



TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

Pekka Kumpulainen
**Anomaly Detection for Communication Network
Monitoring Applications**



Julkaisu 1192 • Publication 1192

Tampereen teknillinen yliopisto. Julkaisu 1192
Tampere University of Technology. Publication 1192

Pekka Kumpulainen

Anomaly Detection for Communication Network Monitoring Applications

Thesis for the degree of Doctor of Science in Technology to be presented with due permission for public examination and criticism in Festia Building, Auditorium Pieni Sali 1, at Tampere University of Technology, on the 14th of March 2014, at 12 noon.

Tampereen teknillinen yliopisto - Tampere University of Technology
Tampere 2014

ISBN 978-952-15-3228-3 (printed)
ISBN 978-952-15-3266-5 (PDF)
ISSN 1459-2045

Abstract

Functioning mobile telecommunication networks are taken for granted in present-day society. The network operator's objective is to optimise the network's capabilities in order to provide fluent connections for subscribers. Network management is based on the huge amounts of data that are recorded from all parts of the network. The data is used to monitor performance, to detect problems and also to provide novel knowledge to be used in future planning. Anomalous events in the network provide a valuable source of information for network management. This thesis presents an interpretation of anomalies and the basic theory of how to detect them when the probability distribution is known. However, since in real life applications the probability distribution is not known, the main focus is on methods that are based on distances.

This thesis proposes procedures for anomaly detection and for summarising the information obtained about the anomalies. The procedures utilise clustering in both the anomaly detection and the further analysis of the anomalies. Scaling of variables affects the distances and the results of clustering. Therefore, methods to incorporate expert knowledge by application specific scaling of variables are presented.

The proposed procedures are exemplified in three use cases. The cases present practical problems from distinct parts of the network; the radio interface between the mobile device and the network, the system logs from the operator's servers, and the traffic through the cells. Each case presents unique characteristics and challenges. The problems are solved utilising the proposed procedures. Two novel anomaly detection methods developed in this thesis are applied in the second case, where anomaly detection is applied to server logs.

All use cases use real data from commercial networks where the ground truth does not exist. Therefore, precise comparisons of the methods are impossible. The results have been verified with network experts and found to be informative and useful.



Preface

The preparation of this doctoral thesis has been a long process, and its completion brings a great sense of relief. I am deeply grateful to my supervisor Prof. Risto Ritala for his patience throughout these years. His guidance and his positive kicking forward have been essential. I also wish to thank Dr. Tuomo Kauranne and Prof. Dominic Palmer-Brown for their valuable comments in the pre-examination phase.

Most of the research was carried out in projects with Nokia Networks, and later Nokia Siemens Networks. I wish to express my special gratitude to Dr. Kimmo Hätönen for his support and for sharing many painful writing experiences. I also thank M.Sc Mikko Kylväjä and M.Sc. Mika Särkioja for their very productive co-operation.

M.Sc Heimo Ihalainen and Dr. Pekko Vehviläinen provided valuable assistance in the early phase of this process. I am grateful to all TUT/MIT, and later TUT/ASE staff for a pleasant working environment, and for constantly reminding me that the thesis was not yet ready yet, just in case I forgot.

Finally, I am deeply grateful to my family; my wife, Tea, and our children Milla, Henna, and Kasper, for their patience during this long process. Now, all of a sudden, the children have grown up, and I have a wonderful grandson, Into, who has cheered me up immensely during the past eighteen months.



Table of Contents

- Chapter 1: Introduction..... 1**
 - 1.1 Research problem 2
 - 1.1.1 Application area: mobile telecommunication network management 2
 - 1.1.2 Anomaly detection 3
 - 1.1.3 Applications..... 4
 - 1.2 Limitations 5
 - 1.3 Objective and contribution..... 6
 - 1.4 Outline 9
- Chapter 2: Application domain: telecommunication network 11**
 - 2.1 Mobile telecommunication networks..... 11
 - 2.1.1 GSM 12
 - 2.1.2 UMTS 13
 - 2.2 Network management 14
 - 2.3 Characteristics of the data..... 15
- Chapter 3: Anomalies and outliers 19**
 - 3.1 What are anomalies and outliers? 19
 - 3.2 Prior probability of observations 20
 - 3.3 Dissimilarity to normal state without distribution 21
 - 3.4 Detecting anomalies..... 21
 - 3.4.1 Univariate anomalies 22
 - 3.4.2 Multivariate anomalies 24
 - 3.4.3 Multimodal distributions (multiple operational states) 27
 - 3.5 Distance measures..... 29
 - 3.5.1 Distance metrics 29
 - 3.5.2 Scaling 31
 - 3.5.3 Scale invariant metrics 32
 - 3.5.4 Robust scaling 35
 - 3.5.5 Nonlinear transformations..... 37
 - 3.6 Types of anomaly..... 37
 - 3.7 Sources of outliers 39
- Chapter 4: Anomaly detection methods..... 41**
 - 4.1 Application areas of anomaly detection..... 41
 - 4.2 Categories by level of supervision..... 43
 - 4.2.1 Supervised 43
 - 4.2.2 Semi-supervised 43
 - 4.2.3 Unsupervised 44
 - 4.3 Global and local anomaly detection..... 44
 - 4.4 Categorisation by methodology 45
 - 4.4.1 Statistical 46
 - 4.4.2 Model- and distribution-based..... 49
 - 4.4.3 Depth-based..... 52
 - 4.4.4 Distance-based..... 52
 - 4.4.5 Clustering-based..... 53
 - 4.4.6 Density-based 57
 - 4.4.7 Classification-based and supervised neural networks 57

4.4.8 Unsupervised neural networks	58
4.4.9 Visual.....	67
4.4.10 Projection pursuit	68
4.4.11 Hybrid methods	68
4.5 Results produced by the AD methods.....	69
4.6 Requirements in real life applications	70
4.7 Assessing the result.....	72
Chapter 5: <i>A priori</i> knowledge in scaling for distance based anomalies	77
5.1 Objective	77
5.2 Use case	78
5.2.1 Data	79
5.2.2 Scaling	80
5.2.3 Selecting the potential problems (distance based anomalies)	83
5.2.4 Clustering the problem cells.....	83
5.2.5 Alternative views on problems.....	85
5.2.6 History of the cells	87
5.3 Comparison using standardised data.....	88
5.4 Parameter sensitivity	92
5.5 Discussion.....	93
Chapter 6: Local anomalies in network management	95
6.1 Objective	95
6.2 Use case	96
6.2.1 Data	97
6.2.2 Scaling	97
6.2.3 Model identification	99
6.2.4 Offline usage: analysing the reference data	100
6.2.5 Online usage: analysing new data	105
6.3 Comparisons with other methods	108
6.4 Parameter sensitivity	110
6.5 Discussion.....	113
Chapter 7: Daily traffic patterns	115
7.1 Objectives	115
7.2 Use cases.....	116
7.2.1 Data compression	116
7.2.2 Exploratory analysis of daily behaviour.....	118
7.3 Daily pattern data and preprocessing.....	118
7.3.1 Scaling	119
7.3.2 Cleaning by removing the most obvious anomalies.....	120
7.4 Application of data compression	123
7.4.1 Identification of the compression model	123
7.4.2 Compression and storage.....	124
7.4.3 Uncompression.....	128
7.4.4 Parameter sensitivity	128
7.5 Exploratory analysis of daily behaviour	131
7.5.1 Visualisation of the main characteristics.....	132
7.5.2 Behaviour profiles	134
7.5.3 Visualisation of anomalies	136
7.6 Discussion.....	140
Chapter 8: Conclusions.....	143
8.1 Scientific novelty and significance	143

8.2 Relevance of results	144
8.3 Discussion and future work	146
References	149

List of Abbreviations

AD	Anomaly Detection
BSS	Base Station Subsystem
GMM	Gaussian Mixture Model
GSM	Groupe Spécial Mobile Global System for Mobile communications
JBGE	Just Barely Good Enough
LOF	Local Outlier Factor
MCD	Minimum Covariance Determinant
MS	Mobile Station
MVE	Minimum Volume Ellipsoid
NE	Network Element
NSS	Network and Switching Subsystem
OC-SVM	One-Class Support Vector Machine
OSS	Operations SubSystem
PCA	Principal Component Analysis
pdf	Probability Density Function
ROC	Receiver Operating Characteristic
SOM	Self Organising Map
SPC	Statistical Process Control
SVM	Support Vector Machine
TMN	Telecommunications Management Network
UMTS	Universal Mobile Telecommunication System



Chapter 1: Introduction

Anomaly detection (AD) is one of the four core tasks in data mining [Tan et al. 2005]. It is also an essential part of process monitoring in many fields of industry. Even where automation systems control processes, there are situations where human expertise is required in making decisions. These situations arise in daily online control and in particular in offline analysis targeted at optimising and improving the process. The amount of data produced by industrial applications is increasing all the time, and it is impossible for process operators to browse all the data manually. Automatic applications are required to find the most essential information to support operators in their decision-making. Anomalies in the data are one of the main sources of such information. Automatic AD applications can be regarded as tools which filter out the vast majority of the data reflecting the normal behaviour of the process, and expose only the most interesting parts to the end user.

Anomalies in the data can be signs of errors or malfunctions in the process, including errors in measurement devices and data transfer or storage (outlier detection, error removal). They can also originate from attempts at unauthorised usage of the system (intrusion detection, fraud detection). In addition, detecting rare or exceptional parts of the data can reveal new and valuable information from the system for further optimisation (novelty detection): *An apparently wild (or otherwise anomalous) observation is a signal that says: "Here is something from which we may learn a lesson, perhaps of a kind not anticipated beforehand, and perhaps more important than the main object of the study"* [Kruskal 1960, p. 1]. Rather than rejecting or ignoring anomalous observations, they should be studied, for they may contain unexpected relevant information [Maronna et al. 2006]. An example of a case where novel information was ignored because it did not match the assumed models is given in [Kandel 1992, p. 110]:

The discovery of the ozone hole was announced in 1985 by a British team working on the ground with "conventional" instruments and examining its observations in detail.

Only later, after reexamining the data transmitted by the TOMS instrument on NASA's Nimbus 7 satellite, was it found that the hole had been forming for several years. Why had nobody noticed it? The reason was simple: the systems processing the TOMS data, designed in accordance with predictions derived from models, which in turn were established on the basis of what was thought to be "reasonable", had rejected the very ("excessively") low values observed above the Antarctic during the Southern spring: as far as the program was concerned, there must have been an operating defect in the instrument. Although researchers were looking for - and were measuring - a generally decreasing trend in ozone levels, they were not prepared to accept something that had not been predicted in the models. If Nature has other such surprises in store for us, will we be able to recognize them in time?

In this thesis the main objective is to provide procedures for finding anomalous observations and their original causes. The proposed procedures consist of anomaly detection methods and the summarisation of results.

1.1 Research problem

The aim of this work is to present an overview of anomaly detection (AD) methods and to highlight issues that affect the results in industrial applications; scaling and weighting of variables in particular. The AD methods themselves are not restricted to specific application areas. However, each application will benefit if process knowledge can be utilised in method selection, fine tuning, scaling and weighting. This thesis concentrates on applications in mobile telecommunication network management, and gives examples of the utilisation of the characteristics of data in AD applications.

1.1.1 Application area: mobile telecommunication network management

Mobile telecommunication networks have become an inevitable part of everyday life. Expectations concerning the reliability and quality of the mobile network are equal, if not higher, than those related to traditional telephone networks. The keys to the reliability, dependability and quality of the telecommunication network are the management and operations of the network [Subramanian 2000]. The radio interface between mobile devices and the network introduces further challenges to network management [Mouly & Pautet 1992]. 3G networks move the behaviour and the management towards that of computer networks [Kaarainen et al. 2001].

The purpose of network management is to optimise a telecommunication network's operational capabilities [Freeman 2004]. This includes keeping the network operating at peak performance, informing the operator of impending deterioration, and tools for finding the causes of performance deterioration.

As the anomalies are often signs of undesired activities, operators are not willing to publish much of the work in this area. Therefore, the number of published results on anomaly detection in mobile network management is somewhat limited [Anisetti et al. 2008]. However, several contributions to the detection of anomalies or problem states in mobile telecommunications networks have been published [Höglund et al. 2000; Vehviläinen 2004; Kylväjä et al. 2005; Laiho et al. 2005; Barreto et al. 2005; Kumpulainen & Hätönen 2007; Kumpulainen & Hätönen 2008c; Anisetti et al. 2008; Hätönen 2009; Rajala 2009; Kumpulainen et al. 2009].

This thesis concentrates on applications for mobile network management. However, the methods can be applied to anomaly detection in any process that produces multivariate data.

1.1.2 Anomaly detection

Anomaly detection is a term used for activities aimed at finding patterns or observations in data that are not considered to present the normal behaviour of the process under study. These abnormal observations are described as exceptions, contaminants or, most often, anomalies or outliers. The last two terms are typically interchangeable. In this thesis *anomaly* is preferred but *outlier* is considered to have an identical meaning.

Anomaly detection has been utilised in a wide range of application domains. These include network intrusion detection [Lazarevic et al. 2003; Zhang & Zulkernine 2006], fraud detection [Fawcett & Provost 1997; Hollmen & Tresp 1998; Bolton & Hand 2002; Kou et al. 2004], and fault diagnostics [Jiang & Papavassiliou 2003; Fujimaki 2008]. Mobile network management is the main application area in this thesis [Höglund et al. 2000; Kylväjä et al. 2005; Kumpulainen & Hätönen 2007; Anisetti et al. 2008]. A relatively extensive list of application areas utilising outlier detection is given, for example in Hodge & Austin [2004] and Chandola et al. [2009].

There are several ways to categorise anomaly detection methods. Categorisations can be based on the techniques applied. Agyeman et al. [2006] present a division into distribution-based, depth-based, and distance-based methods with an additional category of clustering-based techniques. Another categorisation consists of model-based, proximity-based and density-based techniques [Tan et al. 2005].

A common three category division is based on the characteristics of the data available for identification of the required models [Hodge & Austin 2004; Tan et al. 2005; Chandola et al. 2009]. This categorisation is the most general and all the methods can be assigned into one of the categories of *supervised*, *semi-supervised* or *unsupervised* methods.

In order to be able to decide whether an observation *deviates* or *significantly differs* from normal, proximity or distance metrics are required [Agyemang et al. 2006]. This applies to most of the methods in all three categories, including supervised classification. The scales of the variables have an essential influence on the distance metrics [Duda et al. 2001].

In addition to distance metrics, a method for ordering observations is required for multivariate data. Ordering enables the comparison of the levels of deviation from the normal. The AD methods applied or developed in this thesis produce univariate anomaly measures from multivariate data. These measures can be used to create a ranking list of the potentially anomalous observations for the end user.

1.1.3 Applications

It is usually very difficult to obtain reference data with labelled anomalies from real industrial processes. Therefore unsupervised methods are most often chosen for real life applications. Separate training data is not required for unsupervised methods in general [Chandola et al. 2009]. However, a typical anomaly detection application consists of two phases: *training* and *testing* or *detection* [Kruegel & Vigna 2003, Patcha & Park 2007]. In the *training* phase a model for the normal states of the process is identified from the history data. The *testing* or *detection* phase consists of the normal operation. Possible anomalies are detected by comparing the new data received from the process to the identified model. While the number of anomalies can be rela-

tively high, the summarisation of the results is requested in some applications [Kylväjä et al. 2005; Lakhina et al. 2005; Rocco & Zio 2007].

Real life industrial applications have to be robust; not too sensitive to the parameters of the methods or the characteristics of the data [Hätönen 2009]. A suitable model for each purpose should be selected. Optimal or best solutions are not always required, instead sometimes less is good enough, as Nisbet et al. [2009] describe when introducing the concept of JBGE applications (Just Barely Good Enough). Simplicity is also preferred by Vapnik's principle: "*try to solve the problem directly and never solve a more general problem as an intermediate step*" [Vapnik 1998, p. 12].

The application domain and the end users specify the goals of anomaly detection applications and preferred result types may vary among the use cases within one single application area. The results are affected by a number of choices that have to be made based on expertise or history data. These choices include, for example, ways of pre-processing (feature extraction, scaling, feature weighting, sample selection), the AD method, and identification or selection of the parameters.

An additional challenge in real life AD applications is that the ground truth is not known. The end users, experts in the application domain, have to assess the results produced after trying a variety of the choices presented above. The final choices are based on their knowledge. It is impossible to tell beforehand what kind of results are preferred, as Gondek & Hofmann [2005, p 70] put it. "*users are often unable to positively describe what they are looking for, yet may be perfectly capable of expressing what is not of interest to them*". Therefore it is essential to provide them robust, easy to use tools to try out a variety of solutions. Such tools enable experts to select the methods that suit the requirements of the problem at hand in their application environment.

1.2 Limitations

Precise comparisons of the methods are not included in this thesis. Instead, the effort is targeted at introducing and describing the characteristics of the methods, in order to provide some guidelines for the selection of methods in various use cases. Supervised AD methods are not covered in this thesis. Those methods are essentially re-

duced to classification problems, in particular to classification for rare classes [Tan et al. 2005].

The following are the prerequisites for the methods that constitute the contribution of this thesis:

- as the AD model parameters are identified from reference (history) data, a sufficient amount of data must be available
- a set of the most severe anomalies is the desired result
- unsupervised or semi-supervised detection is assumed and hence no labelled reference data set is required to be available, but prior knowledge of the normal state may exist
- distance measures are used, therefore the measurements must be on an interval (or ratio) scale

Comparing the results produced by AD methods, as well as the effects of the parameters and pre-processing procedures, is challenging. In supervised AD methods (classification), receiver operating characteristic (ROC) or cost-based analysis can be used for unambiguous ranking of the results [Lippmann et al. 2000; Stolfo et al. 2000]. In the case of unsupervised or semi-supervised AD from multivariate data the comparisons are far more complicated. As there is no unambiguous ordering in multivariate space [Barnett 1976], the ordering of the most severe anomalies can be disputed even in simple artificial examples, in particular if the variables are not on a comparable scale and their importance depends on the actual meaning they have in the application domain. Unambiguous ranking of the anomalies in real life data is practically impossible, as the ground truth does not exist. The final decisions have to be left to end users.

1.3 Objective and contribution

The general objective of this thesis is to provide procedures which support mobile network operators in everyday decision-making. Questions that are often asked by operators include: “What has happened in my process recently? Is there something I should take a closer look at?” The procedures and methods presented in this thesis help in developing applications that provide answers to these questions. The applica-

tions consist of anomaly detection, including summarisation of the results, intended to help in uncovering the causes of anomalies, and in finding corrective actions.

The specific objective of anomaly detection is to provide importance ordering of the observations. Thus, instead of strict classification whether the observations are anomalies or not, the user receives a list of possible problems which are ranked. The user can then investigate the most anomalous observations first, as they are most likely to represent the most severe problems in the network. The judgement of the importance of the anomalies requires specific domain knowledge about the network and is left to the user. When the user judges a possible anomalous observation as not severe enough to require actions, then all observations that are ranked as less anomalous do not require attention and they can be disregarded. Therefore, an exact classification into anomalies and normal data is not necessary.

Another objective is to summarise information of both, the normal state and the detected anomalies by grouping the similar observations together. This reduces the workload of the operator in investigation of the anomalies, eliminating the need to browse through all individual observations.

The main contributions of this thesis consist of a methodological part and applications of these methods in mobile network monitoring. This thesis provides procedures for utilising expert knowledge in anomaly detection, and for summarising the information obtained about the detected anomalies. The second part concentrates on applications, and provides examples of how to use the procedures.

The methodological contribution consists of:

1. An examination of how scaling of variables affects clustering for anomaly detection and analysis.
2. Methods to incorporate expert knowledge in application specific scaling of variables.
3. Clustering for both anomaly detection and for further analysis of the anomalies.

The use cases that exemplify the procedures are for mobile network monitoring. In particular, three practical problems are addressed and solved:

1. Anomaly detection based on performance variables of the radio interface in the mobile network. The main characteristics of this problem are:
 - known normal state
 - expert knowledge available for scaling
 - clustering of the anomalies required
2. Anomalies in server logs. The main characteristics of this problem are:
 - normal states identified by clustering
 - application specific scaling without expert knowledge
 - local anomaly thresholds
3. Anomaly detection in daily behaviour of traffic data through cells. The main characteristics of this problem are:
 - daily traffic activity per hour constitutes 24-dimensional seasonal data vectors
 - normal states identified by clustering
 - application specific scaling without expert knowledge
 - traffic-dependent anomaly thresholds

Three use cases and examples of corresponding applications are presented. The specific characteristics of the data and suitable scaling methods in each case are presented. The results of the applications are studied. The use cases in this thesis are extended from previous publications, which are referred to in the following.

The first example is a special case of semi-supervised anomaly detection, where the optimal state of the process is known and is assumed to present the normal state for AD [Kylväjä et al. 2005; Kumpulainen et al. 2009]. The performance variables of the radio interface in the mobile network are known to have ideal values at one end of the value range, and unacceptable values at the other end. A simple distance based method is used, with prior expert knowledge incorporated through scaling of the variables. The results are compared to those using conventional normalisation.

The second case concentrates on unsupervised AD multimodal non normal distributions [Kumpulainen & Hättönen 2007; Kumpulainen & Hättönen 2008a; Kumpulainen & Hättönen 2008c]. Local AD methods are applied to server system logs. Clustering and a combination of SOM and clustering are compared to the Gaussian mixture mod-

el and the one-class support vector machine. Adaptive anomaly thresholds are introduced that can be used together with all the applied AD methods.

The last case involves unsupervised AD applied to patterns of time series data [Kumpulainen & Hätönen 2008b; Kumpulainen & Hätönen 2012]. Clustering and dynamic traffic dependent anomaly thresholds are used to detect anomalies in daily patterns of network traffic. This has also been applied to data compression (US Pat. 7,461,037 B2).

The basic ideas in all the publications above have been collaborately constructed by the author and the co-authors of each publication. The author has been responsible for the anomaly detection part, for testing and fine tuning the algorithms, and for producing all the software implementation, including the application prototypes.

1.4 Outline

Chapter 2 gives an overview of mobile network management, the application domain of the industrial examples used in this thesis. Chapter 3 introduces the concept of anomalies or outliers and presents various types of anomalies. A wide selection of detection methods are based on distance measures. Therefore a number of commonly used distance metrics and how they are affected by scaling of the variables are included. Existing anomaly detection methods and various ways to categorise them are discussed in Chapter 4. Some commonly used anomaly detection methods are presented in more detail, including modifications developed in this thesis for mobile network management. The subsequent chapters introduce the three use cases, using real life data. Chapter 5 presents distance based anomaly detection utilising expert knowledge in scaling. The data consists of radio interface performance measurements from a mobile network. The use case in Chapter 6 presents anomaly and novelty detection from data collected from a mobile network operator's server logs. The third use case in Chapter 7 applies anomaly detection to daily traffic data from cells in the mobile network. Conclusions are given in Chapter 8.

Chapter 2: Application domain: telecommunication network

Mobile telecommunication networks have become an inevitable part of everyday life. A high level of reliability and quality in mobile network services is expected. The keys to the reliability, dependability and quality of the telecommunication network are the management and operations of the network [Subramanian 2000]. This chapter gives a brief summary of the history of mobile networks. The main structures of the current digital mobile networks are presented, as well as the key features of the data and the requirements of network monitoring.

2.1 Mobile telecommunication networks

The first mobile telephone service started in St. Louis in 1946, and a few years later in Europe [Mouly & Pautet 1992]. The first mobile networks were manually operated and the terminals were heavy and expensive. The service was restricted to the area covered by a single emission and reception site. The available frequency spectrum was limited and the capacity of the network was soon saturated.

Capacity was increased in the cellular networks introduced in the 1970's. They consisted of several cells that covered relatively small, partially overlapping areas. The same frequencies could be used in cells that were far enough from each other, which enabled a huge gain in capacity. The first cellular network, the AMPS (Advanced Mobile Phone Service), started in the US in 1979. The Nordic Mobile Telephone, NMT, started in Sweden in 1981, and soon after that in the other Scandinavian countries. The TACS system was derived from the AMPS and started in the UK in 1985 [Mouly & Pautet 1992].

All these were based on analogue transmission. The coverage was typically nationwide and, due to different systems, mobile devices could not be used in other net-

works. The demand for mobile services exceeded the estimated capacity of the analogue networks. Several countries in Europe decided in cooperation to create a new system that would be common to, and accessible in, all countries.

2.1.1 GSM

The development of standards for the common European digital mobile network started in 1982, when the *Groupe Spécial Mobile* (GSM) had its first meeting. The actual specification work started in 1987, and the specifications were frozen in 1991. All the major European GSM operators started commercial operations in 1992. The term GSM was chosen as the commercial trademark, standing for the *Global System for Mobile communications* [Mouly & Pautet 1992].

The structure of the GSM network is depicted in *Figure 2.1*, which is modified from Hätönen [2009]. The main components of the network are the *Network and Switching Subsystem* (NSS), the *Base Station Subsystem* (BSS) and the *Operations SubSystem* (OSS) [Mouly & Pautet 1992]. Each subsystem contains a variety of *Network Elements* (NE). Mobile Stations (MS) are connected to the network through a radio interface. The BSS and NSS provide the functionality of the network, and the OSS provides tools for the operator to manage and control the network.

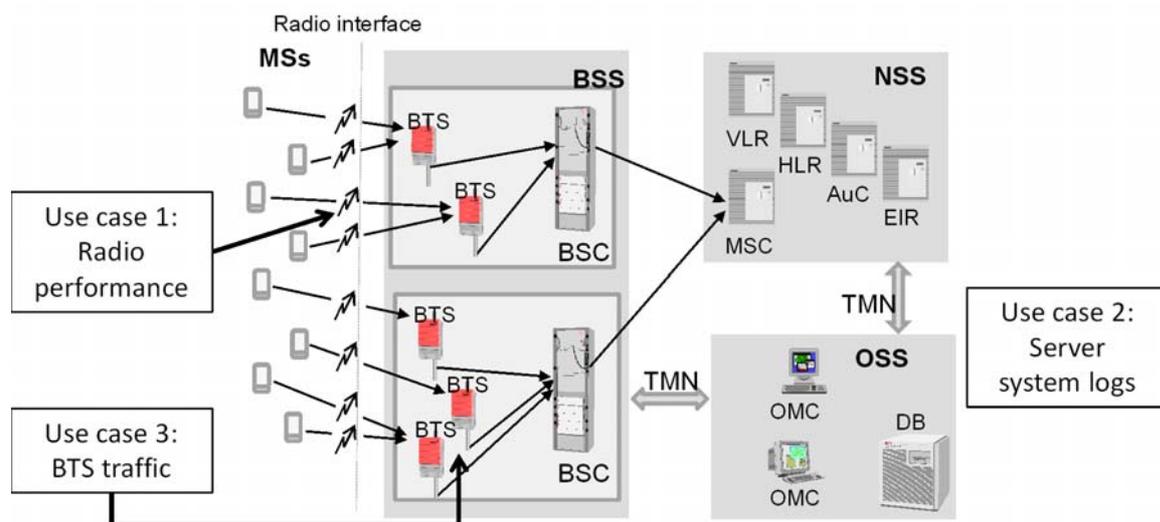


Figure 2.1 Subsystems of the GSM network architecture.

The BSS consists of several *Base Station Controllers* (BSC). Each BSC controls a group of *Base Transceiver Stations* (BTS), which have one or more transmitter-receiver pairs (TRX). A cell is an area covered by one TRX [Vehviläinen 2004]. They provide the radio interface for the MSs. The BSCs are connected to the *Mobile Switching Centre* (MSC), which is the key element of the NSS. Its main tasks are to coordinate the setting-up of calls and to connect the traffic between BSCs. The MSC also works as a bridge between the GSM network and the public switched telephone network. This function is becoming obsolete, as usage of the traditional telephone network has reduced dramatically. Other elements of the NSS are the *Home Location Register* (HLR), the *Visitor Location Register* (VLR), the *Authentication Centre* (AuC) and the *Equipment Identity Register* (EIR).

The main tasks of the OSS include maintaining and operating the network and managing subscriber information. The key element of the OSS is the *Operations and Maintenance Centre* (OMC). The main tasks of the OMC include setup and modification of the parameters of network elements, monitoring of the elements, and software installation. The main tool in the OMC is the *Network Management System* (NMS), a software system which enables operator personnel to monitor and access network elements. The OSS is connected to the NSS and the BSS via a *Telecommunications Management Network* (TMN). The TMN provides management functions and communications between the OSS and other parts of the network [ITU-T 2000]. The measurement data from the network elements are transferred through the TMN, and stored in a database.

2.1.2 UMTS

Work towards creating the next generation network began in 1991, even before GSM networks started. The post-GSM 3G networks are called the UMTS (Universal Mobile Telecommunication System) [Mouly & Pautet 1992]. The UMTS network consists of two main components, the *Universal Terrestrial Radio Access Network* (UTRAN) and the Core Network (CN). The structure and network elements of an UMTS network are depicted in *Figure 2.2* [Kaarainen et al. 2001].

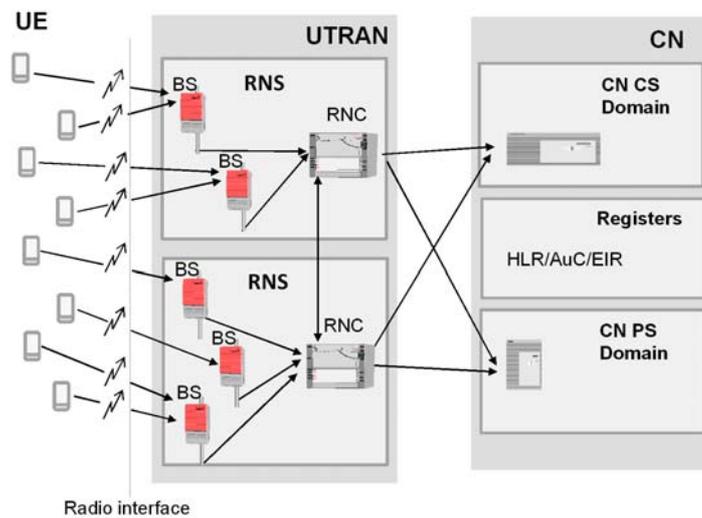


Figure 2.2 Structure and subsystems of UMTS network.

The UTRAN provides the radio interface for the *User Equipment* (UE). The UTRAN is divided into *Radio Network Subsystems* (RNS). Each RNS contains a *Radio Network Controller* (RNC), which controls a group of *Base Stations* (BS). The RNC corresponds to the BSC in the GSM network, but RNCs can also be connected to each other.

The CN contains registers similar to those in the NSS in the GSM network. Two domains, Circuit Switched (CN) and Packet Switched (PS) are shown here, but the network may contain other domains, for example the broadcast messaging domain [Kaarainen et al. 2001].

The structure and the technologies used in the UMTS networks differ from those of the GSM. Nevertheless, they are very similar in many ways. Even though the radio interfaces use different protocols, they share similar requirements and restrictions. The network elements are monitored using the NMS and the databases are used to analyse the functionality of the network.

2.2 Network management

The purpose of network management is to optimise a telecommunication network's operational capabilities [Freeman 2004]. This includes keeping the network operating at peak performance, informing the operator of impending deterioration, and tools

with which to find the causes of performance deterioration. The TMN provides management functions and communications between the OSS and other parts of the network [ITU-T 2000]. The fundamental elements of the TMN physical architecture are physical blocks and physical interfaces. The data communication network is used to transfer information between the TMN and the NEs, for example, measurement data from the NEs, and configuration parameters to the NEs. The TMN contains five management functional areas [ITU-T 2000; Freeman 2004]:

- performance management
- fault management
- configuration management
- accounting management
- security management

Anomaly detection can be utilised in all five management areas. Anomaly detection methods are widely applied in the areas of performance, fault and security management [Höglund et al. 2000; Kylväjä et al. 2005; Kumpulainen & Hätönen 2007; Anisetti et al. 2008; Hätönen 2009]. Configuration management also benefits offline data analysis, including anomaly and novelty detection [Laiho et al. 2005; Barreto et al. 2005]. The detected anomalies can be caused by inferior performance or faults in hardware, software or configuration. Intrusion and frauds will most likely occur as anomalous behaviour in accounting and security management. All these events should be detected; the causes identified and fixed.

2.3 Characteristics of the data

Usage of the data collected from the mobile telecommunications network is two-fold [Hätönen 2009]. The first is to support operational decisions and control. The second use is to accumulate knowledge of the application domain. Anomaly detection methods play an important role in both. Detecting abnormal behaviour is essential in network monitoring to ensure sufficient quality for end users. The information learned from the anomalies, on the other hand, is very helpful in accumulating knowledge of the novel behaviour of the network.

All elementary events occurring in various NEs during the operation of the network are counted, forming the raw low level data called *counters*. Suitable time frames are used in the counting for various management purposes: 15 minutes, one hour or one day. The huge number of counters makes them impractical to use directly. They are aggregated by calculating higher level variables called *Key Performance Indicators* (KPI) [Suutarinen 1994]. KPIs are formed in order to present understandable and easy to interpret functional factors, and they have a descriptive name. The actual formulas for calculating KPIs are confidential, and generally not publicly available. KPIs with the same name are not directly comparable across networks, or even within one operator's network, since for the most common KPIs there are several alternative formulas available to choose from.

The servers inside the OSS are another important source of data. They write log files of specific conditions or events occurring within the system, for example users logging in or out of the system, or applications starting [Hätönen 2009]. The entries in the log files are aggregated by counting them in specific categories over suitable periods of time. The aggregated counters can then be used in various monitoring and analysis tasks, for example in anomaly detection for possible security risks [Höglund et al. 2000; Kumpulainen & Hätönen 2008a].

Distributions of radio performance measurements or log activity counters are not usually known. Some traffic related features have heavy tail distributions and are closely related to ethernet traffic [Williamson et al. 2005], which is self-similar by nature [Leland et al. 2005; Laurikkala 2009]. Poisson models, for example do not fit the network traffic well [Paxson & Floyd 1995] and more complicated models are required, such as mixtures of exponentials [Feldmann & Whitt 1998].

A variety of distributions are produced by both server log activity and radio interface performance measurements. Examples of the distributions of three types of variables are depicted in *Figure 2.3*.

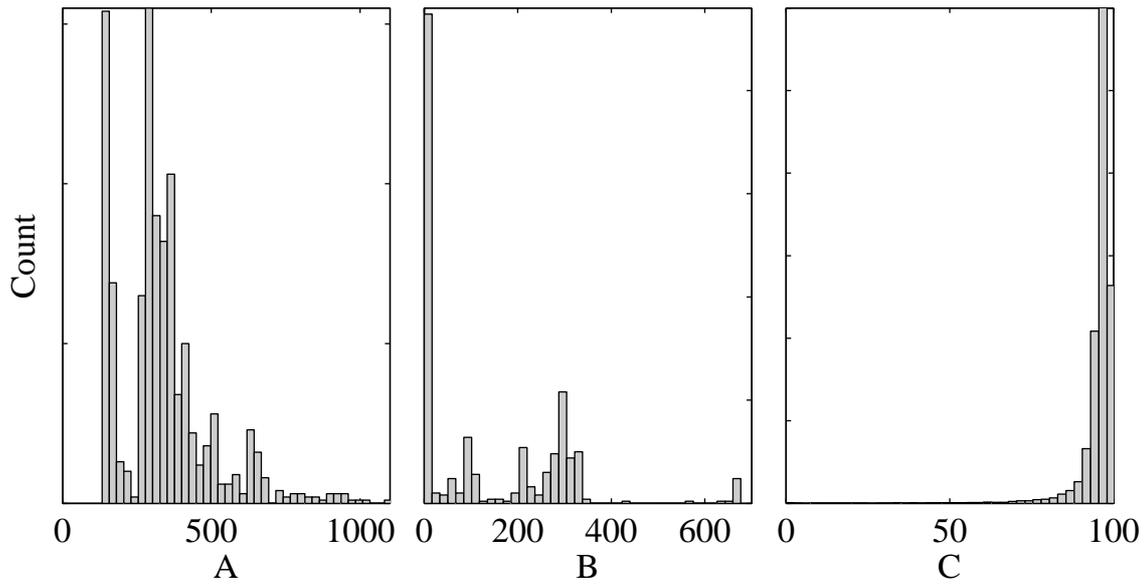


Figure 2.3 Examples of distributions of network management data.

The first two histograms on the left, *A* and *B*, present aggregated log activity variables, and variable *C* is an example of a KPI which represents radio interface performance. These examples include skewed, possibly heavy tailed, and multimodal distributions, and the real data contains a variety of other types of distributions. In practice it is impossible to use any single distribution model, and a mixture of symmetric distributions like the GMM (Gaussian Mixture Model) does not fit these data very well.

Chapter 3: Anomalies and outliers

This chapter introduces the concept of anomalies or outliers. In practical applications these terms refer to similar cases and can be used interchangeably.

3.1 What are anomalies and outliers?

The Oxford Dictionary [Oxford 2005] defines an anomaly as “*something that deviates from what is standard, normal, or expected*”. This definition assumes that a model exists for what is considered to be standard. There also has to be a way to measure the deviation from the expected or normal situation.

An outlier is given a more detailed definition as “*a data point on a graph or in a set of results that is very much bigger or smaller than the next nearest data point*” [Oxford 2005]. This is well in line with the notion that possible outliers are extreme values in the data set [Barnett & Lewis 1987]. This definition requires that the data points can be ordered, and that the outliers appear at either end of the ordered sample.

According to these definitions, the concept of anomaly has a wider scope, covering all abstract phenomena that are not expected, whereas outliers are connected to measured data sets. However, any anomalous event in a process that is monitored will produce outliers in the recorded data set. Therefore it is understandable that both terms are used, depending on the point of view of the study.

A number of additional definitions have been given for outliers in the literature on statistics, for example: “*An outlying observation, or outlier, is one that appears to deviate markedly from other members of the sample in which it occurs*” [Grubbs 1969, p. 1]. Another general definition for an outlier was given by Hawkins: “*An outlier is an observation that deviates so much from other observations as to arouse suspicion that it was generated by a different mechanism*” [Hawkins 1980, p. 1]. A further definition for outlier is: “*An observation (or subset of observations) which appears to be incon-*

sistent with the remainder of that set of data” [Barnett & Lewis 1987]. These definitions are extensive and cover all possible types of anomalies. However, they are very general and give no guidelines on how to decide whether individual observations are outliers or not. All these definitions contain phrases that are very subjective: “*appears to deviate markedly*”, “*deviates so much ... to arouse suspicion*” and “*appears to be inconsistent*”. They all leave the final judgement to the end user.

These definitions also leave open the question of how to measure if one observation deviates significantly from the others. Two ways to measure the deviation are presented in the following sections. The first one is based on the prior probability of the observations, and the second one is based on the dissimilarity to the normal state. These are more precise approaches to anomaly detection than the previous definitions allow.

3.2 Prior probability of observations

An observation can be regarded as an anomaly if the probability of such a value is small. A threshold for abnormality can be drawn from the probability density function (pdf) of a random variable x by selecting the risk level of making a false decision.

Observation x_i is anomalous if the value of the pdf at x_i , $f(x_i)$ is lower than a constant c .

$$(3.1) \quad f(x_i) < c$$

The constant c for the threshold is defined by selected probability p of false positive decision.

$$(3.2) \quad \int_{f(x) < c} f(x) dx = p$$

A measure of deviation from normal can be defined as the information received when observing a value of x , which can be viewed as the ‘degree of surprise’ [Bishop 2006]. Observing an improbable value provides more information and a bigger surprise than a value that is very likely to be observed. Observing a value that is certain to occur with probability one, is an extreme case that provides no new information or surprise.

The measure of information content is a quantity $h(x)$, which is a monotonic function of the probability distribution $f(x)$. Obtaining two values x_1 , and x_2 , that are statistically independent, should provide the sum of the information gained from observing them separately, thus $h(x_1, x_2) = h(x_1) + h(x_2)$. When two observations are statistically independent, then $f(x_1, x_2) = f(x_1) f(x_2)$. The logarithm of $f(x)$ satisfies these relationships, and thus the measure of information and the degree of surprise is given as

$$(3.3) \quad h(x) = -\log f(x).$$

However, this measure requires that the probability distribution is known. In real life applications this is rarely the case.

3.3 Dissimilarity to normal state without distribution

An observation can be regarded as an anomaly if it is distant from normal observations. In other words, the distance between an anomalous observation x_i and a reference point of normal x_R is large. A threshold value D can be given to the distance, thus for an anomalous observation x_i :

$$(3.4) \quad d(x_i, x_R) > D$$

Notation $d(a, b)$ is a distance metric between observations a and b .

The distribution does not have to be known in this presentation of anomalies; only a prototype for what is normal is required. Sometimes there is *a priori* knowledge of normal reference, or *a priori* knowledge of anomalies. Most often in industrial applications the normal reference is identified from data; a mean value of a reference data set, for example.

3.4 Detecting anomalies

Outliers are *extreme* values in data [Barnett & Lewis 1987]. In order to be regarded as an outlier the value has to be *surprisingly extreme*. What is considered *surprisingly extreme* depends on what is expected; the assumed underlying distribution as well as the number of observations in the data set.

3.4.1 Univariate anomalies

If a sample of n univariate observations is sorted in ascending order $x_{(1)}, x_{(2)}, \dots, x_{(n)}$ then $x_{(i)}$ is called the i th order statistic [David & Nagaraja 2003]. The extremes of this sample are the first and the last ones, $x_{(1)}$ and $x_{(n)}$. The cumulative distribution function (cdf) of $x_{(n)}$ is given by $F(x)$, the cdf of the random variable x .

$$(3.5) \quad F_{(n)}(x) = P(x_{(n)} \leq x) = P(\text{all } x_{(i)} \leq x) = F(x)^n$$

The cdf of the first order statistic is $F_{(1)}(x) = 1 - [1 - F(x)]^n$.

Figure 3.1 presents an example of observations that are expected to originate from normal distribution with zero mean and unit variance, $x \sim N(0,1)$. Observation $x_{(i)} = 1.3$ seems to be within a range that can be regarded as normal and is therefore not anomalous. The extreme observations are located at $x_{(1)} = -3$ and $x_{(n)} = 3.3$. In this example, most people would judge them to be lower and upper outliers.

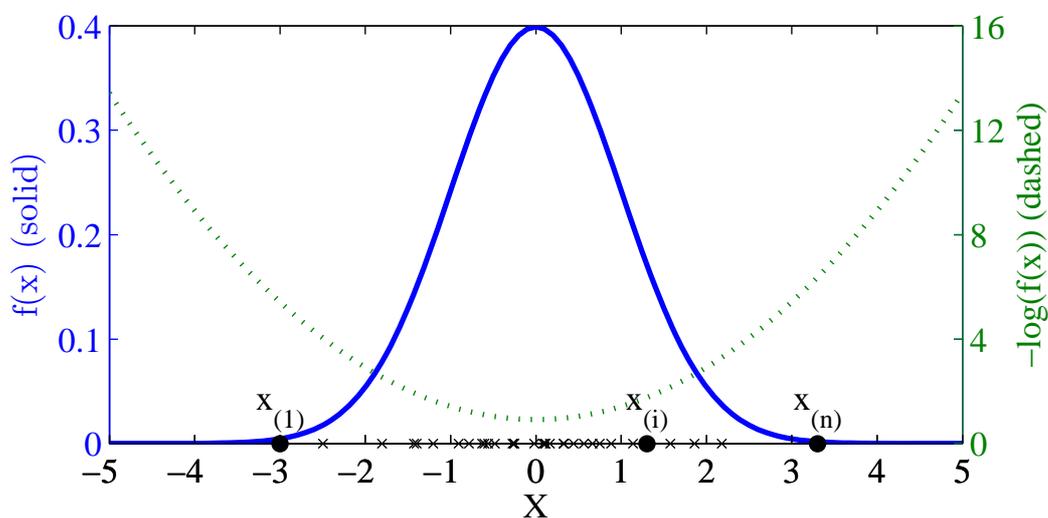


Figure 3.1 Example of lower and upper outliers in normally distributed data.

Now $n = 33$; thus the probability that the largest observation, at least 3.3, equals $P(\text{all } x_{(i)} \leq 3.3) = F(x)^{33} = 0.9842$. If the sample size was 100 or 1000, the

probabilities of $x_{(n)} \geq 3.3$ were 0.9528 and 0.6166 accordingly. Consequently, a value of 3.3 is not at all anomalous if the sample size is 1000 observations.

In order to decide whether the observations are outliers or not, a model of what is normal is required, as well as a measure to describe whether the observation under study fits the model or not. The thresholds c and D can be used to help in the judgement. These thresholds are depicted in *Figure 3.2*, with the observations $x_{(i)}$ and $x_{(n)}$.

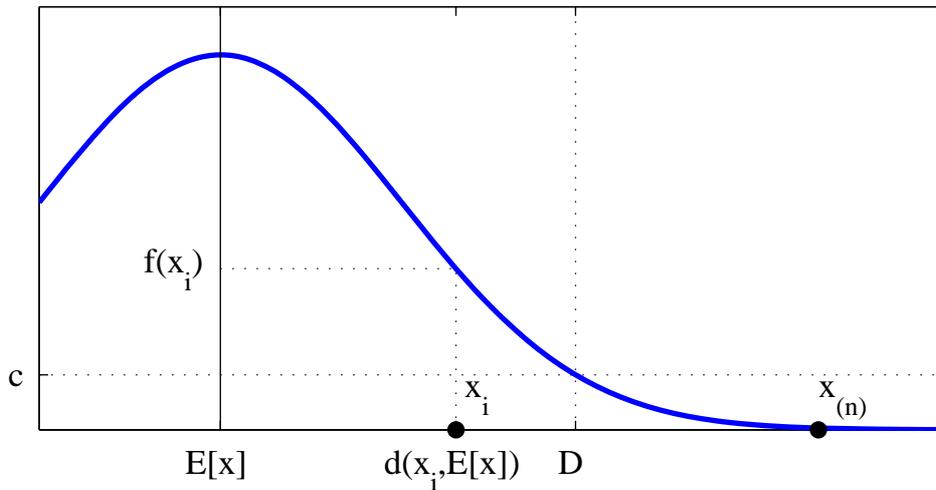


Figure 3.2 Anomaly thresholds described by distance d from the expectation value $E[x]$ and a constant c defining the 0.05 probability of false detection.

Now that the distribution is known, it is straightforward to calculate the threshold value, c , for the pdf after the desired probability of false decision, p , is set. In *Figure 3.2*, $p = 0.05$ and the corresponding constant c for $N(0,1)$ distribution is presented. The value $f(x_{(i)})$ is above the threshold c . Thus observation $x_{(i)}$ can be regarded as normal. However, the value of the pdf at $x_{(n)}$ is clearly below c . Therefore, there is a maximum of five per cent probability of a false decision if $x_{(n)}$ is judged as an anomaly.

The obvious reference value of normal is the expectation value of the distribution, $E[x] = 0$, as depicted in *Figure 3.2*. The distance of observation $x_{(i)}$ from the normal in a univariate case is the difference $d(x_{(i)}, E[x]) = x_{(i)} - E[x] = x_{(i)}$. In this case (symmetric distribution), the probability of false decision can be converted to the threshold

D for the distance. $N(0,1)$ distribution at $p = 0.05$ probability yields $D = 1.96$. The distance of observation $x_{(i)}$ from $E[x]$ is below the threshold D and it can be regarded as normal. However, the distance of observation $x_{(n)}$ exceeds the threshold, and it can be judged as an anomaly.

