Master thesis in Data Mining

REPORT NO. 20008:024
ISSN: 1651-4769

Department of Computer Science

Outlier Detection in Online Gambling

Konstantinos Manikas

IT University of Göteborg
Chalmers University of Technology and University of Gothenburg
Göteborg, Sweden 2008

Abstract

Data mining is field that is increasing in importance and width of application day by day. A sub-domain of data mining, the anomaly detection is also rising in importance the last years. Although discovered a long time ago, the last 5 or 10 years the uses of anomaly detection are increasing, therefore making it a useful technique to discover fraud, network intrusions, medicine side effects and many other useful anomalies within a wide set of data. The task of this master thesis is to find a more optimal anomaly detection technique to uncover fraudulent use or addictive playing in the transaction data of online gambling websites. This work is conducted on behalf of a Swedish company that is occupied in the field of data mining. For the needs of this work an anomaly detection method has been adapted, implemented and tested. The evaluation of this method is done by comparing the results it brings with the anomaly detection technique currently used for the same purpose.

1 Introduction

It is a commonly accepted fact that technology nowadays is advancing rapidly. New or improved versions of devices, programs and methods are making our every day's tasks faster and more efficient. This technological progress is also affecting the way that we manipulate data. Today storage space is more affordable than ever. At the same time, data collection has become easier within the past years. People, nowadays, are more willing to share personal data than 10 years ago. Moreover, the use of internet has increased in such a manner that internet itself can serve as a big data source. Everywhere around us information is being collected from government agencies, scientific institutions and businesses to the tram stop or super market around the corner. But what happens to all these data? The average datasets collected have increased so much that the stored data are really hard, if not impossible, to be processed by human minds. The need of a process to go through these data and come up with new relations, models or patterns has been greater than ever. This process is called *Data Mining*.

1.1 Data Mining

1.1.1 What is Data Mining

Data Mining is the automated process of going through large amounts of data with the intention to discover useful information about the data that is not obvious. Useful information may include special relations between the data, specific models that the data repeat itself, specific patterns, and ways of classifying it or discovering specific values that fall out of the "normal" pattern or model. According to Tan et al., "data mining blends traditional data analysis methods with sophisticated algorithms for processing large volumes of data" [22]. It also provides possibilities to explore data in new ways with the use of artificial intelligence techniques and neural networks.

Data Mining is derived by the combination of different areas: Statistics, database technology, artificial intelligence, pattern recognition, machine learning and visualization [10]. All these fields have very vague borders that define them. This makes it difficult to distinguish where each of these fields overlap and where does one field end data mining begins. A good description of data mining and the fields that influence it is provided in Figure 1

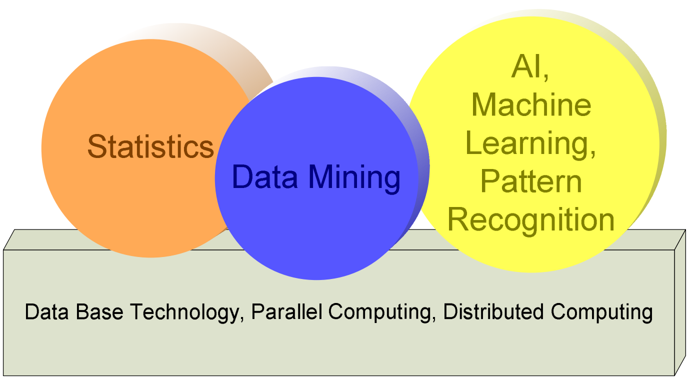

Figure 1. Data Mining and the fields of influence

Data mining is usually applied to data that has already been collected for different reasons. This means that data mining is many times applied to a dataset after the data have been collected and stored and without taking into consideration the purpose of the data mining procedures. Therefore it does not include any data collection principles. This is the difference with statistics. In statistics the data are collected in order to answer specific questions.

Knowledge Discovery in Databases

Data mining is a part of Knowledge Discovery in Databases (KDD). As KDD, we define the process of extracting useful information from "raw" data. This process includes several stages, as shown in Figure 2, where the raw data are input in the process, converted into an appropriate format, applied data mining techniques and then post processed. In the pre-processing procedure the data are selected, adapted to an acceptable format and subjected to "data cleaning". In data cleaning, the data are checked for invalid or double records leaving only the useful data for the next step. In the post processing procedure, the results of the data mining methods are converted into the useful information. This might be different visualisations of the results or display of the data patterns or results.[15]

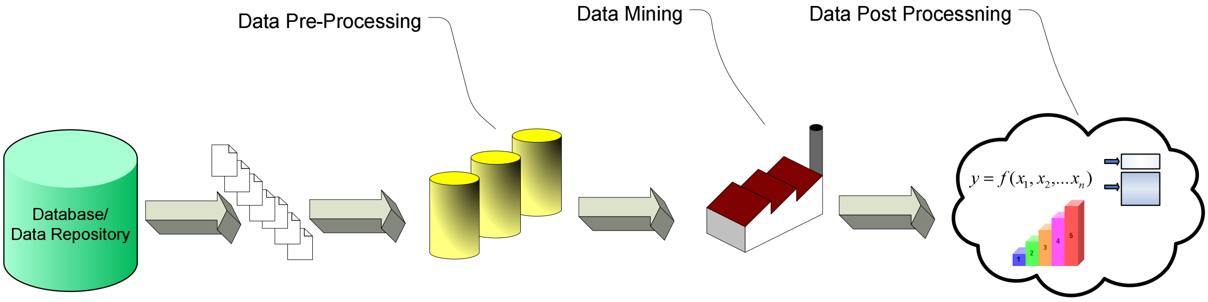

Figure 2. The KDD stages

Additional issues on Data Mining

As it is mentioned earlier data mining is the process of going through large datasets and discovering information about the data. However, not all information retrieval processes can be considered data mining processes. A simple selection of some fields in a table of a database or the hypothesis that since an employee is on pregnancy leave, she would be female cannot be considered as data mining. The information that is revealed about the data should not be clear before the process of the data mining unless the intention is to confirm a hypothesis. Moreover, data mining procedures are aimed to performed tasks that are impossible (or really hard) to be performed by humans. In that sense, most of the cases, data mining is called to work with really big amount of data, without this being restricting. It is possible for a data mining procedure to discover a specific model in a relatively small dataset. This should be held with additional cautiousness, as a model would appear in every dataset if the data mining engine searches enough. This is the so called *data dredging* (might also be referred to as data snooping or fishing), where a model is discovered in a dataset or a group of different datasets that is too general to include the whole data or datasets.[13]

1.1.2 Uses of Data Mining

Data mining, in its different forms, is used widely today and has resulted in the improvement of many areas. Although a big part of it has been focusing on marketing and customer relations there is a big variety of fields that it is spreading. These fields are summarized by Kantardzic in [13].

Mining in Financial Data

In the financial field data mining has been applied to information concerning banking systems for credit fraud detection, bank customers' credibility judgement, maintaining customers by predicting whether the customer is probable to go to a competitive company. Moreover data analysis is applied to predict stock prices and stock tendency in the stock value exchange.[13], [6]

Telecommunications

Telecommunication is a field that is characterised by high competition. The telecommunication companies many times are faced with the challenge to invest big amounts of money into new technologies with uncertain future. In this field, data mining techniques are applied for the classification of the consumer public and the prediction of its actions. Prediction of the customer's actions can be predicting what kind of services would a new customer probably want, how can one make sure that a customer will continue being a customer and how is it possible to predict when a customer will request a new service/product. Moreover, in the domain of the internet services, data mining is applied for network intrusion detection. Specific data mining methods assist in the detection of intrusions and malicious attacks like Denial of Service (DoS) attacks.

Retail Vendors

The product commerce is a domain that is also becoming more efficient and competitive. Since commerce is one of the oldest professions, profit margins today are getting slimmer and every vendor's dream is to become a "market leader". Additionally, with the increase of e-commerce, in the last 10 years, the marked has stretched geographically but also in terms of competition. Once providing goods online, people from all over the world can make purchases but you are also competing with vendors from all over the world. Therefore the need for improved processes and smart business movement is bigger. This is where data mining intervenes. Data mining is applied to assist in the distinction of the advertising target group (who is more probable to be interested in the specific product and how could this person be reached), in the product association (with the sense of after purchasing diapers it is highly possible to purchase baby milk) and in keeping the customer "loyalty".

Medical Research

Data mining has been playing an important role in the research in new diseases and medication. Patients' medical histories are stored in electronic form containing information about treatments and symptoms. Proper processing of this information is proven to be really useful. Moreover, the breaking of the genetic code in the DNA has opened new doors in the treatment of chronicle diseases like cancer where genetic code anomalies can be detected.

National Security

Data mining has been proven to be really useful in maters of national security. After the terrorist attack of September 11th, 2001, the government of the United States made public two projects that were based on data mining and would assist in the prevention of terrorist attacks. These projects were Terrorism Information Awareness (TIA) and Computer Assisted Passenger Pre-screening System (CAPPS II).TIA was a program of DARPA, research and development organization of the US Department of Defence. The aim of this program was to detect plans of terrorist attacks against targets of American interests. That was achieved with the automated language translation of written text and recorded conversations and the pattern recognition of information collected. CAPPS II was a program applying data mining techniques to characterize the people about to fly with an aeroplane with a specific "score": green, yellow and red. The people with green would have their luggage pass from the normal control. The people with yellow would have their luggage pass from a special control check, while the people with red would not be allowed to fly. [21]

1.1.3 Data Mining Techniques

As we saw from the uses of data mining, data mining is a big field with many applications. This makes it an important part of KDD. Data mining is possible to be studied and explained easier if separated into sub categories according to the task that they perform. These are called the *Data Mining Techniques*.

Within the data mining documentation, there are many different perspectives on how the data mining techniques are separated. Most of these perspectives cover the same fields but with different structure. The choice of the most appropriate separation of the data mining techniques is mostly subjective. A quite common separation is that made by Hand [10], who separates data mining into categories according to the outcome of the tasks they perform. These categories are:

1. **Exploratory Data Analysis**. Which intents to explore the data without aiming somewhere specifically but mostly to extract information concerning the data. The information aim mostly in assisting in the visualisation of the data.
2. **Descriptive Modelling**. That tries to describe the data or the processes that create the data. In this category data are classified into different groups with different methods.
3. **Predictive Modelling: Classification and Regression**. That aims to predict new data. This is achieved by creating a model under which the data are reproduced. According to the correctness of this model, the new values are predicted.
4. **Discovering Pattern and Rules**. This category tries to discover specific pattern and rules of the dataset. Having defined a pattern of dataset makes it possible to detect values that are not complying with this pattern and therefore are anomalous.
5. **Retrieval by Content**. In this category a search for a specific pattern is applied. This category includes image search or web search.

Although complete, this perspective is not ideal for the analysis required in this research as many of the fields are interloping quite often. For the works of this master thesis the perspective of Tan et al. [22] is used to divide the data mining techniques and explain their use. According to this perspective, data mining is divided into four main categories: Classification, Association Analysis, Cluster Analysis and Anomaly Detection.

1.1.4 Classification

Classification is the task of separating each record of a data set into a set of predefined classes. Each record has a set of attributes **x** that characterize it. According to the values of x the record is classified to one of the labels in the set of class labels **y**. So the aim of classification is to create a function f that would classify each attribute set **x** (i.e. a record) to one of the labels in the class label set **y**.

Classification can be used for Descriptive or Predictive Modelling. In predictive modelling classification is used to label a set of already existing records in order to describe or distinguish them in a better way by separating them into different classes. In the predictive modelling, classification is used to classify new and unknown records.

How does Classification work

The classification technique creates a classification model that classifies automatically new records according to their attributes. For this model to be created, a *Training Dataset* and a *Learning Algorithm* is needed. A training dataset is a dataset where the records are already classified. This is an input in the learning model where with the use of the learning algorithm, the classification model is created. After the creation of the classification model, a test dataset is used to estimate the accuracy of the classification model. The test dataset is a dataset that the classes of each row are known but not provided to the model. Therefore, the accuracy of the classification can be calculated by comparing how many of the rows were misclassified by the model classification. Many times the test dataset is a part of the training set that is not provided in the learning process.

