Chapter 2

Probabilistic and Statistical Models for Outlier Detection

> "With four parameters, I can fit an elephant, and with five, I can make him wiggle his trunk." – John von Neumann

2.1 Introduction

The earliest methods for outlier detection were rooted in probabilistic and statistical models and date back to the nineteenth century [180]. These methods were proposed well before the advent and popularization of computer technology and were therefore designed without much focus on practical issues such as data representation or computational efficiency. Nevertheless, the underlying mathematical models are extremely useful and have eventually been adapted to a variety of computational scenarios.

A popular form of statistical modeling in outlier analysis is that of detecting *extreme univariate values*. In such cases, it is desirable to determine data values at the tails of a univariate distribution along with a corresponding level of statistical significance. Although extreme univariate values belong to a very specific category of outliers, they have numerous applications. For example, virtually all outlier detection algorithms use numerical scores to measure the anomalousness of data points, and the final step in these algorithms is to determine the extreme values from these scores. The identification of statistically significant extreme values helps in the conversion of outlier scores into binary labels. Some examples of outlier scoring mechanisms, which are used by different classes of algorithms, are as follows:

- In probabilistic modeling, the likelihood fit of a data point to a generative model is the outlier score.
- In proximity-based modeling, the k-nearest neighbor distance, distance to closest cluster centroid, or local density value is the outlier score.
- In linear modeling, the residual distance of a data point to a lower-dimensional representation of the data is the outlier score.

C.C. Aggarwal, *Outlier Analysis*, DOI 10.1007/978-3-319-47578-3_2

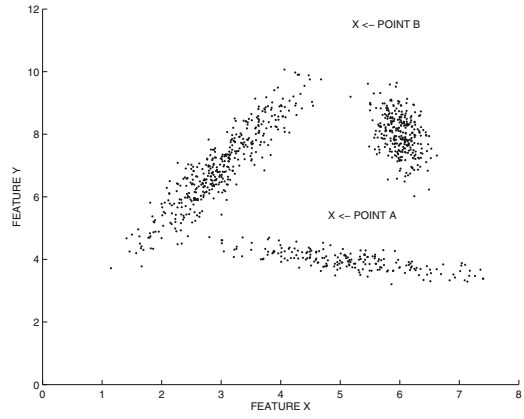


Figure 2.1: Example of the distinction between multivariate extreme values and outliers

- In temporal modeling, the deviation of a data point from its forecasted value is used to create the outlier score.

Thus, even when extreme-value modeling cannot be performed on the original data, the ability to determine the extreme values effectively from a set of outlier scores forms the cornerstone of all outlier detection algorithms as a final step. Therefore, the issue of extreme-value modeling will be studied extensively in this chapter.

Extreme-value modeling can also be easily extended to multivariate data. Data points that lie on the *pareto-extremes* of the data are referred to as multivariate extreme values. For example in Figure 2.1, data point 'B' is a multivariate extreme value. On the other hand, data point 'A' is an outlier but not a multivariate extreme value. Multivariate extreme-value analysis methods are sometimes also used for general outlier analysis. These techniques can *sometimes* perform surprisingly well in real-world outlier analysis applications, although they are not intended to be general outlier analysis methods. The reasons for this behavior are mostly rooted in the fact that real-world feature extraction methods sometimes create representations in which outliers are caused by extremes in values. For example, in a credit-card fraud detection application, it is common to extract features corresponding to the size and frequency of transactions. Unusually large or frequent transactions often correspond to outliers. Even if a subset of features is extracted in this way, it can greatly increase the effectiveness of multivariate extreme-value analysis methods for outlier detection. The drawback of using such methods in the general case is that data points like 'A' in Figure 2.1 are missed by such methods. Nevertheless, such methods should not be ignored in real applications in spite of this obvious drawback. In many cases, such techniques can be added as one or more components of an ensemble method (see Chapter 6) to enhance its accuracy.

It is also possible to use probabilistic modeling for finding general outliers beyond extreme values. For example, in Figure 2.1, one can model the data set as a *mixture* of three Gaussian components and therefore discover both outliers 'A' and 'B.' Mixture models can be considered probabilistic versions of clustering algorithms that discover the outliers as a side-product. A significant advantage of these methods is that they are easy to generalize to different data formats or even mixed attribute types, once a generative model for the data has been defined. Most probabilistic models assume a particular form to the underlying distribution for each mixture component (e.g., Gaussian) to model the normal patterns of data points. Subsequently, the parameters of this model are learned so that the observed

data has the maximum likelihood of being generated by the model [164]. This model is, therefore, a *generative* model for the data, and the probability of a particular data point being generated can be estimated from this model. Data points that have an unusually low probability of being generated by the model are identified as outliers. Mixture models are natural generalizations of multivariate extreme-value analysis; for example, if we modeled the mixture to contain a single Gaussian component, the approach specializes to one of the most well-known multivariate extreme-value analysis methods (see *Mahalanobis method* in section 2.3.4).

This chapter is organized as follows. The next section discusses statistical models for univariate extreme-value analysis. Methods for extreme-value analysis in multivariate data are discussed in section 2.3. Section 2.4 discusses methods for probabilistic modeling of outliers. Section 2.5 discusses the limitations of probabilistic models for outlier analysis. Section 2.6 presents the conclusions and summary.

2.2 Statistical Methods for Extreme-Value Analysis

In this section, we will present probabilistic and statistical methods for extreme-value analysis in univariate data distributions. The extreme values in a probability distribution are collectively referred to as the distribution *tail*. Statistical methods for extreme-value analysis quantify the probabilities in the tails of distributions. Clearly, a very low probability value of a tail indicates that a data value inside it should be considered anomalous. A number of tail inequalities bound these probabilities in cases where the actual distribution is not available.

2.2.1 Probabilistic Tail Inequalities

Tail inequalities can be used in order to bound the probability that a value in the tail of a probability distribution should be considered anomalous. The strength of a tail inequality depends on the number of assumptions made about the underlying random variable. Fewer assumptions lead to weaker inequalities but such inequalities apply to larger classes of random variables. For example, the *Markov* and *Chebychev* inequalities are weak inequalities but they apply to very large classes of random variables. On the other hand, the Chernoff bound and Hoeffding inequality are both stronger inequalities but they apply to restricted classes of random variables.

The *Markov inequality* is one of the most fundamental tail inequalities, and it is defined for distributions that take on only non-negative values. Let X be a random variable, with probability distribution $f_X(x)$, a mean of $E[X]$, and a variance of $Var[X]$.

Theorem 2.2.1 (Markov Inequality) *Let X be a random variable that takes on only non-negative random values. Then, for any constant α satisfying $E[X] < \alpha$, the following is true:*

$$P(X > \alpha) \leq E[X|/\alpha \tag{2.1}$$

Proof: Let $f_X(x)$ represent the density function for the random variable X. Then, we have:

$$\begin{aligned} E[X] &= \int_x x \cdot f_X(x) \cdot \mathrm{d}x = \int_{0 \leq x \leq \alpha} x \cdot f_X(x) \cdot \mathrm{d}x + \int_{x > \alpha} x \cdot f_X(x) \cdot \mathrm{d}x \\ &\geq \int_{x > \alpha} x \cdot f_X(x) \cdot \mathrm{d}x \geq \int_{x > \alpha} \alpha \cdot f_X(x) \cdot \mathrm{d}x \end{aligned}$$

The first inequality follows from the non-negativity of x, and the second follows from the fact that the integral is defined only over the cases in which $x > \alpha$. Furthermore, the term on the right-hand side of the last equation is exactly equal to $\alpha \cdot P(X > \alpha)$. Therefore, the following is true:

$$E[X] \geq \alpha \cdot P(X > \alpha) \tag{2.2}$$

The aforementioned inequality can be re-arranged in order to obtain the final result. ■

The Markov inequality is defined only for probability distributions of non-negative values and provides a bound only on the upper tail. In practice, it is often desired to bound both tails of arbitrary distributions. Consider the case where X is an arbitrary random variable, which is not necessarily non-negative. In such cases, the Markov inequality cannot be used directly. However, the (related) *Chebychev inequality* is very useful in such cases. The Chebychev inequality is a direct application of the Markov inequality to a non-negative derivative of random variable X:

Theorem 2.2.2 (Chebychev Inequality) *Let X be an arbitrary random variable. Then, for any constant α, the following is true:*

$$P(|X - E[X]| > \alpha) \leq Var[X]/\alpha^2 \tag{2.3}$$

Proof: The inequality $|X - E[X]| > \alpha$ is true if and only if $(X - E[X])^2 > \alpha^2$. By defining $Y = (X - E[X])^2$ as a (non-negative) derivative random variable from X, it is easy to see that $E[Y] = Var[X]$. Then, the expression on the left hand side of the theorem statement is the same as determining the probability $P(Y > \alpha^2)$. By applying the Markov inequality to the random variable Y, one can obtain the desired result. ■

The main trick used in the aforementioned proof was to apply the Markov inequality to a non-negative function of the random variable. This technique can generally be very useful for proving other types of bounds, when the distribution of X has a specific form (such as the sum of Bernoulli random variables). In such cases, a parameterized function of the random variable can be used in order to obtain a parameterized bound. The underlying parameters can then be optimized for the tightest possible bound. Several well-known bounds such as the Chernoff bound and the Hoeffding inequality are derived with the use of this approach.

The Markov and Chebychev inequalities are relatively weak inequalities and often do not provide tight enough bounds to be useful in many practical scenarios. This is because these inequalities do not make any assumptions on the nature of the random variable X. Many practical scenarios can however be captured, when stronger assumptions are used on the random variable. In such cases, much tighter bounds on tail distributions are possible. A particular case is one in which a random variable X may be expressed as a sum of other independent bounded random variables.

2.2.1.1 Sum of Bounded Random Variables

Many practical observations, which are defined in the form of *aggregates*, can be expressed as sums of bounded random variables. Some examples of such scenarios are as follows:

Example 2.2.1 (Sports Statistics) *The National Basketball Association (NBA) draft teams have access to college basketball statistics for the different candidate players. For each player and each game, a set of quantitative values describe their various scoring statistics over different games. For example, these quantitative values could correspond to the number*

of dunks, assists, rebounds, and so on. For a particular statistic, the aggregate performance of any player can be expressed as the sum of their statistics over N different games:

$$X = \sum_{i=1}^{N} X_i$$

All values of X_i lie in the range $[l, u]$. The performances of a player over different games are assumed to be independent of one another. The long-term global mean of the statistic represented by X_i over all players is known to be μ. The NBA draft teams would like to identify the anomalous players on the basis of each statistic.