In order to decide on abnormality, even if the distribution is known, the threshold has to be selected. It can be thought of as the risk level of a false alarm of an anomaly. In industrial applications, the costs of false alarms and undetected anomalies have to be taken into account. The balance of these depends on the process, and thus guides the decision regarding the risk level. Univariate methods exist to detect outliers from symmetric unimodal distributions [Davies & Gather 1993]. In general, the underlying distribution is not known, and thus the threshold has to be selected.

3.4.2 Multivariate anomalies

In univariate data the anomalies are the extreme ones, “sticking out at the end” of the data sample [Gnanadesikan & Kettenring 1972]. A multivariate sample has no unambiguous end. Ordering of the observations is required to find the order statistics, including the extremes [Barnett 1976, Barnett & Lewis 1987, Harmeling et al. 2006].

The probability and distance from normal presented in 3.2 and 3.3 can be used for ordering the observations. For multivariate data they both perform a projection to a single dimension where the extremes can be detected.

An example of a data sample of 302 observations in bivariate space is depicted in *Figure 3.3*. Variables x_1 and x_2 are independent and both are from normal distribution with zero mean and unit variance, $N(0,1)$.

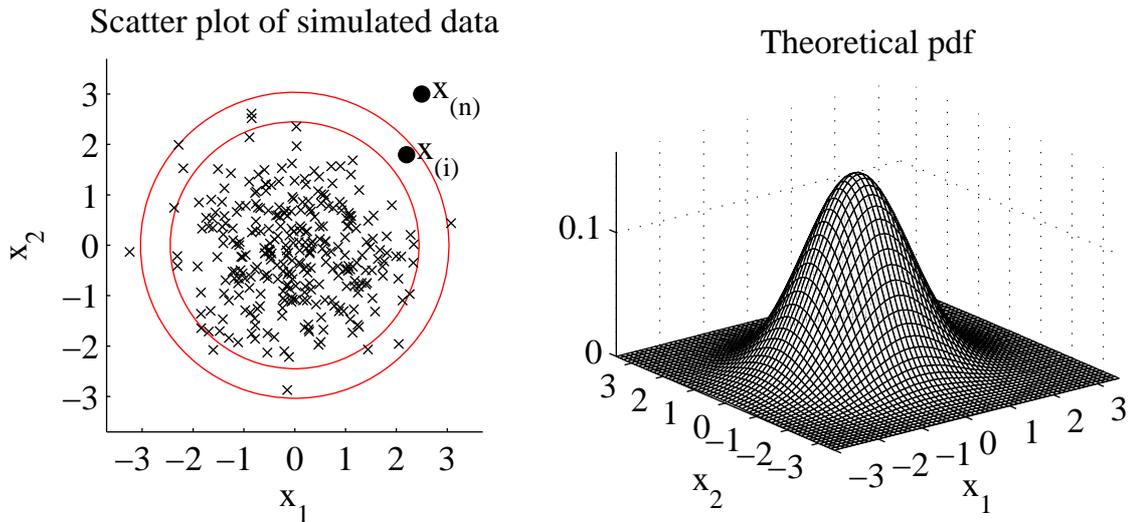


Figure 3.3 Example of two-dimensional data. Scatter plot of 302 observations (left) and theoretical probability density function (right).

The scatter plot contains the contours of constant probability density with 0.05 and 0.01 of total probability outside the enclosed area. In the case of independent normal distribution the contours are circles. Thus the probabilities that an observation falls outside the inner or the outer circle are 0.01 and 0.05 respectively. Correspondingly, 99% and 95% of the observations are located inside the circles.

In two-dimensional space it is possible to judge visually that observation $x_{(n)}$ is the most extreme one and possibly an anomaly. Observation $x_{(i)}$ is located barely inside the 99% circle but outside the 95% circle. The judgement of whether it is an anomaly or not depends on the selection of the risk level.

Visual presentations such as scatter plots are impossible in high-dimensional spaces. If the underlying distribution is known, the values of the probability density function at the observations can be used to order the data sample to find the most extreme observations. Figure 3.4 presents the histogram of the negative logarithm of the pdf values of this two-dimensional example.

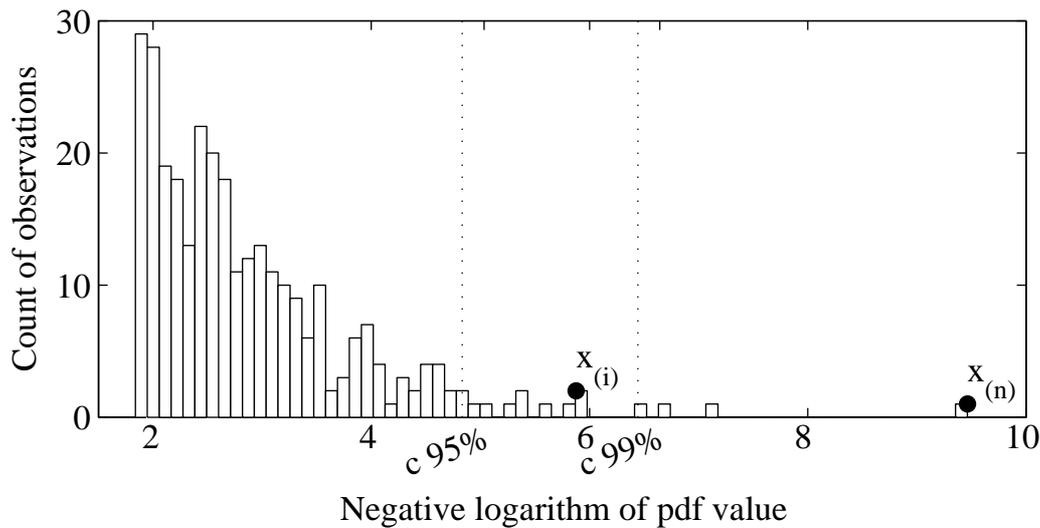


Figure 3.4 Histogram of the values of the pdf at observation points.

In this sample, there are 3 observations that have a pdf value below the 99% threshold, and one just above it. For a sample of 302 observations this complies with the theoretical result, as one out of hundred is expected to be below that threshold. However, the probability density at observation $x_{(n)}$ is so low that it arouses suspicion that it is in fact an anomaly.

Distance is one of the sub-ordering methods for multivariate data [Barnett 1976]. A histogram of the Euclidean distances from the mean is presented in Figure 3.5. The interpretation of the result is similar to that of the pdf values above. The distance of observation $x_{(n)}$ is substantially higher than the others. Observation $x_{(i)}$ is located near the 99% threshold but outside the 95% threshold.

In this case, the variables are from multiple normal distribution with unit covariance matrix. Therefore, the negative logarithm of pdf equals the squared Euclidean distance from the mean with a linear transformation. Thus, the histogram of the squared distance would be the same as the one in Figure 3.4, only with scaled values on the horizontal axis.

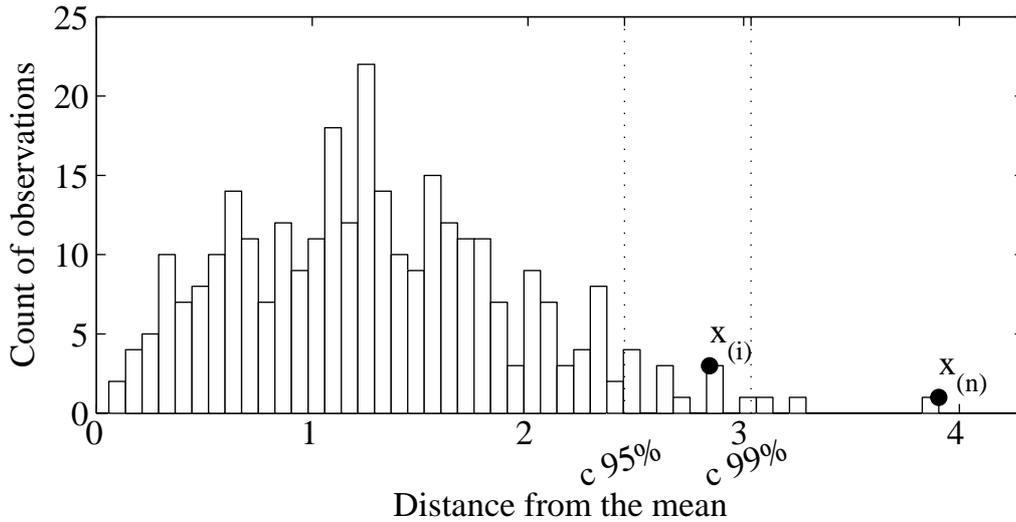


Figure 3.5 Histogram of the Euclidean distances from the mean.

Both the value of probability density function and the distance from a reference are capable of transforming multivariate data into a univariate value that can be used to order the data set. In this example the data were drawn from a known multinormal distribution. Therefore it was possible to calculate the thresholds for given risk levels also for the distance measure. In practical applications, the underlying distributions are not usually known, and the distributions and the probability thresholds have to be identified from the data. Distance measures are often more practical, since they only require simple distance calculation. However, the decision concerning the anomaly threshold for the distance measure has to be based on the data. Expert knowledge of the application domain and the operation of the process is indispensable to this decision.

3.4.3 Multimodal distributions (multiple operational states)

In case of unimodal distributions, the possible anomalies are located at the extremes of the sorted sample. However, not all processes produce unimodal data. There may be several distinct operational states in the process, producing a multimodal distribution or a mixture of distributions. Figure 3.6 presents an example of a process that contains two operational states. The synthetic data set consists of a mixture of two normal distributions, $N(\mu_1, \sigma_1)$ and $N(\mu_2, \sigma_2)$, where $\sigma_2 = 3 * \sigma_1$.

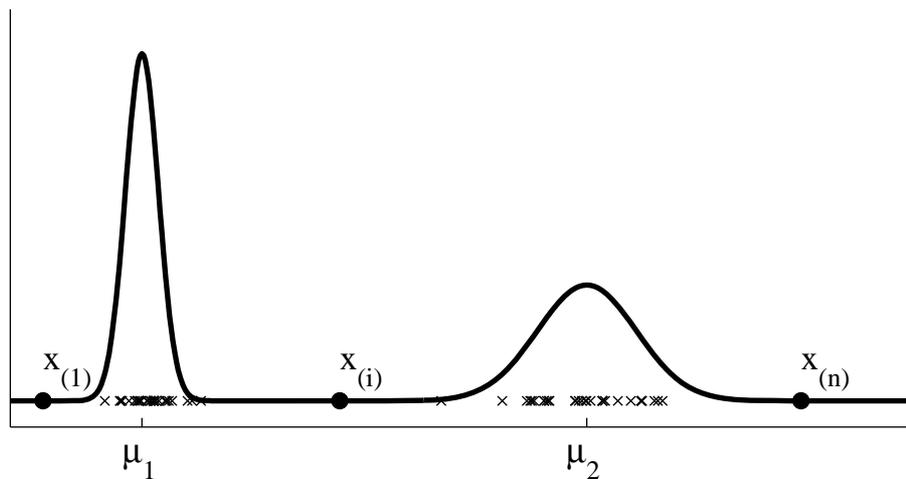


Figure 3.6 Data set with two operational states and anomalies.

Observations $x_{(1)}$ and $x_{(n)}$ are the lower and upper outliers similar to those in Figure 3.1. Observation $x_{(i)}$ is located in the middle of the range and is by no means an extreme value. However, the probability of such value is very low, and appearance of them is very rare. In real life, depending on the application, the transition between the operational states may or may not produce observations between the states. If the transition is slow enough to enable observations to be recorded then the collected data sample will suggest that such values are relatively common. If the transition is rapid, there are rarely observations between the operational states and observations such as $x_{(i)}$, which in this example can be considered anomalies.

If the underlying distribution is known, the decision on the anomalies is straightforward to make. In practice, the distribution is most often not known and it has to be identified from the collected data, for example using mixture models [McLachlan & Peel 2000]. Distance based detection of anomalies requires identification of several reference points, for example by finding clusters in the data [Kaufman & Rousseeuw 1990]. The distance from the nearest reference point determines the ordering and whether an observation is anomalous or not.

3.5 Distance measures

The task of anomaly detection is to find observations that *deviate* or *significantly differ* from normal. In order to measure the deviation or difference, a proximity measure is required [Agyemang et al. 2006]. The proximity measure is also referred as similarity, and its inverse, distance or dissimilarity, can be used equivalently [Tan et al. 2005]. A distance between two points is also the norm of a vector between the points.

While scales of the variables do not affect the methods based on pdf, scaling has an essential influence on distance metrics. Therefore any methods utilising distances are affected by the scales, including a wide range of anomaly detection methods. Variables that have larger numerical values and variance are overemphasised [Duda et al. 2001; Xu & Wunsch 2009]. The variables in industrial processes typically have different scales, and they have to be scaled to equalise the contributions of the variables on the distance measures. Such transformations are referred to as *scaling* in this thesis. Other commonly used terms for this include *normalisation* [Tan et al. 2005; Xu & Wunsch 2009], *standardisation* [Milligan & Cooper 1988; Everitt et al. 2001; Tan et al. 2005] and *rescaling* [Duda et al. 2001]. While the term standardisation is widely used for scaling, it is commonly used to refer to a scaling method into zero mean and unit variance. In this thesis, the term scaling also covers various nonlinear transformations, called re-expressions by Tukey [1977].

3.5.1 Distance metrics

A function $d(a,b)$, the distance between the two points a and b , is a metric if it satisfies three conditions [Dillon & Goldstein 1984]:

positivity: $d(a,b) \geq 0$; $d(a,b) = 0$ iff $a = b$

symmetry: $d(a,b) = d(b,a)$

triangle inequality: $d(a,c) + d(c,b) \geq d(a,b)$

The most commonly used metric is Euclidean distance, also known as straight line distance. If the points a and b are represented by n dimensional coordinate vectors \mathbf{a} and \mathbf{b} , then Euclidean distance

$$(3.6) \quad d(a, b) = \sqrt{(a - b)(a - b)^T}$$

is a special case of a general family of distance metrics, called the Minkowski metric, also referred to as L_k norm, defined as

$$(3.7) \quad d(a, b) = \left(\sum_{i=1}^n |a_i - b_i|^k \right)^{1/k}.$$

Minkowski distance with $k = 2$ yields the Euclidean distance or L_2 norm. The case $k = 1$ is the L_1 norm, also known as city block, Manhattan or taxicab distance.

The L_∞ or L_{max} norm, when k approaches infinity, is also referred to as Supremum or Chebychev distance. The distance is the maximum of the differences between the coordinates of each dimension $d(a, b) = \max(a_i - b_i)$.

A set of points at equal distance from a specific location form a line in two-dimensional space. Examples of three Minkowski distances, L_1 , L_2 and L_∞ are depicted in *Figure 3.7*. The lines present unit distances from the origin, $d(x, \mathbf{0}) = 1$.

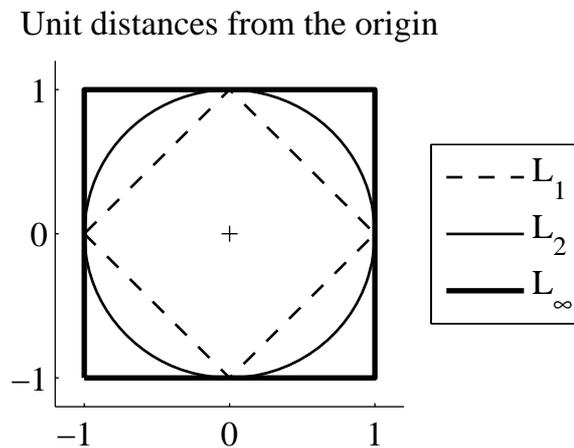


Figure 3.7 Unit distances from the origin.

Overall, the Minkowski distance for the parameter values $1 < k < \infty$ will form convex curves between the squares at the extremes, and the special case of the unit circle at $k = 2$ [Xu & Wunsch 2009].

All Minkowski metrics are translation invariant, but the Euclidean distance L_2 is the only one that is rotation invariant. However, none of them is scale invariant.

3.5.2 Scaling

Scaling of the variables is given surprisingly little attention in anomaly detection considering its huge effect on distance measures. The effect of scaling has been studied in clustering [Milligan & Cooper 1988], and in outlier detection [Knorr et al. 2001]. As could be expected, the results are inconsistent. The influence of scaling is highly dependent on the case and the data at hand. While typically ignored, the importance of scaling is most often brought up with clustering [Kaufman & Rousseeuw 1990; Gnanadesikan et al. 1995; Jain et al. 1999; Everitt et al. 2001; Duda et al. 2001; Xu & Wunsch 2009]. Gnanadesikan et al. [1995, p. 114] have pointed out: *When done efficiently, weighting [scaling] and selection can dramatically facilitate cluster recovery. When not, unfortunately, even obvious cluster structure can be easily missed.* A similar example to that given by Duda et al. [2001] is presented in *Figure 3.8*. The wider range and higher variance of variable x_1 on the left is due to the clustering structure. Scaling both variables to unit variance makes the clusters harder to separate by distance based clustering methods in the scatter plot on the right.

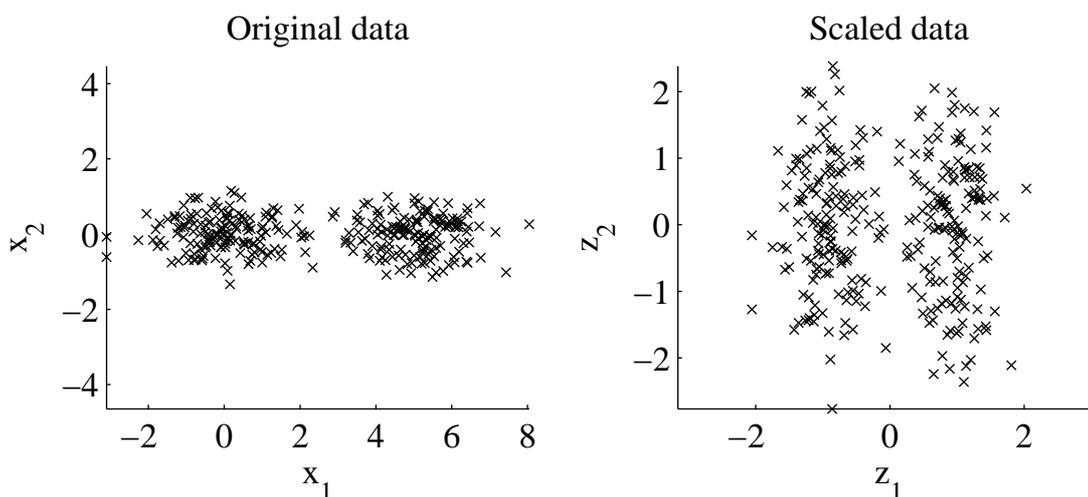


Figure 3.8 Scaling to unit variance can distort the clustering structure of the data.

The most common scaling, which is often done routinely without further explanation, transforms the data so that each scaled variable will have zero mean and unit variance. That is done by subtracting the mean and dividing by the standard deviation,

$$(3.8) \quad z_i = \frac{x_i - \bar{x}_i}{s_i},$$

where \bar{x}_i and s_i are the sample mean and standard deviation of the i^{th} variable respectively. The scaled variables are also referred to as *z-scores* or *standard scores*. This specified type of scaling is often referred to as *standardisation* [Johnson & Wichern 1998; Everitt et al. 2001; Duda et al. 2001; Xu & Wunsch 2009] or *autoscaling* [Wise et al. 2005; Everitt et al. 2001].

Scaling by range, the difference between the maximum and minimum values of the data set, is another common way to equalise the scales of the variables. Often the range is scaled linearly so that each scaled variable z_i will fall between 0 and 1,

$$(3.9) \quad z_i = \frac{x_i - \min(x_i)}{\max(x_i) - \min(x_i)}.$$

This scaling method was found to perform best in revealing clustering structure in artificial generated data [Milligan & Cooper 1988]. However, in outlier detection it was outperformed by scaling to z-scores [Knorr et al. 2001].

3.5.3 Scale invariant metrics

Another way to overcome the problems introduced by the different scales of the variables is to use scale invariant metrics. Standardised Euclidean and Mahalanobis distance are invariant to the variances of the variables.

Standardised Euclidean distance

Standardised Euclidean distance is a scale invariant modification of the L_2 norm. It is defined as

$$(3.10) \quad d(a, b) = \sqrt{(\mathbf{x}_a - \mathbf{x}_b) \mathbf{V}^{-1} (\mathbf{x}_a - \mathbf{x}_b)^T},$$

where \mathbf{V} is a diagonal matrix containing the variances of the variables on the diagonal. This will equalise the variances of the variables. The result is equal to using regular Euclidean distance applied to the z -scores that have zero mean and unit variance. An example of two variables that have different scales is presented in *Figure 3.9*. Variable x_2 has higher variance than x_1 , as seen in the scatter plot on the left, which presents the original values.

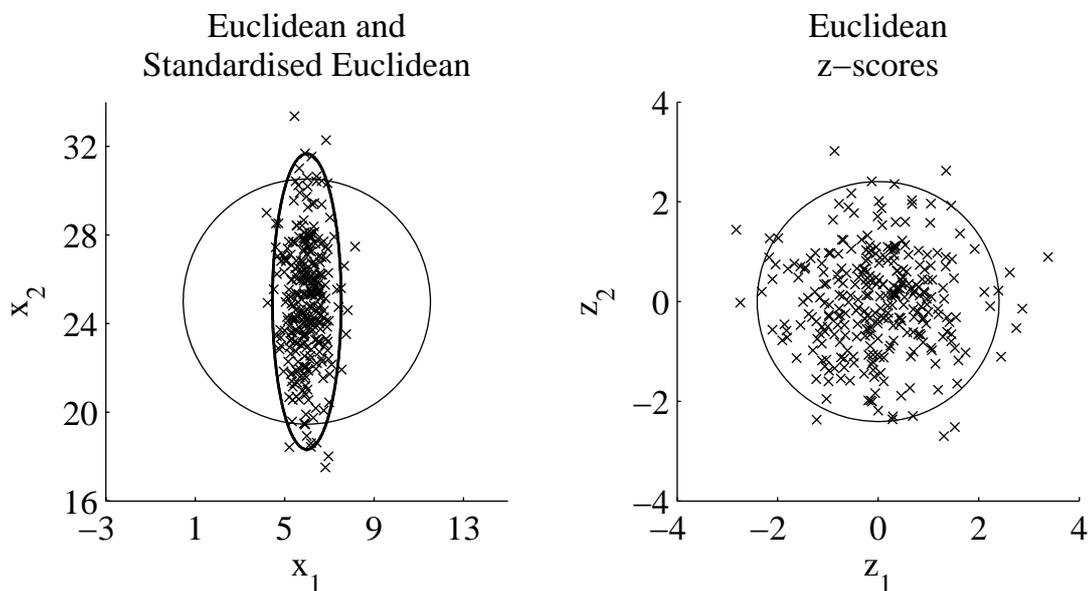


Figure 3.9 Different scales of variables require a scale invariant distance measure (left) or normalising of the data (right).

The circle on the left depicts a contour of equal Euclidean distance from the mean of the data. The distance is selected so that the circle contains 95% of the observations. It is obvious that x_2 dominates the distance differences. Values of x_1 would have to deviate enormously compared to its natural range in order to have any significant effect on the anomaly decision. Standardised Euclidean distance takes into account the variances of the variables, and the circle is squeezed into the ellipse depicted in the left plot. Each direction has an equal influence on the standardised Euclidean distance.

The scatter plot on the right in *Figure 3.9* presents the same data set normalised to zero mean and unit variance; the z-scores. The mean of the data set is shifted to the origin, and the contribution of the variables is equalised so that the circular Euclidean distance can be used.

Mahalanobis distance

Mahalanobis distance [Mahalanobis 1936] is another scale invariant distance measure. It takes into account the covariance of the variables. It is also called *statistical distance* [Johnson & Wichern 1998].

$$(3.11) \quad d(a, b) = \sqrt{(\mathbf{x}_a - \mathbf{x}_b) \mathbf{S}^{-1} (\mathbf{x}_a - \mathbf{x}_b)^T},$$

where \mathbf{S} is the sample covariance matrix.

A scatter plot of two correlating variables, including anomaly thresholds from four distance measures, is presented in *Figure 3.10*. The variables x_1 and x_2 are drawn from normal distribution with standard deviations 1.0 and 1.6, and correlation coefficient 0.8. The straight vertical and horizontal lines present the univariate anomaly thresholds, which equals the L_∞ norm, drawn separately at twice the standard deviation from the mean. The dotted circle and the grey and black ellipses present the equal Euclidean, standardised Euclidean and Mahalanobis distances from the mean, all enclosing 95% of the data.

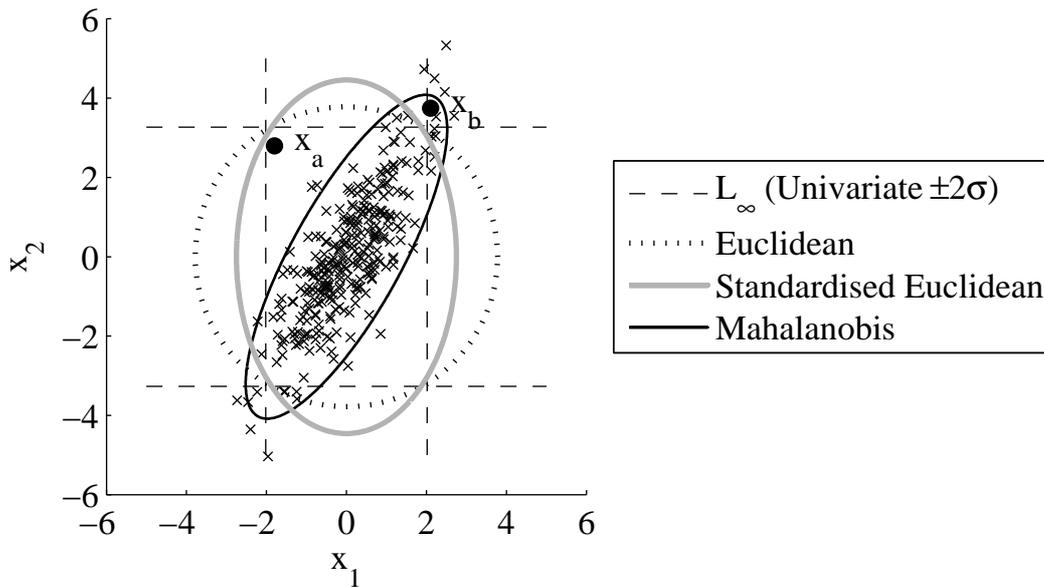


Figure 3.10 Comparison of univariate, Euclidean and Mahalanobis distances when the variables are correlated.

The thresholds determined by univariate and Euclidean distances contain large areas of the data space where there are no observations. Thus, they both are likely to produce false negative results. They fail to detect observations that are potential anomalies, for example the one marked as x_a in Figure 3.10. On the other hand, observations like x_b are detected as anomalies even though they do not deviate from the bulk of the data. The equal distance threshold produced by the Mahalanobis distance is an ellipse, as in the standardised Euclidean distance. The Mahalanobis distance utilises the full covariance matrix of the data, and the resulting ellipse is also rotated according to the correlation of the variables.

Mahalanobis distance is often an appropriate distance for multivariate normal data. However, the presence of outliers in the data does affect the covariance matrix estimate. Therefore robust covariance estimators are often used [Gnanadesikan & Kettenring 1972; Hadi 1992; Maronna & Yohai 1995; Peña & Prieto 2001].

3.5.4 Robust scaling

The presence of anomalies in the data set affects the scaling. A few extraordinary values can distort the estimates of the mean and standard deviation used in z-score scal-

ing. The range scaling which relies on only two values selected from the data, minimum and maximum, is even more sensitive to outliers, and even a single outlier can squeeze the bulk of the data to one end of the range in scaled space.

Median is a robust estimate of the centre of the data compared to the mean. Median absolute deviation about the median (MAD) is a robust way to describe the variation of the data [Maronna et al. 2006],

$$(3.12) \quad \text{MAD}(x) = \text{Med}\{|x - \text{Med}(x)|\} .$$

A robust scaling utilising median and MAD is achieved as,

$$(3.13) \quad z = \frac{x - \text{Med}(x)}{\text{MAD}(x)} .$$

This scaling is used, for example, by Knorr et al. [2001] and Filzmoser et al. [2008]. They both use the “normalised MAD” $\text{MADN}(x) = 1.4826 \cdot \text{MAD}(x)$, which is scaled so that the MADN for normally distributed variables is equal to the standard deviation [Maronna et al. 2006]. Chiang et al. [2003] propose a median-based modified scaling, which provides more accurate estimate of the standard deviation of the normal data, when at most 50 per cent of the data set is considered as outliers.

Another robust estimate of the centre is the trimmed mean, where a predefined proportion α of the largest and the smallest values are excluded from the calculation [Maronna et al. 2006]. When α approaches 0.5, the trimmed mean equals the median. Trimmed estimates can be used also for standard deviation [Kumpulainen & Hätönen 2008a].

The range of the data as an estimate of the deviation is very sensitive to outliers. It can be replaced by more robust trimmed versions. Interquartile range (IQR) is a common estimate that equals the range of the data, excluding one quarter from both ends of the ordered set. It is possible to use any other level of trimming, excluding a smaller proportion of the data set, such as five per cent from both ends.

3.5.5 Nonlinear transformations

None of the metrics presented above are invariant to non-linear transformations. Sometimes such transformations are required to level the relative distances within variables.

Counter variables are common in telecommunications network management data and suitable for logarithm transformations [Kumpulainen & Hätönen 2008a]. Hätönen et al. [2003a] compared z-scores to logarithmic sigmoid (LogSig) functions for scaling the data before analysing it, using the Self Organizing Map (SOM). They found that the z-scores highlight the extreme values, and that LogSig scaling was less influenced by outliers and emphasised the main body of the data.

Kylväjä et al. [2005] presented a piecewise linear scaling method that utilises *a priori* expert knowledge to equalise the importance of the quality variables in the radio interface of mobile telecommunications networks. The properties of that scaling method were further studied by Kumpulainen et al. [2009] and the scaling method is presented in detail in the first use case.

Other commonly used non-linear functions include square and square root.

3.6 Types of anomaly

Williams et al. [2002] divide outliers into three types; *cluster*, *scattered* and *radial* outliers. Cluster outliers occur in small clusters outside the bulk of the data. The outlier clusters are relatively tight, thus the local variance within such a cluster is small compared to that of the bulk of the data. Scattered outliers occur randomly in the data space outside the range of the vast majority of data. Radial outliers are located along the direction of the maximum variance in the data. If the bulk of data occurs in an elongated ellipse, as is the case in multivariate normal distribution, then radial outliers will lie on the major axis of that ellipse, the first eigenvector of the covariance matrix, but separated from and less densely packed than the bulk of data. However, Hardin and Rocke [2004] do not separate scattered and radial outliers, but use the term radial outliers for data that have the same mean as the normal data but larger covariance. These categorisations apply to cases where only one operational state exists in the

process, thus producing unimodal distributions. Cluster outliers, for example, can be originated from acceptable normal states of the process that occur less frequently than the dominant state that produces the bulk of the data. While small clusters are most likely true anomalies, it is not straightforward to draw the threshold when the cluster is small enough to trigger an alarm. The final decision has to be made by the experts in the process.

Chandola et al. [2009] divide anomalies into three categories: *point*, *collective* and *contextual* anomalies.

Point anomalies are single observations not consistent with the rest of the data. These are analogous to both the radial and scattered outliers discussed above.

Collective anomalies are formed by a group of observations where each observation alone is not anomalous but where the whole group occurring simultaneously is anomalous. While point anomalies can occur in any data set, collective anomalies can only exist when consecutive observations are related. Time series data exemplify structured data that can contain collective anomalies [Barnett & Lewis 1987]. Changes in time series data can also be considered collective anomalies [Basseville & Nikiforov 1993]. Another example of collective anomalies can be seen in the passenger profiles in one of the flights used in the terrorist attack against the WTC towers on September 11, 2001 [Kafadar & Morris 2002]. The suspected terrorists shared some common elements: none were U.S. citizens, all had lived in the U.S. for some period of time, all had connections to a particular foreign country, all had purchased one-way tickets at the gate with cash. One such passenger on a flight would not be extraordinary, but the fact that five out of 80 passengers shared those features conforms to the description of collective anomalies.

Contextual anomalies are observations that are not anomalous as such, but only in a special context, or when certain conditions are met. Time series outliers are examples of contextual anomalies [Karioti & Caroni 2002]. Seasonal (cyclic) phenomena form patterns in time series data, and deviations from these patterns are contextual anomalies. The daily patterns of mobile phone traffic exemplify such cyclic phenomena [Kumpulainen & Hätönen 2008b]. Daily average temperature is another example of

a one year cycle. A temperature of 1°C is a normal value in a series of daily averages within a year. However, if it occurs in the middle of summer, as exemplified in *Figure 3.11*, it can be considered as a contextual anomaly.

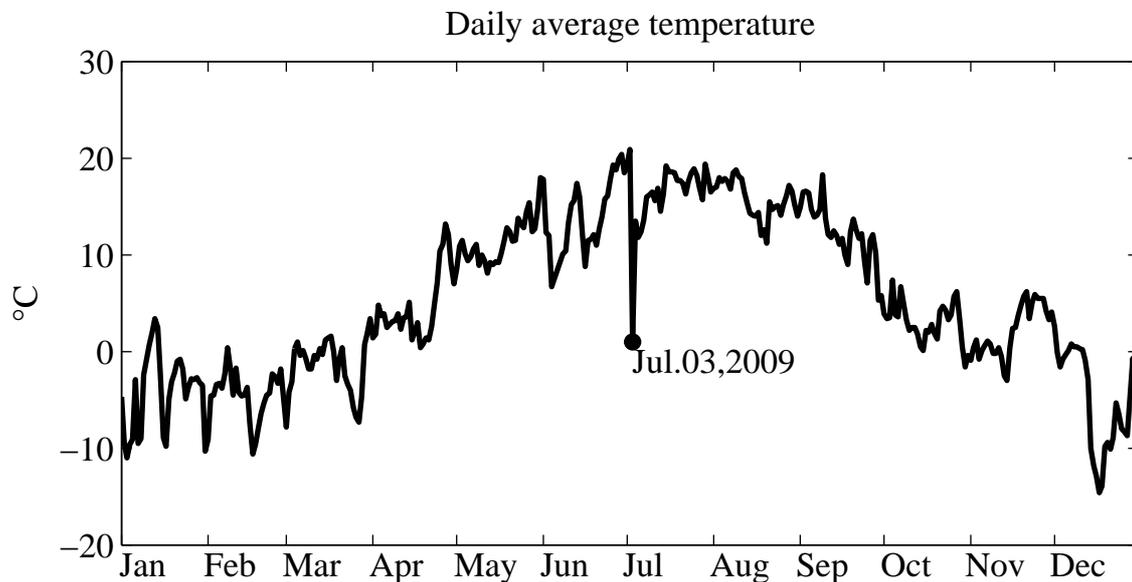


Figure 3.11 Daily average temperatures in Lempäälä 2009 (including a systematic error due to installation of the sensor).

Regression models form a structured case of input and output variables. They are sensitive to outliers, and even one outlier can cause the model to give misleading results. Robust regression methods, including outlier detection, have been widely studied [Gnanadesikan & Kettenring 1972, Rousseeuw & Leroy 1987; Maronna et al. 2006].

3.7 Sources of outliers

Anomalies in a data set can occur for a number of reasons. If the data follows a known distribution, it is normal that one observation out of a hundred will be outside the 99% threshold. Distributions can be divided into outlier-prone and outlier-resistant ones [Green 1976, Hawkins 1980]. In industrial applications, the goal is to detect anomalies produced by malfunctions or errors in the process. Anomalies in the data can present acceptable, but previously unseen, new states of the process. These can provide the users novel information about the process. Further sources of anomalies are errors

in measurements or in recording the data. Some errors may show up as outliers [Chatfield 1988]. It is important to detect such situations and fix the cause of the problem.

Chapter 4: Anomaly detection methods

Generally, anomaly detection requires a model to describe either *normal* or *anomalous* cases or both, as well as a method to decide which of those cases each observation belongs to. Eskin [2000] gives three assumptions that have to be satisfied in order to be able to detect anomalies: *the normal data can be modelled using a probability distribution, the anomalies are sufficiently different from the normal data, and the number of anomalies is small compared to the number of normal observations*. These assumptions are valid in most cases. However, distance based methods assume only the identification of normal states rather than a full distribution model. A wide variety of detection methodologies is presented in surveys [Markou and Singh 2003a; Markou and Singh 2003b; Hodge & Austin 2004, Agyemang et al. 2006, Bakar et al. 2006, Patcha & Park 2007, Chandola et al. 2009; Hadi et al. 2009; Singh & Upadhyaya 2012].

This chapter presents some application areas of anomaly detection and categories of detection methods. The methods are categorised by level of supervision, by local vs. global methods, and by methodology. The categorisations overlap and usually contain all the methods in one of their categories. The characteristics of four methods that are applied in the use case in Chapter 6 are demonstrated with synthetic two-dimensional data. The problem of assessing the performance of methods in real life applications is discussed.

4.1 Application areas of anomaly detection

Extensive lists of application areas that utilise anomaly detection can be found for example in Hodge & Austin [2004]; Tan et al. [2005]; Chandola et al. [2009]; Singh & Upadhyaya [2012].

Detection of anomalies is an essential task in online process monitoring. Early detection of the process getting out of control or of a deterioration in quality is essential in preventing faults in end products. Statistical process control (SPC) is often left out from lists of anomaly detection methods, even though it has been used to detect undesired changes in processes since the early 1940s [Milton 1990]. Detection of anomalies is also essential in offline analysis. Errors or exceptional values in data usually disturb the analysis and modelling methods, and the results are misleading at best [Filzmoser et al. 2008].

Anomaly detection is an essential tool in fraud detection, for example in mobile communication networks [Hollmen & Tresp 1998; Burge & Shawe-Taylor 2001]. Detection of credit card frauds has a substantial financial significance [Ghosh & Reilly 1994]. Kou et al. present a survey of techniques for fraud detection [Kou et al. 2004].

Network intrusion detection has become a very important research field [Scarfone & Mell 2007; Patcha & Park 2007]. Signature based methods have been widely used in misuse detection [Gómez et al. 2009]. They are able to detect previously known types of intrusions only. Therefore, anomaly detection techniques are required to detect new types of intrusions [Lippmann et al. 2000]. Unsupervised anomaly detection methods have been applied lately in network intrusion detection, see for example Leung & Leckie [2005]; Zhang & Zulkernine [2006]. Ghosh et al. [1999] utilise program behaviour profiles in neural network based intrusion detection. Network anomalies have been detected by wavelets [Lu & Ghorbani 2009] and self-organising maps [Ramadas et al. 2003; Zanero 2005]. Kruegel & Vigna [2003] combine several algorithms to detect web-based attacks. Yamanishi et al. [2000] introduced online outlier detection, applied to network intrusion and to detecting rare events in health insurance data. Their method combines categorial and continuous variables and adapts to non-stationary data, but does not provide an explanation of the cause of the detected outliers. The method was further extended to provide explanations for the outliers and for combining supervised and unsupervised learning [Yamanishi & Takeuchi 2001]. Harada et al. [2008] introduced an online unsupervised anomaly detection method for detecting anomalies in cyclic network traffic data.

Anomaly detection is a valuable tool in mobile network management [Anisetti et al. 2008]. AD has been utilised in radio interface [Kylväjä et al. 2005] and server log monitoring [Kumpulainen & Hätönen 2007].

Utilisation of anomaly detection methods is increasing in a wide range of application areas, such as healthcare [Bouarfa & Dankelman 2012; Hauskrecht et al. 2013] and bioinformatics [Bellaachia & Bari 2012; Ochs et al. 2013].

4.2 Categories by level of supervision

A common division into three categories is based on the characteristics of the data available for identification of the required models [Hodge & Austin 2004; Tan et al. 2005; Chandola et al. 2009]. This categorisation is the most general and all the methods can be assigned into one of these categories: *supervised*, *semi-supervised* and *unsupervised*.

4.2.1 Supervised

In supervised anomaly detection the methods require a labelled reference data set for identification of the models. Each observation in the reference set must have a label stating whether it belongs to the *normal* or *anomalous* class. A classifier is identified using the labelled reference data to classify the observations into the correct classes. New, unlabelled data can then be classified accordingly. These methods are essentially reduced to classification problems, in particular to classification for rare classes [Tan et al. 2005].

4.2.2 Semi-supervised

Semi-supervised methods assume the observations of one of the classes, either normal or anomalous, to be known and labelled in the reference data. It is usually easier to have a labelled data set of the *normal* class, and practically impossible to have examples of every possible type of anomaly to label. Therefore the labelled *normal* class is a more common case in real life applications. The semi-supervised AD method identifies the model of normal behaviour, and anything that differs significantly from normal is considered anomalous. Görnitz et al. [2013] propose a method to adapt

semisupervised detection into supervised as new data become available. In some cases the normal state (or possibly several states) of the process is known, and is usually optimal in some sense. The observations that differ significantly from that are considered to be anomalies. An example of one such special case is presented in Chapter 5.

4.2.3 Unsupervised

In practice it is very common that there are no labels available in the data, and unsupervised AD methods have to be used. These methods are the most widely applicable. The methods assume that the normal state of the process is far more common than anomalous states are. Therefore, the observations representing the *normal* class are supposed to appear far more frequently in the data set than anomalies do. These methods need a sufficiently large data set to be able to identify the *normal* states of the process. Identification of these states is based on an adequately large number of observations recorded during normal behaviour. Observations that do not share their characteristics with a sufficient number of observations, i.e. differ from the common behaviour, are considered anomalies. This is the most common AD category and the most widely covered area in this thesis.

4.3 Global and local anomaly detection

Anomaly detection methods can be divided into two categories: *global* and *local*. These apply to unimodal and multimodal distributions discussed in section 3.4.3.

The vast majority of research has concentrated on global methods. Global methods treat the whole data set as one group and assume that there only exists one normal, usually convex, region in the data space, and any sufficient divergence from that is anomalous. Thus, only a single normal operational state is assumed to exist in the process under study. Global methods include conventional process monitoring tests [Grant & Leavenworth 1996] as well as multivariate SPC methods (MSPC) [Fuchs & Kenett 1998]. Many global methods concentrate on robust estimates of location and scale, for example minimum volume ellipsoid (MVE) and minimum covariance determinant (MCD) estimates [Rousseeuw & Leroy 1987; Hadi 1992; Atkinson 1994; Hardin & Rocke 2004]. Rocke and Woodruff [1996] claim that robust estimators can deal with data where up to 40% of the observations are outliers. Chiang et al. [2003]

assume that at least 50 per cent of the data are from unimodal normal state and all the rest of the data (up to 50 per cent) can be outliers. Data recorded from controlled real life industrial processes rarely contain such numbers of outliers. If the data set contains a high number of cluster outliers, for example, it is most likely that the origin of those clusters is other operational states, not outliers. When the process has multiple operating states, the data will be clustered, and the global methods are likely to fail in anomaly detection. The space between the clusters may be regarded as normal or outliers, depending on the method applied. Thus local methods that are able to take the clustering structure into account are required.

Clustering based methods are local, as they utilise the cluster structure of the data by definition. Kumar and Orlin use a robust and scale-invariant clustering method that uses MVE and identifies outliers as a side product [Kumar & Orlin 2008]. Breunig et al. [1999; 2000] introduced the local outlier factor (LOF), based on local density and clustering. Isaksson and Dunham [2009] found an extensible Markov model (EMM), to outperform both LOF and another local density based algorithm, LSC-Mine. However, finding the optimum threshold for the EMM was problematic. SOM-based methods are closely related to clustering, and detect anomalies in local neighbourhoods [Munoz & Muruzabal 1998; Höglund et al. 2000; Kumpulainen & Hätönen 2008a]. Mixture models, such as Gaussian mixture models (GMM), can be used to identify probability distributions, and to detect the observations of low probability as anomalies [McLachlan & Peel 2000]. One-class support vector machines (OC-SVM) identify the support of the distribution, and they are able to detect multiple normal areas in the data space [Schölkopf et al. 2001]. However, the identified threshold is very sensitive to the parameters of the kernel function, and the validity of the support in multivariate space is not easy to verify [Manevitz & Yousef 2001].

4.4 Categorisation by methodology

Several divisions into AD categories have been provided according to the method. Tan et al. list *model-*, *proximity-* and *density-*, *clustering-based*, and *statistical* approaches [Tan et al. 2005]. Agyeman et al. [2006] present a division into *depth-*, *distribution-*, and *distance-based* methods, with an additional category of *clustering-based* techniques as a special case of distance-based techniques. Jin et al. [2001] pres-

ent the same division, but they also add a category of *density-based* methods that detect local anomalies. Six categories are given in an extensive survey by Chandola et al.: *classification-based*, *clustering-based*, *nearest neighbour-based*, *statistical*, *information theoretic* and *spectral* [Chandola et al. 2009]. Their *classification-based* category covers neural networks-based, Bayesian networks-based and rule-based methods.

Hodge & Austin [2004] present four categories: *statistical models*, *neural networks*, *machine learning* and *hybrid systems*. They include a wide range of methods in the category of *statistical models*. In addition to the traditional statistical tests, this category covers the proximity-based (nearest neighbour, clustering), parametric (minimum volume ellipsoid, depth, regression), non-parametric and semi-parametric methods.

Some of these categories are presented in the following, and their relations to the other categorisations are described.

4.4.1 Statistical

This is probably the oldest and most studied category of anomaly detection. Peirce introduced a criterion to reject doubtful observations over a century ago [Peirce 1852]. Classical works on outliers in statistics can be found for example in Grubbs [1969]; Barnett & Lewis [1987]. Robust statistics are often used in anomaly detection [Atkinson 1994]. Davies & Gather [1993] studied robust outlier identifiers on univariate and unimodal symmetrical distributions. Multivariate robust estimates, including outlier detection, were introduced in Gnanadesikan & Kettenring [1972]. Rocke & Woodruff [1996] studied very robust outlier detection in a case which had up to 35% of the observations as outliers. Statistical methods have been used in fraud detection [Bolton & Hand 2002], and intrusion detection in network traffic [Kanaoka & Okamoto 2003]. A review of statistical approaches in novelty detection is presented in Markou & Singh [2003a].

Statistical process control (SPC) is not mentioned in any of the reviews or surveys referred to above. However, it has been used in industry since the early 1940s, and the goal of SPC is essentially the same as in anomaly detection: to detect rare, undesired

states of the process as soon as possible [Shewhart 1931; Milton 1990]. SPC is based on control charts that are used to monitor quality variables [Shewhart 1931, DeVor et al. 1992, Grant & Leavenworth 1996]. Control charts also provide an efficient online tool to detect collective and contextual anomalies [DeVor et al. 1992]. Control charts show the values of the monitored variable and lines for *average* value and *upper* and *lower control limits*, *UCL* and *LCL*, exemplified in *Figure 4.1*. The average values and control limits are calculated from history data. SPC assumes Gaussian distribution, and the control limits are usually set at three standard deviations from the average value. The assumption of normal distribution can be overcome by using control charts on medians [Janacek & Meikle 1997], or methods that assume skewed distributions [Bi et al. 2001].