There are different techniques to apply data classification. These classification techniques are called classifiers and although they obtain the same objective can vary on they way they achieve it and the effectiveness or cost of training.

Classifiers

One of the common classifiers is the *Decision Tree* classifier. It is in fact the most common and simple way to classify data into labels. It is consisted of a root node, some internal nodes and one or more leaf or terminal nodes in tree structure. When a new record is inserted, it passes from the root nodes and then according to the attribute values it makes a route in the internal nodes to end up in a leaf node that is the class label to be classified. The tree structure is created by the learning algorithms for decision tree. The most common algorithms used for the creation of a decision tree model is the Hunt's algorithm, TreeGrowth, Cart, ID3 and its extension C4.5.

A special attention should be paid in the training dataset of the decision tree. If the training dataset has too few records, then the decision tree will have too few nodes therefore we would have *Model Underfitting.* If the training dataset on the other hand, would have too many records, the model would start fitting perfectly to the specific dataset and therefore misclassify new records. This is called *Model Overfitting.* Model underfitting and overfitting is a general issue of the learning datasets and is not only encountered in decision trees. The size of the training dataset depends on how satisfactory the classification error is.

Apart from the decision tree, there is a variety of classifiers used for classification today. Some are as simple and others are more complicated. Some of the most common classifiers are the Artificial Neural Networks, the Bayesian Classifier, the Rule Classifier, the Nearest

Neighbour and the Support Vector Machines. Each one of the classifiers has its own way of classifying so when considering a classifying task, the best trade-off between learning and preparation cost and classification error rate should be considered, although in many cases the combination of different classifiers in an *ensemble method* is more effective.

1.1.5 Association Analysis

Association analysis is the technique of searching for data patterns and associations within records of big datasets. That means that association analysis tries to find if there are special connections between the records of a dataset. This technique is vastly used today for marketing, advertising, inventory management or customer relationship aims. It analyses data that have been collected by purchases, customer transactions or gallops – questioners but also for more scientific reasons like the analysis of diseases and their causes by looking on the patients' history.

Explanation of Association Analysis

In order to achieve its goal, association analysis, addresses two main issues. Firstly, it needs to make sure that associations encountered are real associations and not happening by luck. Secondly, the cost efficiency issue: the datasets that need to be analysed are usually datasets containing big amounts of data, for example all the purchases made the last year in the main branch of a supermarket chain. Searching for data associations in datasets like this can be an extravagant procedure.

The first issue is handled by measuring the *Support* of the Association Rules. In a simplified example of a retail vendor, the purchase dataset would consist of a table containing the Items to be searched for associations (i.e. the products) as columns and the transactions of what items have been purchased. Therefore we have the itemset, which is the list of available items and the separate transactions that contain a set of items which is a subset of the itemset. An association rule would be the speculation that specific items from the itemset are usually being purchased together. When having an association rule we can measure the validity of this rule by measuring the *Support* and *Confidence* of the rule. The support declares how often this rule is appearing in the dataset while the confidence declares how often the items of the rule appear together. Therefore, in a dataset we can find the associations between the items by creating association rules and measuring the support and confidence of these rules.

However, in really big datasets the measure of the support and confidence for all the possible combinations of items can be really extravagant. This brings up the second issue that association analysis is addressing: The cost efficiency. Association analysis is trying to measure the cost of finding associations in a dataset by calculating the items that are most probable to be included in an association rule with two main techniques: either by excluding items that would not be in an association rule or by reducing the number of calculations needed to find the items that would be in an association rule.

The exclusion of items is achieved with the *Apriori* algorithm. This algorithm is based on the idea that if an item would be in an association rule, then this item would be appearing frequently. With the same sense, if an item does not belong to an association rule, this item would no be appearing so frequently. The apriori algorithm calculates the support of each individual item and automatically excludes the items that are found to have support below a threshold.

On the other hand, the reduction of calculations needed to find items that belong to association rules is done with the *FP-Growth* algorithm. The FP-Growth algorithm creates a tree structure representation of the items, ordering them by importance according to their support and with the item combinations (as they appear in the transactions) appearing as branches. This is called the FP-Tree. After creating this "condensed" representation of the data the association rules can be extracted by taking the combinations of items with the highest support starting from the bottom of the tree.

These two methods are the most common methods for calculating the items that are most probable to be in the association rules. After defining the strongest combinations of these items, the so called *candidate itemsets* the association rules are extracted.

1.1.6 Cluster Analysis

Cluster Analysis is the method of grouping data together according to their characteristics in groups (clusters) that characterise and describe the data. The task of cluster analysis is quite similar to the task of classification. The difference is, however that classification tries to separate data in to a set of predefined classes while in cluster analysis the classes (clusters) are created as part of the analysis. That is why many times in the documentation, cluster analysis is also referred to as *unsupervised classification*.

The cluster separation (the product of cluster analysis) can be called *clustering*. Clusterings are separated in:

Partitional or Hierarchical. Clusterings are characterised as partitional when the data points are not overlapping while they are hierarchical if a cluster is consisted from many clusters. In that case a data point belongs to more that one cluster where each cluster is a subset of the other.

Exclusive, Overlapping or Fuzzy. If the data points of a clustering belong only to one cluster, then this is an exclusive clustering, while if a data point belongs to more than one clusterings equally, then it is overlapping. In fuzzy clustering each data point is assigned a probability from 0 to 1 that it belongs to a cluster. The sum of all cluster probabilities for a data points are equal to 1.

Complete or Partial. Clusterings are also separated into complete when they group all the data or partial when they leave data unclustered.

Clustering Algorithms

There are three common techniques of clustering: Applying Prototype-Based clustering, Hierarchical clustering and Density-Based clustering. These three methods are displayed with the most common algorithms for technique.

The most common algorithm for prototype-based clustering is the *K-means* algorithm. This algorithm defines K cluster centres (centroids) and iterates by assigning each data point to the closest centroid and recalculating the centroid until the centroids do not change position. The number of centroid K is a user defined parameter and is the actual number of clusters the specific clustering will have. The centroids in most of the cases are not in the position of a point. Although simple and vastly used, this algorithm fails to cluster effectively data where the points are not rounded-shaped in contrast with the K-metroid algorithm that the centroids are represented by real points (metroids). This algorithm however has a higher cost.

One of the most representative Hierarchical clustering algorithms is the *Agglomerative Hierarchical Clustering* algorithm. In this algorithm, the clustering starts with each point considered as a cluster and gradually grouping together points that are close to each other. This is done until there is only one cluster. The grouping of points is done with the calculation of the proximity matrix. The proximity matrix is a matrix where the distance of two clusters is calculated and stored. The most common ways to calculate the proximity of two clusters is the MIN that calculates the minimum distance, MAX that calculates the maximum distance and Group Average.

Finally, Density-Based clustering is mostly applied with the *DBSCAN* algorithm. This algorithm is based in the centre-based approach where it checks inside the radium R of a point. If the number of points found within the radium is greater than a point threshold Tp, then the point is a core point. If the number of points is smaller that Tp but is inside the area of a core point then the point is border point. In any other case the point is considered to be a noise point. In the evaluation of the points, the core points are the centre of the clusters, the border points are merged with the cluster and the noise points are deleted

1.2 Anomaly Detection

Anomaly detection is the last category of the Data Mining separation as adapted according to the perspective of [22] and the main subject of this work. The aim of anomaly detection is to find objects that are deviating from the majority of the objects. In more detail, anomalies are objects that do not fit to the rest of the data model or do not belong to any class or cluster. This technique is proven to be really useful in fields like credit card fraud detection, where the customers have a specific model of purchasing and fraud is detected by purchases that are outside this model, network intrusion detection, where network intrusions are detected by values that do not fit the normal network function (rapid increase of traffic, numerous remote login failures etc).

Anomalies or outliers or deviations might rise from several different reasons. The most common anomaly is data collection errors or noise. Errors during measurements or data collection might result in the appearance of anomalous values. This is a common reason for anomalies and it is reduced significantly with data cleaning as part of the data pre-processing of the KDD.

Another form of anomalies is the anomalies produced by normal data. These are data that have extreme values without being originally "anomalous" with the sense of being artificial. An example of this is if we take the network intrusion detection systems mentioned above, there might be moments where the network traffic would be really high due to a new software release where all the users would download at the same time. This is an anomalous occasion with the sense that it is not in the usual traffic values but this anomaly does not occur because of an intrusion in the network. This category of anomalies, the anomalies due to extreme values of normal data, is a common reason for "false alarms" and is an important issue that reduces the efficiency of anomaly detection algorithms significantly.

Finally, an anomaly might rise from data that are located outside the data model or belong to a different class. This is the kind of anomalies that anomaly detection is trying to reveal and they might be the purchase of a product with a stolen credit card number or a side effect of a proven medicine that only occurs to persons with a rare allergy.

A significant role in the efficiency of an anomaly detection machine (as in all the data mining techniques) is played by the training dataset. Anomaly detection techniques can be distinguished according the existence or not of a training dataset into:

- Supervised anomaly detection, where there exists a training dataset with the normal and the anomalous data being separated into classes.

- Semi-supervised anomaly detection, where there is a training dataset that separates the normal data into different classes but does not include classes for the anomalies
- Unsupervised anomaly detection, where there is no existence of a training dataset or information about the anomalies.

There exist many theories that apply anomaly detection. Different theories serve better for different natures of data and anomalies. However, in all theories the choice of the correct values for the parameters is important. The different algorithms include parameters that when set correctly according to the data, they reduce the errors that appear either by classifying as anomalous data, data that are normal or by not detecting data that are in a small percentage anomalous. Moreover, the decision of whether an object is anomalous or not is made by a yes/no weight in some algorithms. This does not reflect reality however, as in real data there are different levels of anomalies and an object can be more anomalous than another.

1.2.1 Uses of Anomaly Detection

Today, anomaly detection spreads in a vast collection of different fields and sciences and plays a rather significant role in the improvement of these fields. Some examples of these fields are:

- **Banking Systems' Fraud Detection**. With the wide use of the e-commerce and web banking or even mobile banking (the management of a banking account through text messages) the risk of fraud increase daily. Anomaly detection is used in this field to detect fraud that can be of the form of stolen credit card purchases or hacked web banking accounts. In all of the cases the transaction activities are examined to discover transactions that deviate from the normal transaction use.
- **Network Intrusion Detection**. This is one of the most common fields for anomaly detection with a lot of work related to it. The intrusion in a computer network can cause big damages if not detected and stopped on time. There are different ways of network intrusions. Some intrusions can be detected easier because they aim in the temporal or permanent network destruction like Denial of Service attacks. Others are more difficult to be detected because they aim in the silent existence and quiet collection of information. For some of these ways, anomaly detection and the close monitoring of the networks can be the only way of detection.
- **Healthcare and Medicine.** Anomaly detection is used for the improvement of medicines with the examination of anomalous side effects during the testing of a new medication or even after the medication is promoted to the market with the evaluation of the medical record data. Additionally, anomaly detection is used to make the healthcare services more accurate (e.g. by separating the pixels on a mammogram as carcinogenetic or not).
- **Customer Relation for Marketing Reasons**. Many companies that are dealing with commerce or services and are directed dependent from the customer are facing the need to insure the customers' "loyalty" with the meaning that their customers would not choose their competitors over them. The use of anomaly detection can make it possible to predict whether a customer is about leave and therefore alarm for appropriate actions. To have an example, if in the case of a airline company a customer is travelling at least once every month, the customer might have been going

for a different company if he has not travelled for more than three months. This brings in front the need to make appealing offers to regain the specific customer.

- **Noise or Outlier Removal.** Of course the list of uses of anomaly detection would be incomplete without this field. This is the reason that anomaly detection was created initially and noise removal is considered as a big part of the Data Cleaning process of the KDD. In this field the detection of anomalies is used to make sure that the data is clean from values that are taking extreme values due to errors in the data collection or noise.