In this example, the aggregate statistic is represented as a sum of bounded random variables. The corresponding tail bounds can be quantified with the use of the *Hoeffding inequality*.

In many cases, the individual random variable components in the aggregation are not only bounded, but also binary. Thus, the aggregate statistic can be expressed as a sum of Bernoulli random variables.

Example 2.2.2 (Grocery Shopping) *A grocery store keeps track of the number of customers (from its frequent purchaser program), who have frequented the store on a particular day. The long term probability of any customer i attending the store on a given day is known to be p_i. The behavior of the different customers is also known to be independent of one another. For a given day, evaluate the probability that the store receives more than η (frequent purchase program) customers.*

In the second example, the number of customers can be expressed as a sum of independent Bernoulli random variables. The corresponding tail distributions can be expressed in terms of the *Chernoff bound*. Finally, we provide a very common application of anomaly detection from aggregates, which is that of fault diagnosis in manufacturing.

Example 2.2.3 (Manufacturing Quality Control) *A company uses a manufacturing assembly line to produce a product, which may have faults in it with a pre-defined (low) probability p. The quality-control process samples N products from the assembly line, and examines them closely to count the number of products with defects. For a given count of faulty products, evaluate the probability that the assembly line is behaving anomalously.*

The sample size N is typically large, and, therefore, it is possible to use the *Central Limit Theorem* to assume that the samples are normally distributed. According to this theorem, the sum of a large number of independent and identical normal distributions converges to a normal distribution.

The different types of bounds and approximations will be formally introduced in this section. The Chernoff bounds and the Hoeffding inequality will be discussed first. Since the expressions for the lower tail and upper tails are slightly different, they will be addressed separately. The lower-tail Chernoff bound is introduced below.

Theorem 2.2.3 (Lower-Tail Chernoff Bound) *Let X be random variable that can be expressed as the sum of N independent binary (Bernoulli) random variables, each of which takes on the value of 1 with probability p_i.*

$$X = \sum_{i=1}^{N} X_i$$

Then, for any $\delta \in (0, 1)$, we can show the following:

$$P(X < (1-\delta) \cdot E[X]) < e^{-E[X] \cdot \delta^2/2} \tag{2.4}$$

where e is the base of the natural logarithm.

Proof: The first step is to show the following inequality:

$$P(X < (1-\delta) \cdot E[X]) < \left(\frac{e^{-\delta}}{(1-\delta)^{(1-\delta)}} \right)^{E[X]} \tag{2.5}$$

The *unknown* parameter $t > 0$ is introduced in order to create a parameterized bound. The lower-tail inequality of X is converted into an upper-tail inequality on the exponentiated expression $e^{-t \cdot X}$. This random expression can be bounded by the Markov inequality, and it provides a bound as a function of t. This function of t can be optimized, in order to obtain the tightest possible bound. By using the Markov inequality on the exponentiated form, the following can be derived:

$$P(X < (1-\delta) \cdot E[X]) \leq \frac{E[e^{-t \cdot X}]}{e^{-t \cdot (1-\delta) \cdot E[X]}}$$

By expanding $X = \sum_{i=1}^{N} X_i$ in the exponent, the following can be obtained:

$$P(X < (1-\delta) \cdot E[X]) \leq \frac{\prod_i E[e^{-t \cdot X_i}]}{e^{-t \cdot (1-\delta) \cdot E[X]}} \tag{2.6}$$

The aforementioned simplification uses the fact that the expectation of the product of independent variables is equal to the product of the expectations. Since each X_i is Bernoulli, the following can be shown:

$$E[e^{-t \cdot X_i}] = 1 + E[X_i] \cdot (e^{-t} - 1) < e^{E[X_i] \cdot (e^{-t} - 1)}$$

The second inequality follows from polynomial expansion of $e^{E[X_i] \cdot (e^{-t} - 1)}$. By substituting this inequality back into Equation 2.6, and using $E[X] = \sum_i E[X_i]$, the following may be obtained:

$$P(X < (1-\delta) \cdot E[X]) \leq \frac{e^{E[X] \cdot (e^{-t} - 1)}}{e^{-t \cdot (1-\delta) \cdot E[X]}}$$

The expression on the right is true for any value of $t > 0$. It is desired to determine the value of t that provides the *tightest possible* bound. Such a value of t may be obtained by computing the derivative of the expression with respect to t and setting it to 0. It can be shown that the resulting value of $t = t^*$ from this optimization process is as follows:

$$t^* = \ln(1/(1-\delta)) \tag{2.7}$$

By using this value of t^* in the aforementioned inequality , it can be shown to be equivalent to Equation 2.5. This completes the first part of the proof.

The first two terms of the Taylor expansion of the logarithmic term in $(1-\delta) \cdot \ln(1-\delta)$ can be expanded to show that $(1-\delta)^{(1-\delta)} > e^{-\delta + \delta^2/2}$. By substituting this inequality in the denominator of Equation 2.5, the desired result is obtained. ■

A similar result for the upper-tail Chernoff bound may be obtained, albeit in a slightly different form.

Theorem 2.2.4 (Upper-Tail Chernoff Bound) *Let X be random variable, which is expressed as the sum of N independent binary (Bernoulli) random variables, each of which takes on the value of 1 with probability p_i.*

$$X = \sum_{i=1}^{N} X_i$$

Then, for any $\delta \in (0, 2 \cdot e - 1)$, the following is true:

$$P(X > (1+\delta) \cdot E[X]) < e^{-E[X] \cdot \delta^2/4} \tag{2.8}$$

where e is the base of the natural logarithm.

Proof: The first step is to show the following inequality:

$$P(X > (1+\delta) \cdot E[X]) < \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}} \right)^{E[X]} \tag{2.9}$$

As before, this can be done by introducing the unknown parameter $t > 0$, and converting the upper-tail inequality on X into that on $e^{t \cdot X}$. This can be bounded by the Markov Inequality as a function of t. This function of t can be optimized, in order to obtain the tightest possible bound.

It can be further shown by algebraic simplification that the inequality in Equation 2.9 provides the desired result for all values of $\delta \in (0, 2 \cdot e - 1)$. ■

Next, the Hoeffding inequality will be introduced. The Hoeffding inequality is a more general tail inequality than the Chernoff bound, because it does not require the underlying data values to be drawn from a Bernoulli distribution. In this case, the ith data value needs to be drawn from *the bounded interval* $[l_i, u_i]$. The corresponding probability bound is expressed in terms of the parameters l_i and u_i. Thus, the scenario for the Chernoff bound is a special case of that for the Hoeffding inequality. We state the Hoeffding inequality below, for which both the upper- and lower-tail inequalities are identical.

Theorem 2.2.5 (Hoeffding Inequality) *Let X be a random variable that can be expressed as the sum of N independent random variables, each of which is bounded in the range $[l_i, u_i]$.*

$$X = \sum_{i=1}^{N} X_i$$

Then, for any $\theta > 0$, the following can be shown:

$$P(X - E[X] > \theta) \leq e^{-\frac{2 \cdot \theta^2}{\sum_{i=1}^{N} (u_i - l_i)^2}} \tag{2.10}$$

$$P(E[X] - X > \theta) \leq e^{-\frac{2 \cdot \theta^2}{\sum_{i=1}^{N} (u_i - l_i)^2}} \tag{2.11}$$

Proof: The proof of the upper-tail portion will be briefly described here. The proof of the lower-tail inequality is identical. For any choice parameter $t \geq 0$, the following is true:

$$P(X - E[X] > \theta) = P(e^{t \cdot (X - E[X])} > e^{t \cdot \theta}) \tag{2.12}$$

The Markov inequality can be used to show that the right-hand probability is at most $E[e^{(X - E[X])}] \cdot e^{-t \cdot \theta}$. The expression within $E[e^{(X - E[X])}]$ can be expanded in terms of the

Table 2.1: Comparison of different methods used to bound tail probabilities

Result	**Scenario**	**Strength**
Chebychev	Any random variable	Weak
Markov	Nonnegative random variable	Weak
Hoeffding	Sum of independent bounded random variables	Strong (Exponentially reduces with samples)
Chernoff	Sum of i.i.d. Bernoulli random variables	Strong (Exponentially reduces with samples)
CLT	Sum of **many** i.i.d. variables	Almost exact
Generalized CLT	Sum of **many** independent and bounded variables	Almost exact

individual components X_i. Since the expectation of the product is equal to the product of the expectations of independent random variables, the following can be shown:

$$P(X - E[X] > \theta) \leq e^{-t \cdot \theta} \cdot \prod_i E[e^{t \cdot (X_i - E[X_i])}] \tag{2.13}$$

The key is to show that the value of $E[e^{t \cdot (X_i - E[X_i])}]$ is at most equal to $e^{t^2 \cdot (u_i - l_i)^2/8}$. This can be shown with the use of an argument that uses the convexity of the exponential function $e^{t \cdot (X_i - E[X_i])}$ in conjunction with Taylor's theorem (see Exercise 12).

Therefore, the following is true:

$$P(X - E[X] > \theta) \leq e^{-t \cdot \theta} \cdot \prod_i e^{t^2 \cdot (u_i - l_i)^2/8} \tag{2.14}$$

This inequality holds for any positive value of t. Therefore, in order to find the tightest bound, the value of t that minimizes the right-hand side of the above equation needs to be determined. The optimal value of $t = t^*$ can be shown to be the following:

$$t^* = \frac{4 \cdot \theta}{\sum_{i=1}^{N} (u_i - l_i)^2} \tag{2.15}$$

By substituting the value of $t = t^*$, the desired result may be obtained. The lower-tail bound may be derived by applying the aforementioned steps to $P(E[X] - X > \theta)$ rather than $P(X - E[X] > \theta)$. ■

Thus, the different inequalities may apply to scenarios of different generality, and may also have different levels of strength. These different scenarios are presented in Table 2.1.

An interesting observation is that the Hoeffding tail bounds decay exponentially with θ^2, which is exactly how the normal distribution behaves. This is not very surprising, because the sum of a large number of independent bounded random variables converges to the normal distribution according to the *Central Limit Theorem (CLT)*. Such a convergence is useful, because the bounds provided by an exact distribution (or a close approximation) are much tighter than any of the aforementioned tail inequalities.