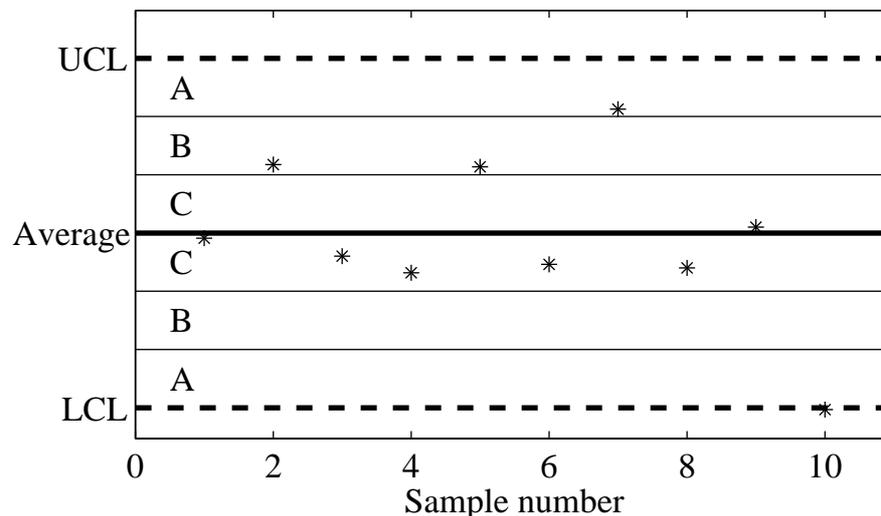


Figure 4.1 Example of a control chart.

The control chart is divided into zones A, B and C as shown in *Figure 4.1* [DeVor et al. 1992]. The borders of the zones are spread at 1σ , 2σ and 3σ from the mean. Additional tests can be applied to control charts to detect various types of abnormal pattern.

1. Extreme points (points outside control limits)
2. Two out of three points in zone A or beyond
3. Four out of five points in zone B or beyond
4. Run of eight or more successive points above or below centreline
5. Linear trend identification (increase or decrease of six successive points)

6. Oscillatory trend identification (14 successive points oscillating)
7. Avoidance of zone C test (eight successive points outside zone C)
8. Run in zone C (fifteen successive points in zone C)

Multivariate statistical process control (MSPC) provides tools for monitoring multivariate data, as monitoring a high number of variables is cumbersome or even impossible. If the variables have correlations, then monitoring each separately is not sufficient, as discussed in section 3.5.3. MSPC is based on principal component analysis (PCA), [Jolliffe 2002; Jackson 1991; Wise et al. 2005]. PCA decomposes the data matrix \mathbf{X} of m measurements and n variables, as the sum of the outer product of vectors \mathbf{t}_i and \mathbf{p}_i and residual matrix \mathbf{E}

$$(4.1) \quad \mathbf{X} = \mathbf{t}_1\mathbf{p}_1 + \mathbf{t}_2\mathbf{p}_2 + \dots + \mathbf{t}_k\mathbf{p}_k + \mathbf{E},$$

where k is the number of components used in the model. Typically, most of the total variance of the data set can be described by a few components, thus $k \ll n$. If all n possible components are included, the residual reduces to zero. Vectors \mathbf{t}_i are called scores. The \mathbf{p}_i are the eigenvectors of the covariance matrix and are called loadings. PCA projects the data on a hyperplane in space spanned by the loadings. The components are ordered according to the corresponding eigenvalues. The control limits can be set on scores, T^2 statistics (distances from the mean in the projection space) or residual Q (norm of the residual vectors in \mathbf{E}). The scores, which are linear combinations of the original variables, are usually more normally distributed than the original variables, in particular when the number of variables is large [Wise et al. 2005]. The causes of the anomalies can be traced back to the original variables by contribution plots [Miller et al. 1998; Conlin et al. 2000].

Martin & Morris [1996] presented non-parametric confidence bounds as an alternative to T^2 statistics in MSPC, acknowledging the distribution of the data instead of assuming normal distribution. Independent component analysis (ICA) [Hyvärinen et al. 2001] has also been utilised in multivariate SPC, relaxing the requirement of normal distribution [Kano et al. 2003]. Hoffmann [2007] introduces novelty detection by kernel-PCA, a non-linear extension of PCA for non-normally distributed data.

4.4.2 Model- and distribution-based

These methods are based on a model identified for the data. Observations that do not fit the model are considered anomalies [Tan et al. 2005]. In a regression model, for example, the observations that have large prediction error are labelled anomalies [Rousseeuw & Leroy 1987]. These methods are closely related to statistical methods, and the division is not clear. PCA, for example, can be regarded as a linear projection model, and statistical methods are based on distribution models.

When a probability distribution model is used, the method can be considered as *distribution-based*. Anomalies are observations that deviate from a standard distribution (e.g., Normal, Poisson, etc.) as presented in 3.4. Multivariate statistical methods have, for example, been used in intrusion detection [Ye et al. 2002, Kanaoka & Okamoto 2003]. Distribution based methods have been used for heavy tailed distributions [Mittnik et al. 2001; Feldmann & Whitt 1998] and in detecting local outliers [Zhang et al. 2008]. Bi et al. [2001] proposed a discrete Gaussian exponential (DGX) distribution to detect outliers in data that has skewed distribution.

The Gaussian mixture model (GMM) is a common distribution model for multimodal distributions [McLachlan & Peel 2000]. GMM models have, for example been utilised in detecting network intrusions online [Lu & Traore 2005] and for diagnosis of liquid rocket engine propulsion [Martin 2007].

Multimodal distributions can be described by a mixture of M component distributions [Nabney 2001]:

$$(4.2) \quad p(x) = \sum_{j=1}^M P(j)p(x|j),$$

where $P(j)$ are the mixing coefficients that describe the proportion of each component density function $p(x|j)$. In the GMM the component distributions are Gaussian distributions. The model is fitted with an expectation maximisation (EM) algorithm to estimate the parameters that maximise the log-likelihood of data [Dempster et al. 1977]. However, generally this maximum does not exist and regularisation heuristics are re-

quired: but there is no rigorous way to set the regularisation parameters [Mukherjee & Vapnik 1999]. The result will depend on the initial guess of the parameters of the Gaussian component distributions. Typically, the model is fitted several times, using random initial parameters, and the model with the largest likelihood is selected. However, in the case of unsupervised anomaly detection there is no proof that the model with the largest likelihood is the most suitable for the task. The number of Gaussians in the model has to be either selected manually or one can use information criteria for the selection, such as Akaike or Bayes information criteria (AIC, BIC) [McLachlan & Peel 2000].

In the following example, four GMM models are fitted to a two-dimensional synthetic data set containing 760 points. The models have the combinations of five or ten components with two values of regularisation constants, which are added to the diagonal of the covariance matrices to make them positive-definite [Nabney 2001]. The results are presented in *Figure 4.2*. The data points are presented by dots and the contour lines enclose 95% of the fitted distribution.

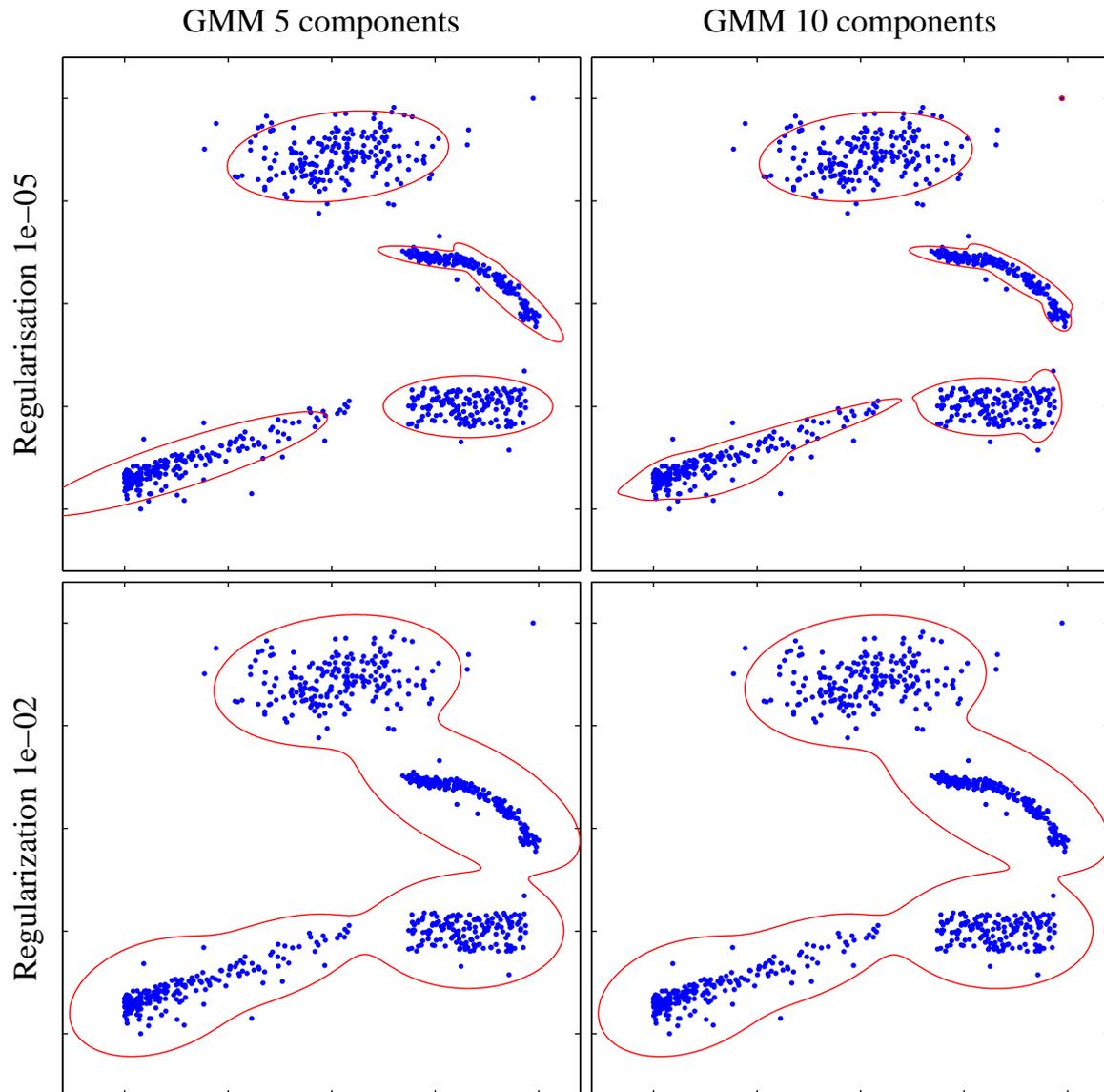


Figure 4.2 GMM fitted to a synthetic data set.

Five components provide a decent presentation of the distribution of these data when the regularisation constant is 10^{-5} . Increasing the number of components to ten provides only minor improvement. A regularisation constant of 10^{-2} produces estimates that leave large areas of empty space around the bulk of the data. These models are only able to detect large variations from the normal. In multivariate space it is not easy to verify the results. Smaller regularisation constants will allow the models to fit the data better, but the convergence of the EM algorithms may require larger values.

4.4.3 Depth-based

Depth-based methods rely on the computation of several layers of convex hulls. Observations on the outer hull are considered anomalies [Jin et al. 2001]. Fast computation of convex hulls in 2-d have been presented [Johnson et al. 1998]. However, when the dimensionality of the data is higher than three the depth based methods become inefficient in practice [Agyemang et al. 2006]. More advanced depth based methods include for example kernelised statistical depth [Cheng et al. 2009] and non parametric multivariate identifiers [Dang & Serfling 2010].

4.4.4 Distance-based

This category of outlier detection methods is based on distance, also known as proximity or similarity measures [Hand et al. 2001]. A very basic distance-based method was presented in 3.4.1. Multivariate Mahalanobis distance (also distribution-based) has been widely used in statistical outlier detection, see for example [Hadi 1992], and further improvements appear in [Hadi 1994].

Knorr and Ng introduced the most widely known concept of distance-based outliers (DB(p,d)-Outlier): an observation in a data set D is a DB(p,d)-outlier if at least a fraction p of the objects in D lies greater than distance d from it [Knorr & Ng 1998, Knorr et al. 2000]. This method is based on the interpoint distances between the observations, and makes no assumptions of distribution. Ramaswamy et al. [2000] introduced an enhanced method of distance based outliers which does not require the distance parameter d in the DB(p,d) outlier. Instead, it is based on the distance of k nearest neighbours of each observation. Harmeling et al. [2006] provided the outlier indices κ , γ and δ that are variants of the k nearest neighbours and are thus also based on distances between observations. While κ is the distance to the single k^{th} nearest neighbour, γ sums the distances to all the k nearest neighbours, giving a more refined view of the local density. The parameter γ also takes into account the directions in the multivariate space by taking the absolute value (vector length) of the mean of the differences of the observation and its k nearest neighbours. Hautamäki et al. [2004] have used k nearest neighbour graphs to detect outliers.

Another variant of distance-based methods utilises the distribution of the interpoint distances of the data set [Šaltenis 2004]. Basic distances, utilising *a priori* knowledge of the application domain, have been used in mobile network radio interface monitoring [Kylväjä et al. 2005; Kumpulainen et al. 2011].

The following categories, *clustering-* and *density-based* methods, can also be considered subsets of distance-based methods. Most clustering methods are based on distance metrics. Also, many of the density-based methods are based on the distances between the observations.

4.4.5 Clustering-based

Clustering is a term for methods that discover groups of similar objects in multivariate data where no predefined classes exist, and thus there are no known right or false results [Everitt et al. 2001]. According to Xu and Wunsch [2009, p. 8]: “*The ultimate goal of clustering is to provide users with meaningful insights from the original data so that they can develop a clear understanding of the data and therefore effectively solve the problems encountered*”. The general purpose of clustering in practice is that the produced groups can summarise the main characteristics of data [de Oliveira & Pedrycz 2007] and the resulting clusters are understandable and meaningful [Everitt et al. 2001]. The clustering results should be evaluated with respect to their usefulness as interpreted by a human researcher [Zimek & Vreeken 2013]. The number of clusters has to be selected. Several criteria exist to evaluate the clustering results of varying numbers of clusters [Davies & Bouldin 1979; Milligan & Cooper 1985; Rousseeuw 1987; Bezdek & Pal 1998]. They typically evaluate the compactness of the clusters and how well they are separated from each other. For data based criteria it is impossible to assess abstract qualities, such as meaningfulness of clusters. Instead of single correct solution, clustering reveals various aspects of the truth, or several truths, and “*the judgement on new clustering results, however, requires difficult and time-consuming validation based on external domain-knowledge beyond the existing class-labels*” [Zimek & Vreeken 2013].

K-means clustering [MacQueen 1967] is the most popular clustering in scientific and industrial applications [Berkhin 2002]. It divides the observations into k clusters

which are represented by a single point, the cluster mean. The objective is to minimise the sum of squared distances between the observations and the nearest cluster centre. The result depends on the initial locations of cluster means, as in the GMM. The algorithm is typically repeated several times with random initial values. Several methods have been proposed for initialisation [Arthur & Vassilvitskii 2007; Steinley & Brusco 2007].

Hierarchical clustering constructs a hierarchical tree, a dendrogram of clustering structure [Johnson & Wichern 1998]. Agglomerative hierarchical clustering starts by assuming each observation as a cluster, combining the ones with minimum distance to a new cluster. The nearest clusters are combined until all the data are in one cluster. Several choices exist to measure the distances between clusters, referred to as linkage methods. Single linkage is the minimum distance between any pair of observations in the clusters, while centroid linkage refers to the distance between the cluster centres. Ward linkage minimises the increment of the sum of squared distances from the cluster centre [Ward 1963]. Basic hierarchical clustering provides information about cluster structure and enables the detection of subclusters and outlier clusters [van der Heijden et al. 2004].

Jin et al. [2001] refer to clustering-based outliers as by-products of robust clustering methods which are developed to find clusters in the presence of outliers. Such clustering algorithms include, for example, DBSCAN [Ester et al. 1996], BIRCH [Zhang et al. 1996] and CURE [Guha et al. 1998]. However, there are several ways to utilise clustering in anomaly detection [Tan et al. 2005]. He et al. [2003] introduced the concept of the *cluster-based local outlier factor* (CBLOF). The CBLOF for each observation is calculated as the distance between the observation and its nearest cluster if the observation belongs to a small cluster, or the distance between the observation and the cluster it belongs to if the observation belongs to a large cluster. The division of the clusters into small and large is parametrised so that a given proportion of the data set belongs to large clusters. The clustering method can be freely selected.

Clustering has been used, for example, in intrusion detection [Portnoy et al. 2001; Leung & Leckie 2005]. Two layered clustering [Kumpulainen & Hätönen 2008c] and

fuzzy clustering [Kumpulainen et al. 2013] have been utilised for AD in mobile telecommunication network management.

The most straightforward approach to cluster based anomaly detection is to identify the normal states as cluster centres, and decide about anomaly based on the distance to the nearest centre. The example on the same generated data as in 4.2 presents an extension with two layers of clustering, an approach developed in this thesis. The clusters from the first layer, L1, identify normal states. Clusters with fewer observations than a predefined minimum are ignored and the observations are reassigned to the next nearest cluster. The centres of the L1 clusters are clustered to form second layer clusters, L2. The L1 clusters that are assigned to the same L2 cluster share the same threshold for anomaly detection. This allows the local variance to be taken in to account in detection, as exemplified in *Figure 4.3*.

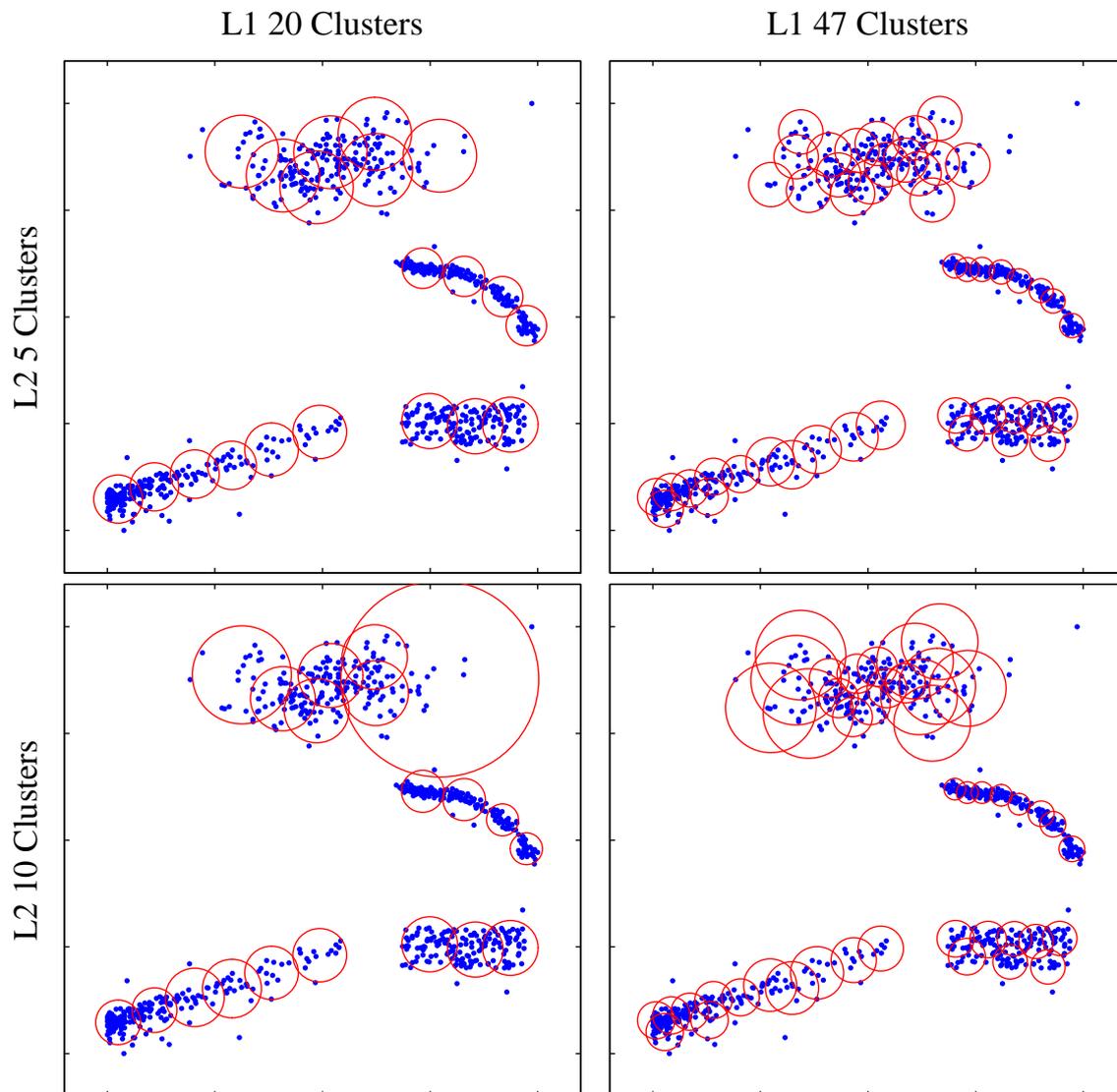


Figure 4.3 Example of anomaly detection by two layer clustering.

In this example, both layers were clustered by hierarchical clustering and Ward linkage. The data points are presented as dots. Each circle represents a cluster from L1. The number of clusters in L1 was 20 in the plots on the left and 47 on the right (initially they were set to 20 and 50 but three clusters with three or fewer data points were removed). The thresholds are set so that five per cent of the data is considered anomalous. The sizes of the circles are defined by the clustering in L2. The plots on top have five L2 clusters and therefore five sizes of threshold circles. The bottom plots have 10 clusters in L2 and circle sizes. A sufficient number of clusters in L1 is essential; nearby clusters with overlapping threshold are not a problem in anomaly detec-

tion. A high number of clusters in L2 can result in large sparse normal regions, as seen in the bottom left plot, where the number of clusters in L2 is close to the number of clusters in L1. According to this example, the best combination is a higher number of clusters in L1 and a lower number in L2.

4.4.6 Density-based

Methods in this category detect local outliers based on the local density of an observation's neighbourhood [Jin et al. 2001]. Breunig et al. [1999] introduced a local outlier factor that is based on local density. Jin et al. [2001] presented an enhanced method to find top n local outliers, the most severe ones, with less computing. DBSCAN was mentioned in clustering based methods, however, it is also brought up here as a density based clustering algorithm which refers to anomalies as *noise points* that do not belong to any of the clusters [Ester et al. 1996]. The density in all these methods is determined by the distances between the observations, and therefore density-based methods can be considered a subcategory of distance-based methods.

4.4.7 Classification-based and supervised neural networks

The task of classification is to identify a model that assigns observations into predefined classes. Classification includes a wide variety of supervised learning methods that require a labelled data set [Duda et al. 2001].

Bayesian classifiers have been combined to multivariate SPC in intrusion detection [Mehdi et al. 2007]. Neural networks have been widely used in anomaly detection, for example in credit card fraud detection [Ghosh & Reilly 1994], novelty detection [Bishop 1994] and intrusion detection [Ghosh et al. 1999]. Stefano et al. [2000] have proposed thresholds for rejecting data in neural network classifiers.

Support vector machines (SVM) are very efficient classifiers [Vapnik 1998], and are therefore a good option for supervised AD. They have been compared to multilayer perceptron (MLP) networks in intrusion detection [Mukkamala et al. 2002]. Enhanced SVM, combined with a genetic algorithm, has been used for network anomaly detection [Shon & Moon 2007]. Using SVD and multiscale transforms was found effective for anomaly detection in self-similar network data [Sastry et al. 2007].

4.4.8 Unsupervised neural networks

Replicator neural networks (RNN) are self-organising feed-forward MLP networks with three hidden layers [Hecht-Nielsen 1995]. An RNN uses the same features as input and output, and identifies a model to reproduce the inputs, using a staircase activation function in the middle hidden layer. RNNs were originally targeted at compression and coding, but they have also been used in outlier detection [Hawkins et al. 2002]. RNNs have been found to perform well in detecting a variety of outliers [Williams et al. 2002]. RNNs have also been used in detecting outliers in speech recognition [Tóth & Gosztolya 2004].

The Self-Organising Map (SOM) is an unsupervised neural network which approximates and visualises multi-dimensional data [Kohonen 1995]. An SOM usually consists of a one- or two-dimensional regular grid of map units, each of which contains a code vector that represents a point in the data space. During the training phase, the code vectors are computed to optimally describe the distribution of the data. The SOM is closely related to k-means clustering. The main difference is that the nodes of the SOM are organised in a low dimensional grid that preserves the topology of the data space. Thus the code vectors that represent points close to each other in the data space will be in nodes that are close to each other in the SOM grid. This allows effective visualisation, which is one of the most used features of the SOM [Vesanto 1999]. It has also been utilised in clustering [Mangiameli et al. 1996; Vesanto & Alhoniemi 2000; Flanagan 2003]. Furthermore, SOMs have been employed in the monitoring of cellular networks [Kylväjä et al. 2004; Barreto et al. 2005; Laiho et al. 2005], and specifically in anomaly detection [Munoz & Muruzabal 1998; Höglund et al. 2000; Kumpulainen & Hätönen 2008a].

The quality of the trained SOM can be evaluated by topographic error [Kiviluoto 1996]. This is calculated by summing the topographic errors of all N observations x_i .

$$(4.3) \quad \varepsilon = \frac{1}{N} \sum_{i=1}^N u(x_i) \quad ,$$

where $u(x_k) = 1$ when best and second best matching units of x_i are not adjacent, and zero otherwise.

Two versions of SOM-based anomaly detection are presented with the generated data set. Both versions use a one-dimensional SOM and quantisation error (QE), the distances from the observations to the nearest code vector, as a measure of anomaly. The map unit providing the minimum QE is referred to as the best matching unit (BMU). The hit count of a map unit is the number of observations that have that unit as the BMU.

The first method, presented by Höglund et al. [2000], features one global threshold and it consists of the following steps:

1. Identify a SOM from a reference data set; exclude map units with no hits.
2. Calculate the QE for the reference data. Set a predefined percentile of the QE as an anomaly threshold.
3. New data is considered anomalous if the QE exceeds the threshold.

The maps were identified by the SOM toolbox for MATLAB [Vesanto et al. 1999], and were initialised according to the first principal component. As this is the best linear approximation of the data, the map converges faster [Kohonen 1995]. This method was applied to the synthetic data set with 20 and 50 map units. Removing the map units with no hits resulted in 20 and 49 map units. The results are presented in *Figure 4.4*.

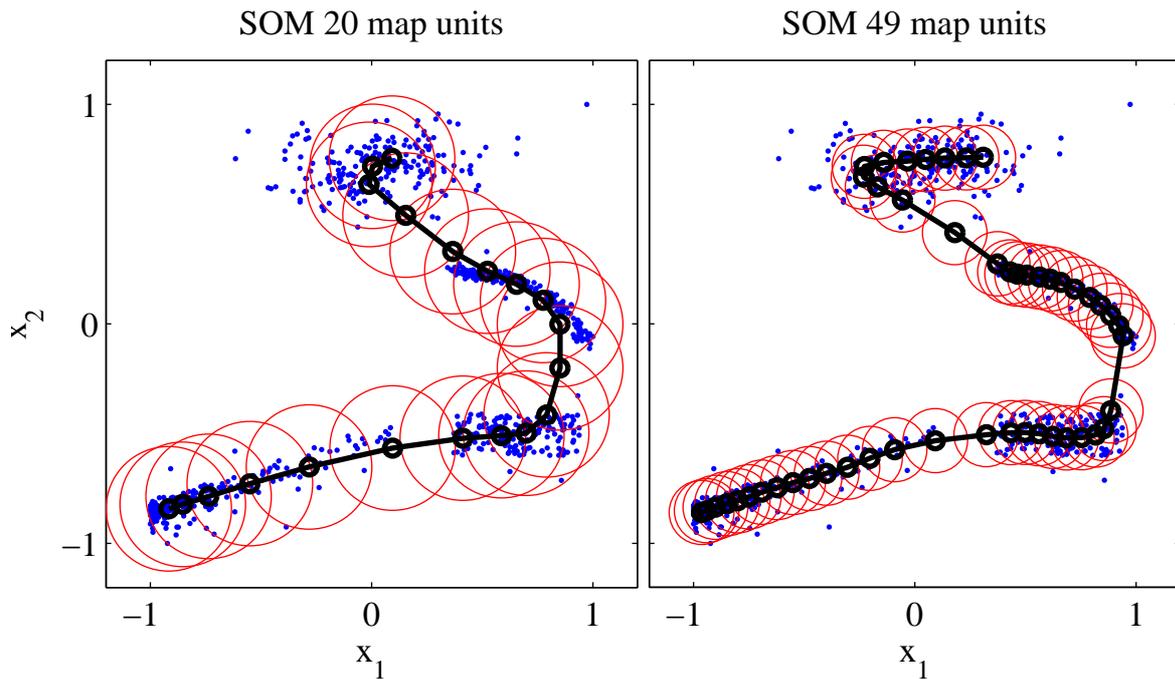


Figure 4.4 Example of anomaly detection by SOM with global threshold.

Small circles, connected with a line, present the code vectors of the map units and are ordered according to the topology. Larger, “normal” circles around each map unit present the 95% threshold, leaving five per cent of the data points outside the circles. The global threshold makes all the circles the same size. That results in the detection of most anomalies in the areas of higher variation in the data. The thresholds in the left plot of the smaller map cover unnecessarily large areas, most likely resulting in false negative detections.

Visualisation is one of the main advantages of the SOM, which is typically constructed as a two-dimensional grid. The most common visualisations of the SOM include component planes, hit histogram and U-matrix [Vesanto 1999]. The component planes visualise the values of the code vectors of each map unit, requiring a separate two-dimensional plane for each variable for a two-dimensional map. The hit histogram value for each map unit is constructed by counting the data points that have the map unit as a BMU, i.e. how many data points hit each map unit. The U-matrix presents the distances between the code vectors of the adjacent map units on the map grid [Ultsch & Siemon 1990]. Both the hit histogram and the U-matrix of a two-dimensional SOM can be presented either colour coded on two-dimensional planes or sur-

faces in three dimensions. The hit histogram can also be presented in markers, the size of the markers set according to the hit count.

A one-dimensional SOM allows more compact visualisation than a two-dimensional map. All two-dimensional planes for visualisation can be replaced by line plots. Component planes of the SOM with 20 map units (left in *Figure 4.4*) are visualised in *Figure 4.5*. The map units on the horizontal axis run from one to 20, and the values of the code vectors' components for the variables x_1 and x_2 are presented on the vertical axis. The value of x_1 starts at -0.91 in the first map unit, reaches the maximum value of 0.85 in map unit 11, and falls to 0.1 in the map unit 20. The value of x_2 increases along the map. In this example of two-dimensional data, the same can be seen in *Figure 4.4*, where the code vectors are presented in the actual data space, and the first map unit is located in the lower left corner. In higher than three-dimensional spaces, the visualisation provided by component planes is valuable.

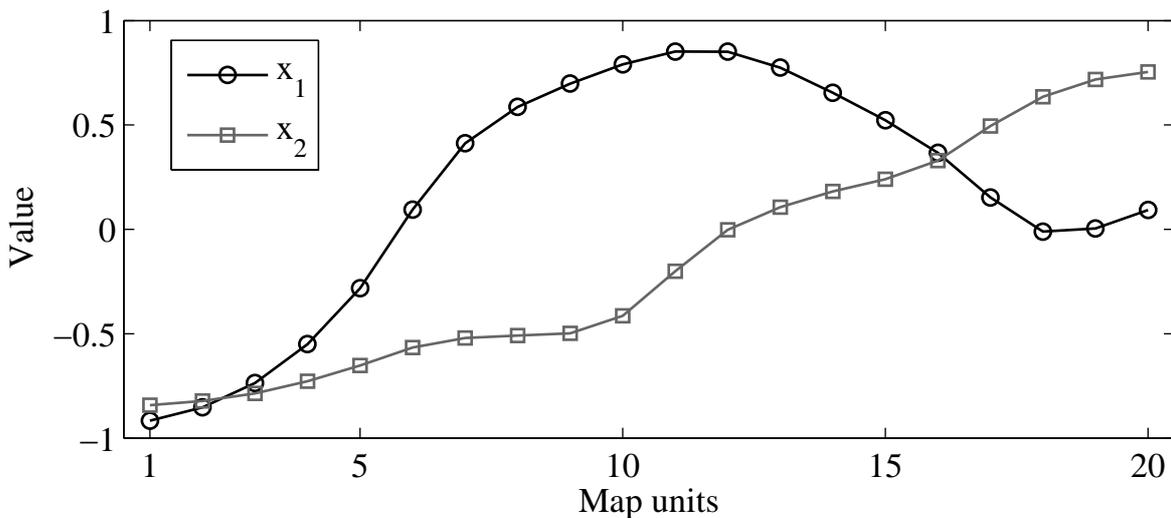


Figure 4.5 Example of component planes, which reduce to lines in one-dimensional SOM

The U-matrix and hit histogram of the one-dimensional SOM are depicted in *Figure 4.6*. The values of the U-matrix at the top are the Euclidean distances between the code vectors of adjacent map units. The higher values at map units 5, 10 and 16 are related to the longer distances between the map units in the data space. These map units are located between the clusters in the data, which can be seen in *Figure 4.4*.

The hit histogram can be presented as a line, but here it is shown as a histogram at the bottom of *Figure 4.6*. The map units in the sparse parts of the data space between the data clusters typically have high values of U-matrix and a low number of hits. This relation is clearly visible for this synthetic data.

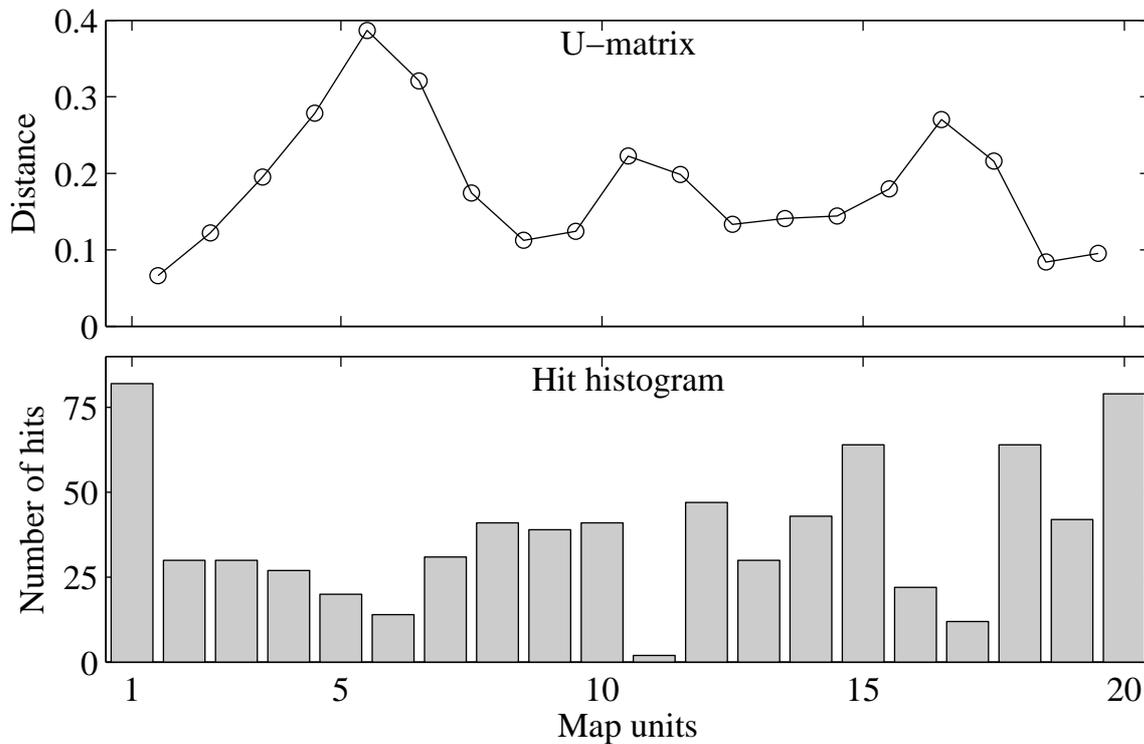


Figure 4.6 U-matrix and hit histogram of one-dimensional SOM can be presented in one dimension.

An extended version which utilises local thresholds for anomalies was developed in this thesis, and was first introduced by Kumpulainen & Hättönen [2007; 2008a]. The thresholds are localised by clustering the SOM code vectors and by identifying the threshold separately for each cluster. This is identical to the two-layered clustering method, only the first clustering is replaced by SOM. The most important benefit of using SOM is that the code vectors, which represent the normal states, are arranged according to the topology, and enable intuitive visualisation of the normal states [Kumpulainen & Hättönen 2008a; 2012]. The method consists of the following steps:

1. Train the SOM and exclude units that have less than a specified number of hits.
2. Cluster the code vectors: these clusters will be called reference groups RG.

3. Set the anomaly threshold to a predefined percentile of the QE in each RG.
4. For new data, calculate the QE, judge as an anomaly if it exceeds the local threshold of the RG, where the BMU of the new data was assigned.

This method was applied to the synthetic data set with 20 and 50 map units. A minimum of three hits was required, which resulted in retaining 19 and 47 map units. Both maps were clustered into five and 10 RGs. The results are presented in *Figure 4.7*.

The anomaly thresholds in this method are adapted to the local variance of the data. The smaller map, with 19 units, does not provide dense enough presentation of the data, resulting in very high threshold (large circles) near the upper part of the plots, where the variation in the data is high. The map with 47 map units fits the data sufficiently, and the anomaly thresholds adapt to the local variance of the data.

A sufficient number of map units is essential. The number of RGs is not that significant but the most appropriate combination of these four examples is a high number of map units and fewer RGs.

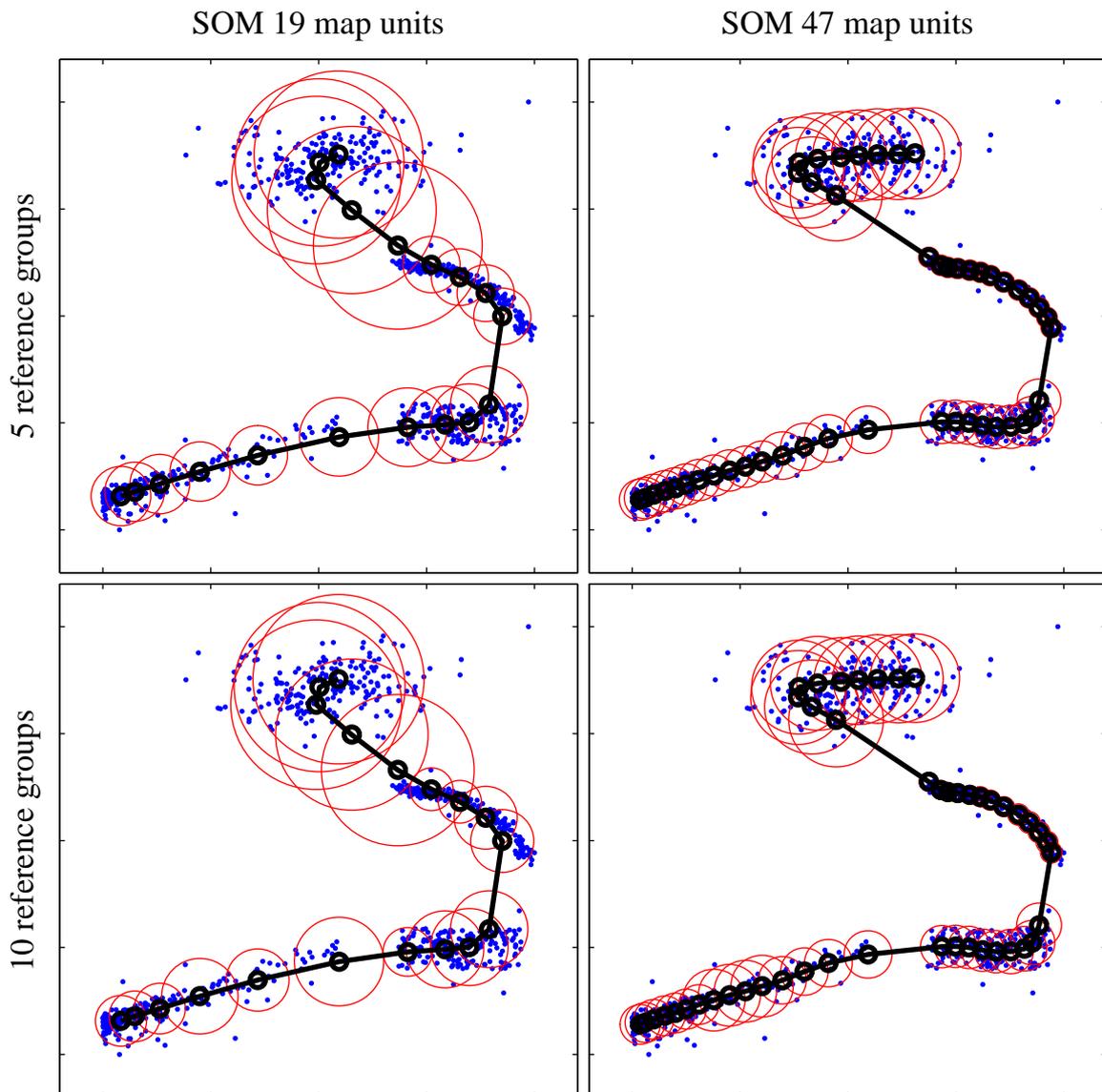


Figure 4.7 Example of anomaly detection by SOM with local threshold.

A One-class SVM (OC-SVM) was proposed by Schölkopf et al. [2001] for estimating the support of a high-dimensional distribution. The OC-SVM does not try to fit the distribution, but instead detects the optimal threshold for including a sufficient portion of the data. The strategy is to separate the data from the origin with a maximum margin in the feature space induced by the kernel. The maximum margin classifier to separate the data set of l observations, x_i , from the origin is achieved by solving the following quadratic program:

$$(4.4) \quad \begin{aligned} \min_{\mathbf{w}, \xi, \rho} \quad & \frac{1}{2} \mathbf{w}^T \mathbf{w} - \rho + \frac{1}{\nu l} \sum_{i=1}^l \xi_i \\ \text{subject to} \quad & \mathbf{w}^T \phi(\mathbf{x}_i) \geq \rho - \xi_i, \xi_i \geq 0, i = 1, \dots, l. \end{aligned}$$

A freely available software package LIBSVM is used in this thesis [Chang & Lin 2001]. Its implementation of OC-SVM solves a scaled version of the dual form of Eq. 4.4:

$$(4.5) \quad \begin{aligned} \min_{\alpha} \quad & \frac{1}{2} \alpha^T \mathbf{Q} \alpha \\ \text{subject to} \quad & 0 \leq \alpha_i \leq 1, i = 1, \dots, l, \sum \alpha_i = \nu l. \end{aligned}$$

In an OC-SVM without any class information, $Q_{ij} = K(\mathbf{x}_i, \mathbf{x}_j) \equiv \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$, thus it equals the kernel K , which can be selected freely. A variety of nonlinear estimators in input space is achieved by using different kernel functions. The decision function classifies the outliers to the class of the origin, represented by -1, and the bulk of the data to class +1 by:

$$(4.6) \quad \text{sgn} \left(\sum_{i=1}^l \alpha_i K(\mathbf{x}_i, \mathbf{x}) - \rho \right).$$

Parameter $\nu \in (0, 1]$ controls the fraction of the observations that is allowed to be classified into the class of the origin. Observations lying outside the support of the distribution are considered anomalies.

The OC-SVM has been applied to unsupervised anomaly and novelty detection [Schölkopf et al. 2001; Manevitz & Yousef 2001; Clifton et al. 2006; Rocco & Zio 2007]. Even though the OC-SVM performed very well with suitable parameters, it was found to be very sensitive to the selection of the kernel function and its parameters [Manevitz & Yousef 2001], causing dramatic differences in novelty detection results [Clifton et al. 2006].

The results of the OC-SVM applied to the synthetic data set are depicted in *Figure 4.8*. The parameter ν was set to include 95% of the data and to leave five per cent outside. A radial basis function (RBF) kernel $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2)$ was used with four values of parameter γ .

The result varies greatly with γ . Lower values result in the acceptance of large areas of empty space as normal. At higher values of γ , the OC-SVM starts producing unwanted holes in the middle of the bulk of the data as seen at $\gamma = 20$ in the lower right corner of *Figure 4.8*. Verification of the results for selecting γ to a functioning value is impossible in the absence of the ground truth in unsupervised anomaly detection. Subjective visual verification is only possible if the dimension of the data space is two or three. Due to high sensitivity to the kernel parameter, applying the OC-SVM for anomaly detection is questionable.

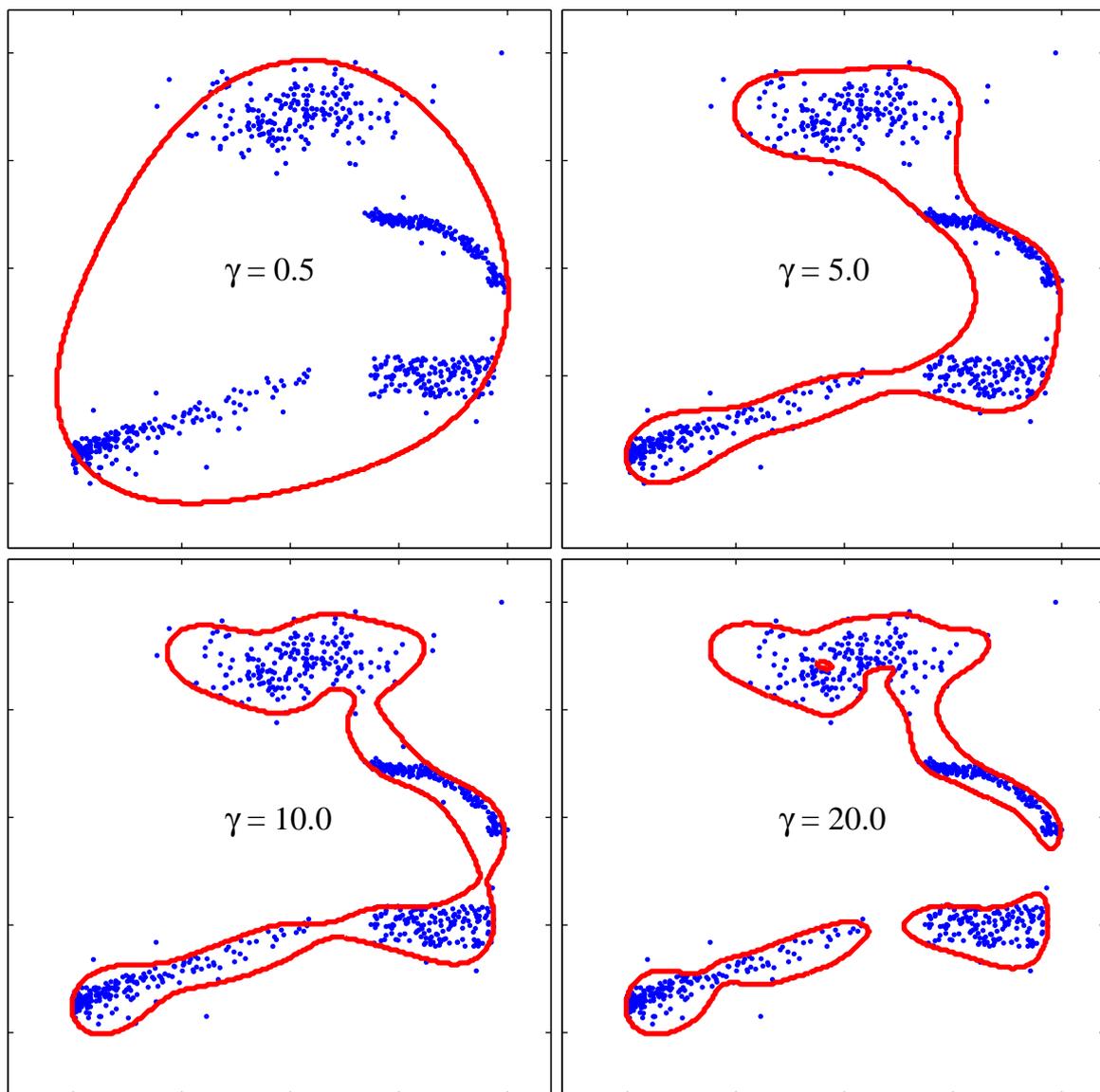


Figure 4.8 Example of anomaly detection by one-class SVM.