1.2.2 Approaches on Related Work

When intending to define the outliers of a dataset, there is a variety of different techniques that can do that. As the field of anomaly detection exists for a fair amount of time, there is significant work made towards the effective detection of anomalies. In the following paragraphs different solutions of the problem of anomaly detection are explained in brief. The solutions are organized in different approaches according to the way that they detect the anomalies. These approaches are based on the separation of the anomaly detection techniques made in [22], while the first two approaches: the statistical approach and the distance-based approached are also found in [13]. In general, the specific separation or similar distinction of the approaches is supported by many researchers [4], [5].

Statistical Approaches

In the statistical approaches the anomalies are detected with the use of statistics. The creation of a model makes the separation of the normal data that fit the model and the anomalies that decline from model. The majority of the statistical anomaly detection methods build a probability distribution data model and evaluate the probability of each data object. Consequently, the objects with low probability are anomalies.

Univariate Normal Distribution

In the family of continuous distribution, the normal or Gaussian distribution is characterised by two values: the mean or average μ and the standard deviation σ so a distribution is of the form $N(\mu, \sigma)$. The standard normal distribution is the distribution that has a mean of zero and a standard deviation of one $N(0,1)$.

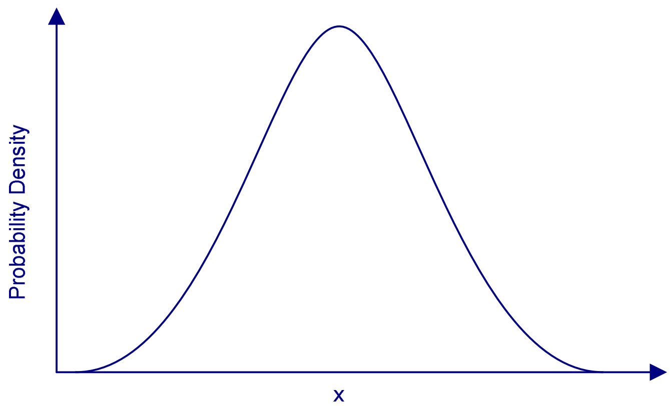

Figure 3. Probability Density function for normal distribution

A univariate distribution is a distribution that has one attribute per data object. That means that there is only one variable *x* that changes. If we consider as *c* a constant, then the probability of $|x| \geq c$ is reversely proportionate to the value of c. If there exists an α that

$$\alpha = P(|x| \geq c) ,$$

where $P(|x| \geq c)$ is the probability of the value of x being greater than or equal to c. In the standard normal distribution the probability that an object is on the tails of the distribution is really low. To give an example α is 0,3173 for c = 1, while for a c= 4, α = 0,0001. The variable α defines how rare the data object is

Although the univariate distribution seems simple and practical is not very applied because of the variable restriction. In real life, the most of the cases have more than one variable that change, therefore the univariate normal distribution cannot offer a solution.

Multivariate normal distributions

In the multivariate normal distributions, a similar approach with the univariate approach is used. The difference, however, is that in order to test the correctness of the results,the distance from the centre is not a reliable measure since there are more than one attributes for each data object. Instead the Mahalanobis distance is used between the point x and the mean point of the data objects ´x that is:

$$\text{Mahalanobis}(x,´x) = (x -´x)S^{-1}(x -´x)^{T}$$

Where S is the covariance matrix

Mixture Model for Anomaly Detection

This model applies anomaly detection by assuming that the probability of an object comes from different probability distributions. In this case, there are two different distributions, one for the normal data and another one for the anomalies. Supposing that we have a distribution for the normal data M, a distribution for the anomalies A and a dataset D that contains the mixture of the other two, the data probability distribution would be:

$$D(x) = (1 - \lambda)M(x) + \lambda A(x)$$

where λ is the anomaly parameter with values from 0 to 1.

This anomaly detection algorithm starts by considering that all the data are initially normal. Therefore, if M_t and A_t are the probability distributions for normal data and anomalies for time *t*, we would have $M_0 = D$. Iteratively, each data object is moved from M to A to create the M_{t+1} and A_{t+1} datasets. After the new datasets have been created the likelihood and log likelihood are calculated according to the following equations:

$$L_t(D) = \prod P_D(x_i) = \left((1-\lambda)^{|M_t|} \prod_{x_i \in M_t} P_{M_t} \right) \left(\lambda^{|At|} \prod_{x_i \in A_t} P_{A_t}(x_i) \right)$$

$$LL_t(D) = |M_t| \log(1-\lambda) + \sum_{x_i \in M_t} \log P_{M_t}(x_i) + |A_t| \log \lambda + \sum_{x_i \in A_t} \log P_{A_t}(x_i)$$

with P_D, P_{Mt} and P_{At} the probability distributions for D, M_t and A_t.

By calculating the difference of the log likelihood before and after and comparing it with a threshold, the algorithm decides whether the data object that was moved is an anomaly or not. In other words if $\Delta = LL_t(D) - LL_{t-1}(D)$ and $\Delta > c$, where c is the threshold, then the point x is an anomalous data point.

Principal Component Analysis

The Principal Component Analysis (PCA) is another method used to detect anomalies. This method is not totally a statistical method as it originates mostly from mathematics. It has been applied in different fields of mathematics and statistics for more than 70 years. It is used to reduce the dimensionality of the data and reveal data patterns. PCA is based on the creation of a covariance matrix of the data and the calculation of the eigenvalues of this matrix. The M eigenvectors corresponding to the M largest eigenvalues of the covariance matrix, define a linear transformation from the N-dimensional space to an M-dimensional space where the features are uncorrelated. After the dimension transformation, (i.e. the mapping of the data onto new axes), patterns are revealed and anomalies are easier to detect. [12]

With the use of PCA, Lakhina et al. [16] are detecting anomalies in computer networks. In their method they are applying the PCA to transfer the data into a new set of axes, the principal components. After the dimension transformation, the projections are separated into two spaces, the normal and the anomalous. In order to classify the projections as normal or anomalous, they have created a method that is based on a threshold value. If the values overcome the threshold then the projection belongs to the anomalous space.

In a similar approach Ali et al. [3], are using the PCA also for detecting network anomalies. In their approach they are collecting the data from the network, converting them to zero-mean data, creating the covariance matrix, retrieving the eigenvectors and eigenvalues with the use of and open source Java library “Colt” and finally creating the matrix of *final data.*This matrix is consisted from the data items as columns and the dimensions as rows that reveals the pattern between them. By examining this matrix, anomalies can be detected in terms of data objects that escape from the patterns.

Distance-Based Anomaly Detection

If an object is an anomaly, then it will be further in distance than the rest of the objects. This is the main idea of the distance-based or proximity-based anomaly detection approach, as implied by the name. A good technique to detect anomalies with this approach is the nearest neighbour technique used for classification.

The nearest neighbour technique for classification classifies a data object according to the majority of the class that the k closest data objects around it belong. So if an object is in the middle of a group of objects that belong to the same group, then this object will also belong to the same group. With the number of nearest neighbours k, being a user defined parameter.

In the same concept, the nearest neighbour technique for anomaly detection defines anomalies as the data objects that have the longest distance from their k closest data objects. In other words, the data objects that are more distant from the rest.

An important issue is the correct value for the parameter k, as in the nearest neighbour for classification. A too small value for k will lead into miss-classifying normal data as anomalies, while a too big value for k will lead into miss-classifying anomalies as normal data. This technique, is performing adequately well in a relative small dataset but as the dataset increases, the algorithm spends more and more resources in calculating the anomalies.

Additionally, the nearest neighbour for anomaly detection fails to find anomalies correctly in datasets where there are groups of data objects with different density. This is because in a dense-allocated group an anomaly might be of the same distance of a normal data object in a sparse-allocated group.

The nearest neighbour method for distance-based anomaly detection is used by Angiulli et al. [4]. In this approach the method of the nearest neighbour for anomaly detection is applied to create a weight *w* of the data objects according to the distance from the k nearest neighbours. In continuation, a Solving Set *S* is created as a learned model containing the data objects with higher score (i.e. the most anomalous data objects). The solving set S is used for the prediction of anomalous objects.

In a similar approach, Ren et al. [19] used a method quite similar to the nearest neighbour to create *Neighborhoods* of data objects. A neighbourhood of a data object is the area around the object in a radius r. The difference is that this method can characterize a whole neighbourhood as anomalous if a data object is anomalous within the neighbourhood. In this approach, the P-Tree, a method of calculating the distances of the different parameters of the data objects is used. This method creates a tree structure of the binary values of the parameters (attributes) of the data objects in order to be processed faster.

Density-Based Anomaly Detection

In a quite similar perspective, anomalies are the data objects that are sparse-situated in concern to the rest of the objects. In this sense, the density of an object with its surrounding objects is the inverse indication of the anomaly level of that object. There are several methods that use this approach.

One technique is to calculate the anomaly level by calculating the average distance of the k nearest neighbours, where the smaller the distance of an object from its nearest neighbours, the higher the density and therefore the smaller the anomaly.

Another technique is measuring the density with the density-based clustering algorithm DBSCAN. As mentioned in the clustering analysis section, this algorithm takes an object as the centre and checks the density of this object by counting the objects that are located inside a radium r. Depending on the number of objects d located inside the radium, the level of anomaly of the object under estimation is calculated.

As it is only natural, in both algorithms the level of accuracy relies on a big percentage on the values chosen for the user specified parameters, the k nearest neighbours and the length of the radium d. However, both of these methods have the same drawbacks with the distance-based approach. Since they only measure the local density, they are not detecting anomalies effectively for datasets that have groups of deferent densities. A good approach that addresses this issue is the approach of the Local Outlier Factor (LOF) technique by Breunig et al. [5]. In this technique the level of anomaly of an object (here called Local Outlier Factor) is calculated as a proportion of the density of each of the minimum nearest neighbour points MinPts with their nearest neighbour points around a k distance. In this sense, as shown in Figure 4, an object O_1 that is located in the core of a sparse cluster can be characterised as non anomalous (low LOF) while an object O_2 that is outside a dense cluster can be characterised as anomalous (higher LOF) although O_2 might have more objects in a smaller distance that O_1. This occurs because although O_2 has more objects in a smaller distance, thus higher density than O_1, the density of the objects close to O_2 with the objects around them is really higher therefore giving a high LOF for O_2. In Figure 4, a normal density-based or distance based algorithm would give classify only to the object O_3 as anomaly and consider O_2 as a cluster object.

In the same approach, Lazarevic et al. [17] is enhancing the Local Outlier Factor method with a method called Feature Bagging for Outlier Detection. This method applies the LOF method many times for different features (attributes) or feature sets of the dataset objects. After collecting the LOF for each attribute it combines the factors to end up with the final anomaly weight for each object. The factor combining method for different attributes of the data objects is similar to the page ranking of the web search engines.

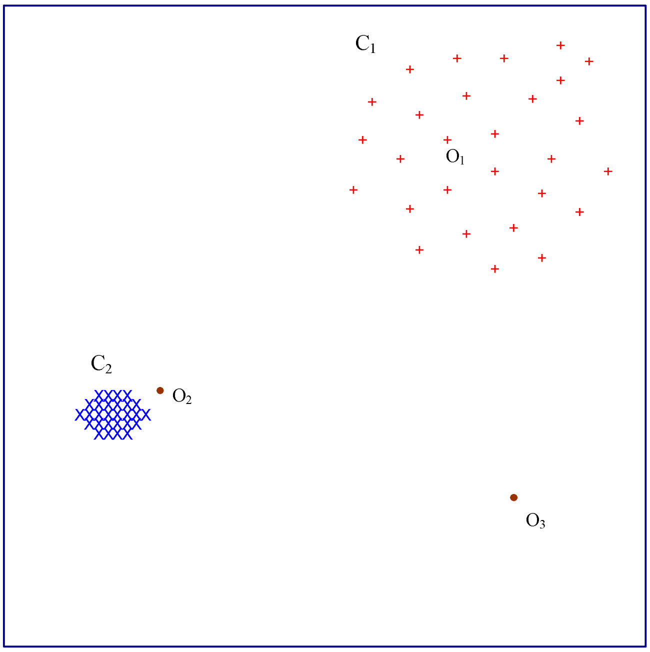

Figure 4. Clusters with different density

Clustering-Based Anomaly Detection

In the previous section, when talking about clustering algorithms, we saw that DBSCAN algorithm classifies the objects in three categories: the core points, the border points and the noise points. Although in cluster analysis the noise points are consider noise and discarded, in anomaly detection some of the noise points might be the anomalous points. In a more general description it would be right to say that in cluster based anomaly detection, the anomalous points are the points that do not belong to a specific cluster.