Theorem 2.2.6 (Central Limit Theorem) *The sum of a large number N of independent and identically distributed random variables with mean μ and standard deviation σ converges to a normal distribution with mean $\mu \cdot N$ and standard deviation $\sigma \cdot \sqrt{N}$.*

A more generalized form of the CLT can also be applied to sums of independent variables (not necessarily identical), in which the variables are sufficiently bounded in terms of underlying moment measures. An example of such a generalization of the CLT is the *Lyapunov* CLT [88]. The basic idea is that the means and variances of a sum of a large number of independent (but not identically distributed) random variables can be approximated by the corresponding sums of the means and variances, respectively. Some weak assumptions on the underlying distributions are also imposed for the condition to hold. Refer to the bibliographic notes.

2.2.2 Statistical-Tail Confidence Tests

The normal distribution has numerous applications such as statistical-tail confidence testing. In statistical-tail confidence tests, the extreme values from a set of data values distributed according to a normal distribution are identified. The assumption of a normal distribution is rather ubiquitous in real domains. This is true not just for variables that are expressed as sums of random samples (as discussed in the previous section), but many variables that are generated by different random processes. The density function $f_X(x)$ for the normal distribution with mean μ and standard deviation σ is defined as follows:

$$f_X(x) = \frac{1}{\sigma \cdot \sqrt{2 \cdot \pi}} \cdot e^{\frac{-(x-\mu)^2}{2 \cdot \sigma^2}} \tag{2.16}$$

In some settings, it is appropriate to assume that the mean μ and standard deviation σ of the modeling distribution are known. This is the case, when a very large number of samples of the data are available for accurate estimation of μ and σ. In other cases, μ and σ might be available from domain knowledge. Then, the Z-value z_i of an observed value x_i can be computed as follows:

$$z_i = (x_i - \mu)/\sigma \tag{2.17}$$

Since the normal distribution can be directly expressed as a function of the Z-value (and no other parameters), it follows that the tail probability of point x_i can also be expressed as a function of z_i. In fact, the Z-value corresponds to a scaled and translated normal random variable, which is also known as the *standard* normal distribution with mean 0 and variance 1. Therefore, the cumulative standard normal distribution can be used directly in order to determine the exact value of the tail probability at that value of z_i. From a practical perspective, since this distribution is not available in closed form, normal distribution tables are used in order to map the different values of z_i to probabilities. This provides a statistical level of significance, which can be interpreted directly as a probability of the data point being an outlier. The underlying assumption is that the data was generated by a normal distribution.

2.2.2.1 t-Value Test

The aforementioned discussion assumes that the mean and standard deviation of the modeling distribution can be estimated very accurately from a large number of samples. However, in practice, the available data sets might be small. For example, for a sample with 20 data points, it is much harder to model the mean and standard deviations accurately. How do we accurately perform statistical-significance tests in such cases?

The *Student's t-distribution* provides an effective way to model anomalies in such scenarios. This distribution is defined by a parameter known as the *number of degrees of freedom* ν, which is closely defined by the available sample size. The t-distribution approximates the

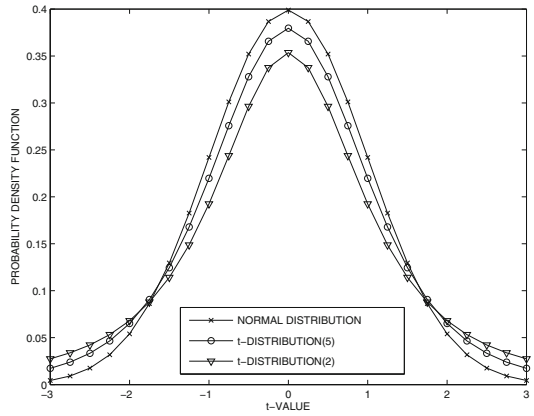


Figure 2.2: The t-distributions for different numbers of degrees of freedom (corresponding to different sample sizes)

normal distribution extremely well for larger degrees of freedom (> 1000), and converges to the normal distribution in the limit where it goes to ∞. For fewer degrees of freedom (or sample size), the t-distribution has a similar bell-shaped curve as the normal distribution, except that it has heavier tails. This is quite intuitive, because the heavier tail accounts for the loss in statistical significance from the inability to accurately estimate the mean and standard deviation of the modeling (normal) distribution from fewer samples.

The t-distribution is expressed as a function of *several independent identically-distributed standard normal distributions*. It has a single parameter ν that corresponds to the number of *degrees of freedom*. This regulates the *number* of such normal distributions, in terms of which it is expressed. The parameter ν is set to $N - 1$, where N is the total number of available samples. Let $U_0 \ldots U_\nu$ be $\nu + 1$, independent and identically distributed normal distributions with zero mean and unit standard deviation. Such a normal distribution is also referred to as the *standard normal distribution*. Then, the t-distribution is defined as follows:

$$T(\nu) = \frac{U_0}{\sqrt{(\sum_{i=1}^{\nu} U_i^2)/\nu}} \tag{2.18}$$

The intuition for using the t-distribution is that the denominator explicitly models the randomness of estimating the standard deviation of the underlying normal distribution with the use of only a *small* number of independent samples. The term $\sum_{i=1}^{\nu} U_i^2$ in the denominator is a χ^2 distribution with parameter ν, and the entire (scaled) denominator converges to 1, when $\nu \Rightarrow \infty$. Therefore, in the limiting case, when a large number of samples are available, the randomness contributed by the denominator disappears, and the t-distribution converges to the normal distribution. For smaller values of ν (or sample sizes), this distribution has a heavier tail. Examples of the t-distribution for different values of ν are provided in Figure 2.2. It is evident that t-distributions with fewer degrees of freedom have heavier tails.

The process of extreme-value detection with a small number of samples $x_1 \ldots x_N$ proceeds as follows. First, the mean and standard deviation of the sample are estimated. This is then used to compute the t-value of each data point directly from the sample. The t-value is computed in an identical way as the Z-value. The tail probability of each data point is computed from the cumulative density function of the t-distribution with $(N-1)$-degrees of

freedom. As in the case of the normal distribution, standardized tables are available for this purpose. From a practical perspective, if more than 1000 samples are available, then the t-distribution (with at least 1000 degrees of freedom) is so close to the normal distribution, that it is possible to use the normal distribution as a very good approximation.

2.2.2.2 Sum of Squares of Deviations

A common situation in outlier detection is the need to unify the deviations along independent criteria into a single outlier score. Each of these deviations is typically modeled as a Z-value from an independent and identically distributed standard normal distribution. The aggregate deviation measure is then computed as the sum of the squares of these values. For a d-dimensional data set, this is a χ^2-distribution with d degrees of freedom. A χ^2-distribution with d degrees of freedom is defined as the sum of the squares of d independent standard normal random variables. In other words, consider the variable V, which is expressed as the squared sum of independent and identically distributed standard normal random variables $Z_i \sim N(0, 1)$:

$$V = \sum_{i=1}^{d} Z_i^2$$

Then, V is a random variable drawn from a χ^2-distribution with d degrees of freedom.

$$V \sim \chi^2(d)$$

Although a detailed discussion of the characteristics of the χ^2-distribution is skipped here, its cumulative distribution is not available in closed form, but it needs to computationally evaluated. From a practical standpoint, cumulative probability tables are typically available for modeling purposes. The cumulative probability tables of the χ^2-distribution can then be used in order to determine the probabilistic level of significance for that aggregate deviation value. This approach is particularly useful when the deviations are modeled to be statistically independent of one another. As we will see in Chapter 3, such situations could arise in models such as principal component analysis, where the errors along the different components are often modeled as independent normal random variables.

2.2.2.3 Visualizing Extreme Values with Box Plots

An interesting approach to visualize univariate extreme values is the use of *box plots* or *box and whisker diagrams*. Such an approach is particularly useful in the context of visualizing outlier scores. In a box-plot, the statistics of a univariate distribution are summarized in terms of five quantities. These five quantities are the "minimum/maximum" (whiskers), the upper and lower quartiles (boxes), and the median (line in middle of box). We have enclosed quotations around two of these quantities because they are defined in a non-standard way. The distance between the upper and lower quartiles is referred to as the *inter-quartile range (IQR)*. The "minimum" and "maximum" are defined in a (non-standard) trimmed way in order to define the location of the whiskers. If there are no points more than 1.5 IQR above the top quartile value (upper end of the box), then the upper whisker is the true maximum. Otherwise, the upper whisker is set at 1.5 times the IQR from the upper end of the box. An exactly analogous rule holds true for the lower whisker, which is set at 1.5 IQR from the lower end of the box. In the special case of normally distributed data, a value of 1.5 IQR more than the top quartile corresponds to a distance of 2.7 times the standard deviation

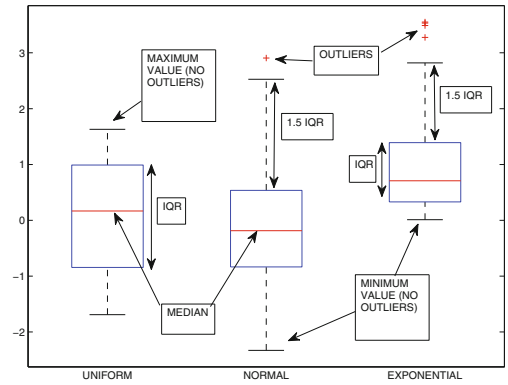


Figure 2.3: Visualizing univariate extreme values with box plots

(from the mean). Therefore, the whiskers are roughly placed at locations similar to the $3 \cdot \sigma$ cut-off points in a normal distribution.

An example of a box plot is illustrated in Figure 2.3. In this case we have shown 100 data points corresponding to each of the (i) uniform distribution with zero mean and unit variance, (ii) standard normal distribution, and (iii) an exponential distribution with unit mean. Note that the first two distributions are symmetric about the mean, whereas the last is not. The corresponding box plots are shown in Figure 2.3. In each case, the upper and lower ends of the box represent[1] the upper and lower quartiles. In the case of the uniform distribution, there no outliers, and therefore, the upper and lower whiskers represent the true maximum and minimum values. On the other hand, there are outliers at the upper end in the case of the normal and exponential distributions. Therefore, the whiskers are placed at 1.5 IQR above the upper ends of the boxes in each of the cases.