4.4.9 Visual

Visual methods can be used to assist human experts. Stalactite plots were introduced to highlight outliers detected by the robust Mahalanobis distance method [Atkinson & Mulira 1993]. Marchette and Solka use images of interpoint distance matrices, making this a distance-based method too [Marchette & Solka 2003]. However, it is only applicable to relatively small data sets. Kandogan introduced star coordinates to visualise trends and outliers [Kandogan 2001]. Caussin et al. [2003] used outlier monitoring displays, based on bi-plots [Gabriel 1971].

4.4.10 Projection pursuit

Filzmoser et al. [2008] divide outlier detection methods into two groups: *distance-based* methods and *projection pursuit* methods. Projection pursuit methods try to find projections of the multivariate data where interesting features would be more obvious [Huber 1985]. In AD the projections should be such that anomalies are easily detectable. It is impossible to investigate all possible projections. Stahel–Donoho is a robust estimator of multivariate location (weighted mean) and scatter (weighted covariance matrix), which has a high breakdown point, i.e. it is robust in the presence of a large proportion of outliers. The weights depend on the outlyingness obtained by considering all univariate projections, and therefore the estimator can be utilised in detecting outliers [Maronna & Yohai 1995; Knorr et al. 2001]. Peña & Prieto [2001] proposed a method to investigate only $2p$ projections of the p -dimensional space: the directions that maximise and minimise the kurtosis coefficient of the projected data. Principal component analysis (PCA) finds projections of maximum variance and is a particularly suitable projection when the dimensionality of the data is high [Shyu et al. 2003; Filzmoser et al. 2008]. Kernel PCA performs a non-linear projection and was found to be less sensitive to noise than the OC-SVM in novelty detection [Hoffmann 2007].

4.4.11 Hybrid methods

Hybrid systems consist of at least two algorithms from the above-mentioned categories [Hodge & Austin 2004]. Penny & Jolliffe [2001] compare multivariate outlier detection methods for clinical laboratory safety data and they suggest the running of a variety of multivariate detection methods in addition to univariate techniques. Barnett & Lewis [1987] also suggest the combination of several methods. Kruegel and Vigna [2003] combined several algorithms for the detection of web-based attacks. An ensemble approach of different soft and hard computing techniques was developed for intrusion detection by Mukkamala et al. [2005]. Combinations of several classification algorithms and the amalgamation of their results using a suitable metric are increasing [Hodge & Austin 2004]. Schubert et al. [2012] propose methods to compare the rankings of anomalies, which enables selecting anomaly detection methods that provide optimally distinct results, thus improving the combined result of the ensemble of detection methods.

4.5 Results produced by the AD methods

Typical results produced by AD methods are either binary labels (*normal* or *anomalous*) or anomaly scores for each observation [Chandola et al. 2009]. Anomaly scores enable ranking of the observations and detection of the “top-N outliers” [Schubert et al. 2012]. The outlier scores can be scaled into probabilities [Gao & Tan 2006]. In some use cases plain labels are sufficient. For example, if AD is an automatic preprocessing phase, rejecting the anomalous observations from further analysis steps, the binary label is the required result.

Hadi et al. [2009] criticise methods that provide ranking of the outliers by some sort of *measure of outlyingness*, such as LOF [Breunig et al. 2000], and instead promote the use of AD methods in providing unambiguous outlier boundaries and thus an exact classification for outliers and non-outliers. On the other hand, in the discussion of Barnett’s paper, Professor Lewis claims that the topic, ordering multivariate data (which also enables ranking of outliers) is “*one of the most important in the human context which one could imagine*” [Barnett 1976, p. 348]. He mentions examples of ranking students or making political decisions where multivariate data is available to support the decisions in selecting the best option.

When AD is used as a decision support tool for human experts, anomaly scores are preferred. The user typically wants to have the anomalies ranked, in order to concentrate on the most severe ones first. The scores or anomaly coefficients enable the ranking. The scores also allow binary labelling by using a threshold for the score. Domain knowledge can be used to select the suitable threshold for each application.

The AD methods used or developed in this thesis are able to create ordering of multivariate data and produce a ranked list of the anomalies. The multivariate data is transformed to a univariate anomaly measure, based on distance or probability measures. Those measures can be used to create the ranking list of the potentially anomalous observations for the end user.

4.6 Requirements in real life applications

In practice it is very difficult to guarantee the performance of any anomaly detection method in a specific application. In their study on the comparison of multivariate outlier detection methods for clinical laboratory safety data, Penny & Jolliffe [2001] noted that the results vary depending on *a) whether the data set is multivariate normal or not, b) the dimension of the data set, c) the proportion of contaminants in the data, d) the type of contamination and e) the degree of separation of the outliers from the rest of the data*. In addition to the list above, the possible cluster structure of the data has a significant influence on the results. They suggest the running of a selection of multivariate detection methods in addition to univariate techniques to highlight the possible anomalies in a data set.

It is useful to try out a wide range of methods, including a variety of pre-processing and scaling options. This applies to research purposes, including offline analysis, in order to increase the knowledge of the application domain. Researchers and process developers will benefit from flexible tools [Kumpulainen & Hätönen 2008c]. These help in selecting the most suitable methods and parameters to use for specific tasks in each environment.

The end users of the automated AD applications are usually experts in the application domain, and it is unrealistic to expect them to have similar experience and knowledge of AD methods to that of researchers and application developers. Therefore, the majority of the free parameters in the detection have to be decided upon in a final application targeted to daily use. The applications have to be robust, reliable, easy to use and understand [Hätönen 2009].

Application domain experts are required to interpret the results provided by AD applications. Automated algorithms should not be trusted exclusively; they should only highlight the potential outliers, and the end user has the freedom to interpret the results [Billor et al. 2000]. Yet the results have to be meaningful and reasonable, otherwise the tools will be rejected.

When anomaly detection software is used as a data analysis tool, finding the hardest-to-detect anomalies is not the most critical task. Rather, it is often more important to

make sure that those anomalies that are reported to the user are in fact interesting. If too many unremarkable data points are returned to the user labeled as candidate anomalies, the software will soon fall into disuse. One way to ensure that returned anomalies are useful is to make use of domain knowledge provided by the user. [Song et al. 2007, p. 631]

While domain knowledge of the users may be invaluable, it is not trivial to integrate all the knowledge into the applications. First of all, it is essential to thoroughly examine the data provided by the system [Hair et al. 1995, p 58]. Domain knowledge can then be used to select the most suitable methods for the AD application. Domain knowledge can be utilised in scaling and weighting of the variables. Using domain knowledge for appropriate scaling of multivariate data can ensure that the deviations in each of the variables will contribute to the final results of multivariate AD methods in meaningful proportions [Kylväjä et al. 2005; Kumpulainen et al. 2009].

Using methods that provide some sort of *anomaly score* or *measure of outlyingness* enables the ranking of detected anomalies, as described in Section 4.5. Assuming that the calculation of such a score is done in a meaningful way, the observations ranked as the most anomalous will be the most interesting ones for the user. This allows the user to concentrate on the most serious problems first. Furthermore, if the user decides that a specific reported anomaly is not serious enough to trigger any actions, then it is likely that none of the anomalies that are ranked less important are worth any action. This will save the user from unnecessarily investigating all the potential anomalies detected by the algorithm used in the application.

As pointed out by Song et al. [2007], it is not essential in industrial AD applications to find the anomalies that are hard to detect and are typically relatively close to the normal data. The results do not have to be exhaustive but merely good enough for the purpose [Nisbet et al. 2009]. According to the principle known as *Occam's razor*, the right model is as simple as possible [Izenman 2008]. In the discussion section of Barnett's paper [1976, p. 348] Dr. G. M. Paddle informs that his medical colleagues were not that interested in the sophisticated methods he presented. Instead they wanted simple techniques with which to browse multivariate data and to pick out the salient features.

In robust statistics, it is essential that the methods tolerate high proportions of outliers. Proportions up to 35%, as studied by Rocke & Woodruff [1996], are not feasible in real life applications. If a process is under control, as it should be in everyday service, the proportion of anomalies is significantly lower. The proportion of anomalies in data mining applications is typically a magnitude less than can be handled by the most robust statistical methods [Williams et al. 2002]. However, the data sets are relatively large and the number of anomalies can still be huge [Hätönen 2009, p. 20]. One way to overcome the mass of anomalies is to cluster the detected anomalies and to provide a summary of anomaly prototypes to the user [Kylväjä et al. 2005; Lakhina et al. 2005]. In the best case, the user can apply similar corrections to similar problems around the network.

In addition to prototypes of anomalies, prototypes of normal behaviour provide valuable information [Harmeling et al. 2006]. Such information is especially important in accumulating process knowledge. Prototypes of the normal behaviour can be extracted, for example, from methods that are based on clustering, the SOM or the GMM [Kumpulainen & Hätönen 2008a; Kumpulainen & Hätönen 2008c].

4.7 Assessing the result

Supervised AD methods are essentially classification problems, and conventional methods from the classification area can be used to assess the results. Receiver operating characteristic (ROC) analysis was originally developed in the field of signal detection, and has been widely applied in evaluating the performance of binary classifiers. A typical ROC plot consists of a curve presenting the number of true positive classifications, versus a number of false positive classifications when the decision threshold of the classifier is varied across its range [Zweig & Campbell 1993]. It can be used in anomaly detection to present detected true anomalies versus false alarms. ROC analysis has been used to illustrate the performance of intrusion detection methods [Lippmann et al. 2000]. Schubert et al. [2012] criticise ROC analysis as it oversimplifies the results using plain true and false positive detections. They suggest using methods that produce outlier scores and comparing the rankings. Stolfo et al. [2000] claim that ROC analysis can be misleading and they suggest cost-based models in the validation of credit card fraud and intrusion detection methods. ROC

analysis can only be used with labelled data when the anomalies to detect are known. However, the costs and weights are company specific, which makes cost-based ROC analysis extremely subjective [Zanero 2007].

In real world applications, labelled data are rarely available and unsupervised AD methods have to be applied. In unsupervised AD and clustering the characteristics of the problem have to be identified from the data. The ground truth does not exist, instead there are multiple truths that may be equally valid [Zimek & Vreeken 2013].

Various procedures have been applied in order to be able to compare the results of unsupervised AD. In medical applications, Hauskrecht et al. [2013] acquire the ground truth from a panel of experts, whereas Bouarfa & Dankelman [2012] identify the normal state, consensus workflow, from data. They prefer data based consensus rather than expert opinion “*which can require a lengthy debate, with no guarantee of reaching a final consensus*”. Thus, they also acknowledge the existence of multiple truths and that the experts do not necessarily agree on the final results. Many authors use the popular labelled data sets from the UCI Machine Learning Repository [Bache & Lichman 2013] to test unsupervised AD methods by treating the small groups as anomalies [Aggarwal & Yu 2001; Wu & Wang 2013]. The DARPA data set generated in 1999 [Lippmann et al. 2000] has been widely used to verify and compare intrusion detection methods, both supervised and AD based unsupervised methods [Portnoy et al. 2001; Mukkamala et al. 2005; Shon & Moon 2007; Lu & Ghorbani 2009]. However, the data set is far from perfect and includes many flaws and artefacts [Zanero 2007; Brown et al. 2009].

Hand [2006] presents several arguments why rigorous comparison of classification methods is not always useful, and can often be misleading. All these arguments also apply to anomaly detection. Fine tuning a detection method using specific data sets in order to show its superiority compared to some other methods is a common practice. However, such comparisons “*often fail to take into account important aspects of real problems, so that the apparent superiority of more sophisticated methods may be something of an illusion*” [Hand 2006, p. 1]. The improvements obtained by the more sophisticated methods are usually marginal and typically only available on specific data sets. Furthermore, the data available for the design do not usually represent the

actual distribution that will be faced in a real life situation. This is the very case for example with the DARPA data set discussed above. This also applies to AD applications targeted for use in real life processes, such as mobile networks, where the real data are typically confidential and not publicly available.

Another problem, brought up by Hand [2006], is that the labels are assumed to be objectively defined, with no arbitrariness or uncertainty. This is not always true, as exemplified by the mistrust in the ability of experts to agree on the consensus mentioned above [Bouarfa & Dankelman 2012]. This also applies in mobile networks, where the judgement of the anomalies is always specific to the individual network and the opinions of the experts do vary.

Hand [2006, p. 3] further suggests that “*no method will be universally superior to other methods: relative superiority will depend on the type of data used in the comparisons, the particular data sets used, the performance criterion and a host of other factors*”. The authors typically know their favourite methods best and are able to extract the best performance out of them. The parameters of the other methods in comparison may not be optimal at all but set to arbitrary values without justification as in Wu & Wang [2013] for example. The selection of the methods for the comparison may be unfair. Chiang et al. [2003] and Filzmoser & Todorov [2013] propose robust methods and compare them with PCA, using data that contains cluster outliers. While PCA is an excellent tool in dimension reduction and process monitoring, it is well known that PCA should not be used globally for clustered data. Thus outperforming PCA with such data is not a big achievement.

In the absence of the knowledge of the true anomalies it is practically impossible to assess the results or measure and compare the performance of the AD methods; “*in summary, outlier detection is, like clustering, an unsupervised classification problem where simple performance criteria based on accuracy, precision or recall do not easily apply*” [Williams et al. 2002, p 13]. In real life applications with no ground truth, the detected anomalies have to be inspected and verified by experts of the application domain, which is the approach selected in this thesis. Examples of similar approach are presented in nuclear power plant [Gupta et al. 2013] and in private corporate networks [Vaarandi 2013]. The most important property of the detected anomalies are

that they provide novel, useful information to the end user; “*the common point of all is that they [outliers] are interesting to the analyst. The ‘interestingness’ or real life relevance of outliers is a key feature of outlier detection*” [Singh & Upadhyaya 2012, p. 308]. The interestingness and real life relevance are factors that can not be unambiguously measured or compared. The final judgement can only be made by end users: only they can decide if the results are informative and useful. Therefore, in this thesis the emphasis is on exploratory analysis, and detection methods that provide anomaly scores. Further, summarising the information on the anomalies allows the end users to verify the results. The experts of the application domain are not usually experts in data mining. Therefore they need easy to use tools for comparing the methods, and to find the suitable parameters, as well as the most appropriate scaling [Kumpulainen & Hätönen 2008c].

Chapter 5: *A priori* knowledge in scaling for distance based anomalies

This chapter presents an example of distance based anomaly detection using radio interface performance measurement data from a mobile network (*Figure 2.1*). This is a special case where the global optimal state of the process is known beforehand. In anomaly detection the optimal state is used as a reference instead of the most common state identified from the data. *A priori* knowledge of the behaviour of the variables is utilised through scaling. The detected anomalies are clustered to provide summarised information about possible groups of problems for the operator. The scaling simplifies the interpretation of the problem groups. As a comparison the same procedure is repeated using normalised data without the expert knowledge. The sensitivity to the parameters of the expert scaling is also studied. This use case covers all three presented methodological contributions.

5.1 Objective

Mobile communications rely on the radio interface between the mobile devices of the subscribers and the cells located in the base stations (BS) in the operator's network. One of the daily monitoring tasks is to detect the cells that have problems which reduce their performance. When poor performance is detected the next task is to find its cause. The operator can then plan actions to improve the performance of the poorly behaving cells. Groups of cells that have similar performance problems can possibly be improved collectively.

The questions that the operator typically wants answered on a daily basis are “How has my network been doing recently (yesterday)? Which cells had the poorest performance? Are there other cells with similar problems that could benefit from the same

actions?” A mobile network contains thousands of cells and therefore efficient applications are necessary to provide information about poor performance cells to the operator.

The general objective of this use case is to introduce a procedure that helps in developing applications for providing that information to the operator in a compact and understandable way. The two detailed objectives are the following. The first objective is to provide ranking of the anomalies so that the most severe ones can be investigated with higher priority. This requires scaling the variables so that their importances in severity are equal. The second objective is to provide a summarised presentation of the anomalies by clustering the observations into groups of similar problems.

5.2 Use case

This use case deals with a commercial GSM network [Kylväjä et al. 2005]. The presented procedure provides information about the most critical poorly performing cells in the network. The procedure analyses the performance of the previous day. The information concerning the most severe problems is compressed by presenting groups of cells that have similar problems. The earlier behaviour of the poorly performing cells is also presented to the operator.

The procedure consists of both offline and online phases. The offline phase collects the expert knowledge for scaling. The expert is asked to define the levels of *worst possible*, *very poor*, *satisfactory* and *best possible* performance of each variable.

The online phase is described in the following list:

1. Scale the performance data from the previous day using expert specifications provided in the offline stage.
2. Calculate distances from the known optimal state.
3. Rank the cells by distance.
4. Consider the observations with largest distances as potential problems (anomalies).
5. Cluster the anomalies to create *problem groups*.
6. View the history of cells that have poor performance.

5.2.1 Data

The data set in this example is collected from 2385 cells. The cells are divided into six classes by their type and activity [Kylväjä et al. 2005]. There are three types of cell: micro, indoor and macro cells. The macro cells are further divided into four classes by splitting their traffic and handover activity to low and high activity. The thresholds for activity are specified by a network expert. The numbers of cells in each class are given in Table 5.1.

Table 5.1 The cells are divided into six classes. Most of the cells are macro cells with low amounts of both traffic and handover.

Cell type	Traffic	Handovers	Number of Cells	% of Cells
Micro			191	8.0
Indoor			26	1.1
Macro	Low	Low	1298	54.4
Macro	Low	High	147	6.2
Macro	High	Low	146	6.1
Macro	High	High	577	24.2

The performance data consist of six KPIs that measure various aspects of the performance of the radio interface between the subscriber's mobile devices and the cells in the operator's network. The data in this example consist of daily averages.

- Dropped Call Rate (DCR); the percentage of calls dropped during a day.
- Handover Success (HO_succ); the percentage of successful handovers to or from the cells during a day.
- Congestion; seconds of the day the element has been in a state that no new calls could be accepted due to lack of resources.
- Radio DownLink Quality (RX_DLqual); the percentage of measured radio quality samples in the "good" quality (classes 1 to 4 in the GSM specification).
- Average Downlink Signal Strength (DL_lev); the average signal strength received by the mobile devices served by the cell during a day, in dBm.

- Call Setup Success Rate (CSSR); the percentage of successful calls setup processes during a day.

5.2.2 Scaling

Experts' knowledge can be integrated in the analysis by scaling. Piecewise linear scaling, introduced here, allows for the integration of the process expert's existing knowledge of the behaviour and importance of the variables. The scaling information for the network performance data is acquired from network experts [Kylväjä et al. 2005].

The *a priori* knowledge for the scaling consists of four values of performance indicators that are defined for each KPI by experts: *worst possible*, *very poor*, *satisfactory* and *best possible*. The quality KPIs are scaled to interval [0, 1] corresponding to the *worst* and the *best possible* performance, respectively. The values for *very poor* and *satisfactory* KPI levels form corner points to continuous piecewise linear scaling. In this example, these values are scaled to 0.2 and 0.9 respectively. The scaling function parameters can be adjusted to different performance indicators, different networks and target performance levels. In this example, the values 0.2 and 0.9 are used as reference parameters for scaling. The sensitivity of the results to these parameters is studied at the end of this Chapter. The scaling is exemplified in *Figure 5.1*.

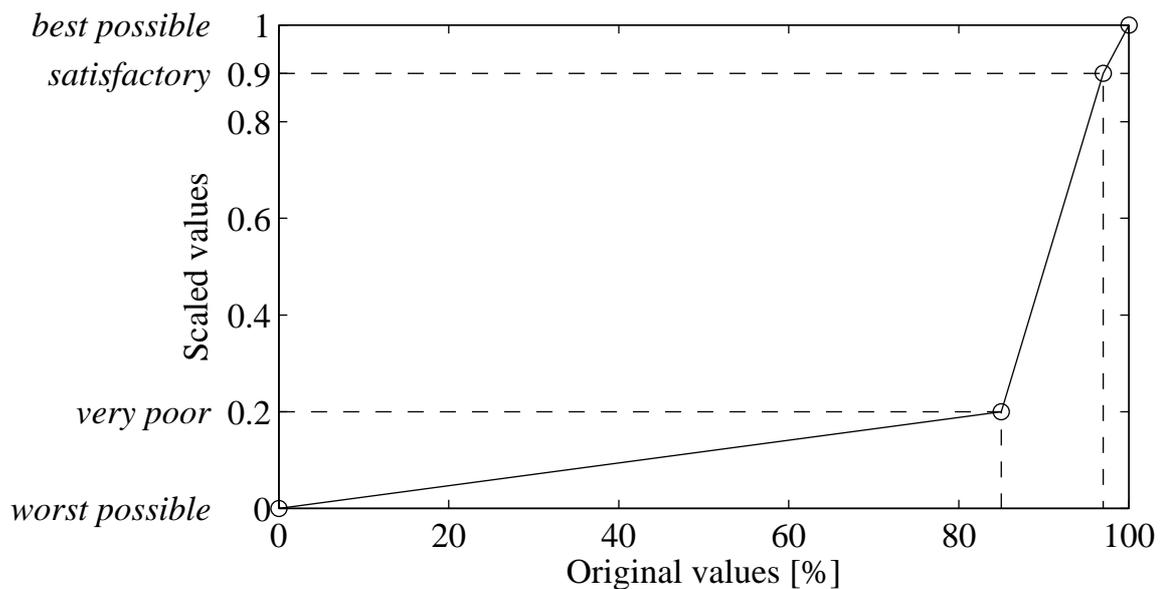


Figure 5.1 Piecewise linear scaling determined by the worst possible, very poor, satisfactory, and best possible values specified by process expert.

This scaling provides two advantages. First, the importance of the variables are equalised. The values of all the scaled performance indicators are within the same range, and the same value refers to the same level of performance in each indicator. Secondly, this scaling incorporates quality into the scaled data, providing meaning to the scaled values: 1 equals good quality and the further the distance from 1 the poorer the quality. This is a major benefit when interpreting results.

Four of the six KPIs in this use case are expressed on a percentage scale, ranging from 0% to 100%. DCR is a failure indicator, thus its best possible value is 0% and the worst possible is 100%. The values for *very poor* and *satisfactory* specified by a network expert are 10% and 1% respectively.

HO_succ, RX_DLqual and CSSR are success KPIs. Their best possible value is 100% and the worst possible is 0%. The *very poor* values of these KPIs are given as 85%, 85% and 82% and *satisfactory* values are 97, 95 and 98% accordingly.

The scaling functions of these four performance KPIs that are originally percentages are presented in *Figure 5.2*.

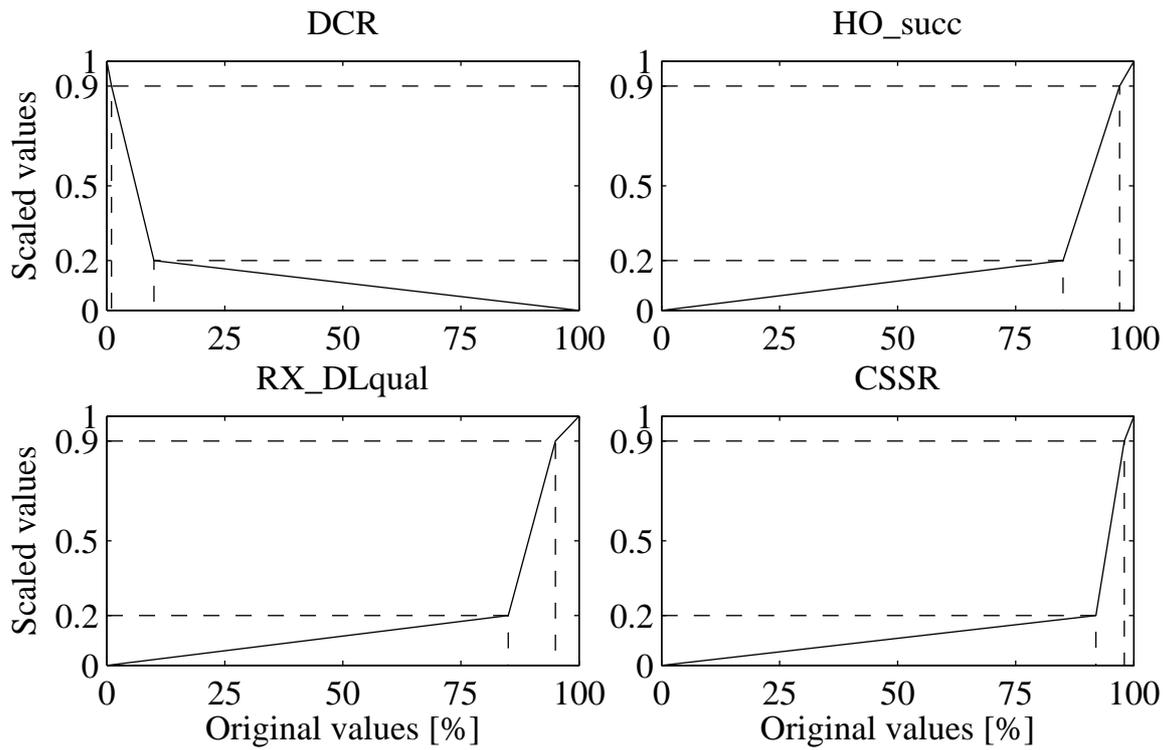


Figure 5.2 Piecewise linear scaling of four percentage type variables.

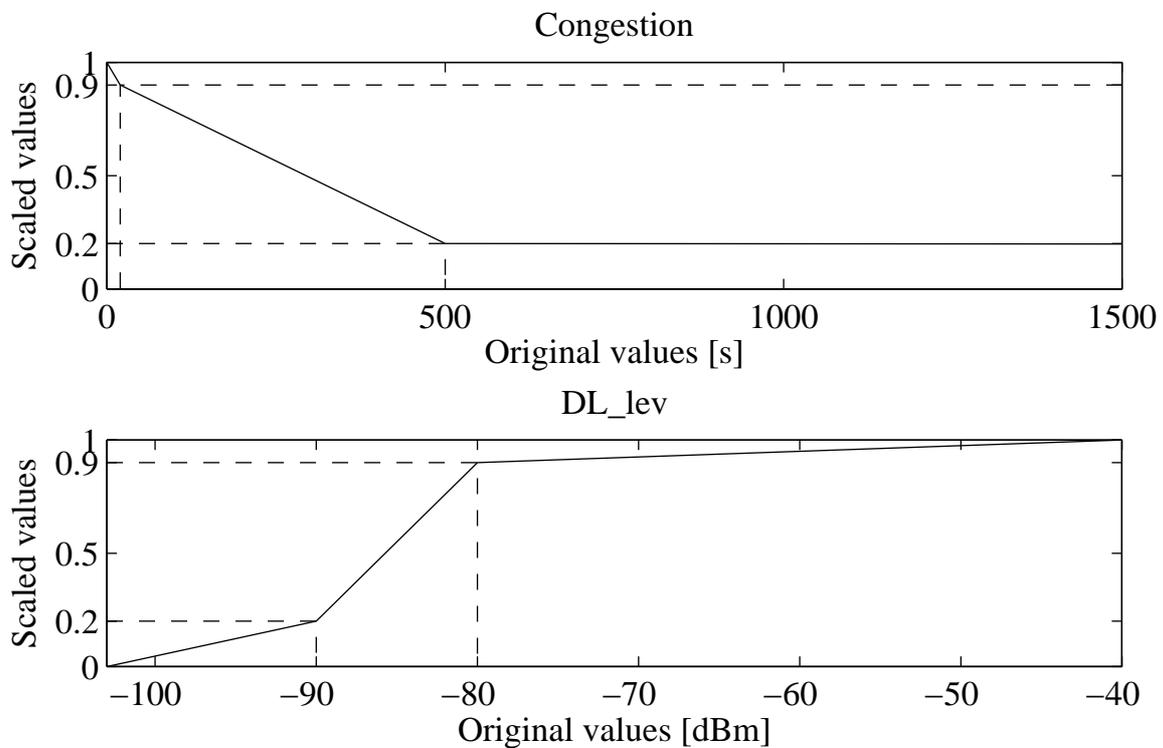


Figure 5.3 Piecewise linear scaling of variables with other than percentage scales.

The two remaining KPIs are not originally expressed in percentages. Congestion is in seconds and DL_lev in dBm, as depicted in *Figure 5.3*.

Congestion has its *best possible* value at 0 s, *satisfactory* at 20 s and *very poor* at 500 s. There is no upper limit for congestion but the *worst possible* is set to 86400 s (24 hours). Such congestion time could only mean that the cell is switched off. Therefore, analysing cells with higher congestion values on a daily time scale is not meaningful, and data from such cells can be ignored.

The *best*, *satisfactory*, *very poor* and the *worst possible* values for DL_lev are set to -40 dBm, -80 dBm, -90 dBm and -103 dBm.

5.2.3 Selecting the potential problems (distance based anomalies)

The anomaly detection is based on the Euclidean distance from the known normal state, which is also known to represent optimal performance. In the scaled space the normal state is a vector of ones. The cells that have the highest distances from the normal are considered potential problems. The worst five per cent of the cells are selected for further analysis. In this use case, this is 119 cells.

5.2.4 Clustering the problem cells

The information about the problem cells is further summarised for the experts by clustering. The selected cells are clustered into six clusters, referred to as Problem Groups. Any clustering method can be used. This example is constructed using hierarchical clustering with Ward linkage [Ward 1963]. Boxplots [McGill et al. 1978] of the data in each problem group are presented in *Figure 5.4*, followed by the expert's analysis of the problems and possible corrective procedures.

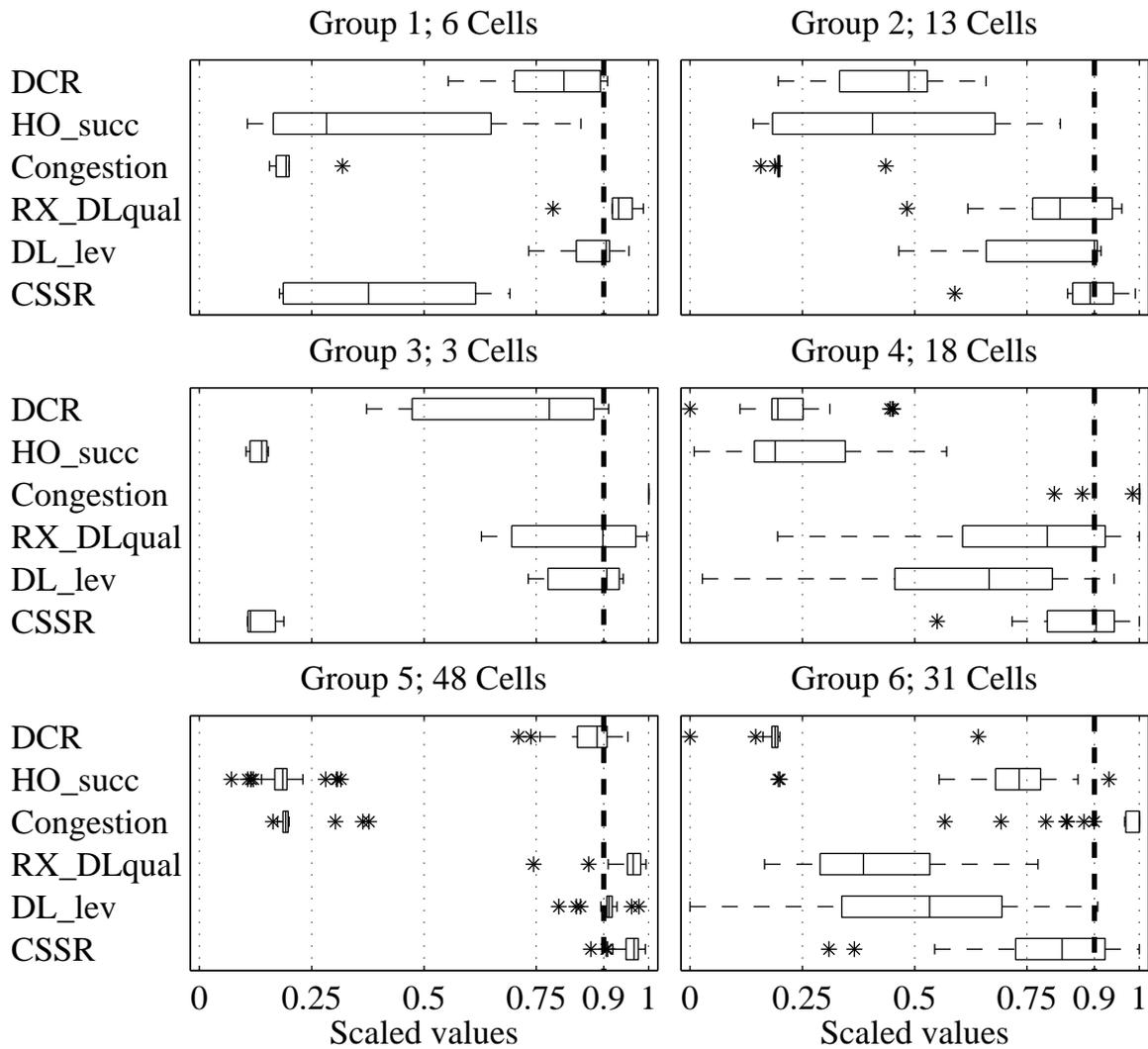


Figure 5.4 Box plots of clustered anomalies after expert scaling.

The main problems in Group1 are caused by congestion, which leads to failures in handovers and poor setup success. Additional radio capacity for cells, or possibly a change of capacity configuration parameters, could fix this problem.

Congestion also causes the problems in Group 2. The setup success is mostly satisfactory but the number of dropped calls is high (poor DCR). Signal level and quality are at a lower performance level than in Group 1.

The most distinctive features in problem Group 3 are poor handover success and call setup success rate. One possible root cause for the problems is hardware failure in the cell or in the transmission links. Another possible cause is a lack of signalling capacity

in the cells. These cells should be checked for failures in hardware and transmission links. In some cases resetting the cell may solve the problems. Fortunately, the number of these cells is low.

The cells in Group 5 also suffer from congestion. In this group that causes poor performance in handovers only.

The cells in Groups 4 and 6 have a high number of dropped calls and reduced success in handovers. The signal levels and radio quality are poor, and are most probably caused by poor coverage. Typically, changing the frequencies of the problematic cells solves these problems. In order to find the problematic frequencies, the situation should be studied on a geographical map. Adding new cells or changing the antenna bearings or tilts might also help.

5.2.5 Alternative views on problems

Problems can be presented to the user from many perspectives. Two tables are presented here as examples. With appropriate user interfaces in the application they allow easy access for investigation of the most critical and interesting parts of the network.

A list of cells sorted by the distance from the normal state is presented in Table 5.2. This allows the operator to concentrate on the cells that have the poorest performance.

Table 5.2 Table of top 10 problems of expert scaled data.

Rank	Distance	Cell ID	Cell Class	Problem group
1	1.711	2070	Macro, Low traffic, Low HO	4
2	1.571	1194	Macro, Low traffic, Low HO	6
3	1.547	884	Macro, High traffic, Low HO	4
4	1.546	1199	Macro, Low traffic, Low HO	6
5	1.541	1350	Macro, High traffic, Low HO	1
6	1.508	617	Indoor	6
7	1.503	1126	Macro, Low traffic, Low HO	6
8	1.473	713	Macro, Low traffic, Low HO	6
9	1.469	885	Macro, Low traffic, Low HO	4
10	1.454	310	Macro, Low traffic, Low HO	4
...

Table 5.3 presents the numbers of detected problem cells in each cell class and the problem group. The distribution of problems in the cell types follows the distribution of the cells in the whole data set.

This table provides the user an easy way to select a set of cells for more detailed analysis.

Table 5.3 Distribution of the poor performance cells in the Problem Groups and the cell classes.

	Group	1	2	3	4	5	6
Cell class	sum	6	13	3	18	48	31
Micro	6	0	0	0	0	5	1
Indoor	2	1	0	0	0	0	1
Macro, Low traffic, Low HO	67	2	10	1	9	23	22
Macro, Low traffic, High HO	8	0	2	1	1	3	1
Macro, High traffic, Low HO	10	1	1	0	3	4	1
Macro, High traffic, High HO	26	2	0	1	5	13	5

5.2.6 History of the cells

It is not always possible to perform this type of detailed analysis on a daily basis, and it is done only occasionally. In such cases, viewing performance during a longer period in history is necessary.

One possibility is to use data from a longer period in identifying the anomaly threshold and the problem groups. An example of such a case is presented by Kumpulainen et al. [2009] where a period of six weeks is used, concentrating only on micro cells.

In this use case, the anomaly threshold and the problem groups are identified using the data from the previous day only, as described earlier in this chapter. Each day in the earlier history of the cells is compared to that threshold. If a day is found to fall outside the threshold, it is assigned to the problem group with minimum distance from its centre.

Performance in the previous two week period of cells 2070, 1194 and 1350 is exemplified in *Figure 5.5*. These cells were ranked as first, second and fifth in Table 5.2. Problem Group 0 represents the “no problem” group.

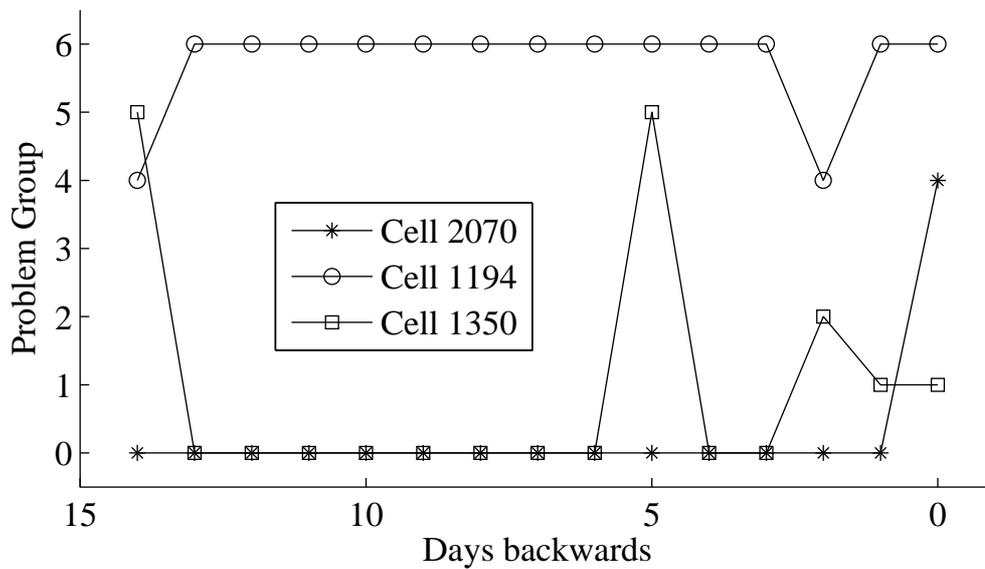


Figure 5.5 Three cells assigned to the Problem Groups in the previous two weeks.

Cell 2070, which had the poorest performance during the previous day, has performed well during the two week period. The recent poor performance suggests that something has changed in that cell.

Cell 1194 is an example of constant poor performance. It alternates between problem groups 4 and 6, which represent similar behaviour. This is most likely caused either by limitations due to the geographical position of the cell or suboptimal configuration of the cell, which should be checked.

Cell 1350 presents an example of a cell that has occasional poor performance. The types of problems vary: the five days of poor performance are spread across three problem groups.

5.3 Comparison using standardised data

This section presents the suggested anomaly detection and problem group identification procedure using normalised data without expert knowledge. Normalising the data to zero mean and unit variance, specified in Eq. 3.8, is one of the most commonly used scaling methods. The mean and standard deviation values of each variable are calculated from the data, and expert knowledge about the process is not required. In anom-

aly detection, the mean is considered as the normal state, and large deviations from that are judged to be anomalies.

In the expert scaling, the known optimal state is used as the normal state. In the normalised case, the mean of the data (zero vector in the scaled space) is used as the normal state. Histograms of the distances from the normal state are presented in *Figure 5.6*.

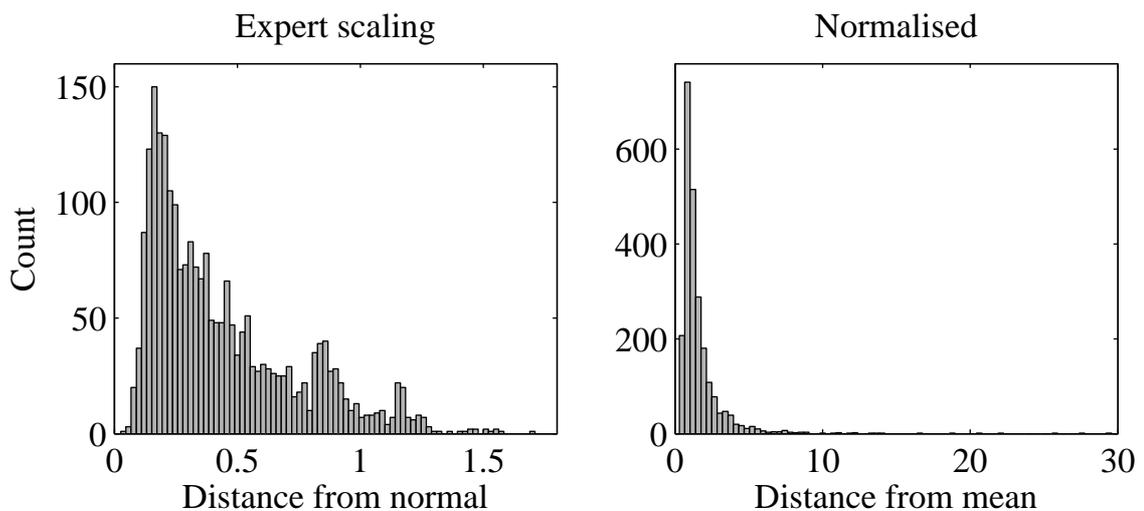


Figure 5.6 Histograms of the distances from the normal state.

The distribution of the distances in normalised space has a very long tail with individual observations that have very large distances. The distances in the expert scaled space are distributed more evenly across the range; the larger distances are compressed closer to the mode. The distribution has local maxima, suggesting that there are groups of observations at the same distance from the normal.

The five per cent of the cells that have the largest distances from the mean are selected for further analysis. They are clustered in the same way as the expert scaled data earlier described. Hierarchical clustering with Ward linkage was used to identify six problem clusters. The boxplots of the clusters are presented in *Figure 5.7*.

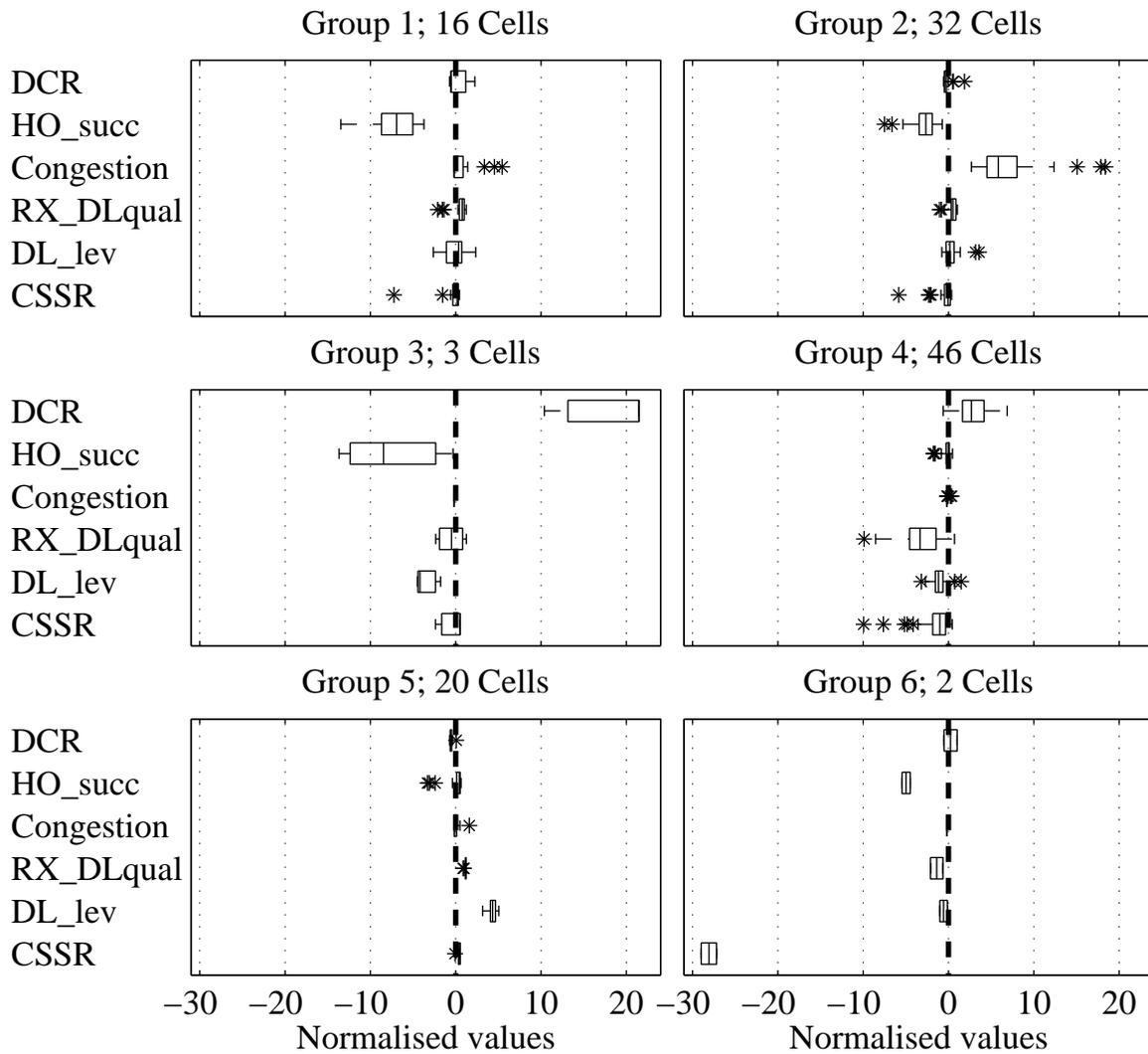


Figure 5.7 Boxplots of clustered anomalies, which were detected from normalised data.

The characteristics of the groups are more difficult to interpret than those after expert scaling. The scaled values of the KPIs are not comparable. Whether a high value represents good or poor performance depends on the KPI: high handover success represents good performance, whereas high dropped call ratio or congestion represents poor performance.

Some characteristic performance patterns can be found. Group 1 has reduced handover performance but no reason for that can be seen. Group 2 has high congestion, causing a drop in handover success. However, it is impossible to estimate the severity of the problem without reviewing the original data. Group 3 has a low signal level,

causing dropped calls and problems in handover success. Group 4, which contains the majority of the cells, has an increased dropped call rate caused by slightly reduced signal quality. Group 6 has only two cells with poor setup and handover success. They are probably caused by reduced signal quality, but judging the significance of the quality level requires a review of the original values.