One of the most common ways to apply cluster-based anomaly detection is with the prototype-based clustering. In prototype-based clustering as previously explained in the K-means algorithm the centre of the cluster is calculated and the anomalous objects are detected by measuring their distance from the cluster centre (centroid in the case of K-means). An example of K-means anomaly detection is encountered [24] by Zanero et al. where they are detecting network intrusions with anomaly detection.

For the same purpose, the network intrusion detection, Leung et al. [18] are applying classification by first mining for frequent datasets. Therefore, they have created the fpMAFIA clustering algorithm that is based on the functionality of the Apriori and FP-Growth association analysis algorithms.

Khan et al. [14] in their proposal, which is also for detecting network intrusions, are applying anomaly detection with the use of the Support Vector Machine. In their suggestion, as the training of the SVM requires many resources, they are using the hierarchical clustering algorithm Dynamically Growing Self-Organization Tree (DGSOT) to detect the training points for the SVM machine.

Finally, in a slightly different approach, Abe et al. [1] are detecting anomalies with the help of a classification algorithm and the *ensemble-based minimum margin active learning*. In this method they are teaching a classification algorithm to detect anomalies from a dataset. The training dataset is a real data dataset with fake anomalies inserted manually.

The cluster-based anomaly detection methods, however, encounter the same issue with the Proximity-based approach. In the case that different clusters have different densities, an anomaly might be closer to the centre in a dense cluster than a normal object in a sparse cluster. This issue is addressed either by calculating the relative object distance that is the distance of the object from the cluster centre in respect to the mean distance of the all objects of the cluster or by the Mahalanobis distance. In the related work mentioned for this approach there is no specific reference to whether they address this issue.

Evolutionary Algorithms

All the above techniques have been noted to bring satisfying results in different natures of data. Since the majority of these techniques, however, are based on detecting outliers by measuring their distance or density from the objects around them, they are proven to perform poorly in very large datasets with many dimensions. That is because locality or neighbourhoods becomes more difficult to define in sparse data. This issue can be faced with the Evolutionary Algorithm approach.

The evolutionary algorithm idea is based on the Darwinian view of evolution. This view supports that since in nature the resources are limited, the different species have to compete for them. This leads to nature having a selection mechanism for the individuals of every species where the fittest survive. In continuation the fittest individuals of each species mate and create even fitter offsprings. With the same concept, in evolutionary algorithms, the solutions of a problem are the individuals that are characterised by a fitness score. The fittest individuals are then chosen through a selection mechanism to survive and produce offsprings (i.e. new solutions). The offsprings are produced by cloning the individuals and re-arranging their genes (i.e. crossover), by cloning the individuals and replace part of the genes with random genes (i.e. mutation) or by combining the genes of two or more individuals (i.e. recombination). [7]

When applying evolutionary algorithms for anomaly detection as in the method proposed by Aggrawal et al. [2] the anomalies are detected by studying the behaviour of projections of the dataset. An anomaly is the lower-dimensional projection that is locally sparse and therefore difficult to detect. The evolutionary algorithm is input with the dimensionality k of the projections and the number of projections m. In continuation, it starts with a number of random solutions and through the processes of selection, crossover and mutation that perform a combination of hill climbing, solution recombination and random search over the space of possible projections stores the anomalies until the termination condition is met.

1.3 Problem Definition

Anomaly detection has been existing for several decades but it was only until the last decade that it has been evolving with a fast pace. That is mostly due to the fact that before there was no or very small practical use of this field of data mining. Within the last years the fields where anomaly detection is used effectively are increasing. However the progress is many times towards specific directions. Today there is a fair amount of different solutions considering the time that it has been efficiently researched. The choice on these solutions is sometimes hard as there is no one solution that is better than the rest, but each solution is suitable under some specific conditions. This comes as a result to the fact that in anomaly detection each new solution implemented is more of a solution tied to the nature of the data than a global solution that would cover a great variance of cases of anomaly detection. Therefore, when choosing an anomaly detection method to adapt to a new problem it is important to look on the particularities of the specific dataset that the method will be applied. Particularities might be the number of parameters of the data objects, the total size of the dataset, how are the data organised (i.e. if they can are grouped in clusters, if they have a specific pattern, if they repeat themselves) and so forth.

This report is the documentation of the effort to encounter the most optimal solution for anomaly detection on online transactions. In other words, to answer the research question:

“ what is the most efficient method for online gambling outlier detection?”

The specific thesis work is conducted on behalf of a Swedish company that is engaged on data mining applications. The main effort of the work is to detect any kind of fraud, misuse or user addiction on the activity held in several web sites that provide online gambling. As a product of this work a new method adapted to the specific needs will be developed and tested. The efficiency of the method will be tested by comparing the results that it brings with the results of the anomaly detection method applied currently. If the results prove to be satisfactory, the substitution of the current method in the company by this method will be examined.

2 Approaching the Problem

In order to provide an answer to the research question, an approach was used to address the problem. This approach is consisted of several steps, which have been modified from the initial planning during the process of this work. The steps have been changing within the work progress to adapt to the new demands in order to address the problem in a more effective way. In brief the steps that were followed can be explained as:

Literature Research

As a beginning, a deep insight on the anomaly detection issue and in data mining methods in general was necessary so there would be an adequate knowledge background. This was obtained through the published literature and online resources that are focusing on these fields. The greatest part of this knowledge background is displayed in the introductory chapter of this report. After a sufficient level of the domain was obtained, a clear view of the advance on this field was necessary. Related work, methods that have been developed for anomaly detection for different domains were studied. The source of information at this stage was mainly published articles on conferences like *The International Conference on Knowledge Discovery on Databases* or the *ACM international conference on Information and knowledge management* and books on anomaly detection and data mining. Most of the methods studied are analysed in the “Approaches on Related Work” section, there have also been several methods that are studied and have not been analysed either because their approach is analysed by a newer method or because they were considered “obsolete”.

Study of the Data to be used

During the study of the related work, it was made clear that there are several approaches that promise effective results but are bound by the characteristics of the data that they are created or tested. In other words, many of the algorithms would bring satisfying results if applied to data with the same characteristics as the data that they are created for but it is not certain that they would be as effective if they would be applied in data of different nature. Consequently, studying the nature of the data was essential for the selection of the proper anomaly detection method.

Choice of the anomaly detection method

Having defined how the expected data would be, the choice of the proper algorithm was an achievable task. The majority of the methods developed are dealing with network intrusion detection in order to raise intrusion alarms and take actions against these intrusions. The datasets, in most of the cases are datasets of the network traffic that do not include many parameters, or at least not as many as the dataset to be used for this work. Therefore, choosing a method that is proven to work for network intrusion detection is not necessary that it will be effective for this work’s task. It should be a method that can effectively handle the variety of a larger number of parameters and would allow the number of parameters to increase.

Method Adoption and Implementation

After settling on the anomaly detection method, the method was modified to depict the specialities of the specific work and then implemented.

Applying the Method

With the termination of the method implementation, several datasets were applied to the method to extract the anomalies. The datasets were of both of the time vertical and user based data groups (see paragraph 2.1).

Method Evaluation

In order to evaluate the method created, it would be compared with the method currently applied for the same task. Therefore the datasets were applied on both methods and the results where compared. According to the similarity or difference of the results, the method would be evaluated. An important issue would be in the case of different results. In this case, manual evaluation of the results would be necessary to decide on whether the anomalies were real anomalies or false alarms.

Method Optimisation

After the completion of the testing and comparing of the method a good feedback for the method's accuracy was created. Using this feedback helped to "fine tune" user parameters so the method would work more effectively and reduce the false alarm rate.

2.1 The Nature of the Data

During the study of the data to be mined, several interesting points came up that prove to be determining on the decision of the chosen method to be implemented. Before examining the points, it would be wise to elaborate on the data and their source. As mentioned previously, the data derive from online gambling activity. The source of this data is several websites that include many transactions by many users everyday. This can make them easily a target of fraudulent activity. The aim of the data mining task is to detect any fraudulent activity as much as any addictive playing.

More specifically, the dataset includes many active users registered that are making a lot of transactions within a short time. These two points result in the size of the dataset being prohibiting large for many anomaly detection algorithms. Moreover, since the websites have such a high traffic rate with active money transactions, they are becoming a target of fraudulent activity easier. Therefore, apart from high requirements in speed and the ability to manipulate big amounts of data, there are also requirements in the effectiveness of the method. In other words, the method should not be very fault tolerant.

One additional issue is that the dataset should also measure the time that the user is spending on the website. This can prove to be useful in both the fraud detection and addictive playing detection.

The anomaly detection analysis would be done in two different axes. Firstly *time vertical*, where all the transactions of all the users within a defined amount of time would be analysed for outlying transactions. Secondly *user based*, where each user would be analysed for changes in the transaction patters or for extremely strange values. Consequently, if a user is executing transactions based upon a specific pattern and suddenly this pattern changes, depending on the differentiation of the pattern and the weight of the pattern, this might be alarming.

Another restricting fact is that the datasets to be used include a big number of variables. This occurs because the amount of the information that needs to be controlled to uncover fraudulent activity is large. Moreover, since there is constantly new fraud methods created, the field of fraud detection is a field that is constantly expanding to be up to date. Therefore, the number of parameters included in the dataset might increase to include detection for different fraud methods. A fact that results into choosing an algorithm that should perform well under multidimensional datasets.

For this reason, the constant update of fraud detection methods, the use of a training set is not advisable. Using an algorithm which requires training, would mean that the training set would have to include all kinds of fraudulent anomaly detection types. An algorithm like this might not perform in anomalies deriving from a fraudulent activity that was not included in the initial training set.

2.2 The Anomaly Detection Method

Choosing the proper method

In the section Approaches on Related Work of the previous chapter, several definitions for the anomaly or outlier in a dataset were given. The most representative definition would be that an anomaly is the data object that declines or that is “blatantly different” [12] from the rest of the objects. In order to choose a proper method to apply, the nature of the data as much as the expected result of the data mining task should be taken into consideration.

As explained in the previous paragraph, the data to be mined have a big number of parameters and this number might change in the future. A fact that by definition gives a great disadvantage to most of the distance and density based methods since the demand in computation resources increases vertically in a large database (either in terms of width i.e. many parameters or in terms of length i.e. many records). This occurs because if we take the example of distance based methods, the distance between two data objects should be measured as the distance between all the parameters of these two objects. Suddenly, with the increase of the parameters the computational resources increase exponentially. Therefore the methods proposed by Anguilli et al. [4] and Ren et al. [19] were not suitable for the specific work.

Additionally, in the previous section it was explained that the specific data mining task is not advisable to include a training dataset. Taking this as a fact, the methods proposed by Abe et al. [1], Aggarwal et al. [2], Khan et al.[14] and Ren et al.[19] are not possible to be implemented for this dataset. The Evolutionary Algorithm method [2] however, appears to be a quite promising approach with positive results in the issue of large databases. The use of this method in the specific work is believed that it would not bring as satisfying results as the method finally applied.

The method of Zanero et al. [24] was considered as inappropriate because the K-means algorithm has the restriction that the clusters should be round-shaped while the similar approach of the K-metroids has big computation cost that would prove to be extravagant in the width and length of the dataset. The method of Leung et al. [18] was excluded for a similar reason, the reason of great resource requirements, as the preprocessing to encounter the most frequent dataset would require a lot of resources.

Finally, the methods of Breuning et al. [5] and the evolution of this by Lazarevich et al. [17] were considered probable solutions but the data processing with the Principal Component Analysis was believed to capture the variability of the data in a more effective way and therefore detect anomalies more efficiently. For this reason, the method adapted for the specific work is an approach quite similar to the method proposed by Lakhina et al. [16] to detect network-wide traffic intrusion.