Many other conventions exist on the placement of whiskers, such as the use of the actual minimum/maximum or the use of particular percentiles of the data distribution. The specific convention used in this book is referred to as the *Tukey box-plot*. Aside from visualizing extreme values, this type of diagram is useful for visualizing the performance of a randomized outlier detection algorithm and is often used in outlier ensemble analysis. We will revisit this issue in section 6.4 of Chapter 6.

2.3 Extreme-Value Analysis in Multivariate Data

Extreme-value analysis can also be applied to multivariate data in a variety of ways. Some of these definitions try to model the underlying distribution explicitly, whereas others are based on more general statistical analysis, which does not assume any particular statistical distribution of the underlying data. In this section, we will discuss four different classes of methods that are designed to find data points at the *boundaries of multivariate data*. The first of these classes of methods (depth-based) is not a statistical or probabilistic approach. Rather, it is based on convex hull analysis of the point geometry. However, we have included it in this chapter, because it naturally fits with the other multivariate extreme-value methods in terms of the *types* of outliers it finds.

[1]It is possible to change the percentile levels of the boxes, although the use of quartiles is ubiquitous.

Algorithm *FindDepthOutliers*(Data Set: $\mathcal{D}$, Score Threshold: r);
begin
  $k = 1$;
  repeat
    Find set S of corners of convex hull of $\mathcal{D}$;
    Assign depth k to points in S;
    $\mathcal{D} = \mathcal{D} - S$;
    $k = k + 1$;
  until(D is empty);
  Report points with depth at most r as outliers;
end

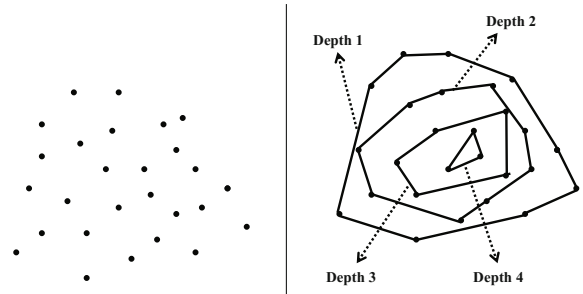
Figure 2.4: Pseudocode for finding depth-based outliers

While the methods discussed in this section are effective in finding outliers at the *outer boundaries* of a data space, they are not good at finding outliers within the inner regions of the data space. Such methods can effectively find outliers for the case illustrated in Figure 2.7, but not the outlier 'A' illustrated in Figure 2.1. Nevertheless, the determination of such outliers can be useful in many specialized scenarios. For example, in cases where multiple deviation values may be associated with records, multivariate extreme-value analysis may be useful. Consider a weather application in which multiple attributes such as temperature and pressure are measured at different spatial locations, and the local spatial deviations from the expected values are computed as an intermediate step. These deviations from expected values on different attributes may need to be transformed into a single meaningful outlier score. An example is illustrated in section 11.2.1.3 of Chapter 11, where deviations are computed on the different measured values of spatial data. In general, such methods are useful for post-processing a multidimensional vector of outlier scores, in which each outlier score is derived using a different and possibly independent criterion. As discussed in Chapter 1, it is particularly common to confuse methods for extreme-value analysis with general outlier analysis methods that are defined in terms of generative probabilities. However, it is important to distinguish between the two, since the application-specific scenarios in which the two kinds of methods are used are quite different.

2.3.1 Depth-Based Methods

In depth-based methods, convex hull analysis is used in order to find outliers. The idea is that the points in the outer boundaries of the data lie at the corners of the convex hull. Such points are more likely to be outliers. A depth-based algorithm proceeds in an iterative fashion. In the kth iteration, all points at the corners of the convex hull of the data set are removed from the data set. These points are assigned a depth of k. These steps are repeated until the data set is empty. All points with depth at most r are reported as the outliers. Alternatively, the depth of a data point may be directly reported as the outlier score. The steps of the depth-based approach are illustrated in Figure 2.4.

The algorithm is also pictorially illustrated on a sample data set in Figure 2.5. A number of efficient methods for finding depth-based outliers have been discussed in [295, 468]. The computational complexity of convex-hull methods increases exponentially with dimensionality. Furthermore, with increasing dimensionality, a larger proportion of data points lie at



(a) Data points (b) Convex hulls

Figure 2.5: Depth-based outlier detection

the corners of a convex hull. This is because the number of points at the corners of a convex hull can be exponentially related to the data dimensionality. Therefore, such methods are not only computationally impractical, but also increasingly ineffectual in higher dimensionality because of increasing ties in outlier scores. Depth-based methods are generally quite different from most of the probabilistic and statistical models discussed in this chapter. In fact, they cannot really be considered probabilistic or statistical methods. However, they are presented here because of their relationship to other multivariate extreme-value methods. Such methods share many characteristics in common, in spite of being methodologically different. For example, they work well only in scenarios where outliers lie at the boundaries of data space, rather than as isolated points in the interior of the data.

2.3.2 Deviation-Based Methods

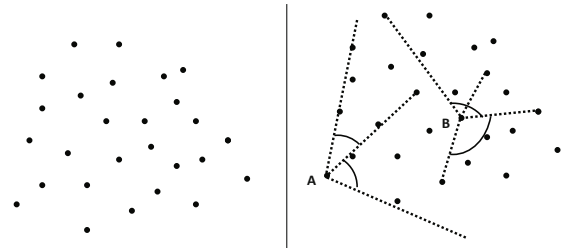
Deviation-based methods measure the impact of outliers on the data variance. For example, the method proposed in [62] tries to measure how much the variance in the underlying data is reduced, when a particular data point is removed. Since the basic assumption is that the outliers lie at the boundary of the data, it is expected that the removal of such data points will significantly reduce the variance. This is essentially an *information-theoretic* method, since it examines the reduction in complexity, when a data point is removed. Correspondingly, the *smoothing factor* for a set of data points R is defined as follows:

Definition 2.3.1 *The smoothing factor $SF(R)$ for a set R is the reduction in the data set variance, when the set of points in R are removed from the data.*

Outliers are defined as exception sets E such that their removal causes the maximum reduction in variance of the data. In other words, for *any* subset of data points R, it must be the case that:

$$SF(E) \geq SF(R)$$

If more than one set have the same reduction in variance, then the smaller set is preferred. This follows the standard information theoretic principle of finding the sets that increase the description length of the data as much as possible, in as little space. The determination of the optimal set E is a very difficult problem, because 2^N possibilities exist for a data set containing N points. The work in [62] uses a number of heuristics such as best-first search and random sampling. One good aspect of this approach is that it is distribution-independent, and can be applied to any kind of data set, as long as an appropriate definition



(a) Data points (b) Spectrum of angles

Figure 2.6: Angle-based outlier detection

of the smoothing factor can be constructed. In the original work in [62], this approach has been applied to the case of sequence data.

2.3.3 Angle-Based Outlier Detection

This method was originally proposed as a general outlier analysis method, although this book has reclassified it to a multivariate extreme-value analysis method. The idea in angle-based methods is that data points at the boundaries of the data are likely to enclose the entire data within a smaller angle, whereas points in the interior are likely to have data points around them at different angles. For example, consider the two data points 'A' and 'B' in Figure 2.6, in which point 'A' is an outlier, and point 'B' lies in the interior of the data. It is clear that all data points lie within a limited angle centered at 'A.' On the other hand, this is not the case for data point 'B,' which lies within the interior of the data. In this case, the angles between different pairs of points can vary widely. In fact, the more isolated a data point is from the remaining points, the smaller the underlying angle is likely to be. Thus, data points with a smaller angle spectrum are outliers, whereas those with a larger angle spectrum are not outliers.

Consider three data points $\overline{X}$, $\overline{Y}$, and $\overline{Z}$. Then, the angle between the vectors $\overline{Y} - \overline{X}$ and the $\overline{Z} - \overline{X}$, will not vary much for different values of $\overline{Y}$ and $\overline{Z}$, when $\overline{X}$ is an outlier. Furthermore, the angle is inversely weighted by the distance between the points. The corresponding angle (weighted cosine) is defined as follows:

$$WCos(\overline{Y} - \overline{X}, \overline{Z} - \overline{X}) = \frac{< (\overline{Y} - \overline{X}), (\overline{Z} - \overline{X}) >}{||\overline{Y} - \overline{X}||_2^2 \cdot ||\overline{Z} - \overline{X}||_2^2}$$

Here, $|| \cdot ||_2$ represents the L_2-norm, and $< \cdot >$ represents the scalar product. Note that this is a weighted cosine, since the denominator contains the squares of the L_2-norms. The inverse weighting by the distance further reduces the weighted angles for outlier points, which also has an impact on the spectrum of angles. Then, the *variance in the spectrum* of this angle is measured by varying the data points $\overline{Y}$ and $\overline{Z}$, while keeping the value of $\overline{X}$ fixed. Correspondingly, the *angle-based outlier factor (ABOF)* of the data point $\overline{X} \in \mathcal{D}$ is defined as follows:

$$ABOF(\overline{X}) = Var_{\{Y,Z \in \mathcal{D}\}} WCos(\overline{Y} - \overline{X}, \overline{Z} - \overline{X})$$

Data points that are outliers will have a smaller spectrum of angles, and will therefore have lower values of the angle-based outlier factor $ABOF(\overline{X})$.

The angle-based outlier factor of the different data points may be computed in a number of ways. The naive approach is to pick all possible triples of data points and compute the $O(N^3)$ angles between the different vectors. The ABOF values can be explicitly computed from these values. However, such an approach can be impractical for very large data sets. A number of efficiency-based optimizations have therefore been proposed.

In order to speed up the approach, a natural possibility is to use sampling in order to approximate this value of the angle-based outlier factor. A sample of k data points can be used in order to approximate the ABOF of a data point $\overline{X}$. One possibility is to use an unbiased sample. However, since the angle-based outlier factor is inversely weighted by distances, it follows that the nearest neighbors of a data point have the largest contribution to the angle-based outlier factor. Therefore, the k-nearest neighbors of $\overline{X}$ can be used to approximate the outlier factor much more effectively than an unbiased sample of the all the data points. It has also been shown in [325] that many data points can be filtered out on the basis of approximate computation, since their approximate values of the ABOF are too high, and they cannot possibly be outliers. The exact values of the ABOF are computed only for a small set of points, and the points with the lowest values of the ABOF are reported as outliers. We refer the reader to [325] for the details of these efficiency optimizations.

