of Institute of Mechanical Engineers*, Vol. 212, pp. 687–703, 1998.

[139] M. Deshpande and G. Karypis. Evaluation of Techniques for Classifying Biological Sequences. *PAKDD Conference*, 2002.

[140] M. Deshpande and G. Karypis. Selective Markov Models for Predicting Web Page Accesses, *ACM Transactions on Internet Technology*, 4(2), pp. 163–184, 2004.

[141] H. Ding, G. Trajcevski, P. Scheuermann, X. Wang, and E. Keogh. Querying and Mining of Time Series Data: Experimental Comparison of Representations and Distance Neasures. *PVLDB*, 1(2), pp. 1542–1552, 2008.

[142] S. Donoho. Early Detection of Insider Trading in Option Markets. *ACM KDD Conference*, 2004.

[143] C. Drummond and R. Holte. C4.5, Class Imbalance, and Cost Sensitivity: Why Undersampling beats Oversampling. *ICML Workshop on Learning from Imbalanced Data Sets*, 2003.

[144] C. Drummond and R. Holte. Explicitly Representing Expected Cost: An Alternative to ROC representation. *ACM KDD Conference*, pp. 198–207, 2001.

[145] P. Domingos. MetaCost: A General Framework for Making Classifiers Cost-Sensitive, *ACM KDD Conference*, 1999.

[146] R. Duda, P. Hart, and D. Stork, Pattern Classification, *Wiley*, 2001.

[147] H. Dutta, C. Giannella, K. Borne, and H. Kargupta. Distributed top-k Outlier Detection in Astronomy Catalogs using the Demac System. *SDM Conference*, 2007.

[148] W. Eberle and L. B. Holder. Mining for Structural Anomalies in Graph-based Data. *DMIN*, 2007.

[149] F. Y. Edgeworth. On Discordant Observations. *Philosophical Magazine*, 23(5), pp. 364–375, 1887.

[150] M. Elahi, K. Li, W. Nisar, X. Lv, and H. Wang. Efficient Clustering-Based Outlier Detection Algorithm for Dynamic Data Stream. *FSKD Conference*, 2008.

[151] C. Elkan. The Foundations of Cost-Sensitive Learning, *IJCAI*, 2001.

[152] C. Elkan, and K. Noto. Learning Classifiers from only Positive and Unlabeled Data, *ACM KDD Conference*, 2008.

[153] D. Endler. Intrusion detection: Applying Machine Learning to Solaris Audit Data, *Annual Computer Security Applications Conference*, 1998.

[154] E. Eskin. Anomaly Detection over Noisy Data using Learned Probability Distributions, *ICML Conference*, 2000.

[155] E. Eskin, A. Arnold, M. Prerau, L. Portnoy, and S. Stolfo. A Geometric Framework for Unsupervised Anomaly Detection, In *Applications of Data Mining in Computer Security*. Kluwer, 2002.

[156] E. Eskin, W. Lee, and S. Stolfo, Modeling System Call for Intrusion Detection using Dynamic Window Sizes, *DISCEX*, 2001.

[157] H. Fan, O. Zaiane, A. Foss. and J. Wu. A Nonparametric Outlier Detection for Efficiently Discovering top-n Outliers from Engineering Data. *PAKDD Conference*, 2006.

[158] W. Fan, S. Stolfo, J. Zhang, and P. Chan. AdaCost: Misclassification Cost Sensitive Boosting, *ICML Conference*, 1999.

[159] T. Fawcett. ROC Graphs: Notes and Practical Considerations for Researchers, *Technical Report HPL-2003-4*, Palo Alto, CA: HP Laboratories, 2003.

[160] T. Fawcett and F. Provost. Activity Monitoring: Noticing Interesting Changes in Behavior. *ACM KDD Conference*, 1999.

[161] D. Fetterly, M. Manasse, and M. Najork. Spam, Damn Spam, and Statistics: using Statistical Analysis to Locate Spam Web Pages, *WebDB*, 2004.

[162] A. Lung-Yut-Fong, C. Levy-Leduc, and O. Cappe. Distributed Detection/localization of Change-points in High-dimensional Network Traffic Data. *Corr*, abs/0909.5524, 2009.

[163] S. Forrest, C. Warrender, and B. Pearlmutter. Detecting Intrusions using System Calls: Alternate Data Models, *IEEE ISRSP*, 1999.

[164] S. Forrest, S. Hofmeyr, A. Somayaji, and T. A. Longstaff. A Sense of Self for Unix Processes, *ISRSP*, 1996.

[165] S. Forrest, P. D'Haeseleer, and P. Helman. An Immunological Approach to Change Detection: Algorithms, Analysis and Implications. *IEEE Symposium on Security and Privacy*, 1996.

[166] S. Forrest, F. Esponda, and P. Helman. A Formal Framework for Positive and Negative Detection Schemes. *IEEE Transactions on Systems, Man and Cybernetics, Part B*, pp. 357–373, 2004.

[167] S. Forrest, A. Perelson, L. Allen, and R. Cherukuri. Self-Nonself Discrimination in a Computer. *IEEE Symposium on Security and Privacy*, 1994.

[168] A. Fox. Outliers in Time Series. *Journal of the Royal Statistical Society. Series B (Methodological)*, 34(3), pp. 350–363, 1972.

[169] A. Frank, and A. Asuncion. UCI Machine Learning Repository, Irvine, CA: University of California, School of Information and Computer Science, 2010. `http://archive.ics.uci.edu/ml`

[170] C. Franke and M. Gertz. ORDEN: Outlier Region Detection and Exploration in Sensor Networks. *ACM SIGMOD Conference*, 2009.

[171] A. Fu, O. Leung, E. Keogh, and J. Lin. Finding Time Series Discords based on Haar Transform. *Advanced Data Mining and Applications*, 2006.

[172] R. Fujumaki, T. Yairi, and K. Machida. An Approach to Spacecraft Anomaly Detection Problem using Kernel Feature Space. *ACM KDD Conference*, 2005.

[173] R. Fujamaki. Anomaly Detection Support Vector Machine and Its Application to Fault Diagnosis, *ICDM Conference*, 2008.

[174] P. Galeano, D. Pea, and R. S. Tsay. Outlier detection in Multivariate Time Series via Projection Pursuit.*Statistics and Econometrics Working Papers WS044221*, Universidad Carlos III, 2004.

[175] P. Gambaryan. A Mathematical Model of Taxonomy. *Izvest. Akad. Nauk Armen*, SSR, 17(12), pp. 47–53, 1964.

[176] V. Ganti, J. Gehrke, and R. Ramakrishnan. CACTUS: Clustering Categorical Data using Summaries. *ACM KDD Conference*, 1999.

[177] B. Gao, H.-Y. Ma, and Y.-H. Yang, HMMs (Hidden Markov Models) based on Anomaly Intrusion Detection Method, *International Conference on Machine Learning and Cybernetics*, 2002.

[178] H. Gao, X. Wang, J. Tang and H. Liu. Network Denoising in Social Media, *Technical Report, Arizona State University*, 2011.

[179] J. Gao and P.-N. Tan. Converting Outlier Scores from Outlier Detection Algorithms into Probability Estimates, *ICDM Conference*, 2006.

[180] J. Gao, F. Liang, W. Fan, C. Wang, Y. Sun, and J. Han. On Community Outliers and their Efficient Detection in Information Networks. *ACM KDD Conference*, pp. 813–822, 2010.

[181] Y. Ge, H. Xiong, Z.-H. Zhou, H. Ozdemir, J. Yu, and K. Lee. Top-Eye: Top-k Evolving Trajectory Outlier Detection. *CIKM Conference*, 2010.

[182] A. Ghosh, J. Wanken, and F. Charron. Detecting Anomalous and Unknown Intrusions against Programs, *Annual Computer Security Applications Conference*, 1998.

[183] A. Ghosh, A. Schwartzbard, and M. Schatz. Learning Program Behavior Profiles for Intrusion Detection, *USENIX Workshop on Intrusion Detection and Network Monitoring*, pp. 51–62, 1999.

[184] S. Ghosh and D. Reilly. Credit Card Fraud Detection with a Neural Network, *International Conference on System Sciences: Information Systems: Decision Support and Knowledge-Based Systems*, 3, pp. 621–630, 1994.

[185] A. Ghoting, S. Parthasarathy, and M. Otey. Fast Mining of Distance-based Outliers in High Dimensional Spaces. *SIAM Conference on Data Mining*, 2006.

[186] D. Gibson, J. Kleinberg, and P. Raghavan. Clustering categorical data: an approach based on dynamical systems. *The VLDB Journal*, 8(3), pp. 222–236, 2000.

[187] D. W. Goodall. A new similarity index based on probability. *Biometrics*, 22(4), pp. 882–907, 1966.