The Anomaly Detection Method

The anomaly detection method chosen to be evaluated is based on the Principal Component Analysis (PCA). Anomalies are detected with PCA using the Subspace approach and the Q-statistic method. To be more concrete, the PCA is transforming the data into new compressed axes while keeping their variability. In continuation, the data are separated in to two subspaces normal and anomalous and from the anomalous subspace the anomalies are detected by defining an anomaly threshold with the Q-static method.

Principal Component Analysis

As explained briefly in the previous chapter, the Principal Component Analysis is a technique used to obtain data compression without losing the useful information that can be extracted by the data concerning their interrelation. It is a multivariate technique that dates back to 1901 where Karl Pearson captured the primary form of the Principal Component Analysis. But it was only until 1933 where Harold Hotelling [9] published a paper explaining the PCA (or alternatively the Hotelling transform) with the form that is known today.

PCA achieves coordinate transformation, where it maps a dataset consisting of interrelated variables into new axes, the Principal Components (PCs). When the data are zero-mean, each axis (i.e. each PC) points towards the maximum variance remaining in the data. The PCs are ordered in such a way that the first PCs contain most of the variation of the original data. In other words, they show how the variables are correlated and reveal patterns in the data.

To get a more clear idea, let us suppose a dataset is consisted of p data objects (rows) and only k parameters (columns). The parameter x of this dataset would be a vector of p variables. PCA is trying to discover a linear function in the data that would be of the form α_1'x with α being the vector of p constants α_{11}, α_{12}, α_{13}, ... , α_{1p} and α' the transpose of α. The linear function should verify the equation:

$$\alpha_1 \mathrm{x} = \alpha_{11}\,\mathrm{x}_1 + \alpha_{12}\,\mathrm{x}_2 + \alpha_{13}\,\mathrm{x}_3 + \ldots + \alpha_{1\mathrm{p}}\mathrm{x}_\mathrm{p} = \sum_{i=1}^{p} \alpha_{1i} x_i$$

In continuation PCA is looking for a linear function α_2'x with maximum variance that is not correlated with α_1'x. After the discovery of the linear function α_n'x with $n \leq p$ that captures the greatest fraction of the data variability, the first n PCs would have been discovered. In general PCA is trying to capture most of the variance of the data in n PCs where $n << p$.

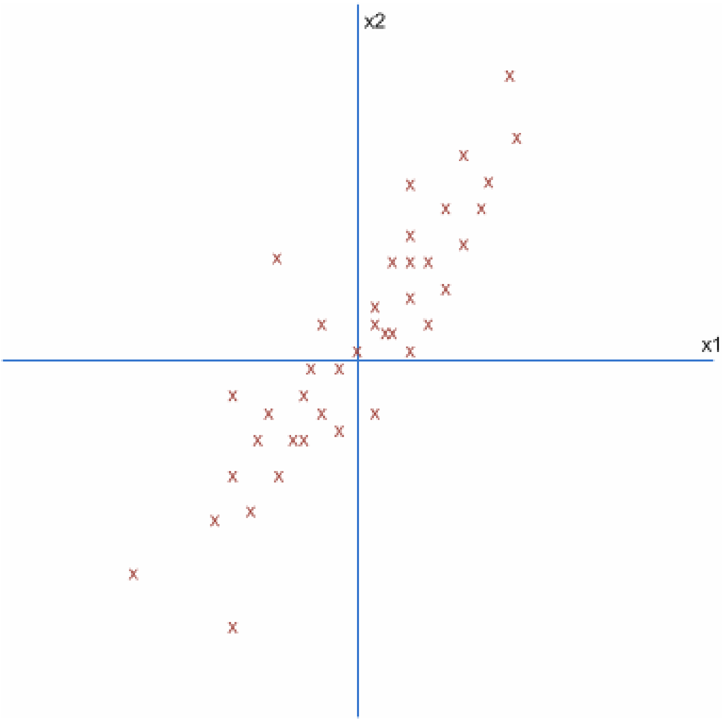

Figure 5. Dataset of 40 objects

To give an example, let us suppose that we have a dataset consisted of two parameters x1 and x2 and plotted as shown in Figure 5. We can see from the plot that there is a big variability in the values more in the direction of x_2 and less in the direction of x_1. After applying PCA in the dataset and transforming the data from x_1, x_2 to z_1, z_2, we would get a plot that would look more like Figure 6. We can note that z_1 the first PC is capturing a great percentage of the variability while z_2, the second, less. It is generally the case in PCA that the first PCs are capturing the greatest percentage of the variability of the dataset.

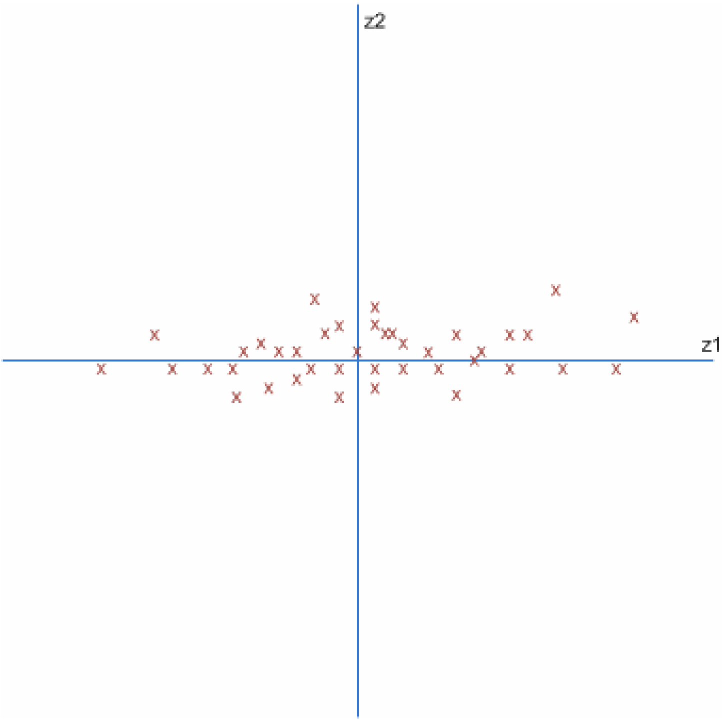

Figure 6. 40 point Dataset after PCA

When we apply PCA into a dataset X with zero empirical mean, we would get a set of *m* principal components. In this set the fist PC would be a vector pointing towards the maximum variance, the second would capture a smaller percentage of the variance that was not captured by the first, the third a percentage that was not captured by the previous two and so on. In general, the first principal component z_1 can be expressed:

$$z_1 = \arg\max_{\|z\|=1} \operatorname{var}\{z^T X\} \Leftrightarrow z_1 = \arg\max_{\|z\|=1} \|Xz\|$$

where arg man signifies the maximum value of the argument.

The kth PC can be found subtracting the first k-1 PCs from X:

$$z_k = \arg\max_{\|z\|=1} \left\| \left(X - \sum_{i=1}^{k-1} X z_i z_i^T \right) z \right\|$$

When having found the PCs we normalise the data and project them into the new set of axes, the PCs. After the data are projected, we can see how much of the total variance is captured by the first PCs in a plot like the plot of Figure 7. Hopefully the greatest percentage of the data variability is captured in the first k PCs where k << m and therefore obtain and adequate level of data dimensionality reduction. In the specific figure, which is the plot of the variability of the PCA analysis of dataset 4 used for the evaluation of the algorithm (see section 2.3.1), it can be noted that more than 99% of the data variability is captured by the first three PCs.

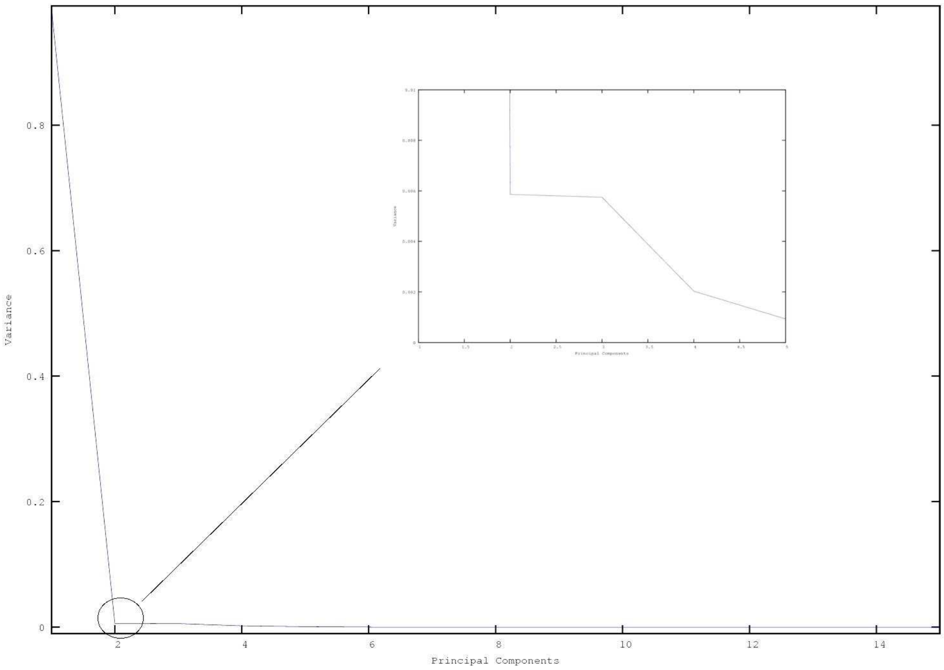

Figure 7. PCA variances of a 17 PC plot

Subspace Approach

After the data have been mapped onto a new set of axes, the Subspace approach is applied to separate the data into two different subspaces. This is an approach, proposed by Dunia et al. [8] for multi-dimensional fault diagnosis while Lakhina et al. [16] adapted it to detect anomalies into network traffic. For the demands of this work, this method is adapted and modified slightly to fit the distinctiveness of the data.

According to this approach, since anomalies are the data values that are really different from the rest of the data, they would be ordered in the last PCs. Therefore, the remapped data can be separated into two subspaces: the *normal S* and the *anomalous Š*. In the process of creating the subspaces, the data are normalised and divided into vectors according to the axis they are mapped. With this sense, the vector u_1 is consisted of the normalised data mapped to the first PC and encapsulates the biggest fraction of variability, *u2* of the data mapped to the second PC and so on. These vectors, in continuation will be classified as either in the normal or in the anomalous space by locating the subspace separation vector where before this vector the normal subspace exists, while from this vector and afterwards there is the anomalous space. This vector is encountered by comparing all the values of the vectors with a specific threshold. If a value is turned out to be greater than this threshold, then the vector that this value belongs to and all the vectors after that are the anomalous space Š. The value of the threshold has been designed as a user defined variable in this work and it derives from the multiplication of the standard deviation of the vector with a number. The most tests in the following section are run with the threshold α being $\alpha = 3*std(u)$.

In this point, let us consider that we have a vector x that was decomposed from the PCA. This vector is consisted of two portions: the vector $\hat{x}$ that is the modelled part and the vector $\tilde{x}$ that is the residual part of x. So x can be expressed in the following form:

$$x = \hat{x} + \tilde{x}$$

To find the modelled part of x, $\hat{x}$ all we have to do is project x onto S while to find the residual part of x, $\tilde{x}$ we project x on Š. To achieve that, we first create a matrix P that is of size m * r, where m is the total number of PCs and r is the number of axes in S, the normal subspace. Therefore, P is the matrix of the eigenvectors resulted from PCA that belong to the normal space or in other words the PCs (z_1, z_2, .., zj) that belong to S. Consequently, the modelled part of x is:

$$\hat{x} = PP^Tx = Cx$$

while the residual part of x is:

$$\tilde{x} = (I - PP^T)x = \tilde{C}x$$

where $C = PP^T$ represents the projection matrix on the modelled subspace and $\tilde{C}$ the projection matrix on the residual subspace.