Because of the inverse weighting by distances, the angle-based outlier analysis method can be considered a hybrid between distance-based and angle-based methods. As discussed earlier with the use of the illustrative example, the latter factor is primarily optimized to finding multivariate extreme values in the data. The precise impact of each of these factors[2] does not seem to be easily quantifiable in a statistically robust way. In most data sets such as in Figure 2.1, outliers lie not just on the boundaries of the data, but also in the interior of the data. Unlike extreme values, outliers are defined by generative probabilities. While the distance factor can provide some impact for the outliers in the interior, the work is primarily focused on the advantage of angular measures, and it is stated in [325] that the degree of impact of distance factors is minor compared to the angular factors. This implies that outliers on the boundaries of the data will be highly favored in terms of the overall score, because of the lower spectrum of angles. Therefore, the angle-based method treats outliers with similar generative probabilities in the interior and the boundaries of the data in a differential way, which is not statistically desirable for general outlier analysis. Specifically, the outliers at the boundaries of the data are more likely to be favored in terms of the outlier score. Such methods can effectively find outliers for the case illustrated in Figure 2.7, but the outlier 'A' illustrated in Figure 2.1 will be favored less. Therefore, while this approach was originally presented as a general outlier analysis method, it has been classified in the section on multivariate extreme-value analysis methods in this book.

It has been claimed in [325] that the approach is more suitable for high-dimensional data because of its use of angles, as opposed to distances. However, it has been shown in earlier work [455], that angle-based measures are not immune to the dimensionality curse, because of concentration effects in the cosine measure. Such concentration effects would also impact the spectrum of the angles, even when they are combined with distances. The variation in the angle spectrum in Figure 2.6 is easy to show visually in 2-dimensional data, but the sparsity effects will also impact the spectrum of angles in higher dimensions. If the main problem, as suggested in [325], is the lack of contrast between pairwise distances, then this is not resolved with the use of angles instead of distances. In a setting where all pairs of distances are similar, all triangles will be equilateral, and therefore all (cosines of) angles

[2]When a random variable is scaled by a factor of a, its variance is scaled by a factor of a^2. However, the scaling here is not by a constant factor.

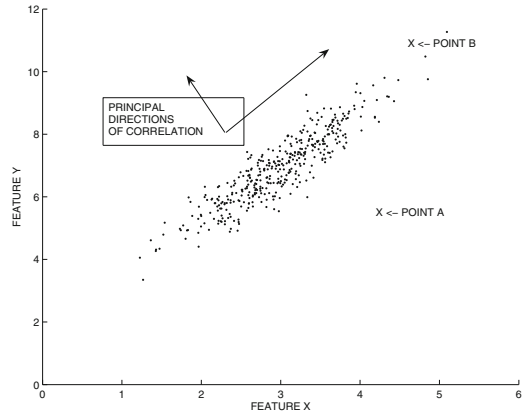


Figure 2.7: Extreme-value analysis in multivariate data with Mahalanobis distance

will converge to 0.5. In fact, the cosine can be shown to be a direct function of Euclidean pairwise distances:

$$\text{Cosine}(\overline{X}, \overline{Y}) = \frac{||X-0||^2 + ||Y-0||^2 - ||X-Y||^2}{2 \cdot ||X-0|| \cdot ||Y-0||} \tag{2.19}$$

If distances retain little information about the relative contrasts, there is little reason to believe that an *indirect* function of the distances (like the cosine spectrum) will do any better. A clear explanation of why the spectrum of angles should be more robust to high dimensionality than distances has not[3] been provided in [325]. More importantly, such methods do not address the issue of locally irrelevant attributes [4], which are the primary impediment to effective outlier analysis methods with increasing dimensionality. Another important point to note is that multivariate extreme-value analysis is much simpler than general outlier analysis in high dimensionality, because the parts of the data to explore are approximately known, and therefore the analysis is global rather than local. The evidence over different dimensions can be accumulated with the use of a very simple classical distance-distribution method [343, 493]. The approach, discussed in the next section, is also suitable for high-dimensional extreme-value analysis, because it implicitly weights globally relevant and irrelevant directions in the data in a different way, and is statistically sound in terms of probabilistic interpretability of the extreme values.

2.3.4 Distance Distribution-based Techniques: The Mahalanobis Method

A *distribution-dependent* approach is to model the entire data set to be normally distributed about its mean in the form of a multivariate Gaussian distribution. Let $\overline{\mu}$ be the

[3]The use of the cosine function in some high-dimensional domains such as text has been cited as an example in a later work [326]. In domains with small and varying non-zero attributes, the cosine is preferred because of important normalization properties, and not because of greater dimensionality resistance. By substituting $||\overline{X}|| = ||\overline{Y}|| = 1$ in Equation 2.19, it is evident that the cosine is equivalent to the Euclidean distance if all points are normalized to lie on a unit ball. The cosine function is not immune to the dimensionality curse even for the unique structure of text [455]. An increasing fraction of non-zero attributes, towards more general distributions, directly impacts the data hubness.

d-dimensional mean (row) vector of a d-dimensional data set, and Σ be its $d \times d$ covariance matrix. In this case, the (i, j)th entry of the covariance matrix is equal to the covariance between the dimensions i and j. Then, the probability distribution $f(\overline{X})$ for a d-dimensional (row vector) data point $\overline{X}$ can be defined as follows:

$$f(\overline{X}) = \frac{1}{\sqrt{|\Sigma| \cdot (2 \cdot \pi)^{(d/2)}}} \cdot \exp\left[-\frac{1}{2} \cdot (\overline{X} - \overline{\mu})\Sigma^{-1}(\overline{X} - \overline{\mu})^T\right] \tag{2.20}$$

The value of $|\Sigma|$ denotes the determinant of the covariance matrix. We note that the term in the exponent is (half) the squared *Mahalanobis distance* of the data point $\overline{X}$ to the centroid $\overline{\mu}$ of the data. This term is used as the outlier score and may be directly computed as follows:

$$\text{Mahalanobis}(\overline{X}, \overline{\mu}, \Sigma) = \sqrt{(\overline{X} - \overline{\mu})\Sigma^{-1}(\overline{X} - \overline{\mu})^T} \tag{2.21}$$

The computation of the Mahalanobis distance requires the inversion of the covariance matrix Σ. In cases where the matrix Σ is not invertible, it is possible to use *regularization* with a $d \times d$ identity matrix I. The basic idea is to replace Σ with $\Sigma + \lambda I$ for some small value of $\lambda > 0$ in Equation 2.21. Here, $\lambda > 0$ represents the regularization parameter.

The Mahalanobis distance of a point is similar to its Euclidean distance from the centroid of the data, except that it normalizes the data on the basis of the inter-attribute correlations. For example, if the axis system of the data were to be rotated to the principal directions (shown in Figure 2.7), then the data would have no inter-attribute correlations. As we will see in section 3.3 of Chapter 3, it is actually possible to determine such directions of correlations generally in d-dimensional data sets with the use of principal component analysis (PCA). The Mahalanobis distance is simply equal to the Euclidean distance between $\overline{X}$ and $\overline{\mu}$ in such a transformed (axes-rotated) data set *after* dividing each of the transformed coordinate values by the standard-deviation of that direction. Therefore, principal component analysis can also be used in order to compute the Mahalanobis distance (see section 3.3.1 of Chapter 3).

This approach recognizes the fact that the different directions of correlation have different variance, and the data should be treated in a statistically normalized way along these directions. For example, in the case of Figure 2.7, the data point 'A' can be more reasonably considered an outlier than data point 'B,' on the basis of the natural correlations in the data. On the other hand, the data point 'A' is closer to the centroid of the data (than data point 'B') on the basis of *Euclidean distance*, but not on the basis of the Mahalanobis distance. Interestingly, data point 'A' also seems to have a much higher spectrum of angles than data point 'B,' at least from an average sampling perspective. This implies that, at least on the basis of the primary criterion of angles, the angle-based method would incorrectly favor data point 'B.' This is because it is unable to account for the relative relevance of the different directions, an issue that becomes more prominent with increasing dimensionality. The Mahalanobis method is robust to increasing dimensionality, because it uses the covariance matrix in order to summarize the high dimensional deviations in a statistically effective way. It is noteworthy that the Mahalanobis method should *not* be merely considered an extreme-value method. In fact, as section 3.3.1 shows, its correlation-sensitive characteristics are more powerful than its extreme-value characteristics.

We further note that each of the distances along the principal correlation directions can be modeled as a 1-dimensional standard normal distribution, which is approximately independent from the other orthogonal directions of correlation. As discussed earlier in this chapter, the sum of the squares of d variables drawn independently from a standard normal distribution, will result in a variable drawn from a χ^2-distribution with d degrees of freedom.

Therefore, the cumulative probability distribution tables of the χ^2 distribution can be used in order to determine the outliers with the appropriate level of significance.

2.3.4.1 Strengths of the Mahalanobis Method

Although the Mahalanobis method seems simplistic at first sight, it is easy to overlook the fact that the Mahalanobis method accounts for the inter-attribute dependencies in a graceful way, which become particularly important in high-dimensional data sets. This simple approach turns out to have several surprising advantages over more complex distance-based methods in terms of accuracy, computational complexity, and parametrization:

1. It is short-sighted to view the Mahalanobis method only as a multivariate extreme-value analysis method because most of its power resides in its use of inter-attribute correlations. The use of the covariance matrix ensures that inter-attribute dependencies are accounted for in the outlier detection process. In fact, as discussed in Chapter 3, one can view the Mahalanobis method as a soft version of PCA. Although it is not immediately obvious, the merits of some of the sophisticated linear models such as one-class support-vector machines (SVMs)[4] and matrix factorization are inherently built into the approach. In this sense, the Mahalanobis method uses a more powerful model than a typical multivariate extreme-value analysis method. A detailed discussion of the connections of PCA with the Mahalanobis method and its nonlinear extension is provided in Chapter 3. Aside from its PCA-based interpretation, it also has a natural probabilistic interpretation as a special case of the EM-method discussed in the next section.

2. The Mahalanobis method is *parameter-free*. This is important in unsupervised problems like outlier detection, in which there is no meaningful way of setting the parameters by testing its performance on the data set. This is because ground-truth is not available for parameter tuning.