[188] F. Grubbs. Procedures for Detecting Outlying Observations in Samples, *Technometrics*, 11(1), pp. 1–21, 1969.

[189] S. Guha, R. Rastogi, and K. Shim. ROCK: A robust clustering algorithm for categorical attributes. *Information Systems*, 25(5), pp. 345–366, 2000.

[190] D. Gunopulos and G. Das. Time-series Similarity Measures, and Time Series Indexing, *ACM SIGMOD Conference*, 2001.

[191] S. Gunter, N. N. Schraudolph, and S. V. N. Vishwanathan. Fast Iterative Kernel Principal Component Analysis. *Journal of Machine Learning Research*, 8, pp 1893–1918, 2007.

[192] M. Gupta, C. Aggarwal, J. Han, and Y. Sun. Evolutionary Clustering and Analysis of Bibliographic Networks, *ASONAM Conference*, 2011.

[193] M. Gupta, C. Aggarwal, and J. Han. Finding Top-k Shortest Path Distance Changes in an Evolutionary Network, *SSDBM Conference*, 2011.

[194] M. Gupta, J. Gao, Y. Sun, and J. Han. Community Trend Outlier Detection Using Soft Temporal Pattern Mining. *ECML/PKDD Conference*, 2012.

[195] M. Gupta, J. Gao, Y. Sun, and J. Han. Integrating Community Matching and Outlier Detection for Mining Evolutionary Community Outliers. *KDD Conference*, 2012.

[196] D. Gusfield. Algorithms for Strings, Trees and Sequences, *Cambridge University Press*, 1997.

[197] D. Guthrie, L. Guthrie, and Y. Wilks. An Unsupervised Approach for the Detection of Outliers in Corpora, *LREC*, 2008.

[198] S. Guttormsson, R. Marks, M. El-Sharkawi, and I. Kerszenbaum. Elliptical Novelty Grouping for Online Short-turn Detection of Excited Running Rotors. *IEEE Transactions on Energy Conversion*, 14(1), pp. 16–22, 1999.

[199] R. Gwadera, M. Atallah, and W. Szpankowski. Markov Models for Identification of Significant Episodes, *SDM Conference*, 2005.

[200] R. Gwadera, M. Atallah, and W. Szpankowskii. Detection of Significant Sets of Episodes in Event Sequences, *IEEE ICDM Conference*, 2004.

[201] R. Gwadera, M. Atallah, and W. Szpankowski. Reliable Detection of Episodes in Event Sequences, *Knowledge and Information Systems*, 7(4), pp. 415–437, 2005.

[202] F. Hampel. A General Qualitative Definition of Robustness, *Annals of Mathematics and Statistics*, 43, pp. 1887–1896, 1971.

[203] J. Haslett, R. Brandley, P. Craig, A. Unwin, and G. Wills. Dynamic Graphics for Exploring Spatial Data With Application to Locating Global and Local Anomalies. *The American Statistician*, 45: pp. 234–242, 1991.

[204] V. Hautamaki, I. Karkkainen, and P. Franti. Outlier Detection using k-nearest neighbor graph. *International Conference on Pattern Recognition*, 2004.

[205] D. Hawkins. Identification of Outliers, *Chapman and Hall*, 1980.

[206] G. G. Hazel. Multivariate Gaussian MRF for Multispectral Scene Segmentation and Anomaly Detection, *GeoRS*, 38(3), pp. 1199–1211, 2000.

[207] J. He, and J. Carbonell. Nearest-Neighbor-Based Active Learning for Rare Category Detection. *CMU Computer Science Department*,

Paper 281, 2007.
`http://repository.cmu.edu/compsci/281`

[208] Z. He, S. Deng, and X. Xu. Outlier Detection Integrating Semantic Knowledge. *Web Age Information Management (WAIM)*, 2002.

[209] Z. He, X. Xu, J. Huang, and S. Deng. FP-Outlier: Frequent Pattern-based Outlier Detection, *COMSIS*, 2(1), 2005.

[210] Z. He, X. Xu, and S. Deng. Discovering Cluster-based Local Outliers, *Pattern Recognition Letters*, Vol 24(9–10), pp. 1641–1650, 2003.

[211] Z. He, X. Xu, and S. Deng. An Optimization Model for Outlier Detection in Categorical Data. *International Conference on Intelligent Computing*, 2005.

[212] Z. He, S. Deng, X. Xu, and J. Huang. A Fast Greedy Algorithm for Outlier Mining. *PAKDD Conference*, 2006.

[213] M. Henrion, D. Hand, A. Gandy, and D. Mortlock. CASOS: A Subspace Method for Anomaly Detection in High Dimensional Astronomical Databases. *Statistical Analysis and Data Mining*, 2012. Online first: `http://onlinelibrary.wiley.com/doi/10.1002/sam.11167`

[214] S. Hido, Y. Tsuboi, H. Kashima, M. Sugiyama, and T. Kanamori. Statistical Outlier Detection using Direct Density Ratio Estimation. *Knowledge and information Systems*, 26(2), pp. 309–336, 2011.

[215] A. Hinneburg, C. Aggarwal, and D. Keim. What is the nearest neighbor in high-dimensional spaces?, *VLDB Conference*, 2000.

[216] H. O. Hirschfeld. A connection between correlation and contingency, *Proc. Cambridge Philosophical Society*, 31, pp. 520–524, 1935.

[217] S.-S. Ho. A Martingale Framework for Concept Change Detection in Time-Varying Data Streams, *ICML Conference*, 2005.

[218] V. Hodge and J. Austin. A Survey of Outlier Detection Methodologies, *Artifical Intelligence Review*, 22(2), pp. 85–126, 2004.

[219] J. Hodges. Efficiency in normal samples and tolerence of extreme values for some estimates of location, *Fifth Berkeley Symposium on Mathematics, Statistics and Probability*, 1, pp. 163–168, 1967.

[220] H. Hoffmann. Kernel PCA for Novelty Detection, *Pattern Recognition*, 40(3), pp. 863–874, 2007.

[221] T. Hofmann. Probabilistic Latent Semantic Indexing. *ACM SIGIR Conference*, 1999.

[222] S. Hofmeyr, S. Forrest, and A. Somayaji. Intrusion Detection using Sequences of System Calls, *Journal of Computer Security*, 6(3), pp. 151–180, 1998.

[223] J. H. Holland. Adaptation in Natural and Artificial Systems. *University of Michigan Press*, Ann Arbor, MI, 1975.

[224] G. Hollier, and J. Austin. Novelty Detection for Strain-gauge Degradation using Maximally Correlated Components. *European Symposium on Artificial Neural Networks*, 2002.

[225] J. Hollmen, and V. Tresp. Call-based Fraud Detection in Mobile Communication Networks using a Hierarchical Regime-switching Model. *NIPS Conference*, pp. 889–895, 1998.

[226] P. Horn, L. Feng, Y. Li, and A. J. Pesce. Effect of Outliers and Nonhealthy Individuals on Reference Interval Estimation. *Clinical Chemistry*, 47(12), pp. 2137–2145, 2001.

[227] L. Huang, M. I. Jordan, A. Joseph, M. Garofalakis, and N. Taft. In-network PCA and anomaly detection, *NIPS Conference*, 2006.

[228] J. W. Hunt and T. G. Szymanski. A Fast Algorithm for Computing Longest Common Subsequences, *Communications of the ACM*, 20(5), pp. 350–353, 1977.

[229] T. Ide, and H. Kashima. Eigenspace-based Anomaly Detection in Computer Systems. *ACM KDD Conference*, 2004.

[230] P. Indyk, R. Motwani, P. Raghavan, and S. Vempala. Locality-preserving Hashing in Multidimensional Spaces. *ACM STOC Conference*, 1997.

[231] D. Jackson, and Y. Chen. Robust Principal Component Analysis and Outlier Detection with Ecological Data, *Environmentrics*, 15, pp. 129–139, 2004.

[232] A. Jain, and R. Dubes. Algorithms for Clustering Data, *Prentice Hall*, 1988.

[233] H. Jagadish, N. Koudas, and S. Muthukrishnan. Mining Deviants in a Time-Series Database, *VLDB Conference*, 1999.

[234] V. Janeja and V. Atluri. Random Walks to Identify Anomalous Free-form Spatial Scan Windows. *IEEE Transactions on Knowledge and Data Engineering*, 20(10), 2008.

[235] P. Janeja and V. Atluri. Spatial Outlier Detection in Heterogeneous Neighborhoods. *Intelligent Data Analysis*, 13(1), 2008.

[236] H. Javitz, and A. Valdez. The SRI IDES Statistical Anomaly Detector. *IEEE Symposium on Security and Privacy*, 1991.

[237] Y. Jeong, M. Jeong, and O. Omitaomu, Weighted Dynamic Time Warping for Time Series Classification, *Pattern Recognition*, 44, pp. 2231–2240, 2010.