Q-Statistic

In the case of an anomaly, x increases its projection to the residual space. This results in $\tilde{x}$ taking large values. Monitoring the values of $\tilde{x}$ facilitates the anomaly detection. The method used in this case to uncover the abnormal conditions is the common statistical method of the squared prediction error (SPE). The squared prediction can be calculated as:

$$SPE = \left\| {}^{\sim} x \right\|^2 = \left\| {}^{\sim} C x \right\|^2$$

The data values are considered to be normal in the case of:

$$SPE \leq Q_\alpha$$

where Q_α is the Q-Statistic value. In this method, Jackson [11] calculates the upper limit Q_α, which the sum of squares of the residuals (i.e. SPE) can reach to be considered as normal values. To calculate the Q_α let us consider:

$$\theta_1 = \sum_{i=r+1}^{p} l_i$$

$$\theta_2 = \sum_{i=r+1}^{p} l_i^2$$

$$\theta_3 = \sum_{i=r+1}^{p} l_i^3$$

and

$$h_o = 1 - \frac{2\theta_1\theta_3}{3\theta_2^1}$$

where *l* is the characteristic root (i.e. the number of PCs resulted from the PCA) and *r* is the number of axes in S. Having the previous the upper limit Q_α would be:

$$Q_\alpha = \theta_1 \left[\frac{c_\alpha \sqrt{2\theta_2 h_o^2}}{\theta_1} + \frac{\theta_2 h_o (h_o - 1)}{\theta_1^2} + 1 \right]^{\frac{1}{h_o}}$$

where c_α is the $1 - \alpha$ percentile in a standard normal distribution that depicts whether or not all the significant components are used or alternatively, a parameter that adjusts the false alarm rate of α.

2.3 Applying the Data – Evaluating the Method

After the Method Adoption and Implementation, a working proof of the algorithm was ready to be provided with data and be evaluated. As already explained in the section concerned with the nature of data, there are two kinds of data analyses: the Time-Vertical and the User-based. Datasets of these two methods were scanned for outliers with the algorithm. This section displays the results of the analyses in these two kinds of datasets and evaluates the efficiency of the algorithm. The evaluation is done by comparing the results of the time-vertical analysis with the results of the company's algorithm and by evaluating the outliers encountered in the User-based analysis.

2.3.1 Time-Vertical Analysis

In the time-vertical analysis, the datasets capture the activity of all the users within a specific amount of time. The analysis is made to find activity that declines from the average activity of the rest of the users in the specific time. This analysis results in detecting suspicious behavior, which would result in fraudulent behavior or addictive gaming.

Analyzing the activity of all the users for a small amount of time, however, includes a big false alarm rate. This occurs, because the anomalies that are to be encountered are not necessarily fraudulent activity. To elaborate on this, let us take the example of a user that is making many more transactions than the rest of the users. In an analysis of this sort, his transaction might appear as an anomaly because he is deviating from the mean transaction rate. This, however, does not make him a fraudulent user as his activity has always been moving in the same rate. This phenomenon of the false alarms (i.e. anomalies that derive from other reasons than the intended detection target) has been reduced with the introduction of parameters that show the activity of the user in the past. Therefore the PCA takes into consideration also how the specific user has been changing his/her activity in specific amounts of time in the past.

For the needs of this work four datasets were extracted. Each dataset contained information of the activity of one week. The datasets included Delta values of the current activity compared with one (Δ_1), five (Δ_5) and ten (Δ_{10}) weeks before the date of the dataset. The datasets were scanned with the algorithm and the resulted anomalies were compared with the set of anomalies detected by the company's algorithm. For each dataset five different detections were made for five different values of the parameter c_α. The parameter would take values c_α = 1,645, c_α = 1,5, c_α = 1 , c_α = 0,5 and c_α = 0 for no false alarm rate. The results of the detection methods of the two algorithms, summarized in numbers, can be seen in Table 1.

```
Dataset 1

Anomalous Space Starts in Principle Component: 1
```

ca	Anomalies from Algorithm	Anomalies from Company's Algorithm	Common Anomalies	% Coverage of the Common Anomalies
1,645	88	258	top 88	100
1,50	94	258	top 94	100

1,00	116	258	top 116	100
0,50	151	258	top 258	100
0,00	220	258	top 220	100
Dataset 2				
Anomalous Space Starts in Principle Component: 1				
1,645	40	234	top 40	100
1,50	42	234	top 42	100
1,00	56	234	top 56	100
0,50	80	234	top 80	100
0,00	116	234	top 116	100
Dataset 3				
Anomalous Space Starts in Principle Component: 1				
1,645	39	241	top 39	100
1,50	42	241	top 42	100
1,00	54	241	top 54	100
0,50	80	241	top 80	100
0,00	116	241	top 241	100
Dataset 4				
Anomalous Space Starts in Principle Component: 1				
1,645	31	157	top 31	100
1,50	32	157	top 32	100
1,00	42	157	top 42	100
0,50	52	157	top 51	98,08
0,00	73	157	72	98,63

Table 1. The results of the time-vertical analysis

As it can be noted from the results, the algorithm did not made use of the subspace separation for none of the datasets applied in this analysis. This means that the algorithm was working as a normal PCA algorithm where all the data projected to the principal components were considered in the anomalous space. This occurred because there were encountered values that overcame the threshold value from the first principle component. This can be seen more clearly in Figure 8, where the PC variance of each dataset is plotted in a chart. As it can be seen, for all the datasets the principal component analysis captures more than 99% of the data variability in the first 2 principal components. Therefore, the probability of an anomaly existing in the variability captured by the first PC is really high. Consequently, the use of the Q-Statistic method was applied to the whole PCA dataset to detect anomalies.

Additionally, it can also be noted that almost all the anomalies detected by the method were detected by the company's algorithm. There is an exception of a data object in the dataset 4 that was detected by the algorithm but not by the company's algorithm. This might be caused by the fact that the company's algorithm is based on the detection of anomalies from their graphical representation or might just be a data fault. The fact that the common anomalies are the top anomalies in the company's detection list means that the anomalies detected in both algorithms are the most anomalous cases as both algorithms sort the anomalies by weight. The difference is that the company's method does not include some sort of false alarm rate variable. This results in the method extracting all the anomalies that is encountering. Therefore, many anomalies that are not originated from fraudulent activity are included in the anomaly list. The oversensitivity of the algorithm might become an issue in case of larger datasets with many more anomalies.

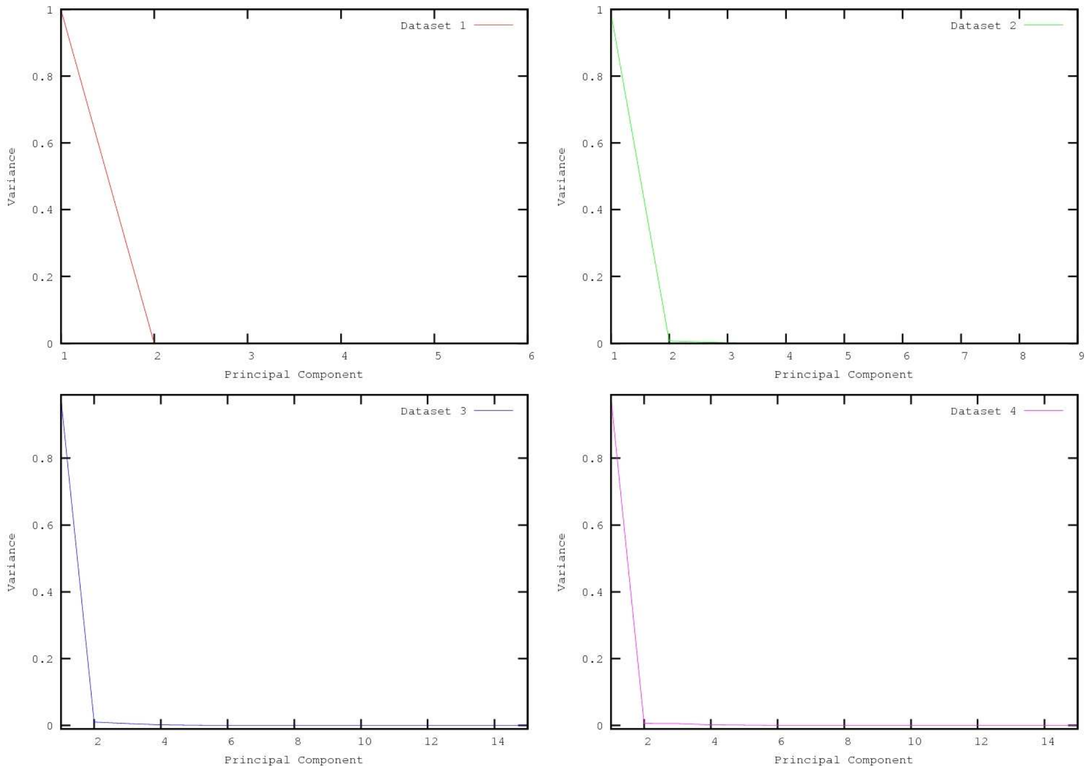

Figure 8. The PC variances of the datasets used

2.3.2 User-based Analysis

In this analysis the activity of each user is scanned for any anomalous behavior within time. The data of the users' activity within the time of one year were accumulated and extracted. Because of the great number of users within the databases, only the users that appeared to have anomalous behavior on the time-vertical analysis were selected to be mined for anomalous behavior. In more detail, the users from the previous four datasets that were detected to have anomalous behavior with ca = 1were analysed for changes within the patterns of their activity. A total of 268 user data with a 78 parameter dataset each and recorded activity of each week were scanned evaluating which principle component the anomalous space was beginning, how many anomalies were there detected and the dates of the detected anomalies. Afterwards, the charts of the most anomalous cases were drawn to visualize the anomalous activity. A list of some of the users and the information that were extracted can be seen on Table 2. In this list for each user is displayed the principal component that the anomalous subspace is starting and the number of anomalies detected. For privacy reasons the user ids are replaced by the string “XXX”.

User	AnomalousPC	No of Anomalies
XXXX	4	1
XXXX	28	3
XXXX	22	1
XXXX	15	0
XXXX	57	2
XXXX	18	0
XXXX	19	0
XXXX	19	0
XXXX	5	0
XXXX	17	0
XXXX	16	1
XXXX	40	0
XXXX	4	0
XXXX	5	1
XXXX	4	2
XXXX	29	1
XXXX	11	0
XXXX	26	0
XXXX	15	0
XXXX	34	3
XXXX	29	0
XXXX	9	0
XXXX	40	2
XXXX	12	0
XXXX	25	0
XXXX	36	0
XXXX	16	0
XXXX	13	2
XXXX	15	2
XXXX	18	2
XXXX	14	2
XXXX	13	1

XXXX	13	1
XXXX	38	2
XXXX	15	3
XXXX	29	3
XXXX	4	3
XXXX	11	2
XXXX	13	2
XXXX	14	2
XXXX	17	2
XXXX	12	2

Table 2. Results of User-based analysis on Dataset 4

One of the most anomalous users from the time-vertical analysis of dataset 4, user A, was analysed in the user-based analysis and has been found to have two anomalous instances. This user's activity has been plotted in the plot shown in Figure 9 with 3 graphs. Each of these graphs displays the user's transactions and timeslots of each week in respect to 1, 5 and 10 weeks before. The transaction field is the number of transactions the specific user made in the specified time, while the timeslots field shows the time that the user has been spending on the specific site. The anomalies detected on the activity of the specific user are located within the third and fourth week. Within this time, there appears a peak in activity in the plot of 1 week while the same peak appears more normalized and after 5 weeks in the second plot and even smoother and with a delay of 10 weeks in the third plot. This can be considered one of the typical behaviors of online gambling customers, where when introduced to the website, the user starts increasing the time that he/she spends on it until they reach a point after a specific time where they either loose interest or move to another site.