3. The features in real data sets are often extracted in such a way that extremes in values expose the outliers, which is an easy case for the Mahalanobis method. If the analyst has an intuitive understanding that extremes in (many of the) feature values are indicative of outliers, then the Mahalanobis method can sometimes be used confidently. Even in cases where all features do not show this characteristic, the natural aggregation effects in the Mahalanobis distance are able to expose the outliers. At the very least, one can leverage the Mahalanobis method as one of the components of an ensemble method (cf. Chapter 6) to exploit the subset of features that are friendly to extreme-value analysis. A simple combination of a nearest-neighbor detector and the Mahalanobis method can perform surprisingly robustly as compared to a variety of other complex detectors. The addition of a distance-based component to an ensemble method also ensures that outliers like data point 'A' in Figure 2.1 are not missed completely.

4. As discussed later in Chapter 4, most distance-based methods require $O(N^2)$ time for a data set containing N points. Even for data sets containing a few hundred thousand points, it often becomes computationally challenging to compute the outlier

[4]Some one-class SVMs [538, 539] learn a circular separator wrapped around the centroid of the data, albeit in a transformed kernel space. As discussed in the next chapter, it is also possible to kernelize the Mahalanobis method because of its PCA-based interpretation. Furthermore, the solutions in the two cases can be shown to be closely related (cf. section 3.4.3).

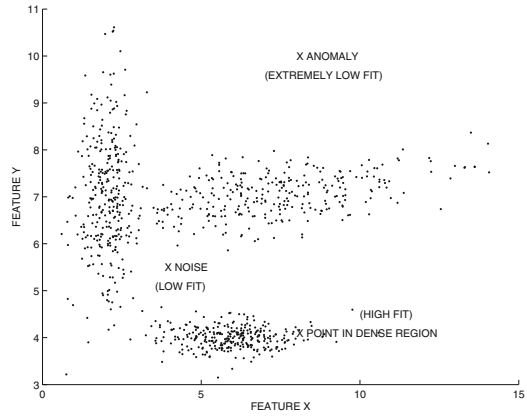


Figure 2.8: Relating fit probabilities to anomalous behavior

scores. On the other hand, the Mahalanobis method is *linear* in the number of data points, although it does require at least quadratic time and space in terms of the data *dimensionality*. Nevertheless, since the number of points is typically orders of magnitude greater than the number of dimensions, the Mahalanobis method has a significant advantage in terms of computational time in most real-world settings.

Therefore, the Mahalanobis method can often be used as an *additive component* of ensemble methods, even when it is not desirable to use it on a stand-alone basis.

2.4 Probabilistic Mixture Modeling for Outlier Analysis

The previous section was focused on the problem of extreme-value analysis for outlier modeling. The simple Mahalanobis method is effective for the example of Figure 2.7, because the entire data set is distributed in one large cluster about the mean. For cases in which the data may have many different clusters with different orientations, such an extreme-value approach may not be effective. An example of such a data set is illustrated in Figure 2.1. For such cases, more general distribution-based modeling algorithms are needed.

The key idea in this generalization is to use probabilistic mixture modeling of the data points. Such models are typically *generative* models, where for each data point, we can estimate the generative probability (or the fit probability) to the model. First, we assume a specific form of the generative model (e.g., a mixture of Gaussians), and then estimate the parameters of this model with the use of the expectation-maximization (EM) algorithm. The parameters are estimated so that the observed data has a *maximum likelihood fit* to the generative model. Given this model, we then estimate the generative probabilities (or fit probabilities) of the underlying data points. Data points that fit the distribution will have high fit probabilities, whereas anomalies will have very low fit probabilities. Some examples of how different types of data points relate to the fit probability in such a model are illustrated in Figure 2.8.

The broad principle of a mixture-based generative model is to *assume* that the data was generated from a mixture of k distributions with the probability distributions $\mathcal{G}_1 \ldots \mathcal{G}_k$ with the repeated application of the following stochastic process:

- Select the rth probability distribution with probability α_r, where $r \in \{1 \ldots k\}$.
- Generate a data point from $\mathcal{G}_r$.

We denote this generative model by $\mathcal{M}$. The value α_r indicates the prior probability, and intuitively represents the fraction of the data generated from mixture component r. We note that the different values of α_r, and the parameters of the different distributions $\mathcal{G}_r$ are not known in advance, and they need to be learned in a data-driven manner. In some simplified settings, the values of the prior probabilities α_r may be fixed to $1/k$, although these values also need to be learned from the observed data in the most general case. The most typical form of the distribution $\mathcal{G}_r$ is the Gaussian distribution. The parameters of the distribution $\mathcal{G}_r$ and the prior probabilities α_r need to be *estimated* from the data, so that the data has the maximum likelihood fit of being generated. Therefore, we first need to define the concept of the fit of the data set to a particular component of the mixture. Let us assume that the density function of $\mathcal{G}_r$ is given by $f^r(\cdot)$. The probability (density function) of the data point $\overline{X_j}$ being generated by the model is given by the following:

$$f^{point}(\overline{X_j}|\mathcal{M}) = \sum_{i=1}^{k} \alpha_i \cdot f^i(\overline{X_j}) \tag{2.22}$$

Then, for a data set $\mathcal{D}$ containing N records denoted by $\overline{X_1} \ldots \overline{X_N}$, the probability of the data set being generated by the model $\mathcal{M}$ is the product of the corresponding individual point-wise probabilities (or probability *densities*):

$$f^{data}(\mathcal{D}|\mathcal{M}) = \prod_{j=1}^{N} f^{point}(\overline{X_j}|\mathcal{M}) \tag{2.23}$$

The log-likelihood fit $\mathcal{L}(\mathcal{D}|\mathcal{M})$ of the data set $\mathcal{D}$ with respect to $\mathcal{M}$ is the logarithm of the aforementioned expression and can be (more conveniently) represented as a sum of values over the different data points.

$$\mathcal{L}(\mathcal{D}|\mathcal{M}) = \log\left[\prod_{j=1}^{N} f^{point}(\overline{X_j}|\mathcal{M})\right] = \sum_{j=1}^{N} \log\left[\sum_{i=1}^{k} \alpha_i \cdot f^i(\overline{X_j})\right] \tag{2.24}$$

This log-likelihood fit needs to be optimized to determine the model parameters, and therefore maximize the fit of the data points to the generative model. The log-likelihood fit is preferable to the likelihood fit because of its additive nature across different data points, and its numerical convenience.

It is noteworthy that it is much easier to determine the optimal model parameters separately for each component of the mixture, if we knew (at least probabilistically), which data point was generated by which component of the mixture. At the same time, the probability of generation of these different data points from different components is dependent on these optimal model parameters. This circularity in dependence naturally suggests an iterative EM-algorithm in which the model parameters and probabilistic data point assignments to components are iteratively refined and estimated from one another. Let Θ be a vector representing the *entire set* of parameters describing all components of the mixture model. For example, in the case of the Gaussian mixture model, Θ would contain all the component mixture means, variances, co-variances, and the parameters $\alpha_1 \ldots \alpha_k$. Then, the EM-algorithm starts off with an initial set of values of Θ (possibly corresponding to random assignments of data points to mixture components), and proceeds as follows:

- **(E-step):** Given current value of the parameters in Θ, determine the *posterior* probability $P(\overline{X_j}|\mathcal{G}_r, \Theta)$ that the point $\overline{X_j}$ was generated by the rth mixture component. This computation is performed for all point-component pairs $(\overline{X_j}, \mathcal{G}_r)$.
- **(M-step):** Given current probabilities of assignments of data points to clusters, use maximum likelihood approach to determine the value of all the parameters Θ, which maximizes the log-likelihood fit on the basis of current assignments. Therefore, in the Gaussian setting, all cluster means, covariance matrices, and prior probabilities $\alpha_1 \ldots \alpha_k$ need to be estimated.

It now remains to explain the details of the E-step and the M-step. The E-step simply computes the probability density of the data point $\overline{X_j}$ being generated by each component of the mixture, and then computes the fractional value for each component. This is defined by the Bayes posterior probability that the data point $\overline{X_j}$ was generated by component r (with model parameters fixed to the current set of the parameters Θ). Therefore, we have:

$$P(\mathcal{G}_r|\overline{X_j}, \Theta) = \frac{\alpha_r \cdot f^{r,\Theta}(\overline{X_j})}{\sum_{i=1}^{k} \alpha_i \cdot f^{i,\Theta}(\overline{X_j})} \tag{2.25}$$

With some abuse of notation, a superscript Θ has been added to the probability density functions in order to denote the fact that they are evaluated at the current set of model parameters Θ.

Next, we describe the parameter estimation of the M-step, which maximizes the likelihood fit. In order to optimize the fit, we need to compute the partial derivative of the log-likelihood fit with respect to corresponding model parameters, and set them to 0 in order to determine the optimal value. The values of α_r are easy to estimate and are equal to the expected fraction of the points assigned to each cluster, based on the current values of $P(\mathcal{G}_r|X_j, \Theta)$. In practice, in order to obtain more robust results for smaller data sets, the expected number of data points belonging to each cluster in the numerator is augmented by 1, and the total number of points in the denominator is $N + k$. Therefore, the estimated value of α_r is $(1 + \sum_{j=1}^{N} P(\mathcal{G}_r|\overline{X_j}, \Theta))/(k + N)$. This approach is a form of regularization, and it is also referred to as *Laplacian smoothing*. For example, if N is extremely small, such an approach pushes the assignment probability towards $1/k$. This represents a natural *prior* assumption about the distribution of points in clusters.

In order to determine the other parameters specific to a particular component r of the mixture, we simply treat each value of $P(\mathcal{G}_r|\overline{X_j}, \Theta)$ as a weight of that data point in that component, and then perform maximum likelihood estimation of the parameters *of that component*. This is generally a much simpler process than having to deal with all components of the mixture at one time. The precise estimation process depends on the probability distribution at hand. For example, consider a setting in which the rth Gaussian mixture component in d dimensions is represented by the following distribution:

$$f^{r,\Theta}(\overline{X_j}) = \frac{1}{\sqrt{|\Sigma_r|} \cdot (2 \cdot \pi)^{(d/2)}} \cdot \exp\left[-\frac{1}{2} \cdot (\overline{X_j} - \overline{\mu_r}) \, \Sigma_r^{-1} \, (\overline{X_j} - \overline{\mu_r})^T\right] \tag{2.26}$$

Here, $\overline{\mu_r}$ is the d-dimensional mean vector and Σ_r is the $d \times d$ co-variance matrix of the generalized Gaussian distribution of the rth component. The value of $|\Sigma_r|$ denotes the determinant of the covariance matrix. When the number of mixture components is large, the non-diagonal entries are often set to 0 in order to reduce the number of estimated parameters. In such cases, the determinant of Σ_r simplifies to the product of the variances along the individual dimensions.