[238] B. Jiang, and J. Pei. Outlier Detection on Uncertain Data: Objects, Instances, and Inferences, *ICDE Conference*, 2011.

[239] M. F. Jiang, S. S. Tseng, and C. M. Su. Two-phase Clustering Process for Outliers Detection. *Pattern Recognition Letters*, 22, 6–7, pp. 691–700, 2001.

[240] R. Jiang, H. Fei, and J. Huan. Anomaly Localization for Network Data Streams with Graph Joint Sparse PCA. *ACM KDD Conference*, 2011.

[241] W. Jin, A. Tung, and J. Han. Mining Top-n Local Outliers in Large Databases. *ACM KDD Conference*, 2001.

[242] W. Jin, A. Tung, J. Han, and W. Wang. Ranking outliers using symmetric neighborhood relationship. *PAKDD Conference*, 2006.

[243] T. Johnson, I. Kwok, and R. Ng. Fast Computation of 2-dimensional Depth Contours. *ACM KDD Conference*, 1998.

[244] I. Jolliffe. Principal Component Analysis, *Springer*, 2002.

[245] M. Joshi, R. Agarwal, and V. Kumar. Mining Needles in a Haystack: Classifying Rare Classes via Two-Phase Rule Induction, *ACM SIGMOD Conference*, 2001.

[246] M. Joshi, V. Kumar, and R. Agarwal. Evaluating Boosting Algorithms to Classify Rare Classes: Comparison and Improvements. *ICDM Conference*, pp. 257–264, 2001.

[247] M. Joshi, and R. Agarwal. PNRule: A Framework for Learning Classifier Models in Data Mining (A Case Study in Network Intrusion Detection), *SDM Conference*, 2001.

[248] M. Joshi, R. Agarwal, and V. Kumar. Predicting Rare Classes: Can Boosting Make Any Weak Learner Strong? *ACM KDD Conference*, 2002.

[249] M. Joshi. On Evaluating Performance of Classifiers on Rare Classes, *ICDM Conference*, 2003.

[250] P. Juszczak and R. P. W. Duin. Uncertainty Sampling Methods for One-class Classifiers. *ICML Workshop on Learning from Imbalanced Data Sets*, 2003.

[251] J. Kang, S. Shekhar, C. Wennen, and P. Novak. Discovering Flow Anomalies: A SWEET Approach. *ICDM Conference*, 2008.

[252] G. Karakoulas and J. Shawe-Taylor. Optimising Classifiers for Imbalanced Training Sets, *NIPS*, 1998.

[253] D. R. Karger. Random sampling in cut, flow, and network design problems, *STOC*, pp. 648–657, 1994.

[254] S. Kasiviswanathan, P. Melville, and A. Banerjee. Emerging Topic Detection using Dictionary Learning, *CIKM Conference*, 2011.

[255] L. Kaufman, and P. Rousseeauw. Finding Groups in Data: An Introduction to Cluster Analysis, *Wiley-Interscience*, 1990.

[256] F. Keller, E. Muller, K. Bohm. HiCS: High-Contrast Subspaces for Density-based Outlier Ranking, *IEEE ICDE Conference*, 2012.

[257] E. Keogh, S. Lonardi, and B. Y.-C. Chiu. Finding Surprising Patterns in a Time Series Database in Linear Time and Space. *ACM KDD Conference*, 2002.

[258] E. Keogh, J. Lin, and A. Fu. HOT SAX: Finding the Most Unusual Time Series Subsequence: Algorithms and Applications, *ICDM Conference*, 2005.

[259] E. Keogh, S. Lonardi, and C. Ratanamahatana. Towards Parameter-Free Data Mining. *ACM KDD Conference*, 2004.

[260] D. Kifer, S. Ben-David, and J. Gehrke. Detecting Change in Data Streams, *VLDB Conference*, 2004.

[261] E. Knorr, and R. Ng. Algorithms for Mining Distance-based Outliers in Large Datasets. *VLDB Conference*, 1998.

[262] E. Knorr, and R. Ng. Finding Intensional Knowledge of Distance-Based Outliers. *VLDB Conference*, 1999.

[263] E. M. Knorr, R. T. Ng, and V. Tucakov. Distance-based Outliers: Algorithms and Applications, *VLDB Journal*, 8(3), pp. 237–253, February 2000.

[264] J. Koh, M.-L. Lee, W. Hsu, and W. Ang. Correlation-based Attribute Outlier Detection in XML. *ICDE Conference*, 2008.

[265] G. Kollios, D. Gunopulos, N. Koudas, and S. Berchtold. Efficient Biased Sampling for Approximate Clustering and Outlier Detection in Large Data Sets, *IEEE Transactions on Knowledge and Data Engineering*, 15(5), pp. 1170–1187, 2003.

[266] M. Kontaki, A. Gounaris, A. Papadopoulos, K. Tsichlas, and Y. Manolopoulos. Continuous Monitoring of Distance-based Outliers over Data Streams, *ICDE Conference*, 2011.

[267] K. Kontonasios and T. Bie. An Information-Theoretic Approach to Finding Noisy Tiles in Binary Databases, *SIAM Conference on Data Mining*, 2003.

[268] Y. Kou, C. T. Lu, and D. Chen. Spatial Weighted Outlier Detection, *SDM Conference*, 2006.

[269] H.-P. Kriegel, M. Schubert, and A. Zimek. Angle-based Outlier Detection in High-Dimensional Data, *ACM KDD Conference*, 2008.

[270] H.-P. Kriegel, P. Kroger, and A. Zimek. Outlier Detection Techniques, *Conference Tutorial at SIAM Data Mining Conference*, 2010. Tutorial Slides at: `http://www.siam.org/meetings/sdm10/tutorial3.pdf`

[271] H.-P. Kriegel, P. Kroger, E. Schubert, and A. Zimek. Interpreting and Unifying Outlier Scores. *SDM Conference*, pp. 13–24, 2011.

[272] H.-P. Kriegel, P. Kroger, E. Schubert, and A. Zimek. A General Framework for Increasing the Robustness of PCA-Based Correlation Clustering Algorithms. *SSDBM Conference*, 2008.

[273] H.-P. Kriegel, P. Kroger, E. Schubert, and A. Zimek. Outlier Detection in Axis-Parallel Subspaces of High Dimensional Data. *PAKDD Conference*, 2009.

[274] H.-P. Kriegel, P. Kroger, E. Schubert, and A. Zimek. Outlier Detection in Arbitrarily Oriented Subspaces, *ICDM Conference*, 2012.

[275] C. Kruegel, and G. Vigna. Anomaly-detection of Web-based Attacks, *CCS*, 2005.

[276] C. Kruegel, D. Mutz, W. Robertson, and F. Valeur. Bayesian Event Classification for Intrusion Detection. *Computer Security Applications Conference*, 2003.

[277] C. Kruegel, T. Toth, and E. Kirda. Service Specific Anomaly Detection for Network Intrusion Detection. *ACM symposium on Applied computing*, 2002.

[278] M. Kubat and S. Matwin. Addressing the Curse of Imbalanced Training Sets: One Sided Selection. *ICML Conference*, 1997.

[279] L. Kuncheva. Change Detection in Streaming Multivariate Data using Likelihood Detectors, *IEEE Transactions on Knowledge and Data Engineering*, Preprint, PP(99), 2011.

[280] A. Lakhina, M. Crovella, and C. Diot. Mining Anomalies using Traffic Feature Distributions, *ACM SIGCOMM Conference*, pp. 217–228, 2005.

[281] A. Lakhina, M. Crovella, and C. Diot. Diagnosing network-wide traffic anomalies. *ACM SIGCOMM Conference*, pp. 219–230, 2004.

[282] G. Lanckriet, L. Ghaoui, and M. Jordan. Robust Novelty Detection with Single Class MPM, *NIPS*, 2002.

[283] J. Lafferty, A. McCallum, and F. Pereira. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data, *ICML Conference*, 2001.

[284] T. Lane and C. Brodley. Temporal Sequence Learning and Data Reduction for Anomaly Detection, *ACM Transactions on Information and Security*, 2(3), pp. 295–331, 1999.

[285] T. Lane and C. Brodley. An Application of Machine Learning to Anomaly Detection, *NIST-NCSC National Information Systems Security Conference*, 1997.

[286] T. Lane, and C. Brodley. Sequence matching and learning in anomaly detection for computer security. *AI Approaches to Fraud Detection and Risk Management*, pp. 43–49, 1997.

[287] R. Lasaponara. On the use of Principal Component Analysis (PCA) for Evaluating Interannual Vegetation Anomalies from SPOT/VEGETATION NDVI Temporal Series. *Ecological Modeling*, 194(4), pp. 429–434, 2006.