The main feature of Group 5 is a high signal level. This group contains cells that have otherwise average performance, but exceptionally high signal levels. Therefore, this group does not reveal cells that have problems. However, they deviate from the mean and are therefore detected as anomalies. Without incorporating any prior knowledge it is impossible for any AD method to detect only poor performance anomalies.

A scatter plot of two variables in expert scaled space is presented in *Figure 5.8*. The piecewise linear expert scaling compresses the long tails of the distributions into more compact clusters.

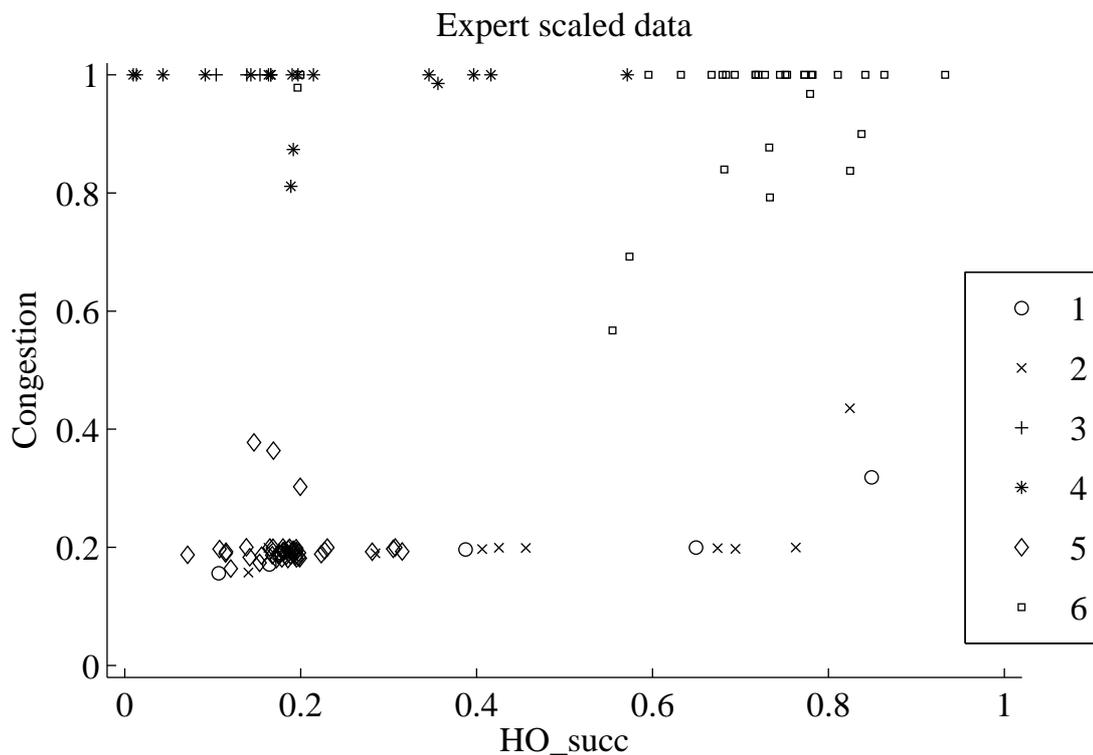


Figure 5.8 Scatter plot of two variables in a priori scaled space. Problem groups specified in the legend.

The scatter plot of normalised KPIs in *Figure 5.9* shows how the long tails form sparse clusters. The bulk of the data is concentrated in a very dense group with no significant distinction.

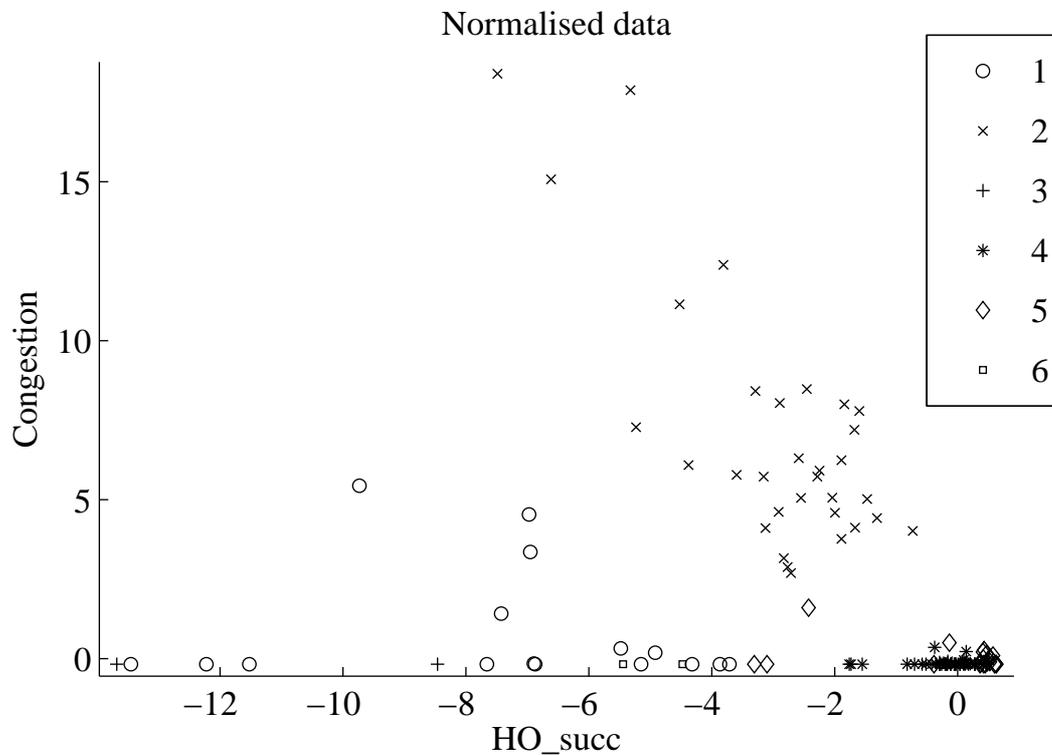


Figure 5.9 Scatter plot in normalised space. Problem groups are specified in the legend.

5.4 Parameter sensitivity

Values 0.2 and 0.9 for the *very poor* and *satisfactory* levels of KPIs have been used as reference values in the expert scaling. The sensitivity of the result to these parameters is studied by varying them and by comparing the sets of the most severe anomalies. Values 0.1 and 0.2 are used for the *very poor* level and 0.8 and 0.9 for the *satisfactory* level. This yields four combinations of the parameters.

The number of common cells compared to the reference values is presented in the following table. The last row contains the comparison with the normalised data.

Table 5.4 Number of common cells within the poorest performance. Comparing expert scaling parameters for {very poor, satisfactory} and normalised data.

Reference {0.2, 0.9}	Rank of the #1	Top 5	Top 10	Top 20	Top 50	All 119
{0.1, 0.9}	3	4	9	19	47	118
{0.1, 0.8}	3	4	9	19	42	116
{0.2, 0.8}	1	5	10	20	44	116
Normalised	3	2	5	8	29	78

The worst cell in the reference scaling is in the top three in all the other scaling methods. The expert scaling is not very sensitive to the parameters; the vast majority of the most severe problems remain the same regardless of the parameters. However, sets produced using normalised data contain only half of the cells given by the reference.

5.5 Discussion

This chapter has presented an example of a simple distance based AD method. It presents how the clustering of anomalies into problem groups provides valuable information for the detection of misbehaving cells in a mobile network, and how expert knowledge can be integrated by piecewise linear scaling. The proposed scaling equalises the importance of the variables and provides a clear meaning of good and poor performance in interpreting the results. This is a major benefit when compared to the results produced from normalised data. Visual inspection of subspaces reveals that the expert scaling equalises the distributions, compressing the long tails, and provides natural clusters of the anomalies, whereas the normalised data leaves the anomalies scattered loosely.

Unfortunately, there are no universally correct answers available for this kind of task. Therefore, an inarguable comparison of methods and the effects of scaling is impossible. Judgement is based on the experience of the user, in this case the network operator's radio expert. The distance from the known optimal state provides the required

ranking of the anomalies, The problem groups of normalised data reveal mostly deviations in single variables, and the meanings of the deviations are left ambiguous. However, the problem groups of the expert scaled data provide unambiguous interpretation of good and poor performance. The operator's radio experts could easily discover explanations and corrective actions for the problem groups. Clustering the expert scaled data summarises the main characteristics of data, and the results were found to be interesting and understandable and meaningful, which are the ultimate goals of clustering and unsupervised anomaly detection [Everitt et al. 2001; de Oliveira & Pedrycz 2007; Singh & Upadhyaya 2012; Zimek & Vreeken 2013].

The proposed expert scaling method was also tested for its parameter sensitivity. Changing the scaled values of the *very poor* and *satisfactory* performance levels had only a minor effect on the ranking of the anomalies. The scaling method seems to be very robust and therefore also easily applicable in other application domains where similar performance indicators are available.

This procedure has been applied to 3G data, comparing this simple, distance based anomaly detection with OC-SVM [Kumpulainen et al. 2011]. The anomalies were clustered into problem groups and analysed by process experts, as this chapter proposed. Some problem groups were detected by both methods (about half of the anomalies). The rest of the detected problem groups were specific to each of the methods. Thus these methods complement each other in anomaly detection, providing the end user with a wider view of anomalous behaviour in the network. These results encourage the application of *combinations* of methods, as suggested in Section 4.4.11.

Chapter 6: Local anomalies in network management

This chapter presents an example of local anomaly and novelty detection procedure using data from server logs in OSS (*Figure 2.1*). First, the AD model is identified from the reference data set. The model is then used online to detect anomalies and novel states. The data are scaled by an application specific method, which is developed with the network experts. The AD model developed in this work is based on SOM and clustering. In addition to local anomaly detection, it provides summarised information about the normal behaviour of the system. The results are compared to the ones produced by its simplified global version and other local AD methods. The sensitivity to the parameters in each of the methods is studied. This use case covers two of the presented methodological contributions.

6.1 Objective

A large number of servers is used to monitor and control a mobile network. The servers themselves need to be monitored for performance and security purposes. Information about various events, such as automatic system processes and manual operator activity, are collected in log files. Major system malfunctions leave clear traces in system logs: processes can for example start to produce huge numbers of log entries or they can stop logging completely. Either one of these cases results in anomalous logging activity compared to normal logging behaviour. Furthermore, anomalies in system component behaviour, user activity or appearance of new software may be signs of a security incident.

The procedure presented in this chapter helps in developing applications for two purposes. The first objective is to monitor and analyse the normal activity of the OSS servers. The second objective is to detect anomalous events which can be malfunctions, security risks or new usage patterns. The anomalies have to be ranked so that

the further study of the most significant one can be prioritised. In order to discover the root cause of the anomalous behaviour, it is required that the contributions of the variables are presented.

6.2 Use case

The procedure developed is a local AD method that is based on a combination of one-dimensional SOM and clustering. The details of the method are published by Kumpulainen and Hätönen [2007, 2008a] and the results when applied to the synthetic example were presented in Section 4.4.8. An application specific scaling method is used to enhance the identification of the normal states and local anomalies. This method is also included in an application prototype developed for the comparison of AD methods [Kumpulainen & Hätönen 2008c].

The procedure consists of offline and online phases. The offline phase uses existing history data and includes identification of the scaling parameters and the AD model, as well as an analysis of the behaviour of the network servers in normal operation. The online phase consists of using the AD model to detect anomalies or novel behaviour in the new collected data. The time scale in the online monitoring may be hours, days or weeks. The details of the phases are listed below.

Identification of the anomaly model offline.

1. Scale the data using robust logarithm scaling.
2. Identify normal states by SOM, ignore map units with less than a specified number of hits, and assign new BMUs to the observations that were assigned to any of the ignored BMUs.
3. Cluster the SOM code vectors to create *reference groups* (RG).
4. Calculate the local anomaly threshold for QE in each RG.
5. Verify the anomaly model by analysing the reference (history) data.

Using the anomaly model online.

6. Scale the new data with the parameters identified in the offline stage.
7. Assign the data to the BMUs and corresponding RGs.
8. Compare the QE to the anomaly threshold of the RG.

6.2.1 Data

The data set has been collected from network management system servers in a small network used for new mobile technology field tests. The data is extracted from a management application server, which controls about a dozen base station controllers providing a full range of network services. About a million log entries are recorded per day. The log entries have been classified by their source application. The data set is constructed as a time series by counting the log entries in each of the classes. The time resolution in this use case is one hour, providing 24 hourly counts for each log class. Seven log entry classes used as variables in this case are listed below. The short names in parenthesis are used in the remaining of this Chapter.

- System log activity (SysLog)
- Authentication activity (Auth)
- Application log activity (App)
- Cron activity (Cron)
- Login activity (Login)
- Remote sessions to network elements (RSess)
- Remote commands to network elements (RComm)

The data set consists of a six week period. In order to simulate the two phases the data set is divided into two parts. The first five week period, a total of 840 observations per variable, is used in the offline phase and referred to as the reference data. The last week of the measurement period, 168 observations, is used in the online phase as new data.

6.2.2 Scaling

For system log data a robust logarithmic scaling that preserves the importance of the variables is used. The scaling first takes a natural logarithm of the variable plus one and then divides by a robust standard deviation.

$$(6.1) \quad x_s = \frac{\ln(x + 1)}{s},$$

where s is the standard deviation, which is calculated using only the observations with some activity, ignoring zero values and also one per cent of the extreme values from the upper tail. The final scaled variable z_i is achieved by subtracting the mean.

$$(6.2) \quad z_s = x_s - \bar{x}_s$$

The effect of scaling on system and application log activity is exemplified in *Figure 6.1*. Syslog has much less variation, with the exception of the three observations with remarkably high values. The conventional normalisation to zero mean and unit variance (Eq. 3.8) is presented in the middle. This scaling is affected by outliers in the data, and possibly hides meaningful variation in this case. The direction of the x variable is dominated by the three outliers, and the rest of the variation in this direction is shrunk to almost negligible. The scatter plot on the right in *Figure 6.1* presents the robust logarithmic scaling used for the log activity data in this case. The small scale variation of syslog is brought out, but the three observations that have higher values are still clearly separated, as well as the group of high values in App.

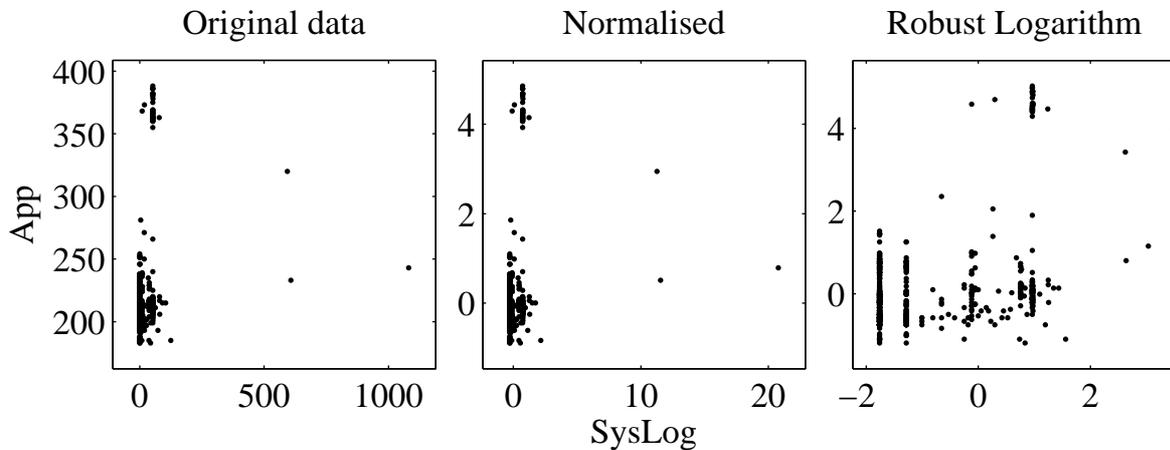


Figure 6.1 Effect of scaling on scatter plots of system and application log activities.

The data used throughout this use case are scaled using robust logarithm scaling. A robust standard deviation and a mean calculated from the reference data set are used for scaling the test set.

6.2.3 Model identification

The AD model is identified using the reference data. The SOM is identified using the SOM Toolbox for MATLAB [Vesanto et al. 1999]. The code vectors are initialised according to the first principal component. The number of map units in the SOM is selected as $N/5$, where N is the number of observations. In this case $N=840$, yielding 168 map units.

Quantisation errors can be used as a simple global AD method. The assumption that five per cent of the data is anomalous gives a global anomaly threshold of 1.84. A histogram of the quantisation errors of the reference data and the anomaly threshold are presented in *Figure 6.2*. This model is used as a global SOM method in the comparison later in this chapter.

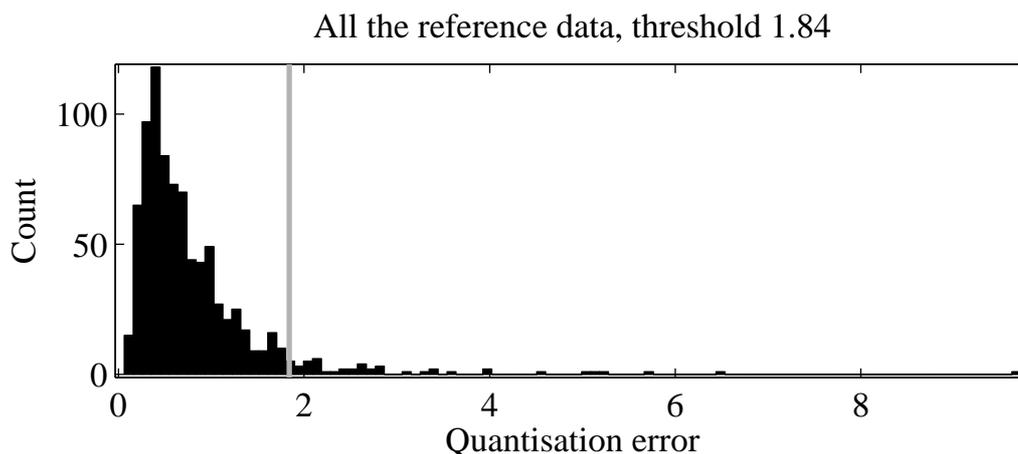


Figure 6.2 Histogram of the quantisation errors of the reference data.

In the local AD method, the map units that do not have a specified number of hits (observations having the unit as their BMU) are ignored. In this case, a minimum of three hits is required. After excluding nodes with less than three hits, the SOM ends up with 132 nodes.

The next step is to form the reference groups by clustering the code vectors of the remaining map units. Any clustering method is applicable. Hierarchical clustering with Ward linkage is selected in this case [Johnson & Wichern 1998, Ward 1963]. The number of clusters is determined by the maximum Davies–Bouldin index [Davies &

Bouldin 1979]. In this case, the number of clusters was limited to between three and ten, and the minimum index was achieved with seven clusters. The clusters are referred to as reference groups.

6.2.4 Offline usage: analysing the reference data

Analysing the reference data serves two purposes. It provides information about the normal behaviour of the system. The second purpose is to verify the model. A network expert should verify that the RGs and the anomalies detected from the reference data are meaningful. Firstly, this section presents examples of analysing the identified normal states according to three levels of detail, followed by examples of the anomalies in the reference data.

Normal states in the reference data

The centres (mean values of the variables) of the RGs provide a very condensed presentation of the main characteristics of the groups, as shown in *Figure 6.3*. The number of observations assigned to each RG are shown below the number of the group.

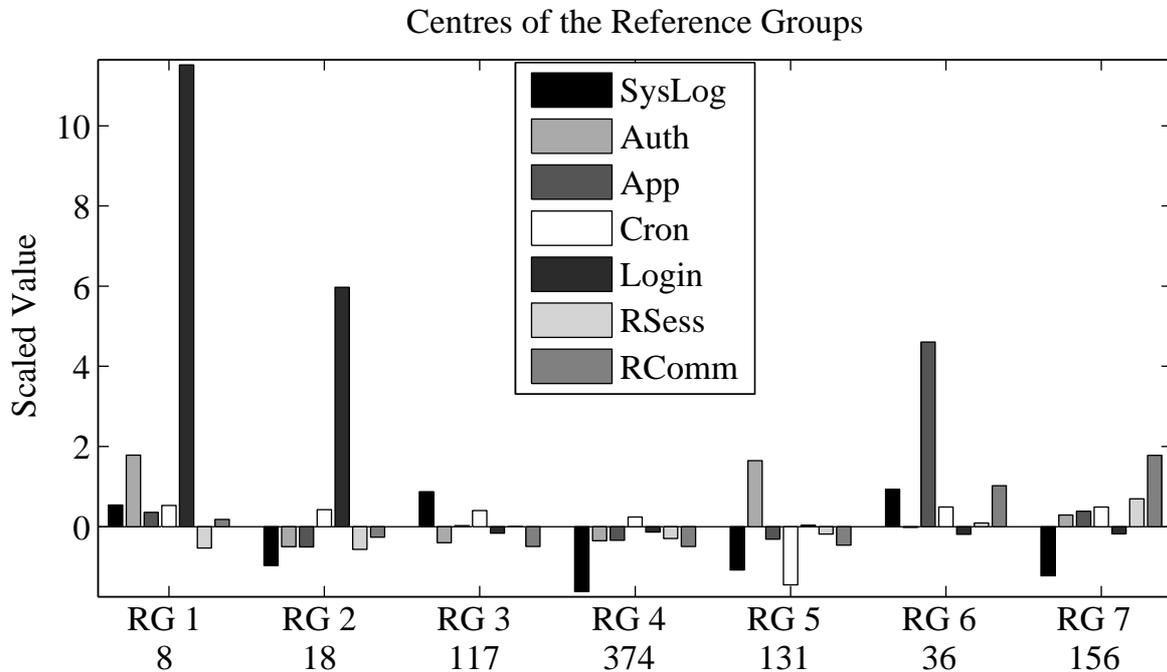


Figure 6.3 Centres of the Reference Groups.

Each RG has distinct characteristics, indicating that the number of groups is not too high. The reference groups 1 and 2 are small with only eight and 18 observations assigned to them. RG 6 is also relatively small with 36 observations. A high value of Login is the main characteristic in both RG1 and RG2. They differ on the levels of SysLog, Auth and App, all of which are above average in RG 1, but below average in RG2. Groups 4, 6 and 7 are the largest and have a low value of SysLog in common, but levels of Auth and RComm separate them.

The second, more detailed presentation of the normal states is provided by box plots, which are able to visualise variation within the RGs. Separate box plots of the code vectors of the SOM for all seven RGs are collected in *Figure 6.4*. The numbers of map units in each RG are given in parentheses in the titles.

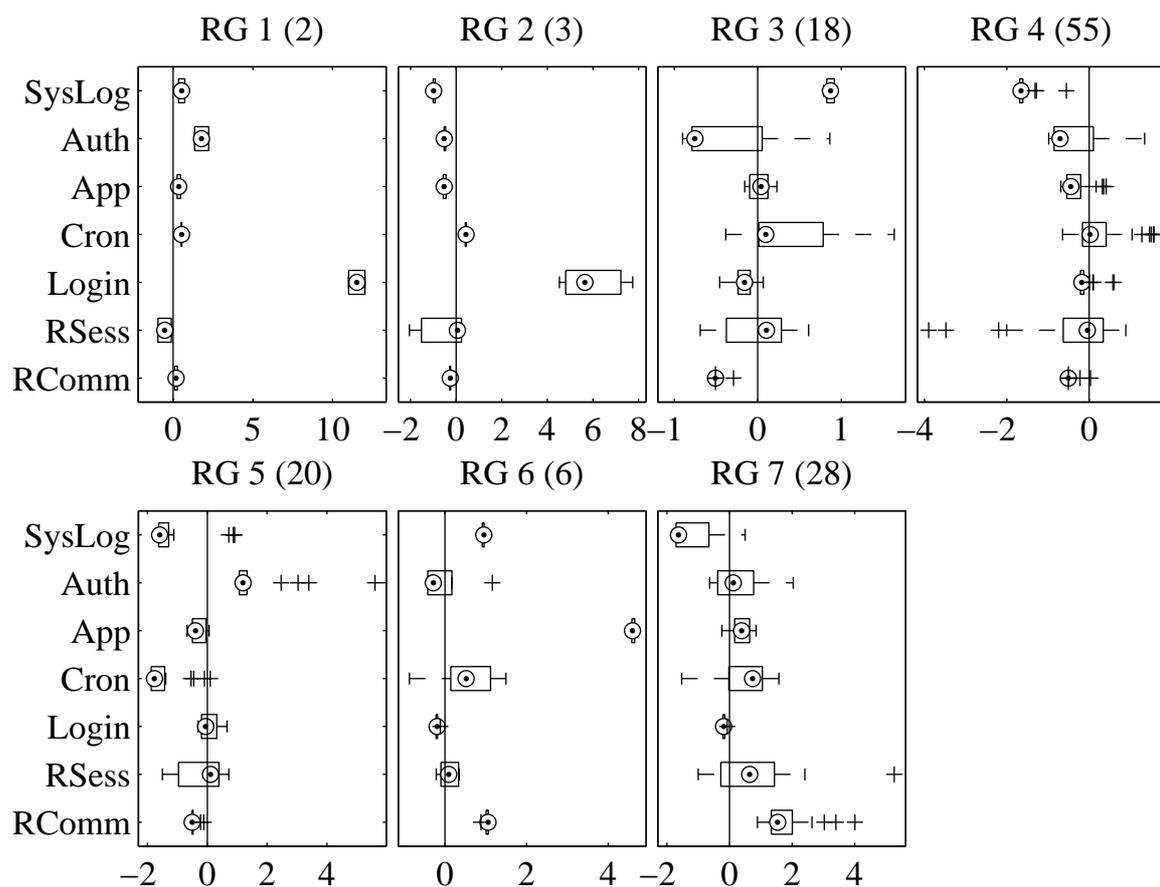


Figure 6.4 Box plots of the SOM code vectors in the reference groups.

The medians of the code vectors, highlighted with circles, are very close to the mean values of the data, shown in *Figure 6.3*. RGs 1 and 2 are represented by only 2 and 3 map units and have only minor variation within the groups. The highest variations are introduced by RSess in RG 4, Auth in RG 5 and RComm in RG 7. In addition, one map unit in RG 7 has a distinctively high value of RSess. All map units have at least three hits from the data, as specified in the identification.

The topology of the SOM is used in the third visualisation, providing the most detailed view of the normal states. One of the advantages of one-dimensional SOM compared to the more popular two-dimensional SOM is its more compact visualisation [Kumpulainen & Hätönen 2012]. Two-dimensional component planes are replaced by component lines which can be shown in one plot, as presented in *Figure 6.5*. The horizontal axis covers the map units from one to 168, as presented in the synthetic example. The vertical lines are the borders of the RGs.

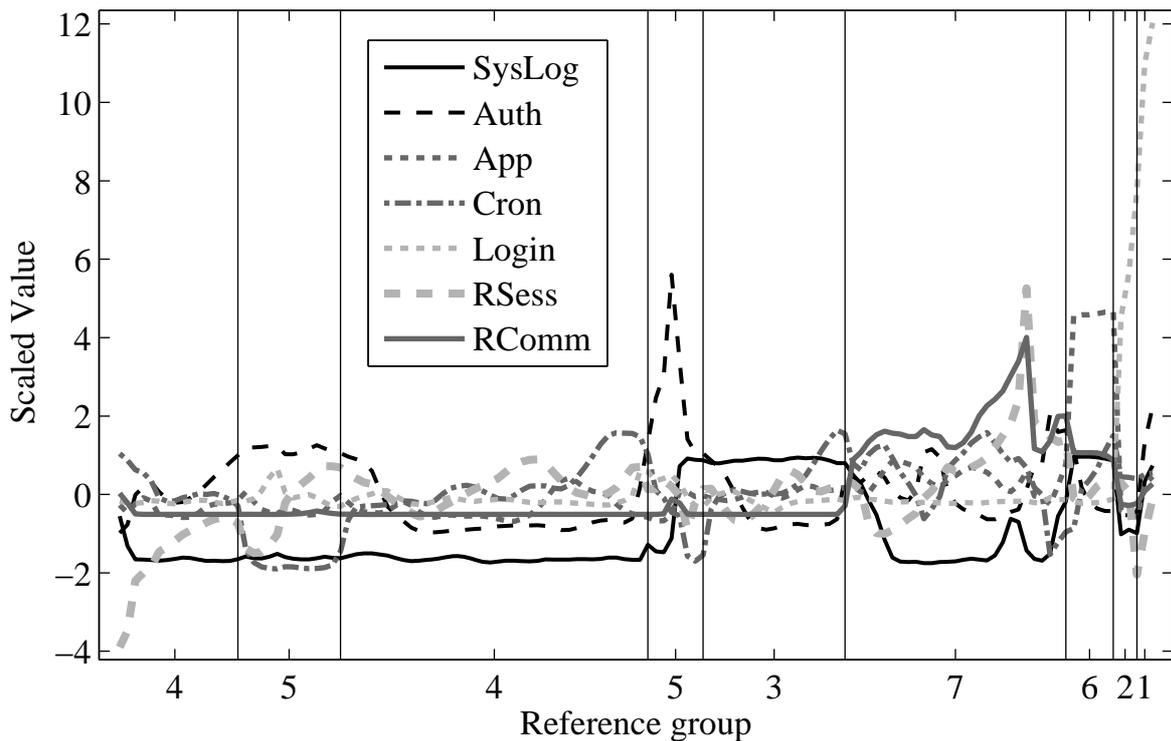


Figure 6.5 Component lines of the one-dimensional SOM.

In addition to the variation within the groups, the component lines show the combinations of the variables in each map unit. The low values of RSess in RG 4 occur to-

gether with the high values of Cron. The single high value of Auth in RG 5 presents a state where all other variables are very close to the mean value.

Anomalies in the reference data

The anomaly threshold for each reference group was determined as the 95th percentile of the quantisation errors in each group. The histograms of the quantisation errors in the reference groups (RG) are presented in *Figure 6.6*. The number of observations assigned to the groups is given by the label on the vertical axis, and the group specific local thresholds are depicted by vertical lines.

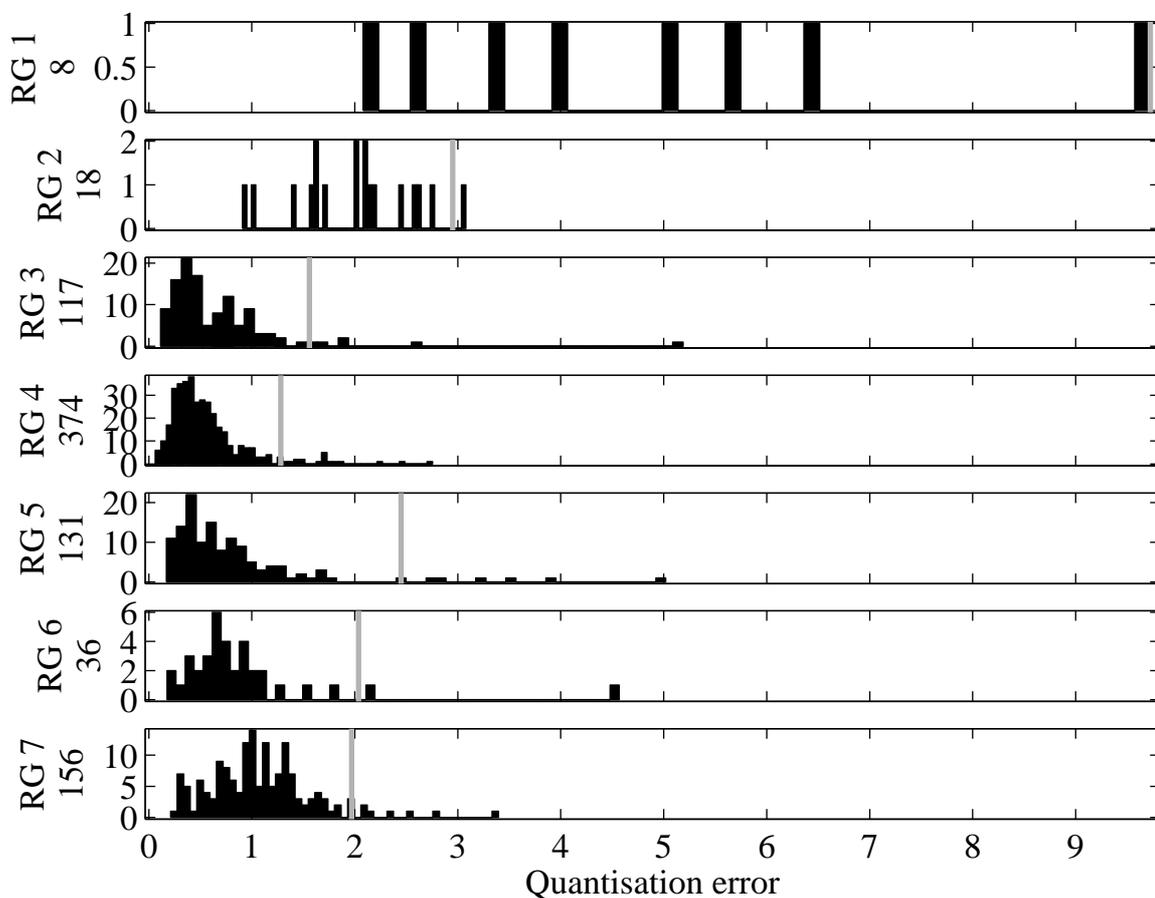


Figure 6.6 Histograms of the quantisation errors of each reference group.

The small groups 1 and 2 represent process states that do not occur very often, yet too often to be considered as anomalies. However, this local anomaly detection method gives an indication of anomalous behaviour in the form of a small reference group.

These groups can represent a rare, possibly acceptable or even desirable behaviour in the process to learn about, or they could be caused by a sustained malfunction that requires immediate attention. In either case, they should be studied further by network experts.

A time series plot is a very common type of visualisation. One day of the reference data set is presented in *Figure 6.7*. The detected anomalies are highlighted with vertical lines and the associated RG for each observation is marked.

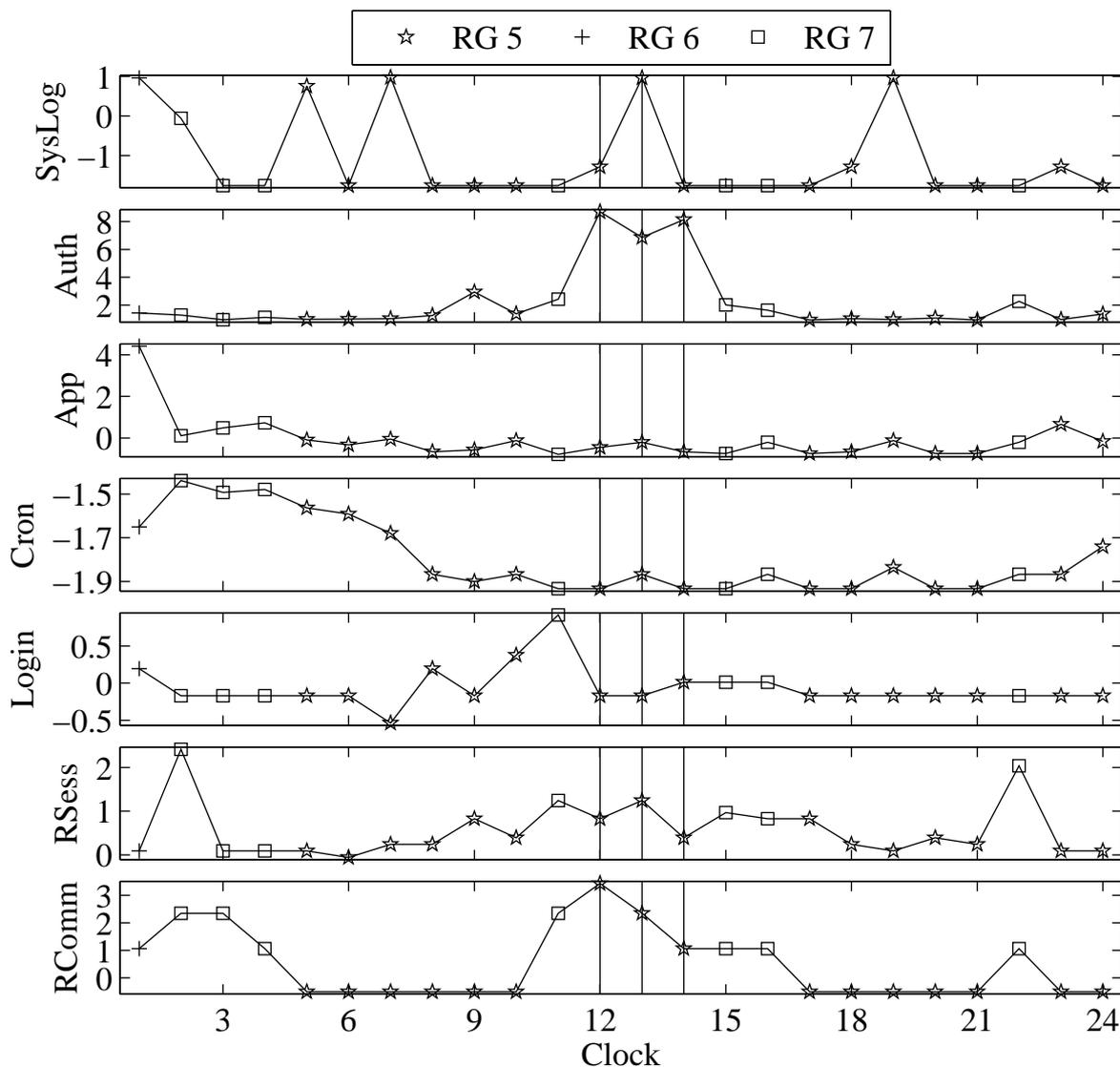


Figure 6.7 Time series plot of one day. Detected anomalies are marked by vertical lines and the reference groups by marker types.

All the observations of that day are assigned to RGs 5, 6 and 7. Anomalies are detected at three consecutive time instances starting at noon. All three anomalies are assigned to RG 7. The contributions of the variables can be assessed by the SOM error, which is the difference between the observation and the code vector of its BMU. The contributions are presented in the following table. High values of Auth and RComm combined with low values of Cron are common to all anomalies. In addition, the one at 13:00 has high values of SysLog and RSess. The contributions are not related to the global values of the variables, but to the nearest local normal state in the corresponding RG.

Table 6.1 Contributions of the variables to the anomalies.

Time	SysLog	Auth	App	Cron	Login	RSess	RComm
12:00	-0.16	3.10	0.06	-1.48	-0.66	0.62	3.55
13:00	2.08	1.25	0.31	-1.41	-0.66	1.04	2.46
14:00	-0.64	2.53	-0.15	-1.48	-0.47	0.19	1.18

6.2.5 Online usage: analysing new data

The test data set is from a period of one week (168 observations) following the reference set. The test set is logarithmically scaled using the robust standard deviation and mean values that were calculated from the reference data. The BMUs are searched for the observations, and the RG of the BMU is assigned to each observation. The quantisation error of each new observation is compared to the local threshold in the assigned RG. If the QE exceeds the threshold, the observation is considered as an anomaly.

The QEs of the test set are presented as a dotted time series in *Figure 6.8*. The local anomaly threshold for each observation is depicted by a solid line. The anomalies are highlighted by circles. The global anomaly threshold (see *Figure 6.2*) is given as a dotted line. The SOM is the same in both local and global detection, except that in the local SOM the map units with less than three hits are removed. Therefore, the QEs of the local model are mostly equal to the ones in the global model, but are higher for those few observations that initially had one of the removed map units as their BMU.

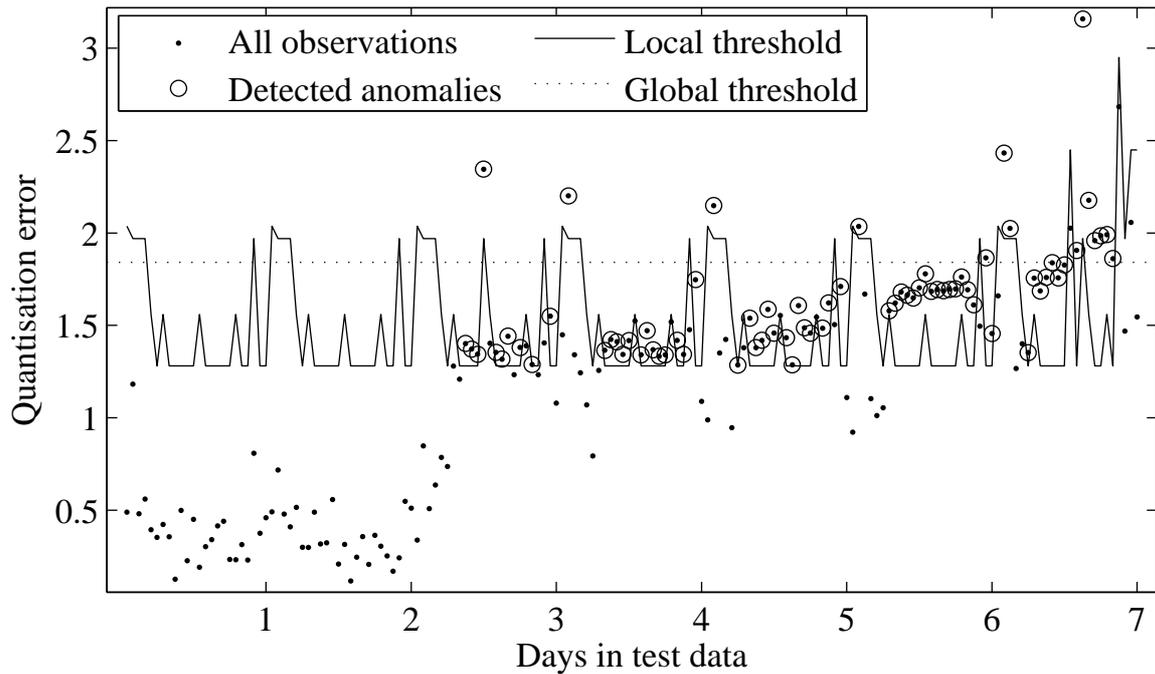


Figure 6.8 Quantisation errors and detected anomalies in the test data.

The distinct shift of the QE level on the third day at 9:00 am indicates a change in the behaviour of the process.

The local thresholds are exceeded by 73 of the observations after the third day, resulting in over 42% of the test set observations being detected as local anomalies. There are two possible reasons for such a high anomaly percentage. Either the behaviour of the system has changed significantly or the local AD model produces false positive detections. Only 17 observations have QEs that exceed the global threshold and are detected as global anomalies, and only five of them occur before the seventh day.

The contributions of the variables on the anomalies can be analysed to verify the origin of the anomalies. A box plot of the contributions in all 73 test set anomalies is presented in *Figure 6.9* (left). A low value of Cron is the main cause of the anomalies. The histograms of Cron values in both the test set anomalies and the reference data are depicted in *Figure 6.9* (right).

The Cron values of the anomalies are on the lower end but not uncommonly low globally. Thus the reason for the detection of the anomalies is a local combination of the variables.

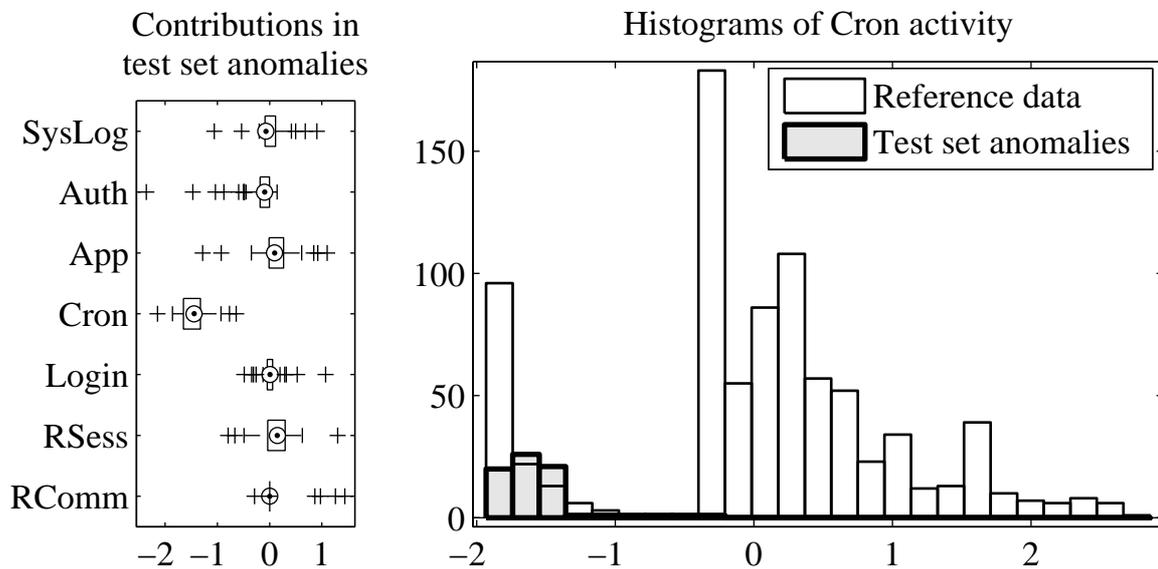


Figure 6.9 Contributions of the variables on the anomalies in the test set (left). Histograms of Cron in the reference data and in the test set anomalies (right).

Examining the time series of both reference and test data sets together reveals the main reasons behind the test set anomalies. At the end of the test date, the low Cron level occurs simultaneously, with a combination of other variables that has not been present earlier.

The most distinct factor is in Auth, as depicted in *Figure 6.10*. The reference and test periods are separated by a vertical grey line. Low Cron values occur during the third week in the reference period. At that time Auth values stay at higher level than in the rest of the measurement period, including the test set. During the third day of the test set the system enters a previously unseen state. The local anomalies in the test set are not false positives, but instead provide an early indication of this novel state of the system.

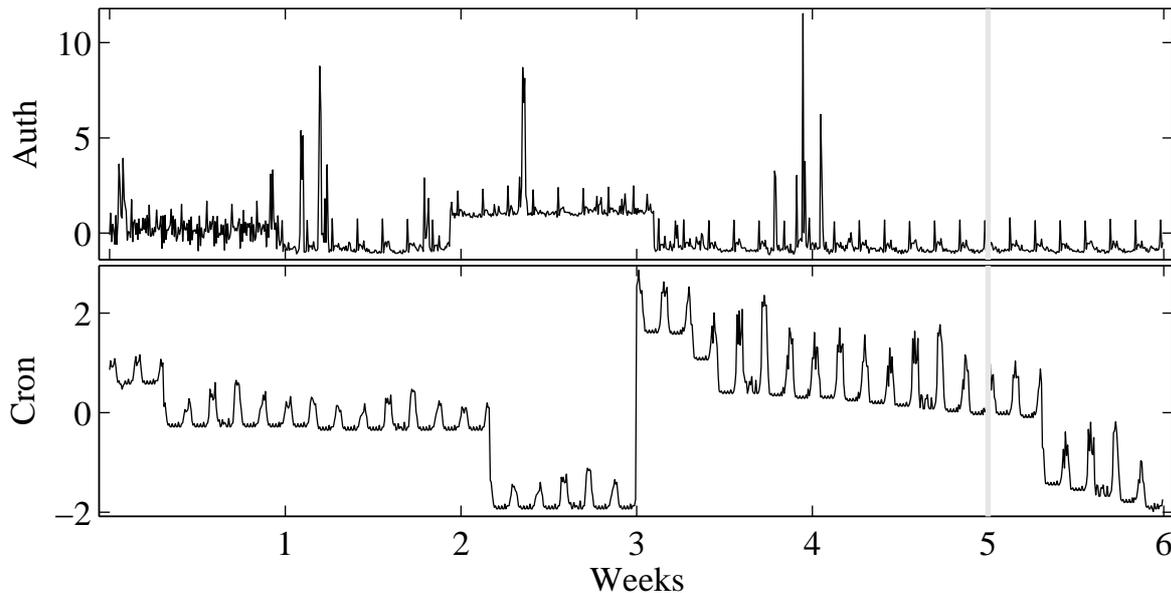


Figure 6.10 Time series of Authentication and Cron activities.