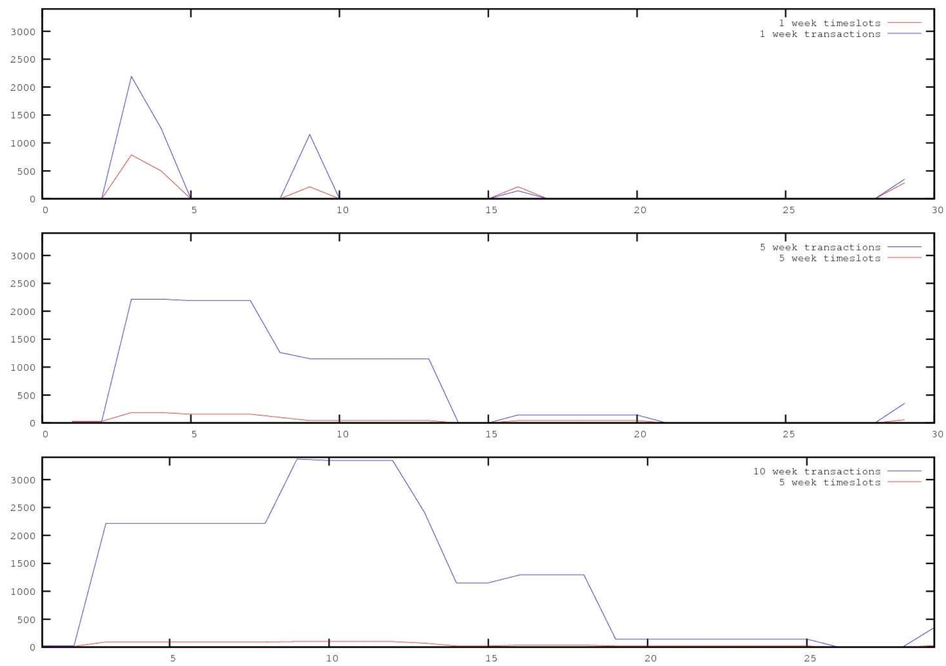

Figure 9. User A plot of transaction and timeslots

In the user B of Figure 10 the plot is some quite different. The user has been demonstrating an increased activity within weeks 13 to 14 and near the end of the time axis there is the appearance of a quite abrupt peak starting from week 24 and peaking in week 27. This peak signifies the outlier behavior for this 2 weeks time. The user is showing a great increase outside of its normal values although the specific user is having increased activity in general. Another important observation is that although the number of transactions rises vertically, the timeslots do not follow the same increasing rate. They do appear increased but not with the same rate.

.

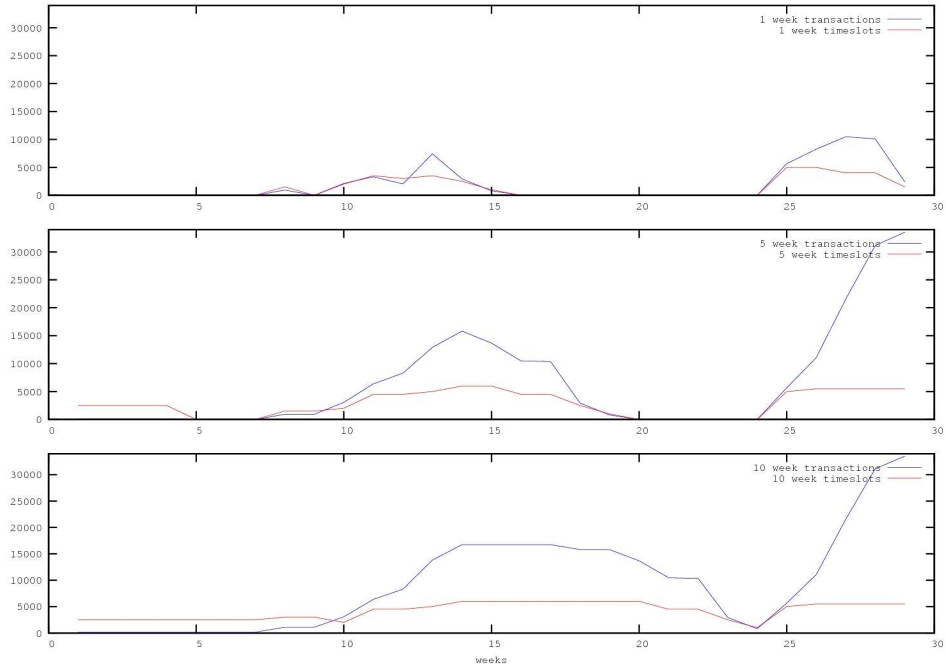

Figure 10. User B transaction and timeslots plot

When comparing the charts of the two users, it can be observed that the transaction axis does not have the same scale. The transactions axis for user A is in the scale of 0 to 3 400 while the transaction axis of user B is from 0 to 30 000. This difference can be noted easier if we plot both the transaction plots for 10 weeks together as done in Figure 11. As it can be seen, the transactions of user B reach values that are 10 times the transactions of user A. The low values in the activity of user A in comparison to the timeslots might be the reason that this user was detected as an anomaly in the time-vertical analysis. The user's transaction values might appear as an anomaly compared to the transaction values of the rest of the active users for the specific time.

User B, after the principal component analysis, gives a dataset of 78 principal components with 2 anomalies. These principal components are separated into the normal space, which is the first 12 and the anomalous space that is consisted of the components 13 to 78. When we plot the principal components of the anomalous space, we realize that the anomalies are located towards the end and, to be more exact, in the last principal component. Figure 12 shows the last 5 principal components where it can be seen clearly that the PC 78 has a big peak spreading towards the negative values of the axis y. When we tried to plot the normal space, however, we realized that the first 12 principal components have zero values. That comes as a result to the fact that within the 78 variables used in the initial dataset, there are many variables that have zero values for all the rows-objects. In an attempt to reduce this noise and test whether we would get different results, we removed any redundant variables that would only be zero and used only the variables that we considered that the anomalies are detected from. In this way, the 78 variable dataset was replaced by a 9 variable, really

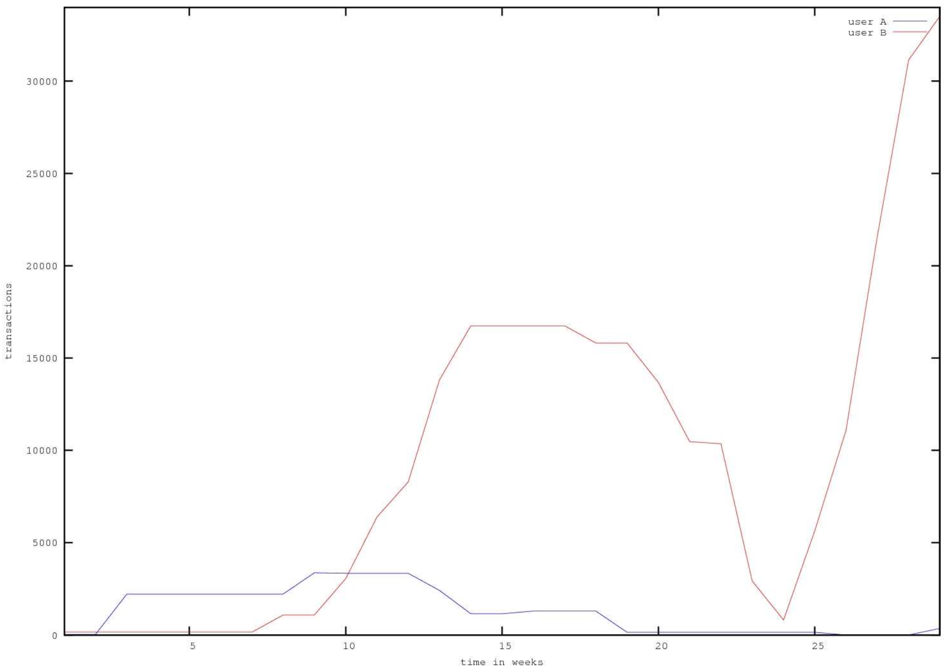

Figure 11. User A and B transactions

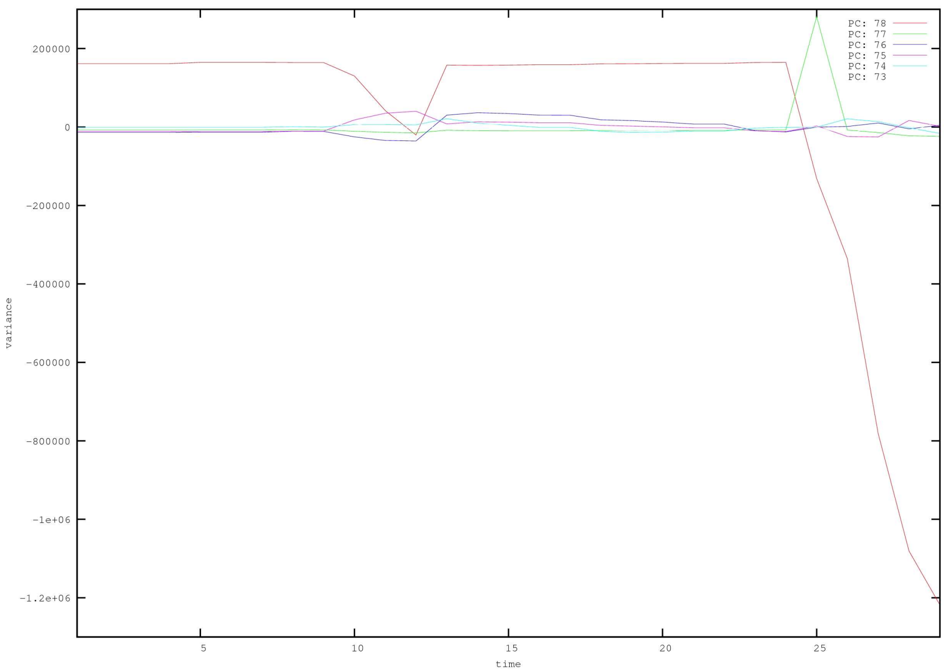

Figure 12. User B anomalous space PCs

compressed, dataset. When running the PCA on the compressed dataset we realized that the method detected the same 2 anomalies, but the anomalous subspace started from principal component 1. In other words, there is no normal space. Plotting the PCs in Figure 13 we realize that this occurs because since the first principal component captures the majority of the variety of the dataset, the anomalies are included in this variety. Fact that does not occur in the 78 parameter dataset because of the noise that the redundant zero values introduces.