[288] J. Laurikkala, M. Juholal, and E. Kentala. Informal Identification of Outliers in Medical Data, *Fifth International Workshop on Intelligent Data Analysis in Medicine and Pharmacology*, pp. 20–24, 2000.

[289] A. Lazarevic, and V. Kumar. Feature Bagging for Outlier Detection, *ACM KDD Conference*, 2005.

[290] A. Lazarevic, L. Ertoz, V. Kumar, A. Ozgur, and J. Srivastava. A Comparative Study of Anomaly Detection Schemes in Network Intrusion Detection. *SIAM Conference on Data Mining*, 2003.

[291] S. Q. Le and T. B. Ho. An Association-based Dissimilarity Measure for Categorical Data. *Pattern Recognition Letters*, 26(16), pp. 2549–2557, 2005.

[292] J.-G. Lee, J. Han, and X. Li. Trajectory Outlier Detection: A Partition-and-detect Framework, *ICDE Conference*, 2008.

[293] W. Lee and B. Liu. Learning with Positive and Unlabeled Examples using Weighted Logistic Regression. *ICML Conference*, 2003.

[294] W. Lee, S. Stolfo, and P. Chan. Learning Patterns from Unix Execution Traces for Intrusion Detection, *AAAI workshop on AI methods in Fraud and Risk Management*, 1997.

[295] W. Lee, S. Stolfo, and K. Mok. Adaptive Intrusion Detection: A Data Mining Approach, *Artificial Intelligence Review*, 14(6), pp. 533–567, 2000.

[296] W. Lee, and S. Stolfo. Data Mining Approaches for Intrusion Setection. *Proceedings of the 7th USENIX Security Symposium*, 1998.

[297] W. Lee, and D. Xiang. Information Theoretic Measures for Anomaly Detection, *IEEE Symposium on Security and Privacy*, 2001.

[298] N. Lesh, M. J. Zaki, and M. Ogihara. Mining Features for Sequence Classification, *ACM KDD Conference*, 1999.

[299] C. Leslie, E. Eskin, and W. Noble. The Spectrum Kernel: A String Kernel for SVM Protein Classification, *Pacific Symposium on Biocomputing*, pp. 566–575, 2002.

[300] X. Li, J. Han, S. Kim, and H. Gonzalez. ROAM: Rule and Motif-based Anomaly Detection in Massive Moving Object Data Sets, *SDM Conference*, 2007.

[301] X. Li, B. Liu, and S. Ng. Negative Training Data can be Harmful to Text Classification, *EMNLP*, 2010.

[302] X. Li, Z. Li, J. Han, and J.-G. Lee. Temporal Outlier Detection in Vehicle Traffic Data. *ICDE Conference*, 2009.

[303] D. Lin. An Information-theoretic Definition of Similarity. *ICML Conference*, pp. 296–304, 1998.

[304] J. Lin, E. Keogh, A. Fu, and H. V. Herle. Approximations to Magic: Finding Unusual Medical Time Series, *Mining Medical Data (CBMS)*, 2005.

[305] J. Lin, E. Keogh, S. Lonardi, and B. Y.-C. Chiu. A Symbolic Representation of Time Series, with Implications for Streaming Algorithms. *DMKD Workshop*, 2003.

[306] B. Liu, W. S. Lee, P. Yu, and X. Li. Partially Supervised Text Classification, *ICML Conference*, 2002.

[307] B. Liu, Y. Dai, X. Li, W. S. Lee, P. Yu. Building Text Classifiers Using Positive and Unlabeled Examples. *ICDM Conference*, 2003.

[308] G. Liu, T. McDaniel, S. Falkow, and S. Karlin, Sequence Anomalies in the cag7 Gene of the Helicobacter Pylori Pathogenicity Island, *National Academy of Sciences of the United States of America*, 96(12), pp. 7011–7016, 1999.

[309] L. Liu, and X. Fern. Constructing Training Sets for Outlier Detection, *SDM Conference*, 2012.

[310] F. T. Liu, K. M. Ting, and Z.-H. Zhou. Isolation Forest. *ICDM Conference*, 2008.

[311] S. Lin, and D. Brown. An Outlier-based Data Association Method for Linking Criminal Incidents. *SIAM Conference On Data Mining*, 2003.

[312] X.-Y. Liu, J. Wu, and Z.-H. Zhou. Exploratory Undersampling for Class-Imbalance Learning. *IEEE Transactions on Systems, Man and Cybernetics– Part B, Cybernetics*, 39(2), pp. 539–550, April 2009.

[313] X. Liu, X. Wu, H. Wang, R. Zhang, J. Bailey, and K. Ramamohanarao. Mining Distribution Change in Stock Order Data Streams, *ICDE Conference*, 2010.

[314] Z. Liu, W. Shi, D. Li, and Q. Qin. Partially Supervised Classification – based on Weighted Unlabeled Samples Support Vector Machine. *ADMA*, 2005.

[315] X. Liu, P. Zhang, and D. Zeng. Sequence Matching for Suspicious activity Detection in Anti-money Laundering. *Lecture Notes in Computer Science*, Vol. 5075, pp. 50–61, 2008.

[316] S. Loncarin. A Survey of Shape Analysis Techniques. *Pattern Recognition*, 31(5), pp. 983–1001, 1998.

[317] C.-T. Lu, D. Chen, and Y. Kou. Algorithms for Spatial Outlier Detection, *ICDM Conference*, 2003.

[318] J. Ma and S. Perkins. Online Novelty Detection on Temporal Sequences, *ACM KDD Conference*, 2003.

[319] J. Ma, L. Saul, S. Savage, and G. Volker. Learning to Detect Malicious URLs, *ACM Transactions on Intelligent Systems and Technology*, 2(3), Article 30, April 2011.

[320] M. Mahoney, and P. Chan. Learning Nonstationary Models of Normal Network Traffic for Detecting Novel Attacks, *ACM KDD Conference*, 2002.

[321] M. Mahoney, and P. Chan. Learning Rules for Anomaly Detection of Hostile Network Traffic, *ICDM Conference*, 2003.

[322] F. Malliaros, V. Megalooikonomou, and C. Faloutsos. Fast Robustness Estimation in Large Social Graphs: Communities and Anomaly Detection. *SDM Conference*, 2012.

[323] L. M. Manevitz and M. Yousef. One-class SVMs for Document Classification, *Journal of Machine Learning Research*, 2: pp, 139–154, 2001.

[324] C. Marceau. Characterizing the Behavior of a Program using Multiple-length n-grams, *Workshop on New Security Paradigms*, pp. 101–110, 2000.

[325] M. Markou and S. Singh. Novelty detection: A Review, Part 1: Statistical Approaches, *Signal Processing*, 83(12), pp. 2481–2497, 2003.

[326] M. Markou and S. Singh. Novelty Detection: A Review, Part 2: Neural Network-based Approaches, *Signal Processing*, 83(12), pp. 2481–2497, 2003.

[327] M. Masud, Q. Chen, L. Khan, C. Aggarwal, J. Gao, J, Han, and B. Thuraisingham. Addressing Concept-Evolution in Concept-Drifting Data Streams. *ICDM Conference*, 2010.

[328] M. Masud, T. Al-Khateeb, L. Khan, C. Aggarwal, J. Gao, J. Han, and B. Thuraisingham. Detecting Recurring and Novel Classes in Concept-Drifting Data Streams. *ICDM Conference*, 2011.

[329] M. Masud, Q. Chen, L. Khan, C. Aggarwal, J. Gao, J. Han, A. Srivastava, and N. Oza. Classification and Adaptive Novel Class Detection of Feature-Evolving Data Streams, *IEEE Transactions on Knowledge and Data Engineering*, to appear, Online verion appeared on May 22, 2012. `http://doi.ieeecomputersociety.org/10.1109/TKDE.2012.109`.

[330] C. McDiarmid. On the Method of Bounded Differences. In *Surveys in Combinatorics*, pp. 148–188, *Cambridge University Press*, Cambridge, 1989.

[331] P. Melville and R. Mooney. Diverse Ensembles for Active Learning, *ICML Conference*, 2004.

[332] C. Michael and A. Ghosh. Two State-based Approaches to Program-based Anomaly Detection, *Computer Security Applications Conference*, pp. 21, 2000.

[333] B. Miller, N. Bliss, and P. Wolfe. Subgraph Detection using Eigenvector L1-Norms. *NIPS Conference*, 2010.

[334] B. Miller, M. Beard, and N. Bliss. Eigenspace Analysis for Threat Detection in Social Networks. *International Conference on Information Fusion*, 2011.

[335] D. Mladenic and M. Grobelnik. Feature Selection for Unbalanced Class Distribution and Naive Bayes. *ICML Conference*, 1999.