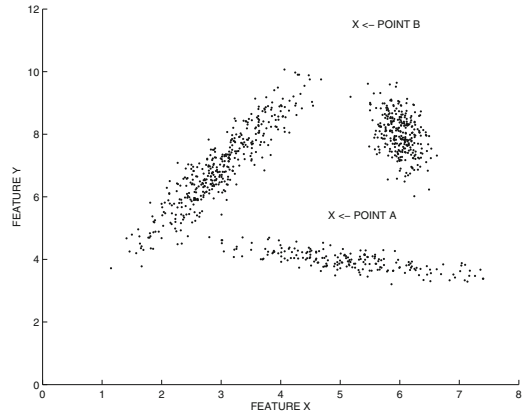


Figure 2.9: EM-Algorithm can determine clusters with arbitrary correlations (Revisiting Figure 2.1)

It can be shown that the maximum-likelihood estimation of $\overline{\mu_r}$ and $[\Sigma_r]_{ij}$ are equal to the (probabilistically weighted) means and co-variances of the data points in that component. Recall that these probabilistic weights were derived from the assignment probabilities in the E-step. Thus, the E-step and the M-step depend on each other and can be probabilistically executed to convergence in order to determine the optimum parameter values Θ.

At the end of the process, we have a probabilistic model that describes the entire data set as the observed output of a generative process. This model also provides a probabilistic fit value for each data point in the form of Equation 2.22. This value provides the outlier score. Thus, we can use this fit in order to rank all the data points, and determine the most anomalous ones. The idea is that points that are far away from the dense regions in the data (such as the one shown in the upper region of Figure 2.8) will have very low fit values. These points are the anomalies in the data. If desired, statistical hypothesis testing can be applied for identification of outliers with unusually low fit values. However, for statistical testing, the logarithm function should be applied to the fit values (i.e., log-likelihood fits should be used) to reduce the relative variance of inliers (large fit values), so that points with very low fit values will pass an extreme-value test.

The approach requires the number of mixture components as an input parameter. In some cases, domain-specific insights about the data can be used to make meaningful choices. In cases where such insights are not available, an ensemble of mixture models with different parameter settings is useful [184]. In particular, the work in [184] averages the point-wise log-likelihood scores obtained on models with different numbers of mixture components. Furthermore, these models are built on different samples of the data set. Excellent results have been reported using this approach.

2.4.1 Relationship with Clustering Methods

Probabilistic mixture modeling is a stochastic version of clustering methods, which can also be used for outlier detection (cf. Chapter 4). It is noteworthy that the fit values in a Gaussian mixture model use the distances of points from cluster centroids in the exponent of the Gaussian. Therefore, the log-likelihood fit of a single Gaussian is the Mahalanobis distance, although the additive fits from multiple Gaussians cannot be simplified in this

manner. Nevertheless, the effect of the nearest cluster is often predominant in the fit values. In the clustering models of Chapter 4, only the distance to the *nearest* cluster centroid is used directly as the outlier score. Therefore, the clustering techniques of Chapter 4 can be viewed as hard versions of the EM-algorithm in which a specific (nearest) cluster is used for scoring the points rather than using the combined fit values from all the clusters in a soft probabilistic combination.

The EM-algorithm can identify arbitrarily oriented clusters in the data, when the clusters have elongated shapes in different directions of correlation. This can help in better identification of outliers. For example, in the case of Figure 2.9, the fit of point 'A' would be lower than that for point 'B,' even though point 'B' is closer to a cluster on the basis of absolute distances. This is because the Mahalanobis distance in the exponent of the Gaussian normalizes for the distances along the different directions of correlation in the data. Indeed, data point 'A' is more obviously an outlier.

2.4.2 The Special Case of a Single Mixture Component

Interestingly, the special case in which the mixture distribution contains a single Gaussian component (cf. Equation 2.26) works surprisingly well in real settings. This is in part because using a single Gaussian component corresponds to the Mahalanobis method of section 2.3.4. The specific merits of this method are discussed in section 2.3.4.1. As we will see in Chapter 3, this leads to a soft version of Principal Component Analysis (PCA), which is known to be effective because of its ability to identify data points that violate interattribute dependencies. The Mahalanobis method can therefore be explained both from the point of view of probabilistic methods and linear models.

Although the use of a single mixture component seems to miss true outliers (like the outlier 'A' of Figure 2.9), it also has the advantage that none of the mixture components can overfit a small but tightly-knit cluster of outliers. When a larger number of mixture components are used, one of the components might correspond to a small tightly knit group of outliers like the outlier cluster illustrated in Figure 2.10. The Mahalanobis method will correctly label the points in this cluster as outliers, whereas a mixture model (with a larger number of components) runs the risk of modeling this small cluster as a legitimate mixture component. Interesting anomalies often occur in small clusters because they might have been caused by similar underlying causes (e.g., a specific disease or type of credit-card fraud). The Mahalanobis method is able to identify such clusters as outliers because they are often inconsistent with the global mean and covariance structure of the data. Because of these typical characteristics of real data sets, very simple methods like the Mahalanobis method sometimes outperform significantly more complex models.

As discussed in section 3.3.8 of the next chapter, one can also combine the Mahalanobis method with kernel methods to model more general distributions. For example, some variations of these methods can correctly identify the outlier 'A' in Figure 2.9. An ensemble-centric version of this approach has been shown to provide high-quality results [35].

2.4.3 Other Ways of Leveraging the EM Model

The EM model discussed in the previous section quantifies the outlier score as the fit of the point to any of the mixture components. Therefore, all mixture components are assumed to be instances of the normal class. A different approach is one in which some domain-specific insights are available about the differences in distribution of the normal classes and the anomalous class. In such a case, different probability distributions are used to model the

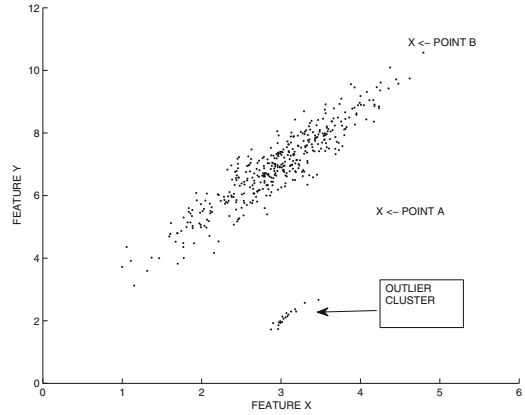


Figure 2.10: The use of a single mixture component is robust to the presence of small outlier clusters

normal and anomalous classes. The outlier score of a point is quantified as its fit to the anomalous class and larger scores are indicative of anomalies. This approach is generally difficult to use in the absence of specific insights about the differences in distribution of the normal and anomalous classes. For example, such an approach has been used for the identification of particular types of outliers such as noise [110]. In the next section, we will provide another example of a setting in which it is possible to model normal and outlier classes with realistic (and different) distributions.

2.4.4 An Application of EM for Converting Scores to Probabilities

Interestingly, EM algorithms can also be used as a final step after many such outlier detection algorithms for converting the scores into probabilities [213]. Note that the fit value returned by the EM algorithm of the previous section (cf. Equation 2.22) is a probability *density* value, and cannot be interpreted as a numerical probability. The ability to characterize an outlier in terms of numerical probabilities is a very useful step for intuition and interpretability.

The idea is that the distribution of the scores can be treated as a univariate data set, which can then be fit to a probabilistic generative model. In this case, the outlier points are explicitly assumed to belong to a component of the mixture model (rather than simply treating them as points with low fit values). Note that one can differentiate the outlier and non-outlier classes in this setting only if some additional insight is available about the natural distributions of the outlier and non-outlier classes. Therefore, different types of distributions can be used to model the outlier and non-outlier components. The work in [213] uses a bimodal mixture of exponential and Gaussian functions. The assumption is that the non-outlier points are distributed according to the exponential distribution, whereas the outlier points are distributed according to the Gaussian distribution. This assumption is made on the basis of the "typical" behavior of outlier scores in real applications. The EM algorithm is used to compute the parameters of each component of the mixture distribution, and the corresponding prior probabilities of assignment. These can be used in order to convert the outlier scores into probabilities with the use of the Bayes rule, since it is now

possible to compute the posterior probability (see Equation 2.25) that the data point belongs to the outlier component. We note that the assignment of a component of the mixture to the outlier class is critical in being able to estimate the probability that a data point is an outlier, which is facilitated by the difference in the distributions (Gaussian versus exponential) of outliers scores in the two classes.

2.5 Limitations of Probabilistic Modeling

Parametric methods are very susceptible to noise and overfitting in the underlying data. Mixture models always assume a specific distribution of the data, and then try to learn the parameters of this distribution. A natural trade-off exists between the generality of this distribution and the number of parameters that need to be learned. If this trade-off is not calibrated carefully, then one of the following two scenarios could occur:

- When the particular assumptions of the model are inaccurate (e.g., inappropriate use of Gaussian distribution), the data is unlikely to fit the model well. As a result, a lot of spurious data points may be reported as outliers.

- When the model is too general, the number of parameters to describe the model increases. For example, when one uses an inappropriately large number of mixture components, a small but tightly-knit outlier cluster may fit one of the mixture components. An example of such a small cluster is illustrated in Figure 2.10. In fact, the technique in the previous section of converting scores to probabilities leverages the fact that univariate outlier scores often cluster together in a Gaussian distribution. Unfortunately, there is no way of generalizing this approach easily to multidimensional data sets. As a result, when reporting points of *low* fit as outliers (rather than a specially modeled outlier class), it is always possible to miss the true outliers as a result of the overfitting caused by small clusters of outliers. One possibility for reducing overfitting is to fix the prior probabilities to $1/k$, although such assumptions might sometimes result in under-fitting.