6.3 Comparisons with other methods

For comparison, anomalies are detected from the same data sets with four other AD methods. The numbers of resulting anomalies are compared to those provided by the local SOM and clustering based method presented above, which is referred to as L-SOM-C. The methods in the comparison are listed below.

- SOM with global threshold (G-SOM)
- Two Layer Clustering (2-LC)
- Gaussian Mixture Model (GMM)
- One-Class Support Vector Machine (OC-SVM)

The details of these methods are presented with the synthetic example in Chapter 4.4. In this comparison the methods are parametrised to resemble the parameters of L-SOM-C when possible. The anomaly threshold is set so that five per cent of the reference data is assumed as anomalous in all the methods.

The SOM in G-SOM is equal to the one used L-SOM-C but including all the map units, whereas in L-SOM-C the map units with fewer than three hits are excluded. The SOM has 163 map units, and a 95 percentile of the QEs is used as the global threshold.

Hierarchical clustering with Ward linkage is used in both phases of 2-LC. The clusters in the first phase are treated in a same way as the map units in the L-SOM-C. The number of clusters in the first phase is 163, equal to the number of map units in both SOM methods. The clusters that are assigned fewer than three observations are excluded, leaving 97 clusters. The observations are reassigned to the remaining clusters. The second phase is similar to the identification of the reference groups in L-SOM-C. The centres of the first phase clusters are further clustered. The number of clusters is selected from the range of three to ten. The minimum Davies–Bouldin index is achieved at nine clusters. The Euclidean distances from the nearest first phase cluster are calculated for each observation. A 95 percentile of the distances is selected separately for each second phase cluster, yielding nine local anomaly thresholds.

The GMM is identified using regularisation coefficient 10^{-6} , which provides reasonable convergence in 100 iterations. The number of components is selected from a range of between three and ten. The model identification is repeated five times for each number of components, starting from random initial values, and the one with the minimum log likelihood is selected. The model with ten components has the minimum AIC and is selected. The anomaly threshold is identified by generating a sample of $5 \cdot 10^6$ random observations according to the distribution identified, and by selecting the value of the PDF that includes 95% of the sample.

OC-SVM is identified using LIBSVM software [Chang & Lin 2001]. An RBF kernel is used, and γ is set as the inverse of the number of variables, which is the default value in LIBSVM, in this case $\gamma = 1/7$.

All the AD models are identified, using the five week reference data set which is scaled as described in section 6.2.2. Anomalies are detected from both the reference and test data sets. The numbers of the detected anomalies are compared to the ones produced by L-SOM-C. The results are collected in Table 6.2. The total number of

detected anomalies is given for each of the methods. Columns titled “Common with L-SOM-C” report the number of anomalies that are also detected by L-SOM-C.

Table 6.2 Numbers of anomalies detected by the methods in the comparison.

	Anomalies in the reference set		Anomalies in the test set	
	Total	Common with L-SOM-C	Total	Common with L-SOM-C
L-SOM-C	43	43	73	73
G-SOM	42	24	17	14
2-LC	41	15	9	7
GMM	40	12	3	3
OC-SVM	48	16	94	72

The number of reference set anomalies that are common with L-SOM-C is surprisingly low for all the methods. G-SOM is the only one that detects more than half of the anomalies detected by L-SOM-C.

The total number of test anomalies detected by 2-LC is nine, 5.4% of the test set, which is very close to that specified for the reference set. Seven of them are also detected by L-SOM-C. GMM detects only three anomalies in the test data. OC-SVM detects 94 anomalies in the test set, which is the highest number of all the methods. This set includes all but one of the anomalies detected by L-SOM-C.

According to this test, L-SOM-C and OC-SVM are the only methods that are sensitive enough to detect the novel behaviour of the system in the test period. 2-LC and GMM detect only the most extreme anomalies, while G-SOM detects more of the novelty. However, as was seen in *Figure 6.8*, most of the detections by G-SOM are concentrated into the very end of the test period. In hourly or even daily monitoring, this will indicate novel behaviour later than L-SOM-C or OC-SVM.

6.4 Parameter sensitivity

The sensitivity to the parameters of the methods is investigated by comparing the anomalies detected in the test data when the parameters are varied. This test is per-

formed on L-SOM-C, 2-LC, GMM and OC-SVM with nine sets of parameters in each method.

L-SOM-C is tested with three numbers of map units: $N/6$, $N/5$ and $N/4$, where N is the number of observations in the reference data. Three numbers of reference groups are also used: 7, 10 and 15. All combinations of these yield the nine test cases. The same numbers are used as the numbers of clusters in the first and second phases (P1 and p2) of 2-LC. The same numbers of the reference groups are used for the numbers of components in GMM. The second varied parameter in GMM is the regularisation coefficient, which is given values 10^{-6} , 10^{-5} and 10^{-4} , giving a total of nine combinations. OC-SVM is tested with nine values of the γ parameter in the RBF kernel. All parameter combinations can be seen in *Figure 6.11*.

The identification and detection procedure as described in 6.3 is repeated with each parameter combination for all the methods. The anomalies detected in the test data are presented in *Figure 6.11*. Time on the horizontal axis covers the seven days, with 24 hourly observations per day. Nine rows per method present the parameter combinations. The anomalies detected by a method are marked as black blocks at the corresponding time instant.

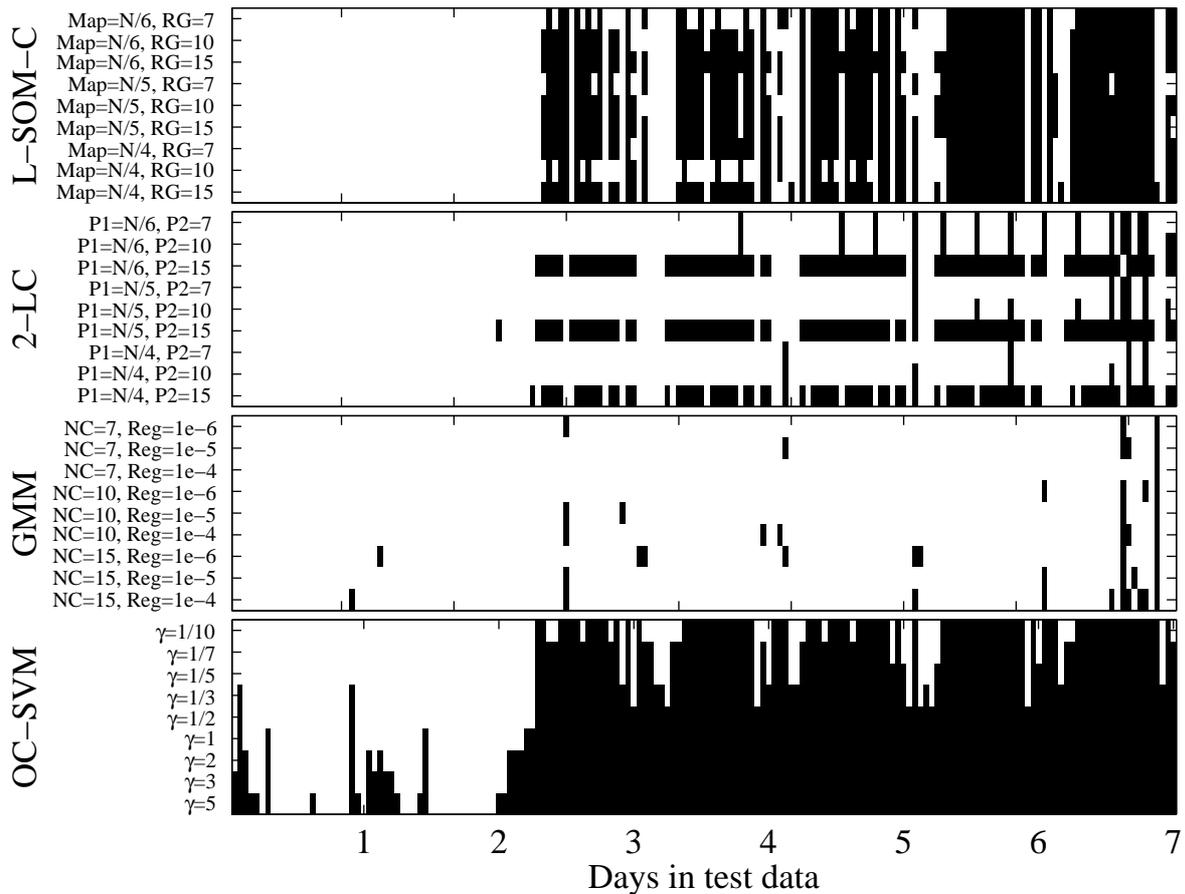


Figure 6.11 Detected anomalies in the test data marked as black blocks.

Altering the parameters affects the anomaly detection of all the methods. The behaviour of L-SOM-C is the most robust of these methods. The major effect is that fewer anomalies are detected with parameter combinations $\{N/6, 7\}$ and $\{N/4, 10\}$. The same effect is present to a lower degree with $\{N/5, 7\}$.

The most significant changes occur for 2-LC and GMM. Increasing the number of clusters in the second phase of 2-LC to 15 makes it more sensitive. With those combinations its detection is similar to that of L-SOM-C. GMM detects only a few anomalies with any parameter combination. Yet many anomalies are detected with only one parameter combination. One observation, on the seventh day at 21:00, is detected with all combinations. On the other hand, this observation is not detected by 2-LC at all, and with only one parameter combination by L-SOM-C.

The number of anomalies detected by OC-SVM increases consistently with an increase of γ . At higher values of γ it also detects anomalies during the first two days, where other methods do not detect any anomalies (except for two detected by GMM). With the value of $\gamma = 1/2$, the OC-SVM detects 116 anomalies, which is 69% of the test data and 138 anomalies; 82% with the highest value $\gamma = 5$.

6.5 Discussion

This chapter has presented an example of the detection of anomalies in server log data. An application specific robust logarithm method was used to scale the data. Logarithm levels the distributions of the counter type variables. Furthermore, robust estimates of the mean and the standard deviation, ignoring the upper tail and zeros, makes the scaling insensitive to the outliers and only takes into account the data with some activity. The robust logarithm scaling was compared to conventional normalisation, which is most often accepted without justification. Visual inspection of the scaled data shows that the robust logarithm scaling is less sensitive to the outliers and better reveals the inherent structure of the data, and thus the robust logarithm scaling is preferred.

A novel local anomaly detection procedure which is based on SOM and clustering, referred to as L-SOM-C, was applied, simulating the offline and online phases. The selection of the number of clusters was based on the minimum Davies–Bouldin index. One-dimensional SOM and the results of clustering, the reference groups, provide users with complementary and meaningful insights into the original data. The quantisation error of the SOM provides the required ranking of the anomalies as well as the contributions of the variable, revealing the sources of the anomalous observations.

The anomalies detected in the online phase were analysed to verify the performance of the L-SOM-C method. The method is able to detect meaningful and interesting novel behaviour of the system at an early stage. Four other methods were applied in the online phase for comparison: G-SOM (based on a global SOM), 2-L-C (based on two layers of clustering), as well as GMM and OC-SVM as more conventional reference methods. OC-SVM was the only one that was able to detect the novel behaviour

as early as L-SOM-C. The other methods detected only a fraction of the novelties in the test data.

The sensitivity to parameter changes was tested for four methods: L-SOM-C, 2-L-C, GMM and OC-SVM. Each method was trained using the reference data with nine sets of parameter values. The anomalies detected from the test data were analysed. GMM was able to detect only very few anomalies with any parameter set. The detection rate of OC-SVM changed a lot with its only parameter, with all the higher values resulting in the detection of all the observations after the first two days as anomalies. The performance of 2-L-C was highly dependent on the parameters. The highest number of clusters in the second layer provided similar results to those of L-SOM-C. All other parameter combinations provided only a few detections. Clearly, the most robust method was L-SOM-C, which performed rather consistently with all the parameter combinations in the test.

According to these comparisons, L-SOM-C, developed in this work, is the one that can be easily applied to other application areas. It is most likely to provide useful results without laborious identification and fine tuning of the parameters of the method.

Chapter 7: Daily traffic patterns

This chapter presents two use cases of applying anomaly detection in daily traffic data from BTS (see *Figure 2.1* for an example). In the first case anomaly detection is utilised in compression of cyclic time series data. The second case presents exploratory analysis of the distribution of daily traffic patterns within a weekly cycle. Both cases apply clustering and anomaly detection methods.

7.1 Objectives

Mobile networks are monitored using data collected from the network elements (NEs). Data collection processes count all the operations the NEs execute to establish data or voice connections. Such operations include voice connection, data context reservation attempts, handovers and connection shutdowns. The NEs also monitor and record the validity of connections by recording detected error rates, signal strengths and other physical quantities that describe connection quality. Each network element produces a time series of values for each observed indicator.

Traffic delivered through the cells is one of the most significant indicators of mobile network performance. Each cell resides in a unique physical and geographical environment and usage varies according to the rhythm of life of the surrounding society [Khedher et al. 2002]. Variations in traffic include daily, weekly, and yearly cycles. The shortest of these cycles is daily traffic, which depends on the location of the cell and the behaviour of subscribers that are connected to it. In addition to the total volume of traffic, finer details of how traffic is distributed during a day, daily patterns, are of great importance. Recognising the behaviour of subscribers helps the operator in managing and developing the services they offer. Valuable information for those purposes can be revealed by analysing the daily traffic patterns which result from the usage habits of subscribers.

The two use cases in this chapter present distinct objectives and applications but share the same data, consisting of daily traffic patterns collected from cells in a commercial mobile network.

The databases of mobile telecommunication operators typically contain hundreds of variables that contain strong cycles. The best known and most important one is the amount of traffic. The objective in the first use case is to reduce storage space needed for data of variables containing strong periodical cycles. The compression has to retain easy access to the data with minimal computational load. The method for compression of cyclic time series data has been published in Kumpulainen & Hätönen [2008b].

Daily traffic patterns have often been observed to depend on the day of the week. On the other hand, some cells provide similar patterns regardless of the day of the week. The objective of the second use case is to analyse and visualise the characteristic daily traffic patterns, as well as to detect anomalies that deviate from the normal behaviour. Part of this study has been published in Kumpulainen & Hätönen [2012].

7.2 Use cases

Both use cases are based on the same data set, which is scaled and preprocessed as described in section 7.3. The procedures for the two use cases are described below.

7.2.1 Data compression

Data compression in this use case is achieved by representing the daily traffic with a limited number of prototype patterns. The prototypes are identified from historical data that contains the shape of each daily traffic series. In this use case the sampling rate is one hour. The absolute amount of traffic is eliminated by dividing each daily series with its mean value. Instead of 24 hourly values per day for each BTS, only the daily mean traffic and reference to the corresponding prototype need to be stored. In addition, the prototype patterns are stored, but the number of the prototypes is significantly lower than the number of the daily time series patterns to be stored.

This is a lossy compression method. Error is reduced by storing separately those single hourly observations that differ significantly from all the prototypes, the anomalies. The mean traffic is used to form dynamic, traffic dependent thresholds for anomaly detection. This allows more deviation (in the scaled space) from the prototype at low traffic, and tighter thresholds where the traffic is high.

The procedure can be divided into three phases. The compression model is identified from historical data. This includes identifying the prototypes and anomaly thresholds. The second phase is compression, either the offline compression of an existing database, or online compression integrated in the collection of new data. Uncompressing the stored data is the third step. The steps in each phase are listed below.

Identification of the compression model.

1. Preprocess (scale and remove the most obvious anomalies).
2. Identify traffic pattern prototypes by clustering.
3. Calculate the hourly standard deviations within each cluster.

The compression model consists of the prototypes and the corresponding standard deviations. The compression model itself requires storage space for $P*2*24$ values where P is the selected number of prototypes.

Compression and storage.

1. Scale by the daily mean.
2. Find the id of the best matching prototype.
3. Calculate the traffic dependent anomaly threshold.
4. Store the daily mean, the prototype id and possible anomalies.

Uncompression.

1. Retrieve the daily mean, the prototype id and anomalies from the database.
2. Retrieve the prototype pattern from the model and scale by the daily mean.
3. Insert possible anomalies separately.

The details of each step and the results are presented in 7.4.

7.2.2 Exploratory analysis of daily behaviour

SOM and clustering are utilised in analysing daily patterns. They are efficient tools in summarising and visualising the characteristics of data. Even though there are no well-separated distinct clusters present in these data, users appreciate the overview of the data provided by SOM and clustering. One-dimensional SOM is used in this analysis. It is very efficient in visualising data where the variables are not independent but form a pattern, a daily traffic pattern in this case [Kumpulainen & Hätönen 2012].

1. Preprocess (scale and remove the most obvious anomalies).
2. Identify one-dimensional SOM for visualisation.
3. Cluster the SOM code vectors.
4. Visualise the weekly variation on the SOM.
5. Detect anomalies and their distribution during the days of the week.

The details of each step and the results are presented in 7.5.

7.3 Daily pattern data and preprocessing

The data set in these use cases consists of the volume of voice calls in the cells of a commercial mobile network. The traffic volume time series of each cell are ordered into daily patterns, sampled once per hour. The row vectors of patterns are concatenated vertically to form a data matrix X that has 24 columns. Each row x_i presents a daily pattern of one cell on one day $x_i = [x_{C,D}(1) \ x_{C,D}(2) \ \dots \ x_{C,D}(24)]$, where subscripts denote a cell C and date D . The number of rows in a full data matrix with no missing values would equal the number of days in the measurement period multiplied by the number of network elements. In real life some rows will be missing.

The data set in this study consists of 100 cells and a measurement period of six weeks. Some values are missing from the database, resulting in a total of 3919 daily patterns in the data set. Due to large number of missing values at midnight, only the hourly traffic from 1 am to 11 pm is included in the analysis.

The one week of traffic of two cells is exemplified in *Figure 7.1*. The grid lines are at midnight. Traffic is heaviest during the daytime, and there is a section of nearly no traffic every night in both cells. The traffic of one week in these cells presents a wide variety of shapes in daily patterns. The patterns in the two cells have similar shapes on some days, for example on Friday. The total traffic through cell B is significantly higher than through cell A, which makes direct comparison of the shapes cumbersome.

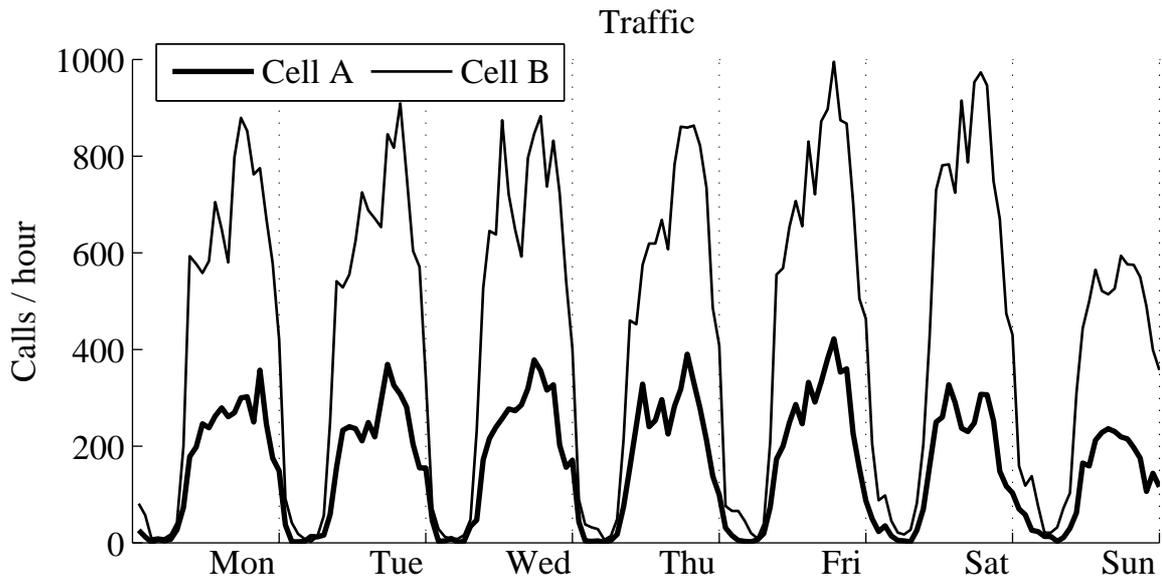


Figure 7.1 Traffic patterns of two cells during one week. The grid lines are located at midnight.

7.3.1 Scaling

Meaningful comparison of the patterns requires that the effect of the average volume of the traffic is removed. Dividing each row of the data matrix by the mean value scales the patterns so that the shapes are comparable. Thus the scaled daily traffic pattern $p_{C,D}$ is

$$(7.1) \quad p_{C,D} = \frac{x_{C,D}}{\mu_{C,D}},$$

where $\mu_{C,D}$ is the mean of the traffic through cell $C = 1 \dots 100$ on day $D = 1 \dots 42$.

The shapes of the scaled patterns can be compared visually and by similarity metrics. Two days from *Figure 7.1* are presented as examples of scaled daily patterns in *Figure 7.2*. Visual inspection suggests that the patterns of these two cells differ more on Wednesday but are more similar on Friday. This is confirmed by similarity measures. For example, the city block distance (Eq. 5.3) between the patterns is 3.41 on Wednesday and 2.75 on Friday.

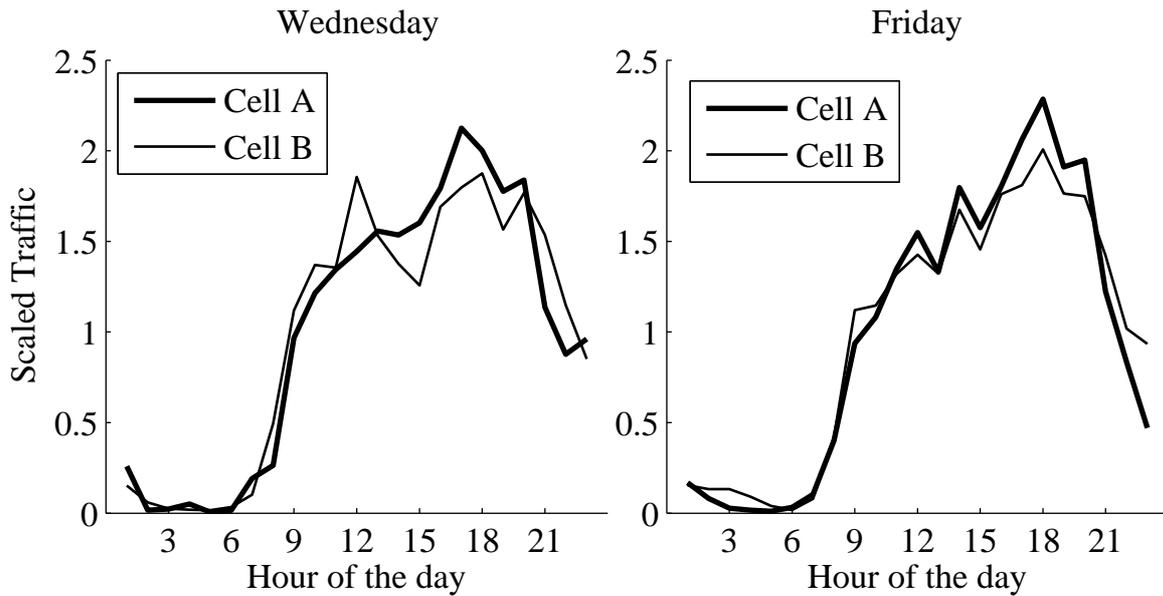


Figure 7.2 Scaled daily traffic patterns of two cells.

7.3.2 Cleaning by removing the most obvious anomalies

The objectives in these use cases include representing and analysing the most common patterns. Therefore the most obvious anomalies which may affect the identification of normal behaviour are first removed. The simplest method is to detect the daily patterns that have the largest Euclidean distance from the mean value in the multivariate space, as described in Section 3.4.2. A clustering based method provides additional information about the anomalies. This method is based on the assumption that small clusters, those with only few observations, consist of anomalies [Gupta et al. 2013]. The results of these methods are compared in the following.

The clustering based method relies on the assumption that normal observations outnumber anomalies. When the data set is clustered, all clusters that contain only a few

observations can be considered as anomalies. This requires that the number of clusters is sufficiently high. An unambiguous rationale for the number of clusters does not exist. Hierarchical clustering and Ward linkage are used in this case to form 80 clusters. If the observations of this data set were evenly distributed into the clusters, there would be 48 observations in each cluster. Another subjective selection is the number of observations required in a normal cluster. In this case, the clusters that have less than ten observations are considered anomalies, and the observations assigned to those clusters are removed.

The clustering results in 15 anomaly clusters, containing a total of 54 observations. For comparison purposes, the same number of anomalies are selected with the distance based method, and 54 observations that have the highest distance from the mean pattern are identified.

Traffic patterns in two anomaly clusters are presented in *Figure 7.3*. All these patterns are included in the distance based anomalies. The main advantage of the clustering method is that it groups the similar rare observations, which helps in verifying and analysing the anomalies.

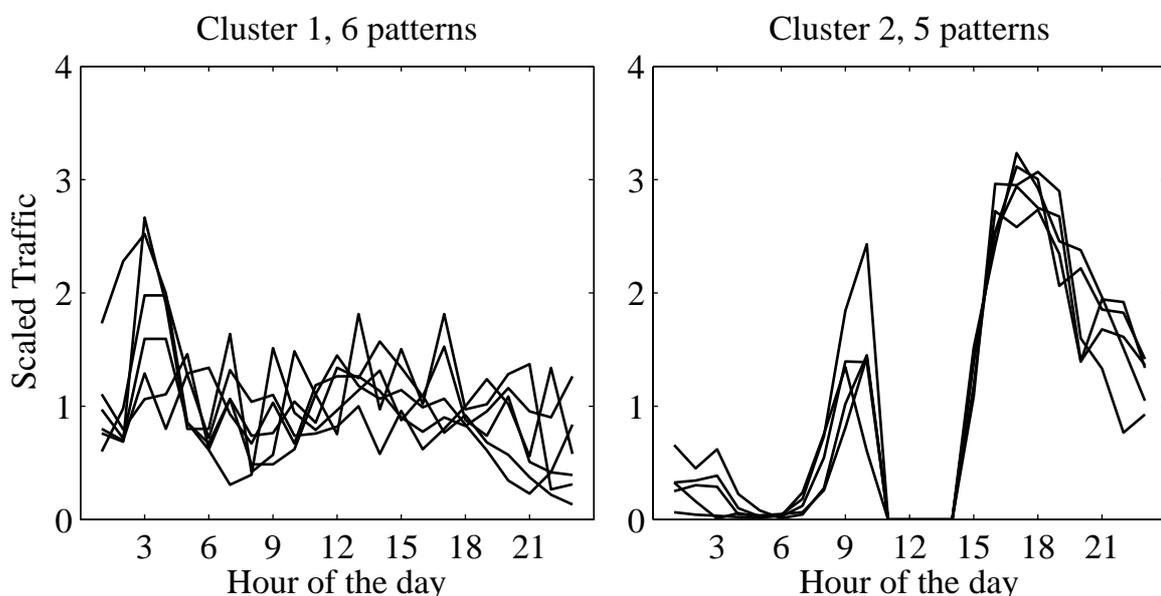


Figure 7.3 Examples of anomalous traffic patterns that were detected with both the cluster and distance based methods.

Cluster 1 contains six daily patterns, which represent relatively constant traffic throughout the day. Four of them have an additional peak at 03:00. All these daily patterns represent the Sundays of one single cell. Such patterns seem to be normal in that specific cell, but not present elsewhere in the network.

Cluster 2 contains five patterns with a zero traffic from 11:00 to 14:00. All the patterns are from distinct cells and four of them are from one day. This suggests that there has been a malfunction in a part of the network. Either those cells have been shut down simultaneously or the collection of data has failed. Experts with knowledge of the network topology will be able to track the problems.

Two more clusters of anomalies are presented in *Figure 7.4*. The distance based method detected none of the six patterns in cluster 3 and only one of the seven patterns in cluster 4, the one marked with circles.

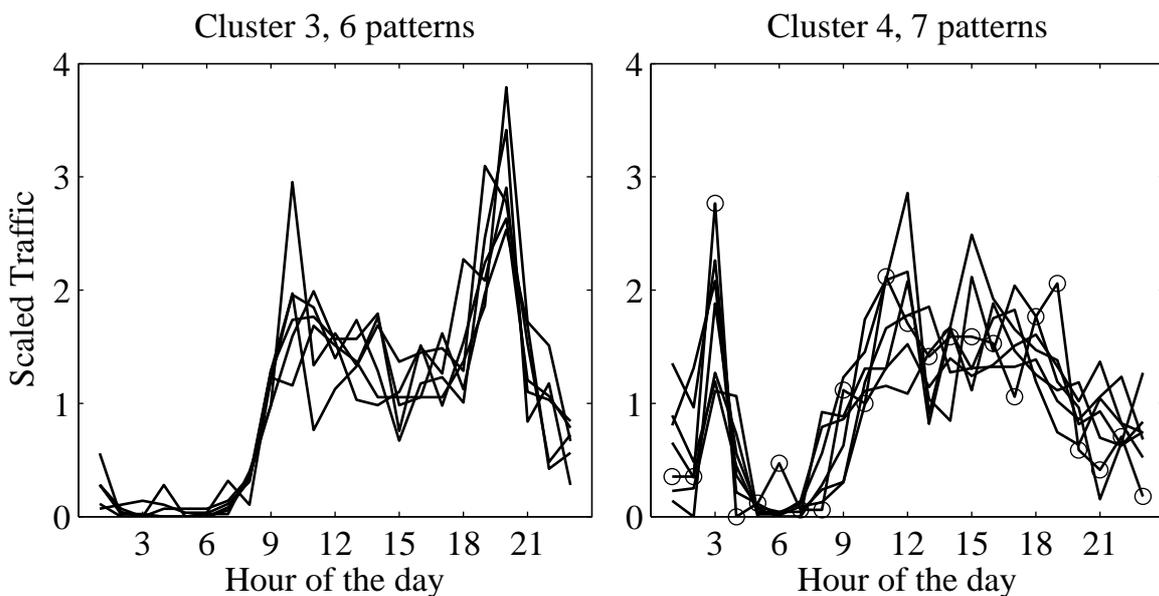


Figure 7.4 Examples of anomalous traffic patterns that were detected with the cluster based method, only one of which was detected with the distance based method.

The six patterns in cluster 3 are from three cells, two patterns from each. A peak of high traffic at 20:00 and a lower peak at 10:00 are common features in this cluster. The most distinguishable feature in cluster 4 is the peak at 03:00.

The patterns detected as anomalies by the clustering method are removed from the data set. The rest of this chapter uses the cleaned data set with the remaining 3865 patterns.

7.4 Application of data compression

This section presents the use case of compressing cyclic data. Compression is based on replacing the daily traffic pattern by a reference to a prototype pattern, detecting individual anomalous time instances within the patterns, and storing them separately.

7.4.1 Identification of the compression model

The compression model consists of the prototypes and the corresponding standard deviations that have to be identified.

The first step after preprocessing is to cluster the data set. The cluster centres are the prototype patterns which represent the main characteristics of the data. The clustering method, as well as the number of clusters, can be freely selected. Examples of prototype patterns are presented in *Figure 7.5*. These prototypes are the results of clustering the data with hierarchical clustering and Ward linkage into ten clusters.

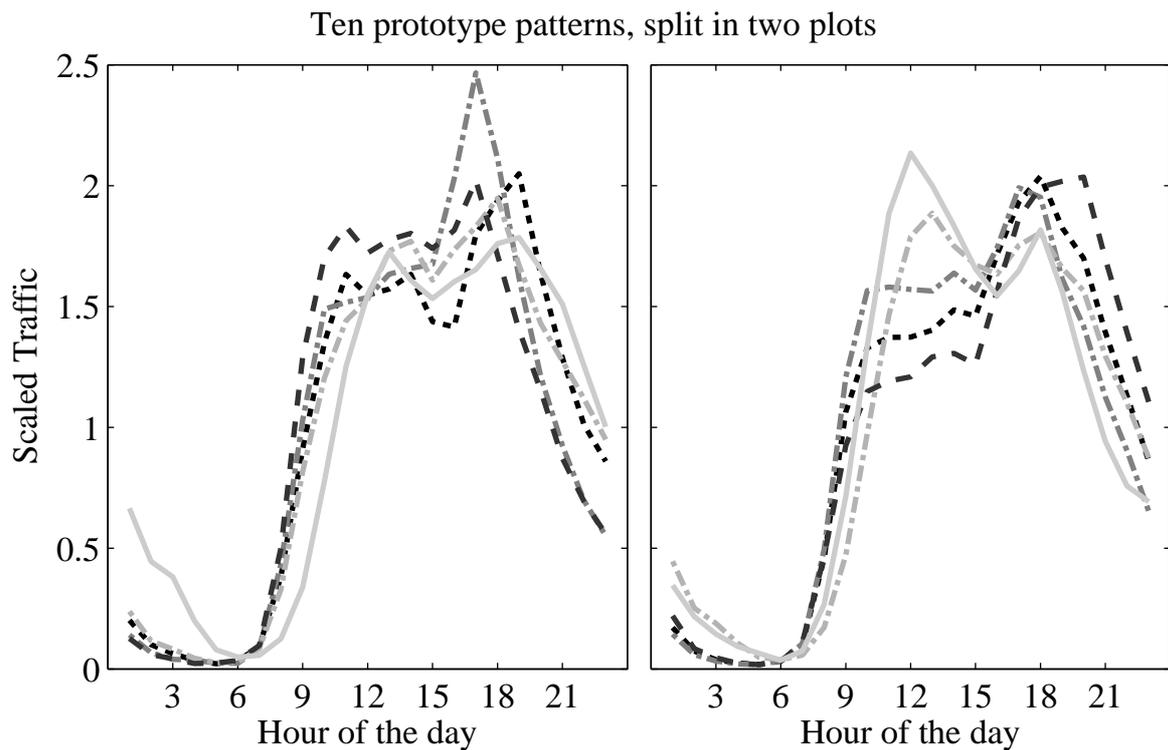


Figure 7.5 Ten prototypes represent distinct shapes of daily patterns.

The second step in the compression model identification is to calculate the standard deviations that are used in the detection of the anomalies to be stored separately. The standard deviations are calculated separately at each hour of the day within each cluster.

In this use case midnight was omitted from the data set. Thus, when the selected number of prototypes is P , the compression model consists of $P*23$ values for both the prototypes and the standard deviations, requiring a total of $2*P*23$ values to be saved as the model.

7.4.2 Compression and storage

The first step in compressing a time series of daily traffic is to calculate its mean. The scaled traffic pattern is calculated by dividing the traffic with the mean as presented in Eq. 7.1.

The second step is to find the best matching prototype pattern. The Euclidean distances between the scaled pattern and the prototypes are calculated. The prototype that provides minimum distance is selected to represent that traffic pattern. The id of the best matching prototype is stored, requiring only one integer value in the data base.

If a single value of a traffic pattern at any individual time instance deviates from the prototype by more than a specified threshold, that value is considered as an anomaly, and the original unscaled value is stored separately. The cluster specific standard deviations are used to calculate the threshold for anomaly detection. The threshold at a time instant $h = 1 \dots 23$ is a product of a coverage factor k and the standard deviation $\sigma_p(h)$ within the cluster of the associated prototype. Thus the values of a pattern $p(h)$ are considered normal if they are within the following range:

$$(7.2) \quad \mu_p(h) - k\sigma_p(h) \leq p(h) \leq \mu_p(h) + k\sigma_p(h) .$$

The mean $\mu_p(h)$ is the value of the corresponding prototype. For normal distribution, coverage factor $k = 1.96$ results in the detection of five per cent of the data as anomalies. This yields one set of thresholds for each prototype cluster. In this use case these are called constant thresholds. The coverage factor k is a selectable parameter, and its effect on the compression ratio and introduced compression error are presented later in this Chapter.

The total volume of traffic is taken into account by dynamic anomaly thresholds. The objective is to allow more deviations from the prototype at low traffic and to require tighter thresholds where the traffic is high. Assuming that the individual calls are independent, the number of calls within a time interval, one hour in this case, follows a Poisson distribution. The standard deviation of a Poisson distributed variable is equal to the square root of the mean. This is utilised to scale the thresholds in the scaled space. Additional scaling by cluster mean provides individual thresholds for each prototype. The dynamic threshold $d_{C,D}(h)$ is calculated at each time instant h of the day for each pattern $p_{C,D}$ recorded from cell C on day D as follows:

$$(7.3) \quad d_{C,D}(h) = \frac{k\sigma_c(h)}{\sqrt{\frac{\mu_{C,D}}{\mu_p}}}.$$

The selected coverage factor k is equal to the one in the constant threshold in Eq. 7.2 and $\sigma_c(h)$ is the hourly standard deviation of the patterns associated into the same cluster as the pattern $p_{C,D}$. The denominator is the square root of the ratio of the mean traffic of the pattern $p_{C,D}$ at hand, $\mu_{C,D}$ used in Eq. 7.1, and μ_p , the mean traffic of all the data associated into the same cluster as the pattern $p_{C,D}$.

When the daily mean equals the mean of the cluster, $\mu_{C,D} = \mu_p$, the dynamic threshold reduces to the constant threshold. During lower traffic, when $\mu_{C,D} < \mu_p$, the denominator in Eq. 7.3 is smaller than one, the result is a wider range in dynamic thresholds. On the other hand, during higher traffic, when $\mu_{C,D} > \mu_p$, the dynamic thresholds will allow less variation from the prototype.

The differences between the constant and dynamic thresholds are exemplified in *Figure 7.6*. The coverage factor is $k = 1$ in both thresholds. The left side presents an example of a pattern that has low total traffic on this day. The constant thresholds are exceeded in 15 time instances of the 23, thus requiring those values to be stored separately in the database. Only one value at 23:00 exceeds the dynamic thresholds.

The pattern on the right side has significantly higher mean traffic than the mean in the cluster it is associated with. Therefore the dynamic thresholds are closer to the prototype than the constant thresholds. No anomalies are detected if the constant thresholds are used. The dynamic thresholds are exceeded in five time instances.

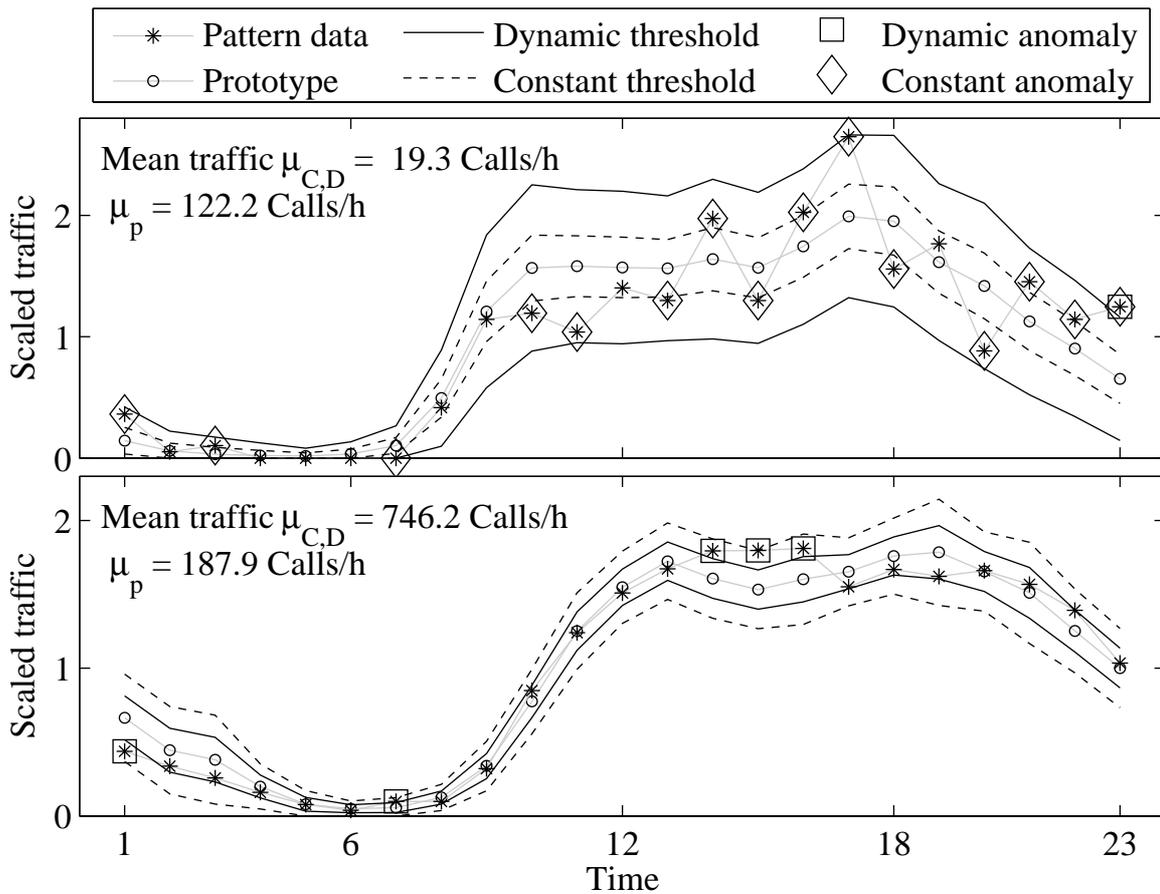


Figure 7.6 Two scaled traffic patterns and corresponding thresholds. Dynamic thresholds allow more variability for low traffic patterns.

Another view of the constant and dynamic thresholds with $k = 1$ is depicted in Figure 7.7. It presents a scatter plot with a mean traffic $\mu_{C,D}$ of a daily pattern on the horizontal axis and scaled value at noon on the vertical axis. The grey solid line presents the value of the prototype at noon. The grey dashed lines are the constant thresholds. They have constant value within a cluster, regardless of the total traffic; thus they form straight lines. The dynamic thresholds, black dotted curves, allow more variation during low traffic but less when the mean daily traffic is high.

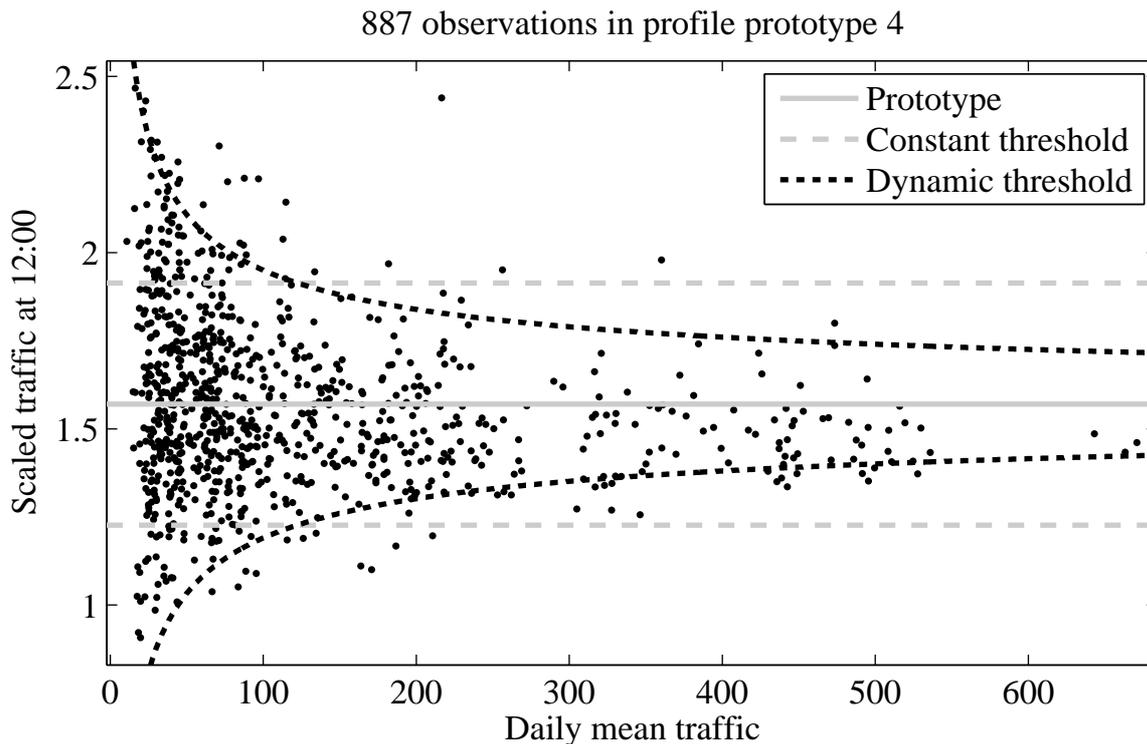


Figure 7.7 Scaled traffic at noon as a function of daily mean of traffic.

7.4.3 Uncompression

Uncompressing the stored data is straightforward. For cell C at date D the data base contains the mean traffic $\mu_{C,D}$, the id of the prototype pattern, and the anomalies if there were any. The associated prototype pattern, specified by the stored id, is scaled back to the original scale by multiplying by the stored mean traffic. Finally, if any separately stored anomalies were included, they are inserted, replacing the values of the scaled prototype. The anomalies are stored as original unscaled values and therefore require no scaling.

7.4.4 Parameter sensitivity

This data compression method has two parameters that have to be selected: the number of clusters and thereafter the prototype patterns and the coverage factor k in the anomaly detection step. The values of these parameters affect the compression ratio and the error introduced in the uncompression.

First, the effect of the number of prototypes, from one up to 50, is studied with the coverage factor $k = 1$. The results are presented in *Figure 7.8*. The space required to store the prototype pattern with corresponding standard deviations has been taken into account when calculating the compression ratios.