[336] M. Mongiovi, P. Bogdanov, R. Ranca, A. Singh, E. Papalexakis, and C. Faloutsos. SIGSPOT: Mining Significant Anomalous Regions from Time-evolving Networks. *ACM SIGMOD Conference*, 2012.

[337] E. Muller, M. Schiffer, and T. Seidl. Statistical Selection of Relevant Subspace Projections for Outlier Ranking. *ICDE Conference*, pp, 434–445, 2011.

[338] E. Muller, M. Schiffer, P. Gerwert, M. Hannen, T. Jansen, and T. Seidl, SOREX: Subspace Outlier Ranking Exploration Toolkit, *Joint ECML PKDD Conference*, 2010.

[339] E. Muller, M. Schiffer, and T. Seidl. Adaptive Outlierness for Subspace Outlier Ranking, *CIKM Conference*, 2010.

[340] E. Muller, F. Keller, S. Blanc, and K. Bohm. OutRules: A Framework for Outlier Descriptions in Multiple Context Spaces. *ECML/PKDD Conference*, 2012.

[341] E. Muller, I. Assent, P. Iglesias, Y. Mulle, K. Bohm. Outlier Analysis via Subspace Analysis in Multiple Views of the Data, *ICDM Conference*, 2012.

[342] R. Motwani, and P. Raghavan. Randomized Algorithms, *Cambridge University Press*, 1995.

[343] A. Mueen, E. Keogh, and N. Young. Logical-Shapelets: An Expressive Primitive for Time Series Classification, *ACM KDD Conference*, 2011.

[344] A. Naftel and S. Khalid. Classifying Spatiotemporal Object Trajectories using Unsupervised Learning in the Coefficient Feature Space. *Multimedia Systems*, 12(3), pp. 227–238, 2006.

[345] K. Narita, and H. Kitagawa. Outlier Detection for Transaction Databases using Association Rules, *WAIM*, 2008.

[346] H. Nguyen, V. Gopalkrishnan, and I. Assent, An Unbiased Distance-based Outlier Detection Approach for High Dimensional Data, *DASFAA*, 2011.

[347] V. Niennattrakul, E. Keogh, and C. Ratanamahatana. Data Editing Techniques to Allow the Applicability of Distance-based Outlier Detection in Streams, *ICDM Conference*, 2010.

[348] H. Ning, W. Xu, Y. Chi, Y. Gong, and T. Huang. Incremental Spectral Clustering With Application to Monitoring of Evolving Blog Communities. *SDM Conference*, 2007.

[349] C. Noble, and D. Cook. Graph-based Anomaly Detection, *ACM KDD Conference*, 2003.

[350] P. Olmo Vaz de Melo, L. Akoglu, C. Faloutsos, and A. Loureiro. Surprising Patterns for the Call Duration Distribution of Mobile Phone Users, *ECML/PKDD Conference*, 2010.

[351] M. Otey, S. Parthasarathy, A. Ghoting, G. Li, S. Narravula, and D. Panda. Towards NIC-based Intrusion Detection. *ACM KDD Conference*, 2003.

[352] M. Otey, S. Parthasarathy, and A. Ghoting. Fast Distributed Outlier Detection in Mixed Attribute Data Sets, *Data Mining and Knowledge Discovery*, 12(2–3), pp. 203–228, 2006.

[353] C. R. Palmer and C. Faloutsos. Electricity based External Similarity of Categorical Attributes. *PAKDD Conference*, 2003.

[354] Y. Panatier. Variowin. Software For Spatial Data Analysis in 2D. *New York: Springer-Verlag*, 1996.

[355] C. Papadimitriou, P. Raghavan, H. Tamakai, and S. Vempala. Latent Semantic Indexing: A Probabilistic Analysis, *ACM PODS Conference*, 1998.

[356] S. Papadimitriou, H. Kitagawa, P. Gibbons, and C. Faloutsos. LOCI: Fast Outlier Detection using the Local Correlation Integral. *ICDE Conference*, 2003.

[357] S. Papadimitriou, J. Sun, and C. Faloutsos. SPIRIT: Streaming pattern discovery in multiple time-series. *VLDB Conference*, 2005.

[358] L. Parra, G. Deco, and S. Andmiesbach. Statistical Independence and Novelty Detection with Information Preserving Nonlinear Maps. *Neural Computation*, 8(2), pp. 260–269, 1996.

[359] Y. Pei, O. Zaiane, and Y. Gao. An Efficient Reference-based Approach to Outlier Detection in Large Datasets. *ICDM Conference*, 2006.

[360] D. Pelleg, and A. Moore. Active Learning for Anomaly and Rare Category Detection, *NIPS Conference*, 2004.

[361] Z. Peng, and F. Chu. Review Application of the Wavelet Transform in Machine Condition Monitoring and Fault Diagnostics, *Mechanical Systems and Signal Processing*, 18(2), pp. 199–221, March 2004.

[362] S. Petrovic, M. Osborne, and V. Lavrenko. Streaming First Story Detection with Application to Twitter. *Proceedings of the ACL Conference*, pp. 181–189, 2010.

[363] N. Pham, and R. Pagh. A Near-linear Time Approximation Algorithm for Angle-based Outlier Detection in High-dimensional Data, *ACM KDD Conference*, 2012.

[364] J. Pickands. Statistical Inference using Extreme Order Statistics. *The Annals of Statistics*, 3(1), pp. 119–131, 1975.

[365] B. Pincombe. Anomaly Detection in Time Series of Graphs using ARMA Processes. *ASOR Bulletin*, 24(4): 2–10, 2005.

[366] M. Pinsky. Introduction to Fourier Analysis and Wavelets, *American Mathematical Society*, 2009.

[367] C. Phua, V. Lee, K. Smith, and R. Gayler. A Comprehensive Survey of Data Mining-based Fraud Detection Research. `http://arxiv.org/abs/1009.6119`.

[368] C. Phua, D. Alahakoon, and V. Lee. Minority Report in Fraud Detection: Classification of Skewed Data, *ACM SIGKDD Explorations Newsletter*, 6(1), pp. 50–59, 2004.

[369] D. Pokrajac, A. Lazerevic, and L. Latecki. Incremental Local Outlier Detection for Data Streams, *CIDM Conference*, 2007.

[370] C. Potter, P. N. Tan, M. Steinbach, S. Klooster, V. Kumar, R. Myneni, and V. Genovese. Major Disturbance Events in Terrestrial Ecosystems detected using Global Satellite Data Sets. *Global Change Biology*, pp. 1005–1021, 2003.

[371] A. Pires, and C. Santos-Pereira. Using Clustering and Robust Estimators to Detect Outliers in Multivariate Data. *International Conference on Robust Statistics*, 2005.

[372] L. Portnoy, E. Eskin, and S. Stolfo. Intrusion Detection with Unlabeled Data using Clustering. *ACM Workshop on Data Mining Applied to Security*, 2001.

[373] B. Prakash, N. Valler, D. Andersen, M. Faloutsos, and C. Faloutsos. BGP-lens: Patterns and Anomalies in Internet Routing Updates. *ACM KDD Conference*, 2009.

[374] M. Prastawa, E. Bullitt, S. Ho, and G. Gerig. A Brain Tumor Segmentation Framework based on Outlier Detection, *Medical Image Analysis*, 8, pp. 275–283, 2004.

[375] C. Priebe, J. Conroy, D. Marchette, and Y. Park. Scan Statistics on Enron Graphs, *Computational and Mathematical Organizational Theory*, 11(3), pp. 229–247, 2005.

[376] F. Provost, and T. Fawcett. Analysis and Visualization of Classifier Performance: Comparison under Imprecise Class and Cost Distributions, *ACM KDD Conference*, 1997.

[377] F. Provost, T. Fawcett, and R. Kohavi. The Case against Accuracy Estimation while Comparing Induction Algorithms, *ICML Conference*, 1998.

[378] G. Qi, C. Aggarwal, and T. Huang. On Clustering Heterogeneous Social Media Objects with Outlier Links, *WSDM Conference*, 2012.

[379] L. R. Rabiner. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, *Proceedings of the IEEE*, 77(2), pp. 257–285, Feb. 1989.

[380] M. Radovanovic, A. Nanopoulos, and M. Ivanovic. On the Existence of Obstinate Results in Vector Space Models, *ACM SIGIR Conference*, 2010.

[381] S. Ramaswamy, R. Rastogi, and K. Shim. Efficient Algorithms for Mining Outliers from Large Data Sets. *ACM SIGMOD Conference*, pp. 427–438, 2000.

[382] B. Raskutti and A. Kowalczyk. Extreme Rebalancing for SVMS: A Case Study. *SIGKDD Explorations*, 6(1): pp. 60–69, 2004.

[383] S. Roberts. Novelty Detection using Extreme Value Statistics, *IEEE Proceedings on Vision, Image and Signal Processing*, 146(3). pp. 124–129, 1999.