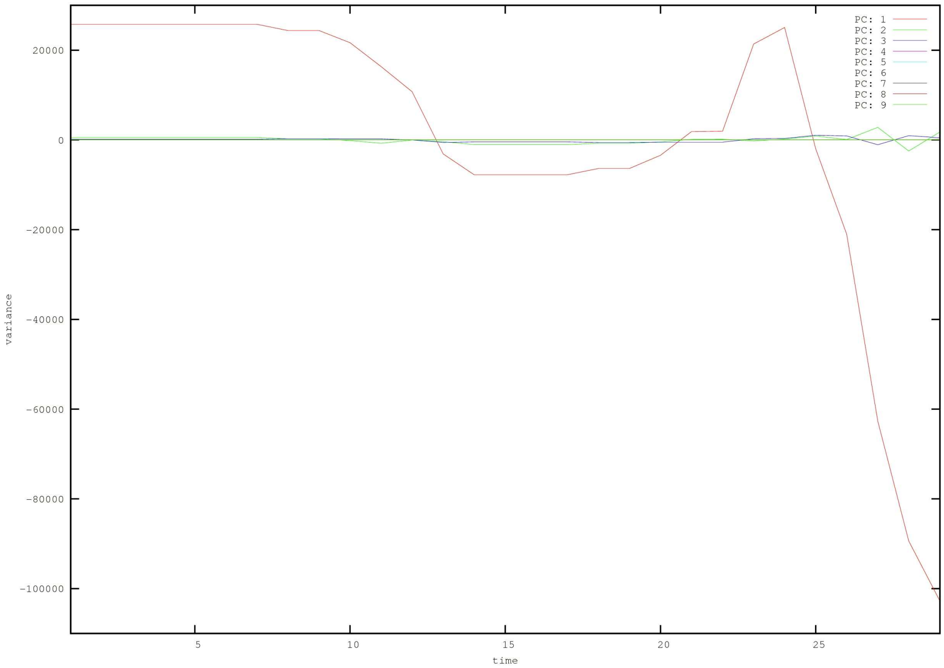

Figure 13. User B PCs for the compressed dataset

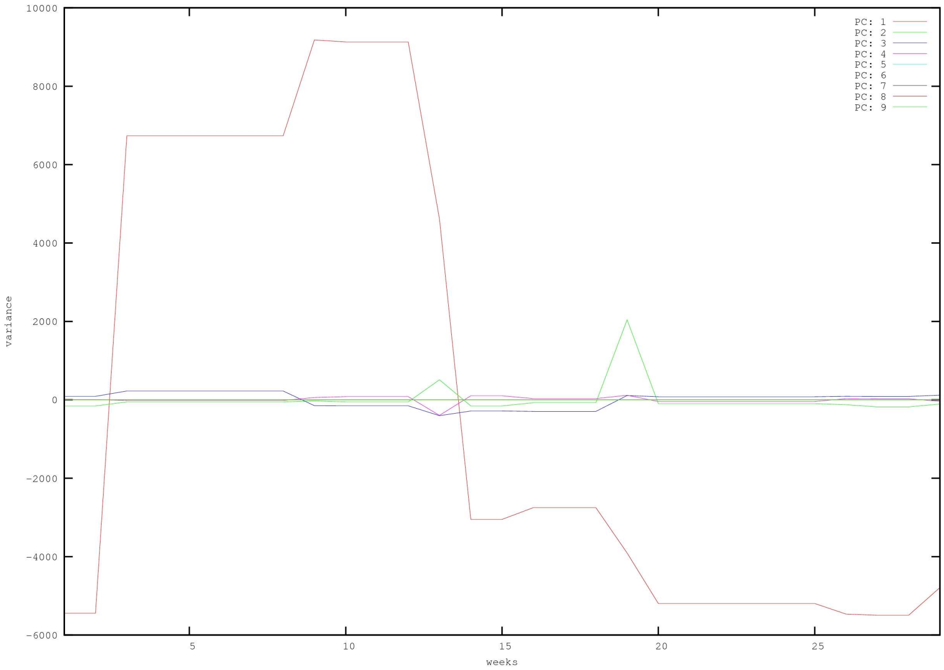

Figure 14. User A PCs for the compressed dataset

When using the same fields for the data of User A we get the plot of Figure 14. In this analysis the anomalous space starts from PC 2, however there are no anomalies detected. The most probable is that within the variables that were excluded from the dataset, there were variables that were exposing the anomalies for this user. Comparing the PC 1 of the two users we see that the scales of the y axis are really different. The variance of user A ranges from –6000 to 10 000 while the variance of user B ranges from -100 000 to 20 000.

3 Conclusion

The purpose of this work was to find the optimal method for the detection of fraudulent behavior and addictive gaming in online gambling data. This was achieved by two main steps: implementing a method for that task and comparing the results of this method with the results of the method already applied in the company for the same reason.

To achieve this task the data mining method of anomaly detection was brought into focus. For the needs of this work, the specific field of anomaly detection was studied as much as the current evolution of the methods applying anomaly detection. During this study, it was clear that there are many methods offering effective anomaly detection but they are all concerning specific natures of data and most of them were developed for detection of specific anomalies. Therefore, before adapting or creating a new method, the cause of the anomaly detection as much as the data to be applied should be analysed. After the analysis of the nature of the data and the purpose of the anomaly detection, the specialties of the data were noted. The datasets to be mined are large datasets with many parameters that might change in the future. Moreover, because of the fact that the purpose is to detect fraudulent transactions, the algorithm should not be really fault-tolerant.

After defining the specifications and restrictions for the method, the method was decided to be based on the Principal Component Analysis because it is effective in manipulating large dataset revealing any patterns on the data. On the principal component analysis method, there is a method proposed by Dunia et al. [8] and used for detection of network intrusion by Lakhina et al. [16]. This method separates the data that are coming out of PCA analysis into two subspaces, the normal and the anomalous. In continuation the anomalous subspace is scanned and with the Q-statistic method [11] the anomalous values are detected.

With the termination of the method implementation, the method was tested with real data to evaluate its efficiency. The evaluation was held in two kinds of data, the time-vertical and the user-based. For the time-vertical analysis the algorithm proved to detect the same anomalies from the top of the list of the company's algorithm although for the specific datasets it did not make use of the subspace approach. In the user-based approach and for all of the datasets applied, the method separated the data successfully into the normal and anomalous subspaces and provided with the anomalous instances. A random selection of the anomalies was evaluated graphically and was proven to have “strange” values for the anomalous point in the number of transactions or the timeslots..

In conclusion, concerning the purpose of this work, as captured in the question put in the beginning of this report, the author finds it extremely difficult to state whether or not the method followed is the most efficient for the specific nature of data. However, the method proposed has proven to be more effective than the currently applied in this data. The proposed method provides the possibility to control the sensitivity of the detection and most of all, it is a more resource-efficient implementation of anomaly detection as in some cases the subspace separation can reach half of the PCA data, which can be translated in the need to scan only half of the dataset for anomalous values.

4 Discussion

During the process of this work, a lot of effort has been spent on studying the principal components analysis. It is many times the case in technical studies when one is studying in depth one specific subject, that this subjects appears more and more as the most probable solution. In the chapter of the method analysis, two methods were taken as possible solutions to the problem: The Local Outlier Factor (LOF) and the use of evolutionary algorithms. The reason that the specific method was chosen over the LOF method was because it was decided that although the LOF method manages to detect effectively anomalies over passing the common issues of the distance-based and density-based algorithms, maybe it would not perform as efficiency in a larger dataset. While, the reason that the specific method was chosen over anomaly detection through the implementation of an evolutionary algorithm, was because of the need for a training set for the evolutionary algorithm. Although after the evolutionary algorithm has been trained, would probably perform better than the selected algorithm, the need for training would be more resource-extravagant and might limit the spectrum of the anomaly detection method. The specific choice, however, is not by any means excluding the possibility that there is a method not included in the related work research that performs anomaly detection for the specific set of data in a more effective way.

Although the method appears to work effectively in the tested datasets, its performance in really big datasets might bring respectable delay. The tested datasets were all of normal size and might prove to be smaller than the datasets that will have to be mined normally. If the method is to be applied for the specific task, it would be advisable to be tested on really big datasets. The code is consisted of several loops into the whole datasets during the subspace separation and the detection of anomalies with the Q-statistic. Possible optimisation of the code might increase the performance of the method.

One additional future work could be the method optimisation. As noted in the user-based analysis, in a dataset of 78 parameters there might be an average of 1 or 2 anomalies found. In the majority of the cases these 2 anomalies are continuous and they detect a high peak in the activity. These anomalies should have a tendency to be located in some part of anomalous space either in the begging or in the end. Maybe the separation of the subspaces into more than two would increase the performance of the method. If apart from the normal and anomalous spaces there is a transitional space that would be anomalous but with less probability of an anomaly existing in this space, the method would mine anomalies faster. Research on how the anomalies tend to appear inside the anomalous space would make this possible.

Finally, as it has been noted in the examples of the user-based analysis some anomalies that have been mined are caused from anomalous behavior but that behavior might not be fraudulent or addictive gaming. The study of how many of the resulted anomalies derive from fraudulent activity or addictive gaming will help in the “tuning” of the false alarm parameter c_α as much as the subspace threshold. This parameter still needs to be adjusted for the correctness of the results while the subspace threshold is believed to be able to perform a better separation.

5 References

[1]. Abe N., Zadrozny B. and Longford J., Outlier Detection by Active Learning, In Proc. of the *12th ACM SIGKDD international conference on Knowledge discovery and data mining*, Philadelphia, PA, USA, 2006

[2]. Aggarwal C. C., Yu P. S., An effective and efficient algorithm for high-dimensional outlier detection, The *VLDB* Journal — The International Journal on Very Large Data Bases Volume 14 , Issue 2, p. 211-221, Secaucus, NJ, USA, 2004

[3]. Ali A., Modood A., Azam H. and Mahmood W., ADAM: A Practical Approach for Detecting Network Anomalies Using PCA, In Proc. of *2004 National Conference on Emerging Technoligies, Shaheed*, Pakistan, 2004

[4]. Angiulli F, Basta S. and Pizzuti C., Detection and Prediction of Distance-Based Outliers, In Proc. of *2005 ACM Symposium on Applied Computing*, Santa Fe, New Mexico, 2005

[5]. Breunig M. M., Kriegel H. P., Ng R. T. and Sander J., LOF: identifying density-based local outliers, In Proc. of *2000 ACM-SIGMOND Intl. Conf. on, ACM Press*, New York, USA, 2000

[6]. Chlink G., Data Mining Taps the Trends, Government Executive Magazine, October 1, 2000, http://govexec.com/story_page_pf.cfm?articleid=18105 [acc. 2008-03-11]

[7]. De Jong K. A., *Evolutionary Computation: A Unified Approach*, The MIT Press, Cambridge, USA, 2006

[8]. Dunia R., Qin S. J., Multi-dimentional Fault Diagnosis Using a Subspace Approach, In *American Control Conference*, 1997

[9]. Hotelling, H., Analysis of a complex of statistical variables into principal components. Journal of *Educational Psychology* vol. 24, pp. 417-441, USA, 1933

[10]. Hand D., Mannila H. and Smyth P., *Principles of Data Mining*, The MIT Press, Cambridge, USA, 2001

[11]. Jackson J. E., *A User's Guide to Principle Components*, John Willey & Sons Inc., New York, USA, 1991

[12]. Jolliffe I. T., *Principal Component Analysis Second Edition*, Springer-Verlag, New York, USA, 2002

[13]. Kantardzic M., *Data Mining—Concepts, Models, Methods, and Algorithms*, Wiley-IEEE Press, Hoboken, USA, 2003

[14]. Khan L., Awad M., Thuraisingham B., A new intrusion detection system using support vector machines and hierarchical clustering, The *VLDB* Journal — The International Journal on Very Large Data Bases, Volume 16 , Issue 4, p. 507-521, Springer-Verlag New York, Inc. Secaucus, NJ, USA, 2007

[15]. Klösgen W. and Zytkow J. M., *Handbook of Data Mining and Knowledge Discovery*, OXFORD University Press, New York, USA, 2002

[16]. Lakhina A., Crovella M. and Diot C., Diagnosing Network-Wide Traffic Anomalies, In Proc. of *2004 conference on Applications, technologies, architectures, and protocols for computer communications*, Portland, Oregon, USA, 2004

[17]. Lazarevic A. and Kumar V., Feature Bagging for Outlier Detection, In Proc. of *11th ACM SIGKDD international conference on Knowledge discovery in data mining*, Chicago, Illinois, USA, 2005

[18]. Leung K., Leckie C., Unsupervised Anomaly Detection in Network Intrusion Detection

Using Clusters, In Proc. of *28th Australian Conference on Computer Science*, Newcastle, Australia, 2005

[19]. Ren D., Rahal I. and Perrizo W. and Scott K., A Vertical Distance-Based Outlier Detection Method with Local Pruning, In Proc. of the *13th ACM international conference on Information and knowledge management*, Washington, D.C., USA, 2004

[20]. Schölkopf B. and Smola A. J., *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*, The MIT Press, Cambridge, USA, 2002

[21]. Seinfert J. W., Data Mining: An Overview, Congressional Research Service, Order Code RL31798, 2004, http://www.fas.org/irp/crs/RL31798.pdf [acc. 2008-03-11]

[22]. Tan P.-N., Steinbach M. and Kunmar V., *Introduction to Data mining*, Addision-Wesley, Boston, USA, 2006

[23]. Witten I. H. and Frank E., *Data Mining: Practical Machine Learning Tools and Techniques*, Morgan Kaufman Publishers, Boston, USA, 2005

[24]. Zanero S., Savaresi S. M., Unsupervised learning techniques for an intrusion detection system, In Proc. of *2004 ACM symposium on Applied Computing*, Nicosia, Cyprus, 2004