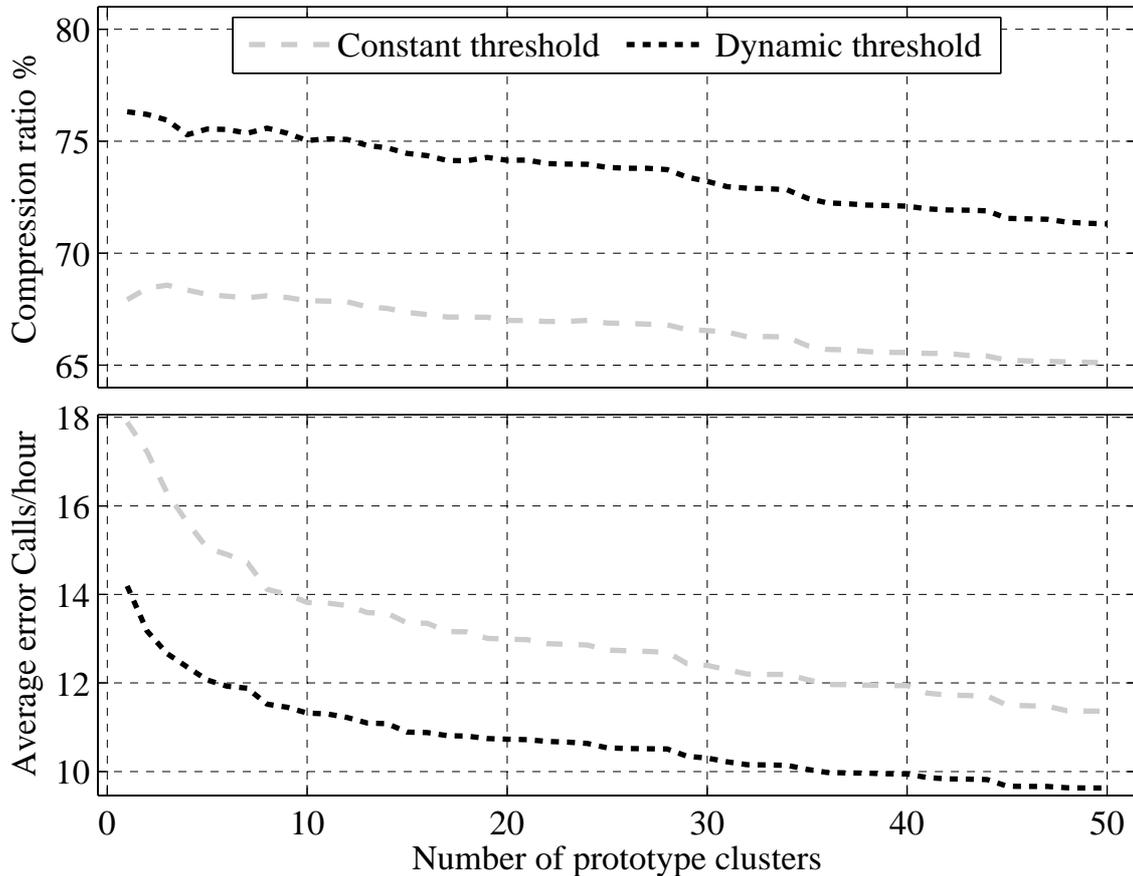


Figure 7.8 Increasing the number of prototype patterns reduces the compression ratio. Dynamic threshold provides better compression than constant threshold.

The best compression ratio is achieved with a low number of prototypes. As the number of prototypes is increased, the compression ratio decreases in a constant slope. The dynamic thresholds provide a higher compression ratio with a margin of about six per cent throughout the whole range of the number of prototypes.

The average error is represented in the actual units calls/h. The error decreases rapidly when the number of prototypes is increased from one. The dynamic threshold provides lower error. The difference reduces at the higher number of prototypes.

The effects of the coverage factor k on the compression ratio and error are presented in *Figure 7.9*. The number of prototypes is 10 and the k ranges from 0.5 up to 3.0.

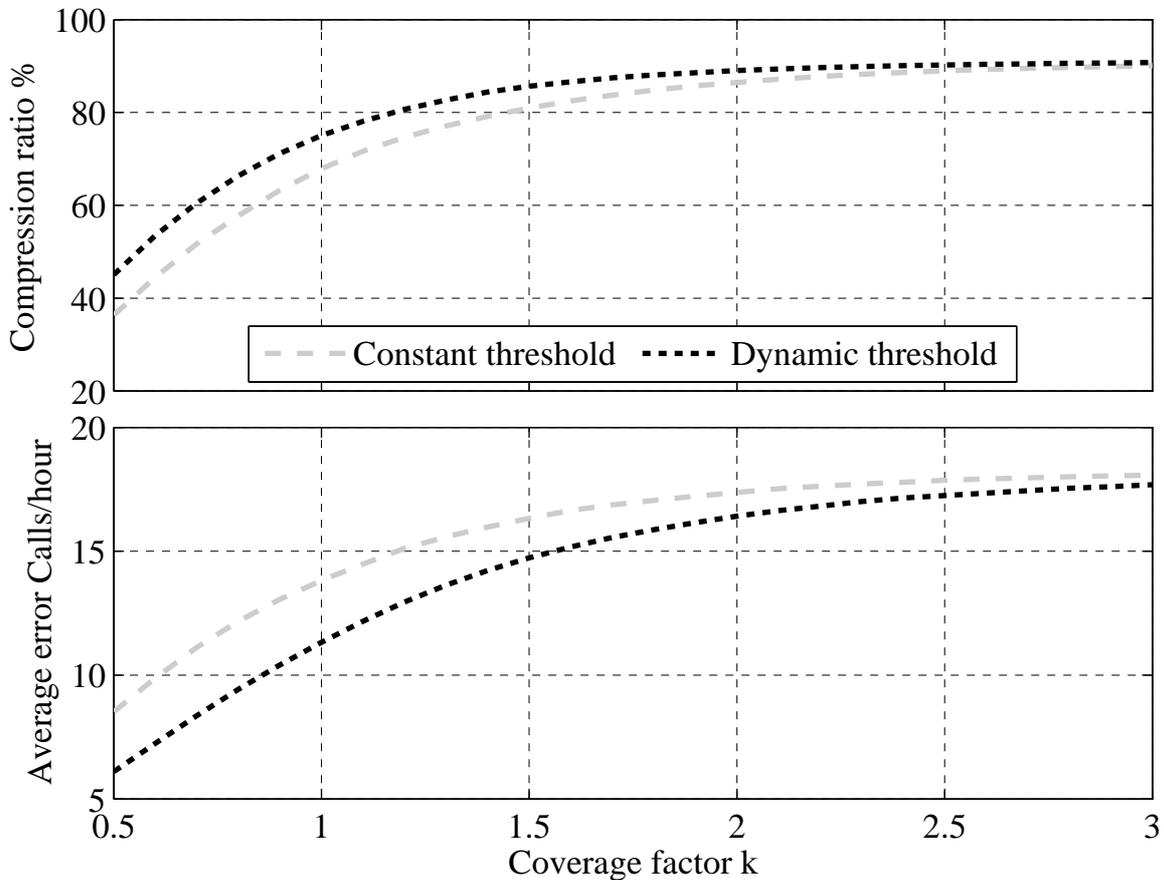


Figure 7.9 Increasing the coverage factor increases the compression ratio. Dynamic threshold provides better compression than constant threshold, in particular at lower values of the coverage factor.

Both the compression ratio and error rise with increasing k . The increase is non-linear in both. The compression ratio starts at around 40% at $k = 0.5$ and levels out to 90% at $k = 2$. The dynamic thresholds provide higher compression, but the difference is hardly noticeable above $k = 2$. The dynamic threshold produces lower error at all values of k , but the difference reduces at higher values.

The effect of both parameters is combined in *Figure 7.10*. The compression ratio is presented as the required storage space as a percentage of the uncompressed data on the vertical axis. The error is on the horizontal axis, thus, the optimal selection of the parameters would be located in the lower left corner of the figure. The number of pro-

prototypes covers a range from one to 100 at 11 values of the coverage factor k . Each k is presented as a line, colour coded by the number of the prototypes.

Judging from the figure, the Pareto optimal state is achieved at between 20 and 30 prototypes. Fewer than ten or more than 50 prototypes is never Pareto optimal.

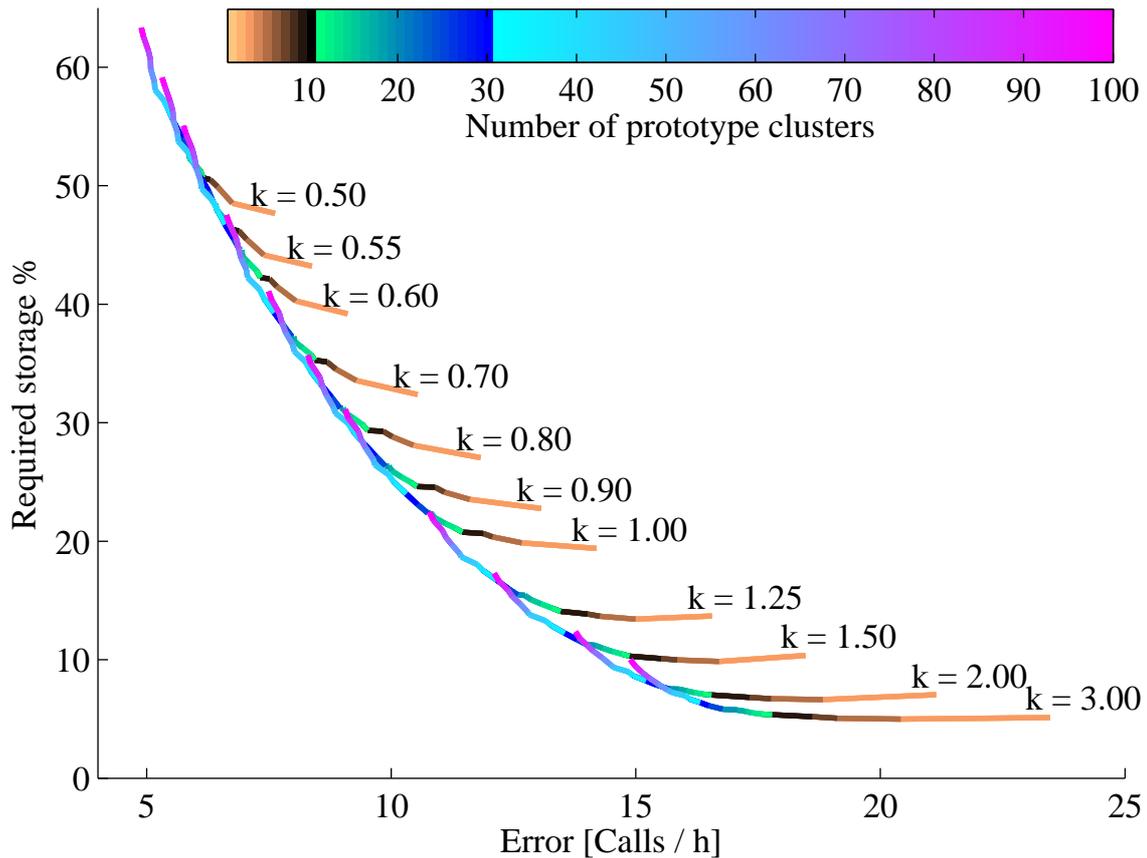


Figure 7.10 Optimal number of prototypes seems to be between 20 and 30.

7.5 Exploratory analysis of daily behaviour

This section presents exploratory analysis of the daily patterns. First, the common behaviour is visualised by one-dimensional SOM and clustering. Three anomaly detection methods are then applied and the characteristics of the detected anomalies are visualised in a one-dimensional SOM.

7.5.1 Visualisation of the main characteristics

A one-dimensional SOM allows more compact visualisation than a two-dimensional map [Kumpulainen & Hätönen 2012]. All two-dimensional planes for visualisation can be replaced by line plots. The 23 dimensional code vectors that represent the daily patterns can be combined into one map that presents the typical patterns. This is an advantage, in particular in this application where the variables are not independent but form a pattern. The topology preservation feature of the SOM ensures that the code vectors of the adjacent units are similar, resulting in a smooth map of prototype patterns.

The component planes of a one-dimensional SOM trained with the daily pattern data are depicted at the top of *Figure 7.11*. The darker grey scale represents higher traffic. The SOM has 55 nodes, which minimises the topographic error given in Eq. 4.3.

Both a U-matrix and a hit histogram of a one-dimensional SOM are presented as single lines at the bottom of *Figure 7.11*. U-matrix values are Euclidean distances between the code vectors of the adjacent map units. The hit histogram line shows the number of data points that hit each map unit. Map units surrounded by high U-matrix values denote that their code vectors are further apart in the data space, representing a sparse part of the space. Such units typically have a relatively small number of hits. Therefore the U-matrix resembles a mirror image of the hit histogram.

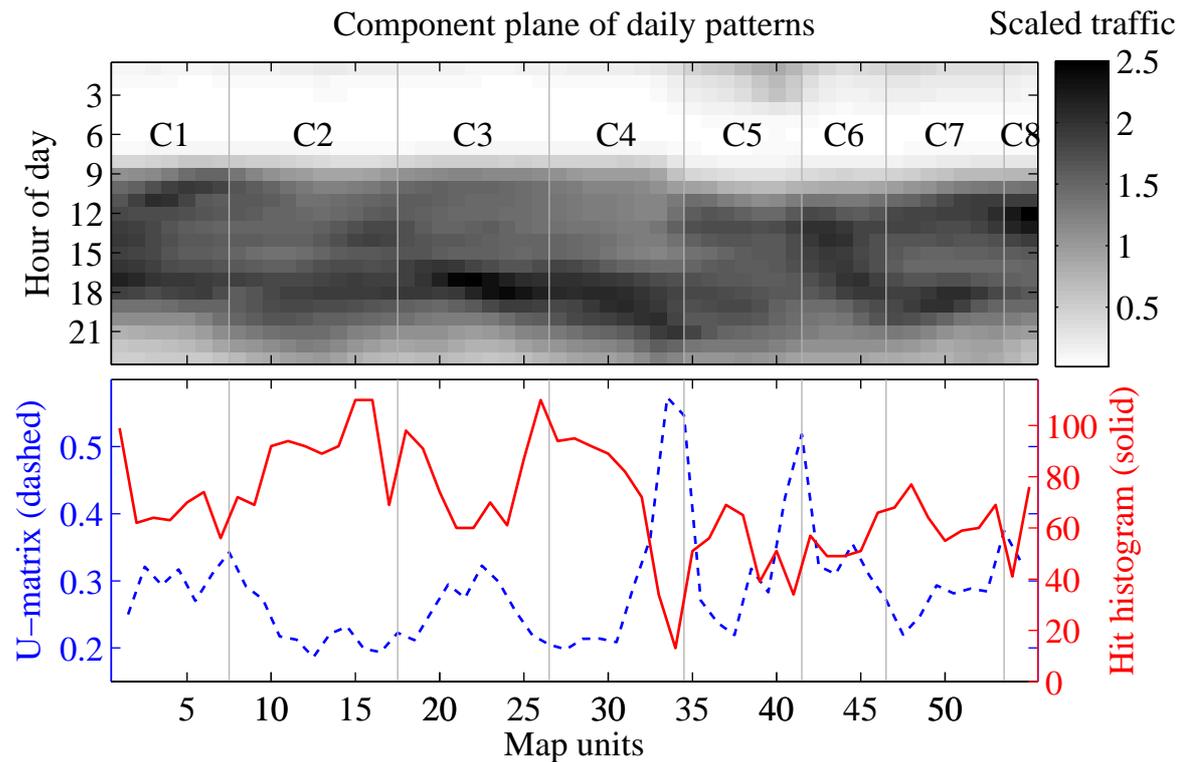


Figure 7.11 Clustered component plane of one-dimensional SOM (top) with U-matrix and hit histogram (bottom).

Clustering the code vectors of the SOM is a common procedure to further summarise the information identified in the trained SOM [Vesanto & Alhoniemi 2000, Laiho et al. 2005]. Here the code vectors of the one-dimensional SOM are clustered by hierarchical clustering with Ward linkage into 8 clusters, which is a subjective selection after the evaluation of several clusterings. A higher number of clusters yields an unnecessary splitting of clusters, whereas fewer clusters combine clusters into larger and less informative ones. The clusters are referred to as SOM clusters C1 to C8. As the U-matrix values show, cluster C5 is well separated from both C4 and C6. Transitions between other clusters are not as clearly separated. However, each of the SOM clusters have their own characteristics. Clusters C3 and C8 present patterns that have one high traffic peak during a day but at a separate time of the day. Clusters C1 and C7 present patterns that have two peaks, although C1 also contains behaviour where the two peaks merge into one, covering a longer period in the afternoon (map units from one to five). Clusters C2 and C5 present patterns with relatively constant traffic from 9:00 until midnight and C5 covers significant amount of traffic after midnight.

A one-dimensional SOM enables compact visualisation of subgroups of the data set, which supports exploration of how the traffic behaviour depends on the day of the week. Hit histograms calculated separately for each day of the week are combined in *Figure 7.12*.

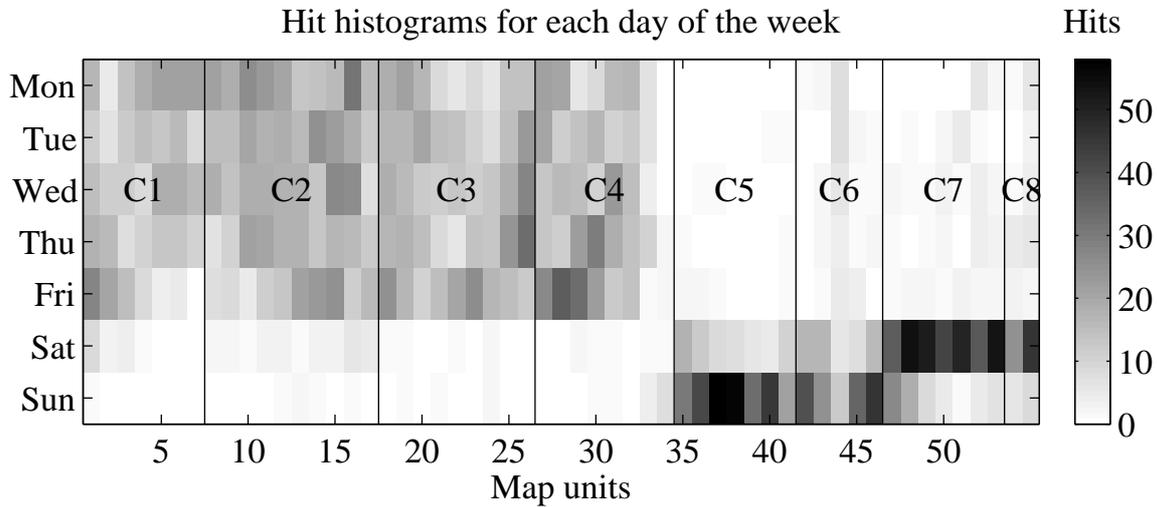


Figure 7.12 Separate hit histograms for each day of the week.

The patterns of weekends are clearly separated from weekdays and are concentrated in the map units from 35 to 55. There are only a few hits on weekdays in that area of the map. Saturday and Sunday partly overlap. Sundays are mostly assigned to clusters C5 and C6, whereas Saturdays are concentrated to the end of the map, in clusters C7 and C8.

7.5.2 Behaviour profiles

This section presents how the daily behaviour can be summarised at the level of individual cells. First, a feature vector is formed for each cell by calculating the number of days that the cell is assigned to each of the SOM clusters. The counts are divided by the total number of days of the cell in the data set, forming an eight-dimensional feature vector of the proportions the cell is assigned in each SOM cluster. The features are clustered and the cluster centres are referred to as behavioural profiles (BP).

In this example, the BPs are constructed by hierarchical clustering with Ward linkage. The number of BPs is a subjective selection, depending on what level of detail is pre-

ferred. The resulting BPs of seven clusters are exemplified in *Figure 7.13*.

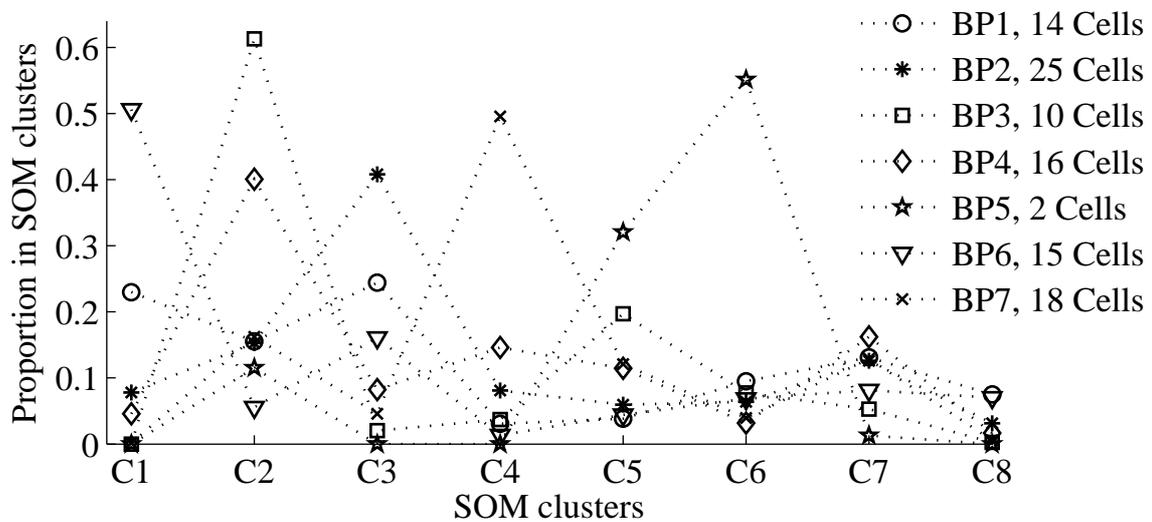


Figure 7.13 Behaviour profiles of the cells.

Most of the behaviour profiles are concentrated into one single SOM cluster on weekdays. The SOM clusters C5, C6, C7 and C8 cover the patterns produced during weekends in all cells and therefore have relatively constant proportions in the BPs.

The two cells in BP5 present anomalous behaviour. Outside of the small proportion in C2, they are concentrated into SOM clusters C5 and C6, which present patterns produced on Sundays. The data set of six weeks contains four Thursdays, five Fridays and six of all the other days of the week for both cells. Thus, regardless of the day of the week, the behaviour of these cells is similar to the behaviour of all the other cells on Sundays only.

BP1 which contains 14 cells is another exception. In addition to the normal weekend clusters, it spreads across three SOM clusters. This group of 14 cells presents more variation in its daily behaviour than all the others.

Detailed analysis and identification of the root causes producing these findings requires knowledge about the topology and geography of the network, which are only available to the operator.

7.5.3 Visualisation of anomalies

The two previous sections presented exploratory analysis and visualisation of normal, average daily behaviour. This section presents summarising visualisations of anomalies, the patterns that deviate from normal behaviour. In this example, anomalies are detected by three methods, referred to as *Local*, *Local k* and *Dynamic threshold*.

The *Local* method is the one presented in 6.2 and is applied to server log data. In this case, the SOM clusters $C1 \dots C2$ are utilised as the *reference groups* of the method. Five percent of the patterns with the highest quantisation errors in each SOM cluster are regarded as anomalies.

Local k is a modification of the *Local* method. The quantisation errors in all SOM clusters are made comparable by scaling them with the average quantisation errors of the clusters. An anomaly coefficient $a_{C,D}$ is calculated for each traffic pattern $p_{C,D}$ as

$$(7.4) \quad a_{C,D} = \frac{QE_{C,D}}{\mu_{e(C,D)}},$$

where $QE_{C,D}$ is the quantisation error of pattern $p_{C,D}$, from cell C on date D and $\mu_{e(C,D)}$ is the mean of the quantisation errors in the SOM cluster (*reference group*) into which the pattern $p_{C,D}$ is assigned. Instead of assuming an equal percentage of anomalies in all clusters, five per cent of the highest coefficients $a_{C,D}$ are assumed to be anomalies. This reduces the risk of false positives in compact clusters, and enhances the detection of anomalies in clusters where more than the selected percentage of data deviates significantly from the normal.

Dynamic threshold detection is based on the method presented in 7.4 and utilised in data compression. In that example, the dynamic thresholds were used to detect individual time instances in traffic patterns. In this use case the anomaly detection needs to be performed on the complete 23 dimensional patterns. In this example, a traffic pattern is assumed to be anomalous if three or more time instances (out of a total of 23) exceed the dynamic thresholds. The same SOM cluster centres as in local methods are used as the prototype patterns; thus the number of prototypes is eight. The cover-

age factor k is set to 1.95, which results in the detection of about five per cent of the data set as anomalies.

The numbers of the common anomalies detected by the three methods are collected in Table 7.1. The total numbers of anomalies are on the diagonal. The local methods detect mostly the same patterns; 151 of the anomalies detected by both. The Dynamic threshold method detects different patterns. Only 24 patterns are in common with the *Local* method, and 20 with the *Local k* method.

Table 7.1 The numbers of the common anomalies detected by the three methods.

Common anomalies	<i>Local</i>	<i>Local k</i>	<i>Dynamic threshold</i>
<i>Local</i>	194	151	24
<i>Local k</i>	151	193	20
<i>Dynamic threshold</i>	24	20	192

The top row of Table 7.2 presents the distribution of patterns in the SOM clusters. C2 and C3 contain the highest proportions of data; over 41% together. The weekend clusters C5, C6, C7 and C8 contain the smallest proportions of data. Nevertheless, all of them in total contain 31.2%, which is slightly more than $2/7$, the proportion of the weekend days.

Table 7.2 Distribution of the data and anomalies in the eight SOM clusters.

	C1	C2	C3	C4	C5	C6	C7	C8
Data	488 12.6%	889 23.0%	711 18.4%	571 14.8%	365 9.4%	272 7.0%	452 11.7%	117 3.0%
Anomalies in each cluster								
<i>Local</i>	24 5%	44 5%	36 5%	29 5%	18 5%	14 5%	23 5%	6 5%
<i>Local k</i>	6 1.2%	56 6.3%	23 3.2%	44 7.7%	33 9.0%	9 3.3%	20 4.4%	2 1.7%
Dynamic threshold	36 7.4%	51 5.7%	29 4.1%	19 3.3%	15 4.1%	20 7.4%	16 3.5%	6 5.1%

The numbers and percentages of detected anomalies are collected at the bottom of Table 7.2. The *Local* method detects five per cent of anomalies in all SOM clusters as defined in the method. The distribution of the anomalies detected by the *Local k* method is not uniform. It detects significantly more anomalies than the *Local* in clusters C4 and C5 and slightly more in C2. Correspondingly, it detects less anomalies in other clusters. The difference is most radical in C1, where *Local k* detects only 1.2% of the patterns as anomalies. The anomaly percentage in C8 is also significantly below five per cent. The *Dynamic threshold* method has the highest anomaly percentages in clusters C1 and C6.

The main characteristics of the anomalies detected by each method are visualised by a one-dimensional SOM. All the patterns detected as anomalies are combined and used to identify the SOM. The patterns that are detected by more than one method are included in the identification data more than once. This allows them to have more weight in identification. The combined component plane and separate hit histograms for each method are presented in *Figure 7.14*.

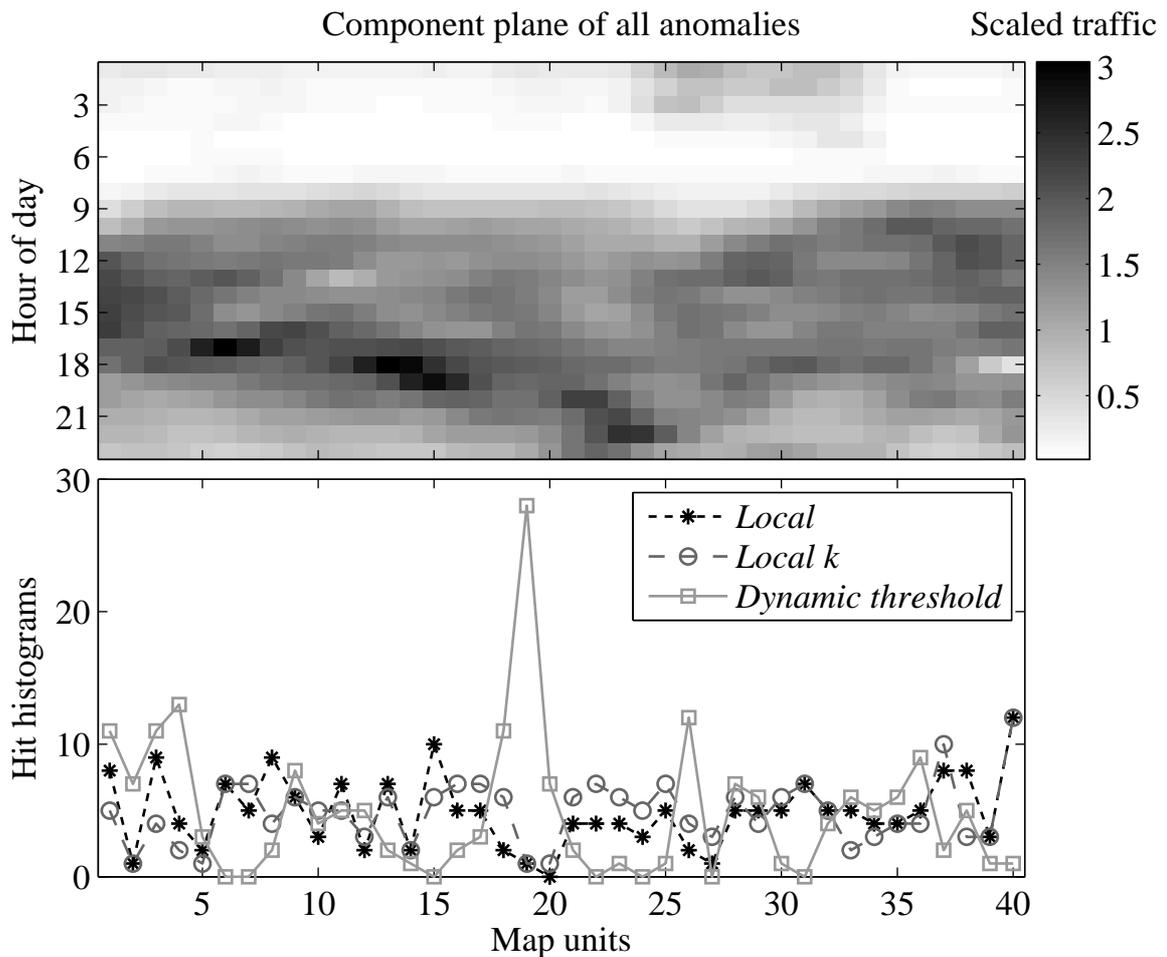


Figure 7.14 SOM component plane and hit histograms of the detected anomalies.

Compared to the component plane of the whole data set in *Figure 7.11*, the anomalies include more traffic between midnight and 05:00. Another difference is that the anomalies have higher traffic peaks, as seen in the colour bars of the scaled traffic.

The hit histograms of the *Local* and *Local k* method show that these methods detect patterns with individual high peak values at different hours of the day. Such patterns are typically not detected by the Dynamic threshold method; its hit histogram remains at low values. Map units 39 and 40 present patterns with a drop in traffic at 18:00, which is a sign of a malfunction in the network or in data collection.

Most of the anomalies detected by the dynamic threshold method seem to have smooth and flat patterns. Such patterns are mostly not detected by either of the local methods. They are most likely more refined variations from the nearest prototype.

The dynamic thresholds are constructed to emphasise higher total traffic. The histograms of the anomalous patterns are depicted in *Figure 7.15*.

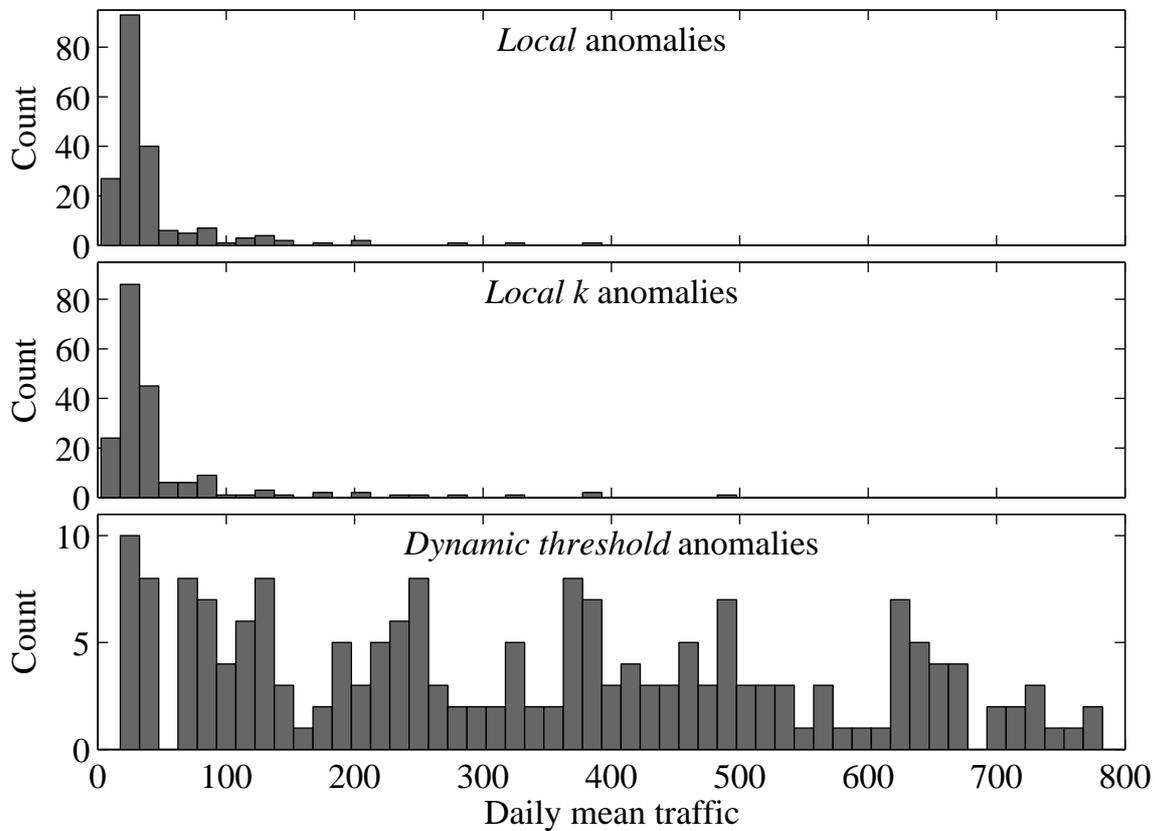


Figure 7.15 Histograms of the daily mean traffic of the detected anomalies.

Both local methods mostly detect patterns that have low traffic. Anomalies in cells that deliver higher traffic are most likely more important for the operator. However, more detailed analysis is not possible without knowledge of the topology and geographical circumstances around the cells. That information is only available to the operator's experts.

7.6 Discussion

This chapter has presented two use cases of anomaly detection applied to daily traffic data from cells in the mobile network. First, the data were scaled by dividing each pattern by its daily average traffic. This removed the volume of the traffic and revealed the shapes of the traffic patterns, which were of interest in this use case.

The most obvious outliers were removed, and analysed utilising clustering. The small clusters were considered to contain rare observations, which were removed and analysed separately. The outlier clusters provided a summarised and informative insight into the outliers, which is the main purpose of clustering, as presented in section 4.4.5. It was straightforward to identify the groups of patterns that shared similar problems.

The first use case was an application for data compression. It was based on representing the majority of the data with prototypes, and on storing individual anomalous instances separately. The prototypes were identified by clustering the data. The achieved compression ratio is determined by two parameters: the number of prototypes, i.e. the clusters, and the hourly coverage factor. However, increasing the compression ratio will also increase the error introduced by the compression, requiring a trade-off between them. The Pareto optimal results in this use case were obtained when the number of prototypes was between ten and 50.

The second use case presented exploratory analysis of traffic patterns through the cells. The behaviour on each day of the week was analysed and visualised by a one-dimensional SOM and clustering. One-dimensional SOM provides compact and informative visualisations of pattern type data, enabling effective visualisation of sub-groups in the data; days of the week in this use case. The results revealed, as expected, that the traffic during weekends differs significantly from that of weekdays. However, the rest of the traffic patterns were surprisingly evenly distributed among weekdays.

The behaviour of individual cells was explored by constructing behaviour profiles. It was found that the behaviour of most of the cells is rather consistent on weekdays regardless of the day of the week. The behaviour profiles also revealed groups of cells with atypical behaviour.

Three anomaly detection procedures were applied and the information in the anomalies was summarised and visualised by the SOM. The resulting anomalies were divergent, one of the methods detecting more anomalies in the patterns of higher daily traffic. A detailed and thorough investigation of the anomalies would require confidential information concerning the network structure and configuration.

The repertoire of methods and procedures for exploratory data analysis is endless. Validation is a significant problem in exploratory analysis, the goal is to find “interesting things”, and there is no gold standard to compare results [Zimek & Vreeken 2013]. The procedures presented in this use case provided novel results, that were interesting, meaningful and useful for end users, thus satisfying the fundamental objectives.

Chapter 8: Conclusions

Anomaly detection (AD) is one of the main tasks in data analysis. Industrial applications for AD are essential in analysing and monitoring processes. However, universally optimal AD methods do not exist, and no single method is applicable to all processes. Even obtaining the most useful results for various parts of a large and complex process, such as a mobile telecommunication network, may require different methods that are fine-tuned to be the most applicable for the specific task. Pre-processing procedures, and scaling in particular, are essential steps in data analysis. Expedient scaling of the data can emphasise the characteristics, that are the most significant with respect to the problem to be solved. Incorporating expert knowledge in scaling enables the subsequent methods to work at their full potential in revealing the most useful results to the end user.

The overall objective of this thesis was to provide procedures which support mobile network operators in solving everyday problems, regarding anomaly and novelty detection in network management. The selected approach led to the following more specific objectives: 1) to provide the users with a ranking of anomalies, enabling the users to prioritise their attention to the most severe problems, and 2) to summarise information about the detected anomalies as well as the normal function of the network.

8.1 Scientific novelty and significance

This thesis presented procedures for anomaly detection to be applied in mobile network management. The key contributions consist of a methodological part and applications of these methods in mobile network monitoring, exemplified in three use cases. The methodological contribution of this thesis consisted of three topics, which were utilised in three use cases. The first topic analysed how scaling of variables affects clustering for anomaly detection and analysis. The second topic provided methods of incorporating expert knowledge into anomaly detection by application specific scaling of variables. The third topic consisted of utilising clustering extensively for

anomaly detection, as well as for summarisation of information about the data and further analysis of anomalies.

The application specific contributions demonstrated in the use cases are as follows:

1. Scaling methods
 - Incorporating expert knowledge in piecewise linear scaling for radio performance data, providing unambiguous interpretation of quality
 - Robust logarithmic scaling for server log data
 - Scaling of the daily traffic profiles to distinguish the shapes
2. Two novel methods for local anomaly detection
 - 2-LC utilising two layers of clustering
 - L-SOM-C utilising the SOM and clustering with local anomaly thresholds
3. Summarisation of the anomalies and normal operation
 - One-dimensional SOM in compact visualisation of multivariate data
 - Clustering the anomalies into meaningful groups
 - Clustering to produce behaviour profiles from daily traffic patterns

The usefulness of anomaly detection and exploratory data analysis methods can be greatly enhanced by the concept of this thesis: combining them with scaling and summarisation into an integral procedure. Appropriate scaling enables the applied analysis algorithms to reveal the maximum amount of useful information from the data. Incorporating expert knowledge in scaling enables standard analysis methods to provide results which are interesting and meaningful regarding the specific problem. Summarisation by SOM and clustering can convert excess information into a reasonable number of groups, thus reducing the required resources in interpreting the results, and allowing the human experts to focus on solving the problems revealed by the data.

8.2 Relevance of results

The methodology was demonstrated by three solved use cases, which are applications in mobile network monitoring. Each use case was targeted to distinct parts of the network, covering a wide range of monitoring applications. The data were collected from commercial mobile networks, thus representing the characteristics and problems of those parts of the network in real life situations. Technological goals were refined in

close collaboration with an industrial partner. The proposed procedures and application prototypes have been used in their internal research, and in supporting their application development. Part of the results of this thesis has been protected (US Pat. 7,461,037 B2).

The first use case dealt with performance measurements in the radio interface. Expert knowledge was utilised in scaling the multivariate data to even out the importance of each of the variables. The distance from a known ideal state in the scaled space was used as a measure of anomaly, which provided ordering of the anomalies by their significance. Summarising the detected anomalies by clustering provided meaningful groups of typical problems in the network. The results were compared to those achieved when traditional normalisation was applied to the data. The results when using the proposed expert scaling were significantly easier to interpret and more informative. The expert scaling provides an unambiguous and understandable mapping from the measurement values to the level of performance, which is an extremely important benefit in practical applications. This is an effective procedure for investigating the performance of a network, and provides the operator with problem groups that can be utilised in on-line monitoring, automatically suggesting corrective actions.

The second use case presented anomaly detection applied to data extracted from the log files produced by the management servers. The proposed procedure included appropriate robust scaling and two anomaly detection methods developed for this application: 2-LC, and L-SOM-C. Three other methods were applied to the same problem: G-SOM (utilising SOM with a global anomaly threshold), GMM and OC-SVM. L-SOM-C and OC-SVM were found to be the most sensitive in detecting the novel behaviour in the test data. The parameter sensitivity of four of the methods (excluding G-SOM) was investigated. The L-SOM-C method was found to be the most robust one, significantly less sensitive to changes in parameter values than the others. Thus, it is most likely applicable to other operator's networks or completely different application areas without excessive and laborious fine tuning. It is a valuable tool in both, analysing the normal behaviour of the network, and in detecting anomalous events. Combined with robust logarithm scaling it is particularly suitable to processes that have multiple normal modes and produce counter data, making it an option in a variety of network security applications.

The third use case dealt with the daily traffic data profiles collected from cells in a mobile network. First, the daily profiles were scaled by dividing by the daily average traffic to reveal the shapes of the profiles. The most obvious outliers were cleaned from the data by clustering the data into a high number of clusters and considering very small clusters (with only a few observations) as outliers. The cleaned data set was used in two separate sub-tasks. The first one presented a procedure for compression of daily profile data. The compression was based on presenting the data in prototypes, provided by clustering, and storing anomalous observations separately. The trade-off between the compression ratio and the introduced error can be adjusted by two parameters: the number of prototypes (i.e. the number of clusters), and the coverage factor, which serves as the threshold in the anomaly detection. The selection of the combination of the parameters was supported by the presented Pareto optimal front. The compression method is applicable to any process that produces cyclic data patterns.

The second task of the third use case presented exploratory analysis of daily behaviour. One objective was to find whether the daily traffic pattern types depended on the day of the week. Utilising SOM and clustering revealed that weekends produce distinct patterns, but the patterns during weekdays showed no structure. The behaviour of individual cells was investigated with help of behaviour profiles. Some cells were found to have consistent patterns regardless of the day of the week, whereas some others showed dependence on the day of the week. Groups of exceptional behaviour were also revealed. The second part of the exploratory analysis consisted of the detection and analysis of anomalous traffic patterns. Three detection methods were applied and the characteristics of the anomalies were visualised with SOM. The methods detected different types of anomalies, that were interesting to the operator's expert. These results justify the claims that no single method is universally optimal, but multiple truths exist, and applying diverse methods in exploratory analysis can provide a variety of new aspects to the task.

8.3 Discussion and future work

Real life data from a mobile network does not contain labels for anomalies to detect. Therefore, unsupervised methods are required, and the ground truth, does not exist.

The ultimate goal of unsupervised methods is to provide results that are understandable, interesting, and useful as presented in sections 4.4.5 and 4.7. Thus performance is based on subjective assessment, and an inarguable comparison of methods and the effects of scaling is impossible. The final judgement of the results is based on the experience of the end user, in this case the network operator's expert. The methodology proposed in this work aims at providing the results to the end user in a summarised way, which is easy to interpret. The results in the presented use cases showed that the proposed procedures are capable of revealing interesting and meaningful results for network monitoring purposes.

Utilisation of the proposed procedures is not restricted to mobile network management. Ever increasing amounts of data are recorded from processes in various areas. Automated procedures for presenting the most interesting and essential information are will be must in the future. Network security has become very important lately. Detecting new types of treats requires unsupervised anomaly detection methods. Huge amounts of data make the summarisation of the results increasingly important.

The use cases in Chapters 5 and 7 demonstrated the existence of multiple truths. Different types of detected anomalies can all be interesting and meaningful. Thus, ensemble methods in both clustering and anomaly detection are the most interesting directions for future work.

References

- [Aggarwal & Yu 2001] Aggarwal, C. & Yu, P. (2001). Outlier Detection for High Dimensional Data. In: Proc of SIGMOD'01, pp. 37-46.
- [Agyemang et al. 2006] Agyemang, M., Barker, K. & Alhajj, R. (2006). A comprehensive survey of numeric and symbolic outlier mining techniques. *Intelligent Data Analysis*, IOS Press, 10, 6, 521–538.
- [Anisetti et al. 2008] Anisetti, M., Ardagna, C., Bellandi, V., Bernardoni, E., Damiani, E. & Reale, S. (2008). Anomalies Detection in Mobile Network Management Data. *Advances in Databases: Concepts, Systems and Applications*. Vol. 4443. pp. 943-948.
- [Arthur & Vassilvitskii 2007] Arthur, D & Vassilvitskii, S (2007). k-means++: the advantages of careful seeding. *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. New Orleans, Louisiana. pp. 1027 - 1035.
- [Atkinson 1994] Atkinson, A.C. (1994) Fast Very Robust Methods for the Detection of Multiple Outliers. *Journal of the American Statistical Association*, Vol. 89, No. 428, pp. 1329-1339.
- [Atkinson & Mulira 1993] Atkinson, A.C. & Mulira, H.-M. (1993). The stalactite plot for the detection of multivariate outliers. *Statistics and Computing*, 3 (1), pp. 27-35.
- [Bakar et al. 2006] Bakar, Z., Mohamad, R., Ahmad, A. & Andderis, M. (2006). A comparative study for outlier detection techniques in data mining. *Proceedings of the IEEE Conference on Cybernetics and Intelligent Systems*. pp. 1–6.
- [Barnett 1976] Barnett, V. (1976). The Ordering of Multivariate Data. *Journal of the Royal Statistical Society. Series A (General)*, Vol. 139, No. 3, pp. 318-355.
- [Barnett & Lewis 1987] Barnett, V. & Lewis, T. (1987). *Outliers in statistical data*. Wiley, Chichester. 463 p.

- [Barreto et al. 2005] Barreto, G.A., Mota, J.C.M., Souza, L.G.M., Frota, R.A. & Aguayo, L. (2005). Condition monitoring of 3G cellular networks through competitive neural models. *IEEE Transactions on Neural Networks*, Vol. 16, Iss. 5, Sept. 2005. pp. 1064 - 1075.
- [Basseville & Nikiforov 1993] Basseville, M. & Nikiforov, I.V. (1993). *Detection of Abrupt Changes - Theory and Application*. Prentice-Hall, Inc., Eaglewood Cliffs, New Jersey. 528 p.
- [Bellaachia & Bari 2012] Bellaachia, A., Bari, A. (2012). A flocking based data mining algorithm for detecting outliers in cancer gene expression microarray data. In *proc. Information Retrieval & Knowledge Management (CAMP)*, pp. 305 - 311. DOI: 10.1109/InfRKM.2012.6204996
- [Berkhin 2002] Berkhin, P. (2002). *Survey of clustering data mining techniques*. Technical report, Accrue Software, San Jose, CA, 2002.
- [Bezdek & Pal 1998] Bezdek, J.C., Pal, N.R.(1998). Some new indexes of cluster validity, *IEEE Trans. Syst., Man, Cybern. B*, vol. 28, pp. 301-315
- [Bi et al. 2001] Bi, Z., Faloutsos, C. & Korn, F. (2001). The “DGX” Distribution for Mining Massive Skewed Data. *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, California, USA, 2001, pp. 17–26.
- [Billor et al. 2000] Billor N, Hadi AS. & Velleman PF. (2000). BACON: Blocked adaptive computationally-efficient outlier nominators. *Computational Statistics & Data Analysis*. Vol. 34, Iss. 3, pp. 279-298
- [Bishop 1994] Bishop, C.M. (1994). Novelty detection and neural network validation. *Vision, Image and Signal Processing, IEEE Proceedings* Vol. 141, Iss. 4, Aug. 1994, pp. 217 - 222.
- [Bishop 2006] Bishop, C.M. (2006). *Pattern Recognition and Machine Learning*, Springer. 738 p.
- [Bolton & Hand 2002] Bolton, R. & Hand, D. (2002). Statistical Fraud Detection: A Review. *Statistical Science*, Vol. 17, No. 3, pp. 235–255.