[384] S. Roberts. Extreme Value Statistics for Novelty Detection in Biomedical Signal Processing. *International Conference on Advances in Medical Signal and Information Processing*. pp. 166–172, 2002.

[385] J. Rogan, J. Miller, D. Stow, J. Franklin, L. Levien, and C. Fischer. Land-Cover Change Monitoring with Classification Trees Using Landsat TM and Ancillary Data. *Photogrammetric Engineering and Remote Sensing*, 69(7), pp. 793–804, 2003.

[386] D. Ron, Y. Singer, and N. Tishby. The Power of Amnesia: Learning Probabilistic Automata with Variable Memory Length, *Machine Learning*, 25(2–3) pp. 117–149, 1996.

[387] P. Rousseeuw and A. Leroy. Robust Regression and Outlier Detection. *Wiley*, 2003.

[388] I. Ruts, and P. Rousseeuw, Computing Depth Contours of Bivariate Point Clouds. *Computational Statistics and Data Analysis*, 23, pp. 153–168, 1996.

[389] M. Sahami, S. Dumais, D. Heckerman, E. Horvitz. A Bayesian approach to filtering junk e-mail. *AAAI Workshop on Learning for Text Categorization. Tech. Rep. WS-98-05.* `http://robotics.stanford.edu/users/sahami/papers.html`

[390] R. K. Sahoo, A. J. Oliner, I. Rish, M. Gupta, J. E. Moreira, S. Ma, R. Vilalta, and A. Sivasubramaniam. Critical Event Prediction for Proactive Management in Large-scale Computer Clusters. *ACM KDD Conference*, 2003.

[391] G. Salton, and M. J. McGill. Introduction to Modern Information Retrieval, *McGraw Hill*, 1986.

[392] S. Sarawagi, R. Agrawal, and N. Megiddo. Discovery-driven Exploration of OLAP Data Cubes. *EDBT Conference*, 1998.

[393] G. Scarth, M. McIntyre, B. Wowk, and R. Somorjai. Detection of Novelty in Functional Images using Fuzzy Clustering. *Meeting of International Society for Magnetic Resonance in Medicine*, 1995.

[394] R. Schapire and Y. Singer. Improved Boosting Algorithms using Confidence-rated Predictions. *Annual Conference on Computational Learning Theory*, 1998.

[395] E. Schubert, R. Wojdanowski, A. Zimek, and H.-P. Kriegel. On Evaluation of Outlier Rankings and Outlier Scores, *SDM Conference*, 2012.

[396] B. Scholkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson. Estimating the support of a high-dimensional distribution. *Neural Computation*, 13(7), pp. 1443–1472, 2001.

[397] B. Scholkopf, R. C. Williamson, A. J. Smola, J. Shawe-Taylor, and J. C. Platt. Support-vector Method for Novelty Detection, *NIPS Conference*, 2000.

[398] R. Schoen, T. Habetler, F. Kamran, and R. Bartfield. Motor Bearing Damage Detection using Stator Current Monitoring. *IEEE Transactions on Industry Applications*, 31(6), pp. 1275–1279, 1995.

[399] K. Sequeira, and M. Zaki. ADMIT: Anomaly-based Data Mining for Intrusions, *ACM KDD Conference*, 2002.

[400] H. Seung, M. Opper, and H. Sompolinsky. Query by Committee. *ACM Workshop on Computational Learning Theory*, 1992.

[401] S. Shekhar, C. T. Lu, and P. Zhang. Detecting Graph-based Spatial Outliers: Algorithms and Applications, *ACM KDD Conference*, 2001.

[402] S.Shekhar, C. T. Lu, and P. Zhang. A Unified Approach to Detecting Spatial Outliers, *Geoinformatica*, 7(2), pp. 139–166, 2003.

[403] S. Shekhar and S. Chawla. A Tour of Spatial Databases. *Prentice Hall*, 2002.

[404] S. Shekhar, C. T. Lu, and P. Zhang. Detecting Graph-based Spatial Outliers, *Intelligent Data Analysis*, 6, pp. 451–468, 2002.

[405] P. Showbridge, M. Kraetzl, and D. Ray. Detection of Abnormal Change in Dynamic Networks. *Proceedings of the Intl. Conf. on Information, Decision and Control*, pp. 557–562, 1999.

[406] M.-L. Shyu, S.-C. Chen, K. Sarinnapakorn, and L. Chang. A Novel Anomaly Detection Scheme based on Principal Component Classifier, *ICDM Conference*, 2003.

[407] A. Siebes, J. Vreeken, and M. van Leeuwen. Itemsets than Compress, *SIAM Conference on Data Mining*, 2006.

[408] J. Silva, and R. Willett. Detection of Anomalous Meetings in a Social Network, *SocialCom*, 2008.

[409] B. W. Silverman. Density Estimation for Statistics and Data Analysis. *Chapman and Hall*, 1986.

[410] K. Smets and J. Vreeken. The Odd One Out: Identifying an Characterising Anomalies, *SIAM Conference on Data Mining*, 2011.

[411] E. S. Smirnov. On exact methods in systematics. *Systematic Zoology*, 17(1), pp. 1–13, 1968.

[412] R. Smith, A. Bivens, M. Embrechts, C. Palagiri, and B. Szymanski. Clustering Approaches for Anomaly Based Intrusion Detection. *Intelligent Engineering Systems through Artificial Neural Networks*, 2002.

[413] P. Smyth. Clustering Sequences with Hidden Markov Models, *Neural Information Processing*, 1997.

[414] P. Smyth. Markov Monitoring with Unknown States. *IEEE Journal on Selected Areas in Communications*, 12(9), pp. 1600-1612, 1994.

[415] H. Solberg, and A. Lahti. Detection of Outliers in Reference Distributions: Performance of Horn's Algorithm, *Clinical Chemistry*, 51(12), pp. 2326–2332, 2005.

[416] X. Song, M. Wu, C. Jermaine, and S. Ranka. Conditional Anomaly Detection, *IEEE Transaction on Knowledge and Data Engineering*, 19(5), pp. 631–645, 2007.

[417] X. Song, M. Wu, C. Jermaine, and S. Ranka. Statistical Change Detection for Multidimensional Data, *ACM KDD Conference*, 2007.

[418] C. Spence, L. Parra, and P. Sajda. Detection, Synthesis and Compression in Mammographic Image Analysis with a Hierarchical Image Probability Model. *IEEE Workshop on Mathematical Methods in Biomedical Image Analysis*, 2001.

[419] A. Srivastava. Discovering System Health Anomalies using Data Mining Techniques, *Joint Army Navy NASA Airforce Conference on Propulsion*, 2005.

[420] A. Srivastava. Enabling the Discovery of Recurring Anomalies in Aerospace Problem Reports using High-dimensional Clustering Techniques. *Aerospace Conference*, 2006.

[421] A. Srivastava, and B. Zane-Ulman. Discovering Recurring Anomalies in Text Reports regarding Complex Space Systems. *Aerospace Conference*, 2005.

[422] A. Srivastava, A. Kundu, S. Sural, and A. Majumdar. Credit card fraud detection using Hidden Markov Model. *IEEE Transactions on Dependable and Secure Computing*, 5(1), pp. 37–48, 2008.

[423] P. Srivastava, D. Desai, S. Nandi, and A. Lynn. HMM-ModE-Improved Classification using Profile Hidden Markov Models by Optimizing the Discrimination Threshold and Modifying Emission Probabilities with Negative Training Sequences, *BMC Bioinformatics*, 8 (104), 2007.

[424] M. Stephens. Use of the Kolmogorov-Smirnov, Cramer-von Mises and Related Statistics without Extensive Tables, *Journal of the Royal Statistical Society*. Series B, pp. 115–122, 1970.

[425] S. Stolfo, D. Fan, W. Lee, A.L. Prodromidis, and P. Chan. Credit Card Fraud Detection Using Meta-Learning: Issues and Initial Results, *AAAI Workshop AI Methods in Fraud and Risk Management*, pp. 83–90, 1997.

[426] S. Stolfo, D. Fan, W. Lee, A. Prodromidis, and P. Chan. Cost-Based Modeling for Fraud and Intrusion Detection: Results from the JAM Project, *DARPA Information Survivability Conf. and Exposition*, 2, pp. 130–144, 2000.

[427] S. Subramaniam, T. Palpanas, D. Papadopoulos, V. Kalogeraki, and D. Gunopulos. Online Outlier Detection in Sensor Data using Non-parametric Models. *VLDB Conference*, 2006.

[428] H. Sun, Y. Bao., F. Zhao, G. Yu, and D. Wang. CD-Trees: An Efficient Index Structure for Outlier Detection. *Web-Age Information Management (WAIM)*, pp. 600–609, 2004.