The proper selection of simplifying assumptions is always tricky. For example, the clusters in the data may be of arbitrary shape or orientation, and may not fit a simplified Gaussian assumption in which the data values along different dimensions are independent of one another. This corresponds to setting the non-diagonal entries of Σ_r to 0 in the Gaussian case. In real data sets, significant correlations may exist among the different dimensions. In such cases, one cannot assume that the matrix Σ_r is diagonal, which would necessitate the learning of $O(d^2)$ parameters for *each cluster*. This can cause overfitting problems when the number of points in the data set is small. On the other hand, efficiency remains a concern in the case of larger data sets, especially if a larger number of parameters are estimated. This is because these methods use the iterative EM algorithm, which needs to scan the entire data step in each iteration of the E- and M-steps. However, these methods are still more efficient than many point-to-point distance-based methods, which require $O(N^2)$ time for a data set containing N points. These methods will be discussed in Chapter 4.

Finally, the issue of *interpretability* remains a concern for many parametric methods. For example, consider the generalized Gaussian model, which tries to learn clusters with non-zero covariances. In such a case, it is difficult to intuitively interpret the clusters with the use of these parameters. Correspondingly, it is also difficult to define simple and intuitive rules that provide critical ideas about the underlying outliers. We note that this issue

may not necessarily be a problem for *all* parametric methods. If the parameters are chosen carefully enough, then the final model can be described simply and intuitively. For example, simplified versions of the Gaussian model without co-variances may sometimes be described simply and intuitively in terms of the original features of the data. On the other hand, such simplifications might cause under-fitting and other qualitative challenges. These trade-offs are, however, endemic to almost all outlier detection methods and not just probabilistic models.

2.6 Conclusions and Summary

In this chapter, a number of fundamental probabilistic and statistical methods for outlier analysis were introduced. Such techniques are very useful for confidence testing and extreme-value analysis. A number of tail inequalities for extreme-value analysis were also introduced. These methods can also be generalized to the multivariate scenario. Extreme-value analysis has immense utility as a final step in converting the scores from many outlier analysis algorithms into binary labels. In many specific applications, such techniques turn out to be very useful even for general outlier analysis. The EM approach for probabilistic mixture modeling of outliers can be viewed as a generalization of the Mahalanobis method. This technique can also be viewed as one of the clustering methods that are commonly used for outlier detection.

2.7 Bibliographic Survey

The classical inequalities (e.g., Markov, Chebychev, Chernoff, and Hoeffding) are widely used in probability and statistics for bounding the accuracy of aggregation-based statistics. A detailed discussion of these different methods may be found in [407]. A generalization of the Hoeffding's inequality is the McDiarmid's inequality [393], which can be applied to a more general function of the different values of X_i (beyond a linearly separable sum). The main restriction on this function is that if the ith argument of the function (i.e., the value of X_i) is changed to any other value, the function cannot change by more than c_i.

The central limit theorem has been studied extensively in probability and statistics [88]. Originally, the theorem was proposed for the case of sums of independent and identically distributed variables. Subsequently, it was extended by Aleksandr Lyapunov to cases where the variables are not necessarily identically distributed [88], but do need to be independent. A weak condition is imposed on these distributions, ensuring that the sum is not dominated by a few of the components. In such a case, the sum of the variables converges to the normal distribution as well. Thus, this is a generalized version of the Central Limit Theorem.

Statistical hypothesis testing has been used widely in the literature in order to determine statistical levels of significance for the tails of distributions [74, 462]. A significant literature exists on hypothesis testing, where the anomalous properties of not just individual data points, but also the collective behavior of groups of data points can be tested. Such techniques are also used in online analytical processing scenarios where the data is organized in the form of data cubes. It is often useful to determine outliers in different portions of a data cube with the use of hypothesis testing [474].

The statistical method for deviation detection with variance reduction was first proposed in [62]. Angle-based methods for extreme-value analysis in multivariate data were proposed in [325]. The multivariate method for extreme-value analysis with the use of the

Mahalanobis distance was proposed in [343, 493]. This technique does not work well when the outliers lie in sparse regions between clusters. A number of depth-based methods have been proposed in [295, 468]. These methods compute the convex hull of a set of data points, and progressively peel off the points at the corners of this hull. The depth of a data point is defined as the order of convex hull that is peeled. These techniques have not found much popularity because they suffer the same drawback as the method of [343] for finding internally located outliers. Furthermore, convex hull computation is extremely expensive with increasing dimensionality. Furthermore, with increasing dimensionality an increasing proportion of the points will lie on the outermost convex hull. Therefore, such methods can only be applied to 2- or 3-dimensional data sets in practice.

It should be noted that the use of probabilistic methods for outlier detection is distinct from the problem of outlier detection in probabilistic or uncertain data [26, 290, 559]. In the former case, the data is uncertain, but the methods are probabilistic. In the latter case, the data itself is probabilistic. The seminal discussion on the EM-algorithm is provided in [164]. This algorithm has a particularly simple form, when the components of the mixture are drawn from the exponential family of distributions. The work in [578] proposed an *online* mixture learning algorithm, which can handle *both* categorical and numerical variables. An interesting variation of the EM-algorithm treats one component of the mixture model specially as an anomaly component [187]. Correspondingly, this component is drawn from a uniform distribution [187], and is also assigned a low a priori probability. Therefore, instead of determining the anomalous points that do not fit any mixture component well, this approach tries to determine the points which fit this special component of the mixture. Such an approach would generally be more effective at modeling noise rather than anomalies, because the special component in the mixture model is likely to model the noise patterns. Finally, a Gaussian mixture model has also been used recently in order to create a global probabilistic model for outlier detection [583].

The EM-algorithm has also been used for clutter removal from data sets [110]. In this case, noise is removed from the data set by modeling the derived data as a mixture of Poisson distributions. We note that the approach in [110] is designed for noise detection, rather than the identification of true anomalies. It was shown in [110] that the improvement in data quality after removal of the clutter (noise) was significant enough to greatly ease the identification of relevant features in the data. The approach of using a special component of the mixture in order to convert the distribution of outlier scores into probabilities has been used in [213]. In addition to the approach discussed in section 2.4.4, a different modeling approach with the use of the logistic sigmoid function is discussed in [213]. Methods for converting outlier scores into probabilities in the supervised scenario have been discussed in [599].

An implicit assumption made by EM methods is that the attributes are conditionally independent of one another once the mixture component has been selected. A probabilistic approach that makes stronger assumptions on attribute interdependence is the *Bayesian Network*. The Bayesian network approach for outlier detection [66] models dependencies among attributes with an off-the-shelf network and uses these dependencies to score points as outliers based on the violations of these dependencies.

2.8 Exercises

1. **[Upper-Tail Chernoff Bound]** The chapter provides a proof sketch of the upper-tail Chernoff bound, but not the full proof. Work out the full proof of bound on the upper

tail using the lower-tail proof as a guide. Where do you use the fact that $\delta < 2 \cdot e - 1$?

2. Suppose you flip an "unbiased" coin 100 times. You would like to investigate whether the coin is showing anomalous behavior (in terms of not being "unbiased" as claimed). Determine the mean and standard deviation of the random variable representing the number of "tails", under the assumption of an unbiased coin. Provide a bound on the probability that you obtain more than 90 tails with the use of the (i) Markov Inequality (ii) Chebychev Inequality (iii) Chernoff Upper Tail Bound, (iv) Chernoff Lower Tail Bound and (v) Hoeffding Inequality. [Hint: Either the upper-tail or lower-tail Chernoff bound can be used, depending on which random variable you look at.]

3. Repeat Exercise 2, when you know that the coin is rigged to show "tails" every eight out of nine flips.

4. Use the central limit theorem to approximate the number of tails by a normal distribution. Use the cumulative normal distribution to approximate the probability that the number of "tails" should be more than 90 for both the cases of Exercises 2 and 3.

5. A manufacturing process produces widgets, each of which is 100 feet long, and has a standard deviation of 1 foot. Under normal operation, these lengths are independent of one another.

- Use the normal distribution assumption to compute the probability that something anomalous is going on in the manufacturing process, if a sampled widget is 101.2 feet long?
- How would your answer change, if the sampled widget was 96.3 feet long?

6. In the example above, consider the case where 10,000 widgets from the assembly line were sampled, and found to have an average length of 100.05. What is the probability that something anomalous is going on in the manufacturing process?

7. Use MATLAB or any other mathematical software to plot the t-distribution with 100 degrees of freedom. Superimpose a standard normal distribution on this plot. Can you visually see the difference? What does this tell you?

8. Work out the steps of the EM-algorithm, when all non-diagonal elements of the covariance matrix Σ are set to zero, and each diagonal element in a given component has the same value. Furthermore, the prior probabilities of assignment are equal to $1/k$, where k is the number of mixture components. Now perform the following modifications:

- Change the E-step, so that each data point is deterministically assigned to the cluster with the highest probability (hard assignment), rather than a soft probabilistic assignment. Under what distance-based conditions does a data point get assigned to a cluster?
- How does this algorithm relate to the k-means algorithm?
- How would your answers change, if all components were constrained to have the same cluster variance?

9. Using the insights gained from Exercise 8, work out how the EM-algorithm with a Gaussian mixture model with a complete set of covariance matrices Σ_r, and a fixed set of priors, relates to a generalized k-means algorithm. [Hint: Consider the concept of Mahalanobis distance computations for assignments in k-means. How should the prior probabilities be defined?]

10. Download the KDD Cup 1999 data set from the UCI Machine Learning Repository [203]. Extract the quantitative attributes from the data set. Apply the EM-algorithm with 20 mixture components, when non-diagonal elements are set to 0.

- Determine the fit of each data point to the learned distribution. Determine the top 10 points with the least fit. Do these data points correspond to intrusion attacks or normal data?
- Repeat the process while allowing non-zero non-diagonal elements. How does your answer change?
- Randomly sample 990 points from the data set, and then add the 10 points found in the first case above. Repeat the procedure on this smaller data set. Do you find significant anomalies in terms of fit probabilities? Do the lowest fit probabilities correspond to the same data points as in the first case above?
- Repeat the same procedure with the second case above.

11. Repeat the first two portions of Exercise 10 on the Ionosphere data set from the UCI Machine Learning Repository. Note that the Ionosphere data set has much higher dimensionality (of quantitative attributes) and smaller number of records. Do you determine the same top-10 anomalies in the two cases? What are the absolute fit probabilities? What does this tell you about applying such algorithms to small and high dimensional data sets?

12. Let Z be a random variable satisfying $E[Z] = 0$, and $Z \in [a, b]$.

- Show that $E[e^{t \cdot Z}] \leq e^{t^2 \cdot (b-a)^2/8}$.
- Use the aforementioned result to complete the proof of the Hoeffding inequality.