-
- [Bouarfa & Dankelman 2012] Bouarfa, L. & Dankelman, J. (2012). Workflow mining and outlier detection from clinical activity logs. *Journal of Biomedical Informatics* 45. pp. 1185-1190.
- [Breunig et al. 1999] Breunig, M. M., Kriegel, H.-P., Ng, R. & Sander, J. (1999). OPTICS-OF: Identifying Local Outliers. *Proc. Conf. on Principles of Data Mining and Knowledge Discovery, Prague. Lecture Notes in Computer Science, Springer, Vol. 1704.* pp. 262-270.
- [Breunig et al. 2000] Breunig M. M., Kriegel H.-P., Ng R., Sander J. (2000). LOF: Identifying Density-Based Local Outliers. *Proc. ACM SIGMOD Int. Conf. on Management of Data (SIGMOD 2000), Dallas, TX.* pp. 93-104.
- [Bro & Smilde 2003] Bro, R. & Smilde, A. (2003). Centering and scaling in component analysis. *Journal of Chemometrics. Vol. 17 Iss. 1.* pp 16 - 33.
- [Brown et al. 2009] Brown, C., Cowperthwaite, A., Hijazi, A. & Somayaji, A. (2009). Analysis of the 1999 DARPA/Lincoln Laboratory IDS evaluation data with NetADHICT. *Symposium on Computational Intelligence for Security and Defense Applications, 2009. CISDA 2009. IEEE.* pp. 1-7. DOI:10.1109/CISDA.2009.5356522
- [Burge & Shawe-Taylor 2001] Burge, P., & Shawe-Taylor, J. (2001). An unsupervised neural network approach to profiling the behavior of mobile phone users for use in fraud detection. *Journal of Parallel and Distributed Computing, 61(7),* pp. 915–925.
- [Caussinus et al. 2003] Caussinus, H., Fekri, M., Hakam, S. & Ruiz-Gazen, A. (2003). A monitoring display of multivariate outliers. *Computational Statistics & Data Analysis, Vol. 44, Iss.s 1-2,* pp. 237-252.
- [Chandola et al. 2009] Chandola, V. Banerjee, A. & Kumar, V. (2009). Anomaly detection: A survey. *ACM Computing Surveys (CSUR), Vol. 41, (3),* 58 p.
- [Chang & Lin 2001] Chang C-C, Lin C-J (2001) LIBSVM: a library for support vector machines. <http://www.csie.ntu.edu.tw/~cjlin/papers/libsvm.pdf>, Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>

- [Chatfield 1988] Chatfield, C. (1988). Problem solving: a statistician's guide. London, Chapman and Hall. 261 p.
- [Cheng et al. 2009] Chen, Y., Dang, X., Peng, H. & Bart, Jr., H.L. (2009). Outlier Detection with the Kernelized Spatial Depth Function. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 31, no. 2, pp. 288-305.
- [Chiang et al. 2003] Chiang, L.H., Pell, R.J. & Seasholtz, MB. (2003). Exploring process data with the use of robust outlier detection algorithms. *Journal of Process Control* 13. pp. 437-449.
- [Clifton et al. 2006] Clifton, L.A., Yin, H. & Zhang, Y. (2006). Support Vector Machine in Novelty Detection for Multi-channel Combustion Data. J. Wang et al. (Eds.): *ISNN 2006, LNCS 3973*, pp. 836–843.
- [Conlin et al. 2000] Conlin, A., Martin, E. & Morris, A. (1999). Confidence limits for contribution plots. *Journal of Chemometrics*, Vol. 14 Iss. 5-6, pp. 725 - 736.
- [Dang & Serfling 2010] Dang, X. & Serfling, R. (2010). Nonparametric depth-based multivariate outlier identifiers, and masking robustness properties. *Journal of Statistical Planning and Inference*. 140 (1). pp. 198-213.
- [David & Nagaraja 2003] David, H.A. & Nagaraja, H.N. (2003). *Order statistics*. New Jersey, John Wiley & Sons Inc. 458 p.
- [Davies & Bouldin 1979] Davies, D.L. & Bouldin, D.W. (1979). A cluster separation measure, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1 (2). pp. 224–227.
- [Davies & Gather 1993] Davies, L. & Gather, U. (1993). The Identification of Multiple Outliers. *Journal of the American Statistical Association*, Vol. 88, No. 423, pp. 782-792.
- [Dempster et al. 1977] Dempster, A.P., Laird, N. M. & Rubin, D. B. (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, Vol. 39, No. 1. pp. 1-38.
- [DeVor et al. 1992] DeVor R.E., Chang, T. & Sutherland, J.W. (1992). *Statistical Quality Design and Control*. New York, Macmillan Publishing Company. 809 p.

-
- [Dillon & Goldstein 1984] Dillon, W. R. & Goldstein, M. (1984). *Multivariate Analysis, Methods and Applications*, John Wiley & Sons Inc. 587 p.
- [Duda et al. 2001] Duda, R.O., Hart, P.E. & Stork, D.G. (2001). *Pattern Classification*, 2nd Edition, John Wiley & Sons. 680 p.
- [Everitt et al. 2001] Everitt, B., Landau, S. & Leese, M. (2001). *Cluster analysis*. Edition: 4, Arnold, London. 237 p.
- [Eskin 2000] Eskin, E. (2000). Anomaly Detection over Noisy Data using Learned Probability Distributions. *Proceedings of the Seventeenth International Conference on Machine Learning*. pp. 255 - 262.
- [Ester et al. 1996] Ester, M., Kriegel, H.-P., Sander, J. & Xu, X. (1996). A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In: *Proc. of 2nd International Conference on Knowledge Discovery and Data Mining (KDD'96)*. pp. 226-231.
- [Fawcett & Provost 1997] Fawcett, T. & Provost, F. (1997). Adaptive Fraud Detection. *Data Mining and Knowledge Discovery*. Vol. 1, Iss. 3, pp. 291-316.
- [Feldmann & Whitt 1998] Feldmann, A. & Whitt, W. (1998). Fitting mixtures of exponentials to long-tail distributions to analyze network performance models, *Performance Evaluation* 31, (3-4), pp. 245-279.
- [Filzmoser et al. 2008] Filzmoser, P., Maronna, R. & Werner, M. (2008). Outlier identification in high dimensions. *Computational Statistics & Data Analysis*, Vol. 52, Iss. 3, pp. 1694-1711.
- [Filzmoser & Todorov 2013] Filzmoser, P. & Todorov, V. (2013). Robust tools for the imperfect world. *Information Sciences*, Vol. 245, 1, pp. 4-20.
- [Flanagan 2003] Flanagan, A. (2003). Unsupervised Cluster Discovery using the Self-Organizing Map. In *Proc. of International Conference on Artificial Neural Networks and Neural Information Processing (ICANN/ICONIP 2003)* Istanbul, Turkey, pp. 9-12.
- [Freeman 2004] Freeman, L. (2004). *Telecommunication system engineering*. John Wiley & Sons, Inc. 991 p.

- [Fuchs & Kenett 1998] Fuchs, C. & Kenett, R. (1998). *Multivariate Quality Control*, Marcel Dekker, Inc., New York, 212 p.
- [Fujimaki 2008] Fujimaki, R. (2008). Anomaly Detection Support Vector Machine and Its Application to Fault Diagnosis. Eighth IEEE International Conference on Data Mining. pp. 797-802.
- [Gabriel 1971] Gabriel, K. R. (1971). The Biplot Graphic Display of Matrices with Application to Principal Component Analysis. *Biometrika*, Vol. 58, No. 3, pp. 453-467.
- [Gao & Tan 2006] Gao, J., & Tan, P. N. (2006). Converting output scores from outlier detection algorithms into probability estimates. In Proceedings of the 6th IEEE international conference on data mining (ICDM), Hong Kong, China, pp. 212-221. doi:10.1109/ICDM.2006.43.
- [Ghosh & Reilly 1994] Ghosh, S. & Reilly, D. L. (1994). Credit card fraud detection with a neural-network. In Proceedings of the 27th Annual Hawaii International Conference on System Science. vol. 3. pp. 621-630.
- [Ghosh et al. 1999] Ghosh, A. K., Schwartzbard, A., & Schatz, M. (1999). Learning program behavior profiles for intrusion detection. In Proceedings of the 1st USENIX Workshop on Intrusion Detection and Network Monitoring. pp. 51–62.
- [Gnanadesikan & Kettenring 1972] Gnanadesikan, R. & Kettenring, J.R. (1972). Robust Estimates, Residuals, and Outlier Detection with Multireponse Data. *Biometrics*, Vol. 28, No. 1, pp. 81-124.
- [Gnanadesikan et al. 1995] Gnanadesikan, R., Kettenring, J.R. & Tsao, S.L. (1995). Weighting and selection of variables for cluster analysis. *Journal of Classification*. Springer New York, Vol. 12, Num. 1 / March, 1995 pp. 113-136.
- [Gómez et al. 2009] Gómez, J., Gil, C., Padilla, N., Baños, R. & Jiménez, C. (2009). Design of a Snort-Based Hybrid Intrusion Detection System. *Distributed Computing, Artificial Intelligence, Bioinformatics, Soft Computing, and Ambient Assisted Living*. Vol. 5518/2009. pp. 515-522.

-
- [Gondek & Hofmann 2005] Gondek, D. & Hofmann, T. (2005). Non-redundant clustering with conditional ensembles. In Proceedings of the 11th ACM international conference on knowledge discovery and data mining (SIGKDD), Chicago, IL. pp. 70-77. doi:10.1145/1081870.1081882
- [Grant & Leavenworth 1996] Grant, E.L. & Leavenworth, R.S. (1996). Statistical quality control. 7th ed. New York, McGraw-Hill. 764 p.
- [Green 1976] Green, R.F. (1976). Outlier-Prone and Outlier-Resistant Distributions. *Journal of the American Statistical Association*, Vol. 71, No. 354, pp. 502-505.
- [Grubbs 1969] Grubbs, F. E. (1969). Procedures for Detecting Outlying Observations in Samples. *Technometrics* 11. pp. 1–21.
- [Guha et al. 1998] Guha, S., Rastogi, R & Shim, K. (1998). CURE: an efficient clustering algorithm for large databases. Proceedings of the 1998 ACM SIGMOD international conference on Management of data. pp. 73 - 84.
- [Gupta et al. 2013] Gupta, A., Toshniwal, D., Gupta, P.K., Khurana, V, & Upadhyay, P. (2013). Extracting anomalies from time sequences derived from nuclear power plant data by using fixed width clustering algorithm. *Advances in Computing, Communications and Informatics (ICACCI)*. pp. 1587 - 1592.
- [Görnitz et al. 2013] Goernitz, N., Kloft, M., Rieck, K. & Brefeld, U. (2013). Toward Supervised Anomaly Detection. *Journal of Artificial Intelligence Research*, 46, pp. 235-262.
- [Hadi 1992] Hadi, A. S. (1992). Identifying Multiple Outliers in Multivariate Data. *Journal of the Royal Statistical Society. Series B (Methodological)*, Vol. 54, No. 3, pp. 761-771.
- [Hadi 1994] Hadi. A.S. (1994). A modification of a method for the detection of outliers in multivariate samples. *Journal of the Royal Statistical Society, B*, 56(2), pp. 393-396.
- [Hadi et al. 2009] Hadi, A. S., Rahmatullah Imon, A. H. M. & Werner, M. (2009). Detection of outliers. *Wiley Interdisciplinary Reviews: Computational Statistics*, Vol. 1 Iss. 1, pp. 57 - 70.
- [Hair et al. 1995] Hair, J.F., Anderson, R.E., Tatham, R.L. & Black, W.C., (1995). *Multivariate Data Analysis: With Readings*, Prentice Hall College Div; 4th edition. 745 p.

- [Hand 2006] Hand, D.J. (2006). Classifier Technology and the Illusion of Progress. *Statistical Science* 21(1). 1-15.
- [Hand et al. 2001] Hand, D., Mannila, H. & Smyth, P. (2001). *Principles of Data Mining*. Cambridge, MA., The MIT Press. 546 p.
- [Harada et al. 2008] Harada, S., Kawahara, R., Mori, T., Kamiyama, N., Hasegawa, H. & Yoshino, H. (2008). A Method of Detecting Network Anomalies in Cyclic Traffic. *Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008*. IEEE. Nov. 30 2008-Dec. 4 2008. pp. 1 - 5.
- [Hardin & Rocke 2004] Hardin, J. & Rocke, D. M., (2004). Outlier detection in the multiple cluster setting using the minimum covariance determinant estimator, *Computational Statistics & Data Analysis*, Vol. 44, Iss. 4, 28 January 2004, pp. 625-638.
- [Harmeling et al. 2006] Harmeling, S., Dornhege, G., Tax, D., Meinecke, F. & Müller, K-R. (2006). From outliers to prototypes: Ordering data. *Neurocomputing*, Vol. 69, Iss. 13-15, pp. 1608-1618.
- [Hauskrecht et al. 2013] Hauskrecht, M., Batal, I., Valko, M., Visweswaran, S., Cooper, G.F. & Clermont, G. (2013). Outlier detection for patient monitoring and alerting. *Journal of Biomedical Informatics* 46(1), pp. 47-55.
- [Hautamäki et al. 2004] Hautamaki, V., Kärkkäinen, I. & Fränti, P. (2004). Outlier detection using k-nearest neighbour graph. *Proceedings of the 17th International Conference on Pattern Recognition, ICPR 2004*. Vol.3. pp. 430 - 433.
- [Hawkins 1980] Hawkins, D. (1980). *Identification of Outliers*, Chapman & Hall, London. 188 p.
- [Hawkins et al. 2002] Hawkins, S., He, H., Williams, G. & Baxter, R. (2002). Outlier Detection Using Replicator Neural Networks. *Data Warehousing and Knowledge Discovery*, 2454, pp. 113-123.
- [He et al. 2003] He, Z., Xu. X. & Deng, S. (2003). Discovering cluster-based local outliers. *Pattern Recognition Letters*, Vol. 24, Iss. 9-10, pp. 1641-1650.
- [Hecht-Nielsen 1995] Hecht-Nielsen, R. (1995). Replicator neural networks for universal optimal source coding. *Science*, 269 (5232), pp. 1860-1863.

-
- [van der Heijden et al. 2004] van der Heijden, F., Duin, R.P.W., de Ridder, D. & Tax, D.M.J. (2004). Classification, parameter estimation and state estimation - an engineering approach using Matlab. West Sussex, England, John Wiley & Sons, 424 p.
- [Hodge & Austin 2004] Hodge, V. & Austin, J. (2004). A Survey of Outlier Detection Methodologies, *Artificial Intelligence Review*, Springer Netherlands, Vol. 22, Num. 2 / October, 2004, pp. 85-126.
- [Hoffmann 2007] Hoffmann, H. (2007). Kernel PCA for novelty detection. *Pattern Recognition*, Vol. 40, pp. 863-874.
- [Hollmen & Tresp 1998] Hollmen, J. & Tresp, V. (1999). Call-based Fraud Detection in Mobile Communication Networks using a Hierarchical Regime-Switching Model. *Proceedings of the 1998 Conference (NIPS'11) Advances in Neural Information Processing Systems*, MIT Press. pp. 889–895.
- [Huber 1985] Huber, P. (1985). Projection Pursuit. *Annals of Statistics*. Vol. 13 (2), pp. 435-475.
- [Hyvärinen et al. 2001] Hyvärinen, A., Karhunen, J. & Oja, E. (2001). *Independent Component Analysis*. New York, John Wiley & Sons, Inc. 481 p.
- [Hätönen et al. 2003a] Hätönen, K., Laine, S. & Similä T. (2003). Using the Log-Sig-function to integrate expert knowledge to Self-Organising Map (SOM) based analysis. *IEEE International Workshop on Soft Computing in Industrial Applications*, Birmingham University, New York, June 23-25, 2003. pp. 145- 150.
- [Hätönen et al. 2003b] Hätönen, K., Kumpulainen, P. & Vehviläinen, P. (2003). Pre- and Post-processing for Mobile Network Performance Data. In: Tuokko, R. (ed). *Automaatio03 Seminar. Automation Makes it Work - Finnish Society of Automation*. September 9-11, 2003, Helsinki. pp. 311-316.
- [Hätönen 2009] Hätönen, K. (2009). *Data mining for telecommunications network log analysis*. Doctoral Thesis, Helsinki University.

- [Höglund et al. 2000] Höglund, A.J., Hätönen, K. & Sorvari, A.S. (2000). A computer host-based user anomaly detection system using the self-organizing map, Proc. IEEE-INNS-ENNS International Joint Conference on Neural Networks (IJCNN), vol. 5, IEEE, 2000, pp. 411–416.
- [Isaksson & Dunham 2009] Isaksson, C. & Dunham, M. (2009). A Comparative Study of Outlier Detection Algorithms. Machine Learning and Data Mining in Pattern Recognition. Springer Berlin / Heidelberg. pp. 440-453.
- [ITU-T 2000] ITU-T (2000). Principles for a telecommunications management network. ITU-T Recommendation M.3010. 44 p.
- [Izenman 2008] Izenman, A. J. (2008). Modern Multivariate Statistical Techniques. Springer, New York. 731 p.
- [Jackson 1991] Jackson, J.E. (1991). A User's Guide to Principal Components. New York, John Wiley & Sons, Inc. 569 p.
- [Jain et al. 1999] Jain, A. K., Murty, M. N. & Flynn, P. J. (1999). Data clustering: a review. ACM Computing Surveys. Vol 31 (3). pp. 264-323.
- [Janacek & Meikle 1997] Janacek, G. J. & Meikle, S. E., (1997). Control Charts Based on Medians. The Statistician, Vol. 46, No. 1. pp. 19-31.
- [Jiang & Papavassiliou 2003] Jiang, J. & Papavassiliou, S. (2003). A network fault diagnostic approach based on a statistical traffic normality prediction algorithm. Global Telecommunications Conference, 2003. GLOBECOM '03. IEEE, Vol. 5, pp. 2918 - 2922.
- [Jin et al. 2001] Jin, W., Tung, A., & Han, J. (2001). Mining top-n local outliers in large databases. Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining San Francisco, California. pp. 293 - 298.
- [Johnson & Wichern 1998] Johnson, R.A. & Wichern, D.W. (1998). Applied multivariate statistical analysis. 4th Edition. New Jersey, Prentice-Hall Inc. 816 p.

-
- [Johnson et al. 1998] Johnson, T., Kwok, I. & Ng, R. (1998). Fast Computation of 2-D depth Contours. Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 1998. pp. 224–228.
- [Jolliffe 2002] Jolliffe, I.T. (2002). Principal Component Analysis. 2nd Edition. New York, Springer-Verlag. 487 p.
- [Kaaranen et al. 2001] Kaaranen, H., Ahtiainen, A., Laitinen, L., Naghian, S. & Niemi, V. (2009). UMTS Networks. Chichester, Wiley. 302 p.
- [Kafadar & Morris 2002] Kafadar, K. & Morris, M. (2002). Data-based detection of potential terrorist attacks on airplanes. White paper. Available at: <http://www.amstat.org/sections/sdns/kafada.pdf>
- [Kanaoka & Okamoto 2003] Kanaoka, A. & Okamoto, E. (2003). Multivariate statistical analysis of network traffic for intrusion detection, in: Proceedings of the 14th International Workshop on Database and Expert Systems Applications, IEEE, 1–5 September 2003, pp. 472–476.
- [Kandel 1992] Kandel, R. (1992). Our changing climate. McGraw-Hill, New York. 126 p.
- [Kandogan 2001] Kandogan, E. (2001). Visualizing multi-dimensional clusters, trends, and outliers using star coordinates. Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 107 - 116.
- [Kano et al. 2003] Kano, M., Hasebe, S. & Hashimoto, I. (2003). Evolution of Multivariate Statistical Process Control: Application of Independent Component Analysis and External Analysis. Proceedings of The Foundations of Computer Aided Process Operations Conference (FOCAPO 2003). Coral Springs, US, Jan. 12-15 2003. pp. 385-388.
- [Karioti & Caroni 2002] Karioti, V. & Caroni, C. (2002). Detecting outlying series in sets of short time series. Computational Statistics & Data Analysis, Vol. 39, Iss. 3, pp. 351-364.
- [Kaufman & Rousseeuw 1990] Kaufman, L. & Rousseeuw, P. J. (1990). Finding Groups in Data: An Introduction to Cluster Analysis. Wiley-Interscience. 342 p.

- [Khedher et al. 2002] Khedher, H., Valois, F. & Tabbane, S. (2001). Traffic characterization for mobile networks. Proceedings of the 56th Vehicular Technology Conference, 24-28 Sept. 2002, IEEE, Vol. 3, pp. 1485 - 1489.
- [Kiviluoto 1996] Kiviluoto, K. (1996). Topology Preservation in Self-Organizing Maps. In: International Conference on Neural Networks (ICNN) 294-299
- [Knorr & Ng 1998] Knorr, E. & Ng, R. (1998). Algorithms for mining distance-based outliers in large datasets. In Proc. of the VLDB Conference, New York, USA, September 1998, pp. 392-403.
- [Knorr et al. 2000] Knorr, E. M., Ng, R. T. & Tucakov, V. (2000). Distance-Based Outliers: Algorithms and Applications. The VLDB Journal The International Journal on Very Large Data Bases, 8(3-4), Springer Berlin / Heidelberg. pp. 237-253.
- [Knorr et al. 2001] Knorr, E., Ng, R. & Zamar, R. (2001). Robust space transformations for distance-based operations. In: Proceedings of the 7th International Conference on Knowledge Discovery and Data Mining 2001. pp. 126-35.
- [Kohonen 1995] Kohonen, T. (1995). Self-Organizing Maps. Series in Information Sciences, vol. 30. Springer, Heidelberg. 362p.
- [Kou et al. 2004] Kou, Y., Lu, C-T., Sirwongwattana, S. & Huang, Y-P. (2004). Survey of fraud detection techniques, IEEE International Conference on Networking, Sensing and Control, 2004 Vol.: 2, pp. 749- 754.
- [Kruegel & Vigna 2003] Kruegel, C. & Vigna, G. (2003). Anomaly detection of web-based attacks. Proceedings of the 10th ACM conference on Computer and communications security, Washington D.C., USA. pp. 251 - 261.
- [Kruskal 1960] Kruskal, W. H. (1960). Some Remarks on Wild Observations, Technometrics, Vol. 2, No. 1 , pp. 1-3.
- [Kumar & Orlin 2008] Kumar. M. & Orlin, J. B. (2008). Scale-invariant clustering with minimum volume ellipsoids. Computers & Operations Research, Vol. 35, Iss. 4, pp. 1017-1029.

-
- [Kumpulainen & Hätönen 2007] Kumpulainen, P. & Hätönen, K. (2007). Local Anomaly Detection for Network System Log Monitoring. In: Konstantionis, M. & Lazaros, I. (eds.). EANN 2007. Proceedings of the 10th International Conference on Engineering Applications of Neural Networks. 29-31 August 2007, Thessaloniki, Greece. pp. 34-44.
- [Kumpulainen & Hätönen 2008a] Kumpulainen, P. & Hätönen, K. (2008). Local anomaly detection for mobile network monitoring. *Information Sciences*. Vol. 178, Iss. 20, pp. 3840-3859.
- [Kumpulainen & Hätönen 2008b] Kumpulainen, P. & Hätönen, K. (2008). Compression of Cyclic Time Series Data. In: Benoit, E. (ed.). Proceedings of the 12th IMEKO TC1 & TC7 Joint Symposium on Man Science & Measurement. September 3-5, 2008, Annecy, France. pp. 413-419.
- [Kumpulainen & Hätönen 2008c] Kumpulainen, P. & Hätönen, K. (2008). Anomaly Detection Algorithm Test Bench for mobile Network Management. *MathWorks/MATLAB User Conference Nordic. The MathWorks Conference Proceedings*. November 20-21, 2008, Stockholm, Sweden. 8 p.
- [Kumpulainen et al. 2009] Kumpulainen, P., Kylväjä, M. & Hätönen, K. (2009). Importance of Scaling in Unsupervised Distance-Based Anomaly Detection. *Proceedings of IMEKO XIX World Congress. Fundamental and Applied Metrology*. September 6-11, 2009, Lisbon, Portugal. pp. 2411-2416.
- [Kumpulainen et al. 2011] Kumpulainen P., Särkioja, M., Kylväjä M., Hätönen K. (2011). Finding 3G Mobile Network Cells with Similar Radio Interface Quality Problems. *Proceedings of 12th INNS EANN-SIG International Conference, Part I. IFIP AICT 363*. pp. 392-401.
- [Kumpulainen & Hätönen 2012] Kumpulainen, P. & Hätönen, K. (2012). Characterizing Mobile Network Daily Traffic Patterns by 1-Dimensional SOM and Clustering. In: Jayne, C. Yue, S. & Lazaros, I. (eds.). *Proceedings of the 13th International Conference on Engineering Applications of Neural Networks*. 20-23 September 2012, London, UK. CCIS Vol 311, Springer, pp. 325-333.

- [Kumpulainen et al. 2013] Kumpulainen P., Särkioja, M., Kylväjä M., Hätönen K. (2013). Analysing 3G radio network performance with fuzzy methods. *Neurocomputing* 107 (2013) pp. 49-58. DOI: 10.1016/j.neucom.2012.07.033
- [Kylväjä et al. 2004] Kylväjä, M., Hätönen, K., Kumpulainen, P., Laiho, J., Lehtimäki, P., Raivio, K. & Vehviläinen, P. (2004). Trial report on self-organizing map based analysis tool for radio networks. *IEEE 59th. Vehicular Technology Conference. VTC 2004-Spring. Vol. 4.* 17-19. pp. 2365- 2369.
- [Kylväjä et al. 2005] Kylväjä, M., Kumpulainen, P. & Hätönen, K. (2005). Information Summarization for Network Performance Management, In: M. Laszlo, J.V. Zsolt, (eds.). *Proceedings of the 10th IMEKO TC10 International Conference on Technical Diagnostics, Budapest, Hungary*, pp. 167-172.
- [Laiho et al. 2005] Laiho, J., Raivio, K., Lehtimäki, P., Hätönen, K. & Simula, O. (2005). Advanced analysis methods for 3G cellular networks. *IEEE Transactions on Wireless Communications, Vol. 4, Iss. 3*, pp. 930 - 942.
- [Lakhina et al. 2005] Lakhina, A., Crovella, M. & Diot, C. (2005). Mining anomalies using traffic feature distributions. In *SIGCOMM '05: Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications*, pp. 217-228.
- [Laurikkala 2009] Laurikkala, M. (2009). Goodness-of-fit tests and heavy-tailed distributions in network traffic data analysis. PhD Thesis, Tampere University of Technology.
- [Lazarevic et al. 2003] Lazarevic, A., Ertöz, L., Kumar, V., Ozgur, A. & Srivastava, J. (2003). A comparative study of anomaly detection schemes in network intrusion detection. In *Proceedings of the Third SIAM International Conference on Data Mining*. pp. 25-36.
- [Leland et al. 2005] Leland, W.E., Taqqu, M.S., Willinger, W. & Wilson, D.V. (1994). On the self-similar nature of Ethernet traffic (extended version), *IEEE/ACM Transactions on Networking* 2 (1), pp. 1–15.

-
- [Leung & Leckie 2005] Leung, K. & Leckie, C. (2005). Unsupervised Anomaly Detection in Network Intrusion Detection Using Clusters, Proceedings of the Twenty-eighth Australasian Conference on Computer Science, Vol. 38, Newcastle, Australia. pp. 333 - 342.
- [Lippmann et al. 2000] Lippmann, R.P., Fried, D.J., Graf, I., Haines, J.W., Kendall, K.R., McClung, D., Weber, D., Webster, S.E., Wyszogrod, D., Cunningham, R.K. & Zissman, M.A. (2000). Evaluating intrusion detection systems: the 1998 DARPA off-line intrusion detection evaluation. DARPA Information Survivability Conference and Exposition. DISCEX '00. Proceedings. Vol. 2, 25-27, pp. 12 - 26.
- [Lu & Ghorbani 2009] Lu, W. & Ghorbani, A.A. (2009). Network Anomaly Detection Based on Wavelet Analysis. EURASIP Journal on Advances in Signal Processing. Vol. 2009, 16 p.
- [Lu & Traore 2005] Lu, W & Traore, I. (2005). A New Unsupervised Anomaly Detection Framework for Detecting Network Attacks in Real-Time. Cryptology and Network Security, Vol. 3810. pp. 96-109.
- [MacQueen 1967] Macqueen, J. (1967). Some methods for classification and analysis of multivariate observations. In: Proc. Fifth Berkeley Symp. on Math. Statist. and Prob. Vol. 1. pp. 281-297.
- [Mahalanobis 1936] Mahalanobis P.C. (1936). On the generalized distance in statistics. Proceedings of the National Institute of Science of India, vol. 2, 1936. pp. 49-55.
- [Manevitz & Yousef 2001] Manevitz, L. & Yousef, M. (2001). One-class SVMs for document classification. The Journal of Machine Learning Research, Vol. 2, Special issue on kernel methods. pp. 139 - 154.
- [Mangiameli et al. 1996] Mangiameli, P., Chen, S.K. & West, D. (1996). A comparison of SOM neural network and hierarchical clustering methods. European Journal of Operational Research. Vol. 93, Iss. 2, pp. 227-448.
- [Marchette & Solka 2003] Marchette, D.J. & Solka J.L. (2003). Using data images for outlier detection. Computational Statistics & Data Analysis 43, pp. 541 - 552.

- [Markou & Singh 2003a] Markou, M. & Singh, S. (2003a). Novelty detection: A review-part 1: Statistical approaches. *Signal Processing*, Vol. 83, Iss. 12, pp. 2481–2497.
- [Markou & Singh 2003b] Markou, M. & Singh, S. (2003b). Novelty detection: A review-part 2: Neural network based approaches. *Signal Processing*, Vol. 83, Iss. 12, pp. 2499–2521.
- [Maronna et al. 2006] Maronna, R.A., Martin, R.D. & Yohai, V.J. (2006). *Robust Statistics: Theory and Methods*. John Wiley & Sons, Chichester. 436 p.
- [Maronna & Yohai 1995] Maronna, R. & Yohai, V. (1995). The Behavior of the Stahel-Donoho Robust Multivariate Estimator. *Journal of the American Statistical Association*, Vol. 90, No. 429, pp. 330-341.
- [Martin 2007] Martin, R.A. (2007). Unsupervised Anomaly Detection and Diagnosis for Liquid Rocket Engine Propulsion. *Aerospace Conference, 2007 IEEE*. pp. 1-15.
- [Martin & Morris 1996] Martin, E. B. & Morris, A. J. (1996). Non-parametric confidence bounds for process performance monitoring charts. *Journal of Process Control*, Vol. 6, Iss. 6, pp. 349-358.
- [McGill et al. 1978] McGill, R., Tukey, J.W., Larsen, W.A. (1978). Variations of Boxplots. *The American Statistician*, 32, pp. 12-16.
- [McLachlan & Peel 2000] McLachlan, G. & Peel, D. (2000). *Finite Mixture Models*, John Wiley & Sons, New York. 456 p.
- [Mehdi et al. 2007] Mehdi, M. Zair, S. Anou, A. & Bensebti, M. (2007). A Bayesian Networks in Intrusion Detection Systems. *Journal of Computer Science* 3 (5). pp. 259-265.
- [Miller et al. 1998] Miller, P., Swanson, R.E. & Heckler, C.F. (1998). Contribution plots: a missing link in multivariate quality control. *Applied Mathematics & Computer Science*. Vol. 8, No. 4. pp. 775–792.
- [Milligan & Cooper 1985] Milligan, G.W. & Cooper M.C. (1985). An examination of procedures for determining the number of clusters in a data set. *Psychometrika*, 50. pp. 159–179.

-
- [Milligan & Cooper 1988] Milligan, G. W. & Cooper, M. C. (1988), A Study of Standardization of Variables in Cluster Analysis. *Journal of Classification*, 5, 181-204.
- [Milton 1990] Milton, J.S. & Arnold, J.C. (1990). *Introduction to Probability and Statistics Principles and Applications for Engineering and the Computing Sciences*. Second edition. Singapore, McGraw-Hill. 700 p.
- [Mittnik et al. 2001] Mittnik, S., Rachev, S. & Samorodnitsky, G. (2001). The distribution of test statistics for outlier detection in heavy-tailed samples. *Mathematical and Computer Modelling*, Vol. 34, Iss. 9-11, pp. 1171-1183.
- [Mouly & Pautet 1992] Mouly, M. & Pautet, M.-B. (1992). *The GSM System for Mobile Communication*. 701 p.
- [Mukherjee & Vapnik 1999] Mukherjee, S. & Vapnik, V. (1999). Multivariate density estimation: a support vector machine approach, Massachusetts Institute Of Technology, Artificial Intelligence Laboratory and Center For Biological and Computational Learning Department of Brain And Cognitive Sciences, A.I. Memo No. 1653 April 1999, C.B.C.L Paper No. 170, <ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-1653.pdf>
- [Mukkamala et al. 2002] Mukkamala, S., Janoski, G. & Sung, A. (2002). Intrusion detection using neural networks and support vector machines. *IJCNN '02. Proceedings of the 2002 International Joint Conference on Neural Networks*. Vol. 2, pp. 1702 - 1707.
- [Mukkamala et al. 2005] Mukkamala, S., Sung, A.H. & Abraham, A. (2005). Intrusion detection using an ensemble of intelligent paradigms. *Journal of Network and Computer Applications*, Vol. 28, Iss. 2, pp. 167-182.
- [Muñoz & Muruzábal 1998] Muñoz, A. & Muruzábal, J. (1998). Self-organizing maps for outlier detection. *Neurocomputing*, Vol. 18, Iss. 1-3, pp. 33-60.
- [Nabney 2001] Nabney, I.T. (2001). *NETLAB Algorithms for Pattern Recognition*, *Advances in Pattern Recognition*, Springer. 420 p.

- [Nisbet et al. 2009] Nisbet, R., Elder, J. & Miner, G. (2009). Handbook of statistical analysis and data mining applications. ACADEMIC PRESS 864 p.
- [Ochs et al. 2013] Ochs, M.F., Farrar, J.E., Considine, M., Wei, Y., Meschini, S. & Arceci, R.J. (2013). Outlier Gene Set Analysis Combined with Top Scoring Pair Provides Robust Biomarkers of Pathway Activity. Pattern Recognition in Bioinformatics LNCS Volume 7986, pp. 47-58.
- [de Oliveira & Pedrycz 2007] de Oliveira, J. V. & Pedrycz, W. Eds. (2007). Advances in Fuzzy Clustering and its Applications, John Wiley & Sons Ltd, Chichester, West Sussex
- [Oxford 2005] The Oxford Dictionary of English, Revised Edition © Oxford University Press 2005.
- [Patcha & Park 2007] Patcha, A. & Park, J.-M. (2007). An overview of anomaly detection techniques: Existing solutions and latest technological trends. Computer Networks. Vol. 51, Iss. 12, pp. 3448–3470.
- [Paxson & Floyd 1995] Paxson, V. & Floyd, S. (1995). Wide area traffic: the failure of Poisson modeling, IEEE/ACM Transactions on Networking 3 (3), 226–244.
- [Peirce 1852] Peirce, B. (1852). Criterion for the Rejection of Doubtful Observations. Astronomical Journal II 45, pp. 161-163.
- [Peña & Prieto 2001] Peña, D. & Prieto, F.J. (2001). Multivariate Outlier Detection and Robust Covariance Matrix Estimation. Technometrics, Vol. 43, No. 3, pp. 286–310.
- [Penny & Jolliffe 2001] Penny, K.T. & Jolliffe, I.T. (2001). A Comparison of Multivariate Outlier Detection Methods for Clinical Laboratory Safety Data. The Statistician, Vol. 50, No. 3, pp. 295-308.
- [Portnoy et al. 2001] Portnoy, L., Eskin, E. & Stolfo, S. (2001). Intrusion Detection with Unlabeled Data Using Clustering. In Proceedings of ACM CSS Workshop on Data Mining Applied to Security (DMSA-2001). 14 p.
- [Rajala 2009] Rajala M. (2009). Data-based modelling and analysis of coherent networked systems with applications to mobile telecommunications networks. Doctoral Thesis, Tampere University of Technology.

-
- [Ramadas et al. 2003] Ramadas, M. & Tjaden, S.O.B. (2001). Detecting anomalous network traffic with self-organizing maps. Proceedings of the 6th International Symposium on Recent Advances in Intrusion Detection, Pittsburgh, PA, USA, 2003, pp. 36–54.
- [Ramaswamy et al. 2000] Ramaswamy, S., Rastogi, R. & Shim, K. (2000). Efficient algorithms for mining outliers from large data sets, in: Proceedings of the ACM SIGMOD International Conference on Management of Data, Dallas, TX, USA, 2000, pp. 427–438.
- [Rocke & Woodruff 1996] Rocke, D. M, & Woodruff, D. L. (1996). Identification of outliers in multivariate data. Journal of the American Statistical Association. Vol. 91, Iss. 435; pp. 1047-1060.
- [Rocco & Zio 2007] Rocco S., C.M. & Zio, E. (2007). A support vector machine integrated system for the classification of operation anomalies in nuclear components and systems. Reliability Engineering & System Safety, Vol. 92, Iss. 5, pp. 593-600.
- [Rousseeuw 1987] Rousseeuw, P.J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis, Journal of Computational and Applied Mathematics 20. pp. 53–65.
- [Rousseeuw & Leroy 1987] Rousseeuw, P. J. & Leroy, A. M. (1987). Robust Regression and Outlier Detection. John Wiley & Sons, Inc., New York, 329 p.
- [Šaltenis 2004] Šaltenis, V. (2004). Outlier Detection Based on the Distribution of Distances between Data Points. Informatica, Vol. 15 , Iss. 3, pp. 399-410.
- [Sastry et al. 2007] Sastry,C.S., Rawat, S., Pujari, A.K. & Gulati, V.P. (2007). Network traffic analysis using singular value decomposition and multiscale transforms, Information Sciences 177 (23) pp. 5275–5291.
- [Scarfone & Mell 2007] Scarfone, K. & Mell, P. (2007). Guide to Intrusion Detection and Prevention Systems (IDPS). NIST Special Publication 800-94. Recommendations of the National Institute of Standards and Technology. 127 p.

- [Schubert et al. 2012] Schubert, E., Wojdanowski, R., Zimek, A., Kriegel, H.-P.. (2012). On Evaluation of Outlier Rankings and Outlier Scores. In Proceedings of the 12th SIAM International Conference on Data Mining (SDM), Anaheim, CA. pp. 1047-1058.
- [Schölkopf et al. 2001] Schölkopf, B., Platt, J., Shawe-Taylor, J., Smola, A.J. & Williamson, R. (2001). Estimating the support of a high-dimensional distribution. *Neural computation*, Vol. 13, No. 7, pp. 1443-1472.
- [Shewhart 1931] Shewhart, W.A. (1931). *Economic Control of Quality of Manufactured Product*. New York, D. Van Nostrand Company, Inc. 501 p.
- [Shon & Moon 2007] Shon, T. & Moon, J. (2007). A hybrid machine learning approach to network anomaly detection, *Information Sciences* 177. pp. 3799–3821.
- [Shyu et al. 2003] Shyu, M.-L., Chen, S.-C., Sarinnapakorn, K. & Chang, L.-W. (2003). A novel anomaly detection scheme based on principal component classifier. In Proceedings of the IEEE Foundations and New Directions of Data Mining Workshop. 10 p.
- [Singh & Upadhyaya 2012] Singh, K., Upadhyaya, S. (2012). Outlier Detection: Applications And Techniques. *International Journal of Computer Science Iss.s*, Vol. 9, Iss. 1, No 3. pop. 307-323.
- [Song et al. 2007] Song, X. Wu, M., Jermaine, C. & Ranka, S. (2007). Conditional Anomaly Detection. *IEEE Transactions on Knowledge and Data Engineering*, Vol. 19, Iss. 5, pp. 631–645.
- [Stefano et al. 2000] De Stefano, C., Sansone, C. & Vento, M. (2000). To reject or not to reject: that is the question-an answer in case of neural classifiers. *Systems, Man, and Cybernetics, Part C: Applications and Reviews*, *IEEE Transactions on*, 30 (1), pp. 84 - 94.
- [Steinley & Brusco 2007] Steinley, D. & Brusco, M. (2007). Initializing K-means Batch Clustering: A Critical Evaluation of Several Techniques. *Journal of Classification*. Springer New York, Vol. 24, Num. 1. pp. 99-121.

-
- [Stolfo et al. 2000] Stolfo, S.J., Fan, W., Lee, W., Prodromidis, A. & Chan, P.K. (2000). Cost-based modeling for fraud and intrusion detection: results from the JAM project. DARPA Information Survivability Conference and Exposition. DISCEX '00. Proceedings Vol. 2, 25-27 Jan. 2000. pp. 130 - 144.
- [Subramanian 2000] Subramanian, M. (2000). Network management: An introduction to principals and practise. Addison-Wesley. 644 p.
- [Suutarinen 1994] Suutarinen, J. (1994). Performance Measurements of GSM Base Station System. Thesis (Lic.Tech.) Tampere University of Technology.
- [Tan et al. 2005] Tan, P., Steinbach, M. & Kumar, V. (2005). Introduction to Data Mining. Addison-Wesley. 769 p.
- [Tóth & Gosztolya 2004] Tóth, L. & Gosztolya, G. (2004). Replicator Neural Networks for Outlier Modeling in Segmental Speech Recognition. Advances in Neural Networks, 3173. pp. 996-1001.
- [Tukey 1977] Tukey, L.W. (1977). Exploratory Data Analysis. Reading, Ma.: Addison-Wesley. 506 p.
- [Bache & Lichman 2013] Bache, K. & Lichman, M. (2013). UCI Machine Learning Repository [<http://archive.ics.uci.edu/ml>]. Irvine, CA: University of California, School of Information and Computer Science.
- [Ultsch & Siemon 1990] Ultsch, A., Siemon, H.P. (1990). Kohonen's Self-Organizing Feature Maps for Exploratory Data Analysis. In: International Neural Network Conference. pp. 305-308.
- [Vaarandi 2013] Vaarandi, R. (2013). Detecting anomalous network traffic in organizational private networks. IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA). pp. 285-292.
- [van der Heijden et al. 2004] van der Heijden, F., Duin, R., de Ridder, D., Tax, D.M.J. (2004) Classification, Parameter Estimation and State Estimation: An Engineering Approach Using MATLAB. Wiley. 440 p.
- [Vapnik 1998] Vapnik, V. (1998). Statistical learning theory. New York: Wiley. 736 p.

- [Vehviläinen 2005] Vehviläinen P. (2004). Data mining for managing intrinsic quality of service in digital mobile telecommunications networks. Doctoral Thesis, Tampere University of Technology.
- [Vesanto 1999] Vesanto, J. (1999). SOM-based data visualization methods, *Intelligent Data Analysis*, 3 (2), pp. 111-126.
- [Vesanto & Alhoniemi 2000] Vesanto, J. & Alhoniemi, E. (2000). Clustering of the Self-Organizing Map. *IEEE Transactions on Neural Networks*, Vol. 11, No. 3, pp. 586-600.
- [Vesanto et al. 1999] Vesanto, J., Himberg, J., Alhoniemi, E. & Parhankangas, J. (1999). Self-organizing Map in Matlab: the SOM Toolbox. In *Proceedings of the Matlab DSP Conference 1999*, Espoo, Finland, November 16–17, pp. 35–40.
- [Ward 1963] Ward Jr., J.H. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association* 58 (301), pp. 236-244.
- [Williams et al. 2002] Williams, G., Baxter, R., He, H., Hawkins, S. & Gu, L. (2002). A comparative study of RNN for outlier detection in data mining. In *Proceedings of the 2nd IEEE Int. Conf. on Data Mining*, Maebashi City, Japan, pp. 709-712.
- [Williamson et al. 2005] Williamson, C., Halepovic, E., Sun, H. & Wu, Y. (2005). Characterization of CDMA2000 cellular data network traffic, in: *Proceedings of the IEEE Conference on Local Computer Networks*, Washington, DC, USA, pp. 712–719.
- [Wise et al. 2005] Wise, B.M., Shaver, J.M., Gallagher, N.B., Windig, W., Bro, R. & Koch, R.S. (2005). *PLS_Toolbox 3.5 for use with MATLAB*, Eigenvector Research, Inc. 292 p.
- [Wu & Wang 2013] Wu, S., Wang, S. (2013). Information-Theoretic Outlier Detection for Large-Scale Categorical Data. *IEEE Transactions on Knowledge and Data Engineering* 25(3). pp. 589 - 602.
- [Xu & Wunsch 2005] Xu, R. & Wunsch, D.C. II. (2005). Survey of clustering algorithms. *IEEE Transactions on Neural Networks*, Vol. 16, Iss. 3, pp. 645 - 678.
- [Xu & Wunsch 2009] Xu, R. & Wunsch, D.C. II. (2009). *Clustering*. IEEE Press. 358 p.

-
- [Yamanishi et al. 2000] Yamanishi, K., Takeuchi, J., Williams, G. J. & Milne, P. W. (2000). On-line unsupervised outlier detection using finite mixtures with discounting learning algorithms. Proceedings of the sixth ACM SIGKDD international conference on knowledge discovery and data mining. pp. 320 - 324.
- [Yamanishi & Takeuchi 2001] Yamanishi, K. & Takeuchi, J. (2001). Discovering outlier filtering rules from unlabeled data. Proceedings of the seventh ACM SIGKDD international conference on knowledge discovery and data mining. pp. 389 - 394.
- [Ye et al. 2002] Ye, N., Emran, S.M., Chen, Q. & Vilbert, S. (2002). Multivariate statistical analysis of audit trails for host-based intrusion detection, IEEE Transactions on Computers 51 (7). pp. 810–820.
- [Zanero 2005] Zanero, S. (2005). Analyzing TCP Traffic Patterns using Self Organizing Maps, ICIAP 05 - Special session on Pattern Recognition in Computer Security. LNCS vol. 3617. pp. 83-90.
- [Zanero 2007] Zanero, S.(2007). Flaws and frauds in the evaluation of IDS/IPS technologies, Forum of Incident Response and Security Teams, Sevilla, Spain, June 2007. pp. 1-18.
- [Zhang et al. 1996] Zhang, T., Ramakrishnan, R., & Livny, M. (1996). BIRCH: an efficient data clustering method for very large databases. Proceedings of the 1996 ACM SIGMOD international conference on management of data 1996, Montreal, Quebec, Canada. pp. 103-114.
- [Zhang & Zulkernine 2006] Zhang, J. & Zulkernine, M. (2006). Anomaly based network intrusion detection with unsupervised outlier detection. IEEE International Conference on Communications, vol. 5, 2006, pp. 2388–2393.
- [Zhang et al. 2008] Zhang, Y., Yang, S. & Wang, Y. (2008). LDBOD: A novel local distribution based outlier detector. Pattern Recognition Letters, Vol. 29, Iss. 7. pp. 967-976.
- [Zimek & Vreeken 2013] Zimek, A. & Vreeken, J. (2013). The blind men and the elephant: on meeting the problem of multiple truths in data from clustering and pattern mining perspectives. Machine Learning, DOI: 10.1007/s10994-013-5334-y.

- [Zweig & Campbell
1993] Zweig, M. & Campbell, G. (1993). Receiver-Operating
Characteristic (ROC) Plots: A Fundamental Evaluation
Tool in Clinical Medicine, Clin. Chem. 39/4. pp. 561-577

Tampereen teknillinen yliopisto
PL 527
33101 Tampere

Tampere University of Technology
P.O.B. 527
FI-33101 Tampere, Finland

ISBN 978-952-15-3228-3
ISSN 1459-2045