[429] J. Sun, S. Papadimitriou, P. Yu, and C. Faloutsos. Graphscope: Parameter-free Mining of Large Time-Evolving Graphs, *ACM KDD Conference*, 2007.

[430] J. Sun, D. Tao, and C. Faloutsos. Beyond Streams and Graphs: Dynamic Tensor Analysis, *ACM KDD Conference*, 2006.

[431] J. Sun, H. Qu, D. Chakrabarti, and C. Faloutsos. Neighborhood Formation and Anomaly Detection in Bipartite Graphs. *ICDM Conference*, 2005.

[432] J. Sun, Y. Xie, H. Zhang, and C. Faloutsos. Less is More: Compact Matrix Representation of Large Sparse Graphs. *SIAM Conference on Data Mining*, 2007.

[433] P. Sun, and S. Chawla. On Local Spatial Outliers, *IEEE ICDM Conference*, 2004.

[434] P. Sun, S. Chawla, and B. Arunasalam. Mining for Outliers in Sequential Databases, *SIAM International Conference on Data Mining*, 2006.

[435] Y. Sun, Y. Yu, and J. Han. Ranking-based Clustering of Heterogeneous Information Networks with Star Network Schema. *ACM KDD Conference*, 2009.

[436] C. Surace, and K. Worden. A Novelty Detection Method to Diagnose Damage in Structures: An Application to an Offshore Platform. *International Conference of Offshore and Polar Engineering*, 4, pp. 64–70, 1998.

[437] C. Surace, K. Worden, and G. Tomlinson. A Novelty Detection Approach to Diagnose Damage in a Cracked Beam. *Proceedings of SPIE*, Vol. 3089, pp. 947–953, 1997.

[438] E. Suzuki, T. Watanabe, H. Yokoi, and K. Takabayashi. Detecting Interesting Exceptions from Medical Test Data with Visual Summarization. *International Conference on Data Mining*, pp. 315–322, 2003.

[439] T. Takahashi, R. Tomioka, and K. Yamanishi. Discovering Emerging Topics in Social Streams via Link Anomaly Detection. *ICDM Conference*, 2011.

[440] P.-N. Tan and V. Kumar. Discovery of Web Robot Sessions based on their Navigational Patterns, *Data Mining and Knowledge Discovery*, 6(1), pp. 9–35, 2002.

[441] Y. Tao, X. Xiao, and S. Zhou. Mining Distance-based Outliers from Large Databases in any Metric Space. *ACM KDD Conference*, 2006.

[442] J. Tang, Z. Chen, A. W.-C. Fu, D. W. Cheung. Enhancing Effectiveness of Outlier Detections for Low Density Patterns. *PAKDD Conference*, 2002.

[443] Y. Tang, Y.-Q. Zhang, N. V. Chawla, and S. Krasser. SVMs Modeling for Highly Imbalanced Classification, *IEEE Transactions on Systems, Man and Cybernetics- Part B: Cybernetics*, 39(1), pp. 281–288, 2009.

[444] M. Taniguchi, M. Haft, J. Hollmen, and V. Tresp. Fraud Detection in Communications Networks using Neural and Probabilistic Methods. *IEEE International Conference in Acoustics, Speech and Signal Processing*, 2, pp. 1241–1444, 1998.

[445] D. Tax. One Class Classification: Concept-learning in the Absence of Counter-examples, *Doctoral Dissertation, University of Delft*, Netherlands, 2001.

[446] L. Tarassenko. Novelty detection for the Identification of Masses in Mammograms. *IEEE International Conference on Artificial Neural Networks*, 4, pp. 442–447, 1995.

[447] P. Thompson, D. MacDonald, M. Mega, C. Holmes, A. Evans, and A. Toga. Detection and Mapping of Abnormal Brain Structure with a Probabilistic Atlas of Cortical Surfaces. *Journal of Computer Assisted Tomography*, 21(4), pp. 567–581, 1997.

[448] M. Thottan, and C. Ji. Anomaly Detection in IP Networks. *IEEE Transactions on Signal Processing*, 51(8), pp. 2191–2204, 2003.

[449] S. Tian, S. Mu, and C. Yin. Sequence-similarity Kernels for SVMs to Detect Anomalies in System Calls, *Neurocomputing*, 70(4–6), pp. 859–866, 2007.

[450] K. M. Ting. An Instance-weighting Method to Induce Cost-sensitive Trees. *IEEE Transaction on Knowledge and Data Engineering*, 14: pp. 659–665, 2002.

[451] M. E. Tipping, and C. M. Bishop. Probabilistic Principal Component Analysis. *Journal of the Royal Statistical Society*, B 61, pp. 611–622, 1999.

[452] H. Tong, and C.-Y. Lin. Non-Negative Residual Matrix Factorization with Application to Graph Anomaly Detection. *SDM Conference*, 2011.

[453] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. N. Kavuri. A Review of Process Fault Detection and Diagnosis Part I: Quantitative Model-based Methods. *Computers and Chemical Engineering*, 27(3), pp. 293–311, 2003.

[454] J. S. Vitter. Random sampling with a reservoir, *ACM Trans. Math. Softw.*, vol. 11(1), pp. 37–57, 1985.

[455] W. Tobler. Cellular geography. In *Philosophy in Geography, Dordrecht Reidel Publishing Company*, pp. 379–386, 1979.

[456] S. Viaene, R. Derrig, B. Baesens, and G. Dedene. A Comparison of State-of-the-Art Classification Techniques for Expert Automobile Insurance Claim Fraud Detection, *Journal of Risk and Insurance*, 69(3), pp. 373–421, 2002.

[457] N. Wale, X. Ning, and G. Karypis. Trends in Chemical Data Mining, *Managing and Mining Graph Data*, *Springer*, 2010.

[458] X. Wang, C. Zhai, X. Hu and R. Sproat. Mining Correlated Bursty Topic Patterns from Coordinated Text Streams. *ACM KDD Conference*, 2007.

[459] B. Wang, G. Xiao, H. Yu, and X. Yang, Distance-based Outlier Detection on Uncertain Data, *International Conference on Computer and Information Technology*, 2009.

[460] B. Wang, X. Yang, G. Wang, and G. Yu. Outlier Detection over Sliding Windows for Probabilistic Data Streams, *Journal of Computer Science and Technology*, 25(3), pp. 389–400, 2010.

[461] L. Wei, W. Qian, A. Zhou, and W. Jin. HOT: Hypergraph-based Outlier Test for Categorical Data. *PAKDD Conference*, 2007.

[462] L. Wei and E. Keogh. Semi-supervised Time Series Classification, *ACM KDD Conference*, 2006.

[463] G. Weiss and F. Provost. Learning When Training Data are Costly: The Effect of Class Distribution on Tree Induction, *Journal of Artificial Intelligence Reserach*, 19: pp. 315–354, 2003.

[464] G. Williams, R. Baxter, H. He, S. Hawkings, and L. Gu. A Comparative Study of RNN for Outlier Detection in Data Mining. *IEEE ICDM Conference*, 2002.

[465] W.-K. Wong, A. Moore, G. Cooper, and M. Wagner. Rule-based Anomaly Pattern Detection for Detecting Disease Outbreaks, *National Conference on Artificial Intelligence*, 2002.

[466] K. van Leemput, F. Maes, D. Vandermeulen, A. Colchester, and P. Suetens. Automated Segmentation of Multiple Sclerosis Lesions by Model Outlier Detection, *IEEE Transactions on Medical Imaging*, vol. 20, pp. 677–688, August 2001.

[467] T. De Vries, S. Chawla, and M. Houle. Finding Local Anomalies in Very High Dimensional Space, *ICDM Conference*, 2010.

[468] M. Wang, C. Zhang, and J. Yu. Native API-based Windows Anomaly Intrusion Detection Method using SVM, *International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing*, 2006.

[469] L. Wei, E. Keogh, and X. Xi. SAXually Exaplicit Images: Finding Unusual Shapes, *ICDM Conference*, 2006.

[470] G. Wu and E. Y. Chang. Class-boundary Alignment for Imbalanced Dataset Learning. *Proceedings of the ICML Workshop on Learning from Imbalanced Data Sets*, 2003.

[471] M. Wu, and C. Jermaine. Outlier Detection by Sampling with Accuracy Guarantees. *ACM KDD Conference*, 2006.

[472] E. Wu, W. Liu, and S. Chawla. Spatio-temporal Outlier Detection in Precipitation Data, *Knowledge Discovery from Sensor Data, Springer, LNCS 5840*, 2008.

[473] M. Wu, X. Song, C. Jermaine, S. Ranka, and J. Gums. A LRT Framework for Fast Spatial Anomaly Detection. *ACM KDD Conference*, 2009.

[474] X. Xi, E. Keogh, C. Shelton, L.Wei, and C. Ratanamahatana. Fast Time Series Classification using Numerosity Reduction, *ICML Conference*, 2006.

[475] Z. Xing, J. Pei, and E. Keogh. A Brief Survey on Sequence Classification, *ACM SIGKDD Explorations*, 12(1), 2010.

[476] L. Xiong, X. Chen, and J. Schneider. Direct Robust Matrix Factorization for Anomaly Detection. *ICDM Conference*, 2011.

[477] L. Xiong, B. Poczos, J. Schneider, A. Connolly, and J. VanderPlas. Hierarchical Probabilistic Models for Group Anomaly Detection, *Artificial Intelligenece and Statitistics*, 2011.

[478] K. Yaminshi, J. Takeuchi, and G. Williams. Online Unsupervised Outlier Detection using Finite Mixtures with Discounted Learning Algorithms, *ACM KDD Conference*, 2000.

[479] K. Yaminshi, and J. Takeuchi. A Unified Framework for Detecting Outliers and Change Points from Time Series Data, *ACM KDD Conference*, 2002.

[480] R. Yan, Y. Liu, R. Jin, and A. Hauptmann. On Predicting Rare Classes with SVM Ensembles in Scene Classification. *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2003.

[481] J. Yang, and W. Wang. CLUSEQ: Efficient and Effective Sequence Clustering, *ICDE Conference*, 2003.

[482] P. Yang, and Q. Zhu. Finding Key Outlying Subspaces for Outlier Detection, *Knowledge-based Systems*, 24(2), pp. 269–274, 2011.

[483] X. Yang, L. Latecki, and D. Pokrajac. Outlier Detection with Globally Optimal Exemplar-based GMM. *SDM Conference*, 2009.

[484] Y. Yang, J. Carbonell, R. Brown, T. Pierce, B. T. Archibald, and X. Liu. Learning Approaches for Detecting and Tracking News Events. *IEEE Intelligent Systems*, 14(4):32–43, 1999.

[485] Y. Yang, T. Pierce, and J. Carbonell. A Study on Retrospective and On-line Event Detection. *ACM SIGIR Conference*, 1998.

[486] Y.Yang, J. Zhang, J. Carbonell, and C. Jin. Topic-conditioned Novelty Detection. *ACM KDD Conference*, 2002.

[487] D. Yankov, E. Keogh, and U. Rebbapragada. Disk-aware Discord Discovery: Finding Unusual Time Series in Terabyte Sized Data Sets, *ICDM Conference*, 2007.

[488] N. Ye. A Markov Chain Model of Temporal Behavior for Anomaly Detection, *IEEE Information Assurance Workshop*, 2004.

[489] N. Ye, and Q. Chen. An Anomaly Detection Technique based on a Chi-square Statistic for Detecting Intrusions into Information Systems. *Quality and Reliability Engineering International*, 17, pp. 105–112, 2001.

[490] L. Ye and E. Keogh. Time Series Shapelets: a New Primitive for Data Mining. *ACM KDD Conference*, 2009.

[491] B.-K. Yi, N. D. Sidiropoulos, T. Johnson, H. Jagadish, C. Faloutsos, and A. Biliris. Online Data Mining for Co-evolving Time Sequences. *ICDE Conference*, 2000.

[492] D. Yu, G. Sheikholeslami, and A. Zhang. Findout: Finding Outliers in Very Large Datasets. *Knowledge And Information Systems*, 4(4), pp. 387–412, 2002.

[493] H. Yu, J. Han, and K. C.-C. Chang. PEBL: Web Page Classification without Negative Examples. *IEEE Transactions on Knowledge and Data Engineering*, 16(1), pp. 70–81, 2004.

[494] J. X. Yu, W. Qian, H. Lu, and A. Zhou. Finding Centric Local Outliers in Categorical/Numeric Spaces. *Knowledge and Information Systems*, 9(3), pp. 309–338, 2006.

[495] B. Zadrozny, and C. Elkan. Transforming Classifier Scores into Accurate Multiclass Probability Estimates. *ACM KDD Conference*, 2002.

[496] B. Zadrozny, J. Langford, and N. Abe. Cost-Sensitive Learning by Cost-Proportionate Example Weighting, *ICDM Conference*, 2003.

[497] B. Zadrozny, and C. Elkan. Learning and Making Decisions when Costs and Probabilities are Unknown, *KDD Conference*, 2001.

[498] J. Zhang, Q. Gao, and H. Wang. SPOT: A System for Detecting Projected Outliers from High-Dimensional Data Stream, *ICDE Conference*, 2008.

[499] J. Zhang, M. Lou, T. W. Ling and H. Wang. HOS-Miner: A System for Detecting Outlying Subspaces of High-dimensional Data. *VLDB Conference*, 2004.

[500] J. Zhang, Q. Gao and H. Wang. A Novel Method for Detecting Outlying Subspaces in High-dimensional Databases Using Genetic Algorithm. *ICDM Conference*, 2006.

[501] J. Zhang and H. Wang. Detecting Outlying Subspaces for High-Dimensional Data: the New Task, Algorithms and Performance. *Knowledge and Information Systems*, 10(3), pp. 333–355, 2006.

[502] Y. Zhang, P. Meratnia, and P. Havinga. Outlier Detection for Wireless Sensor Networks: A Survey. *IEEE Communications Surveys and Tutorials*, 12(2), 2010.

[503] J. Zhang, Z. Ghahramani, and Y. Yang. A Probabilistic Model for Online Document Clustering with Application to Novelty Detection. *NIPS*, 2005.

[504] D. Zhang, and G. Lu. Review of Shape Representation and Description Techniques. *Pattern Recognition*, 37(1), pp. 1–19, 2004.

[505] T. Zhang, R. Ramakrishnan, and M. Livny. BIRCH: An Efficient Data Clustering Method for Very Large Databases. *ACM SIGMOD Conference*, 1996.

[506] J. Zhang and I. Mani. KNN Approach to Unbalanced Data Distributions: A Case Study Involving Information Extraction. *Proceedings of the ICML Workshop on Learning from Imbalanced Datasets*, 2003.

[507] D. Zhang and W. S. Lee. A Simple Probabilistic Approach to Learning from Positive and Unlabeled Examples. *Annual UK Workshop on Computational Intelligence*, pp. 83–87, 2005.

[508] X. Zhang, P. Fan, and Z. Zhu. A New Anomaly Detection Method based on Hierarchical HMM, *International Conference on Parallel and Distrbuted Computing, Applications, and Technologies*, 2003.

[509] Y. Zhang, J. Hong, and L. Cranor. CANTINA: A Content-based Approach to Detecting Phishing Web Sites. *WWW Conference*, 2007.

[510] J. Zhao, C.-T. Lu, and Y. Kou. Detecting Region Outliers in Meteorological Data. *ACM GIS Conference*, 2003.

[511] Z. Zheng, X. Wu, and R. Srihari. Feature Selection for Text Categorization on Imbalanced Data. *SIGKDD Explorations*, 6(1), pp. 80–89, 2004.

[512] C. Zhu, H. Kitagawa, S. Papadimitriou, and C. Faloutsos. OBE: Outlier by Example, *PAKDD Conference*, 2004.

[513] C. Zhu, H. Kitagawa, and C. Faloutsos. Example-based Robust Outlier Detection in High Dimensional Data Sets, *ICDM Conference*, 2005.

[514] A. Zimek, A. Schubert, and H.-P. Kriegel. A Survey on Unsupervised Outlier Detection in High-dimensional Numerical Data, *Journal on Statistical Analysis and Data Mining*, Preprint available at the online Wiley library: `http://onlinelibrary.wiley.com/doi/10.1002/sam.11161/abstract`, 2012.

[515] `http://www.itl.nist.gov/iad/mig/tests/tdt/tasks/fsd.html`

[516] D. D. Lewis. Reuters-21578 Data Set. `http://www.daviddlewis.com/resources/test-collections/reuters21578`.

[517] `http://kdd.ics.uci.edu/databases/20newsgroups`

[518] `http://www.informatik.uni-trier.de/~ley/db/`

[519] `http://www.kdnuggets.com/software/deviation.html`

[520] `http://www.kdnuggets.com/software/index.html`

[521] `http://www.cs.waikato.ac.nz/ml/weka/`

[522] `http://www.cs.ucr.edu/~eamonn/time_series_data/`

[523] `http://www.cs.ucr.edu/~eamonn/SAX.htm`

[524] `http://www-935.ibm.com/services/nz/en/it-services/ibm-proventia-network-anomaly-detection-system-ads.html`

[525] `http://www.ibm.com/software/analytics/spss`

[526] `http://www-01.ibm.com/software/analytics/spss/products/statistics/`

[527] `http://www.sas.com/`

[528] `http://www.sas.com/software/security-intelligence/index.html`

[529] `http://www.oracle.com/technetwork/database/options/advanced-analytics/odm/index.html`

[530] http://www.wizsoft.com

Index