International Journal of Principles and Applications of Information Science and Technology
February 2010, Vol.3, No.1

Importance of Feature Isolation in Detecting Fraud

John C. Fouhy[1], Paul J. Bracewell[1], Graeme L. Gee[1] and Chris H. Messom[2]

[1]Offlode, PO Box 8024, Wellington, New Zealand
[2]School of Information technology, Monash University Sunway Campus, Malaysia
{john, paul, graeme}@offlode.com, C.H.Messom@infotech.monash.edu

Abstract. Fraud detection requires more than powerful analytical techniques. Feature isolation is an important pre-processing step enabling abnormal behaviors, often indicative of fraud, to be readily identified. A generic process is presented which describes this approach to data manipulation. This method enables standard analytical methods to be used with several benefits ranging from ease of interpretation to real-time deployment. Three case studies are provided which show the importance of manipulating the data to enable the successful identification of fraud. In each instance, data pre-processing is required to isolate the characteristics of the underlying fraud which leads to the development of real-time processing for on-going fraud prevention.

Keywords: Internal Fraud, Application Fraud, Credit Card Fraud, BIN Attack, Analytics, Pre-processing, Feature Isolation.

1 Introduction

More often than not, the emphasis on improving predictive performance rests with fine-tuning applicable algorithms. In this paper a generic process for manipulating data to enable indications of change to be more readily identified is outlined. Application of this process is illustrated using three fraud-related case studies.

Manipulating data to isolate the underlying features allows standard supervised learning techniques, such as regression and decision trees, to be used in an out-of-the-box fashion. However, the data must be manipulated in such a way that the underlying meaning of the information described by the data is retained whilst isolating the features of interest. A suitable analogy is ripping the content from a compact disc and converting it to an mp3. There is some information loss (including some loss in quality), but there is no discernible loss of meaning and tremendous savings in storage.

Effective data manipulation exposes trends and isolates features leading to insight regarding the process of interest. The benefits are multiple and include ease of interpretation, real-time deployment and meaningful between and within cohort comparisons. Fraud detection uses techniques from a variety of fields. Seemingly unrelated techniques provide good examples of different processing techniques. For instance, the pre-processing in text-mining is an excellent example. Structure is imposed on unstructured text by first manipulating it into an information-rich, unit-

by-term matrix. This essentially counts the number of times a particular term appears in a document. Terms can be thought of as columns or dimensions. Singular Value Decomposition (SVD) is used to compress this large, unwieldy matrix into something more manageable. During this dimension reduction process unimportant terms are ignored, and highly relevant terms are emphasized. This enables the underlying concepts to be expressed as a combination of terms. These concepts are the features isolated as a consequence of data manipulation. Calculation of ratings is another method of simplifying the data whilst enriching the associated content. An example of this is summarizing rugby player performance to create a single statistic from more than 130 measures suitable for use by coaches and selectors enabling them to monitor the performance of individual rugby players using control charts [1].

Ratings allow similar behavior to be grouped together as a cohort. Departure from established behavior in credit card transactions or payment processing, observed in statistically significant changes in ratings, is often indicative of fraud.

Understanding the meaning of the data and how different variables relate makes the interpretation of results easier; this is crucial for selling results into a business [2]. The ability to use simpler methods, such as regression without interaction terms, reduces the effort of the analyst to explain to the consumers of the output. In the business environment, more emphasis is placed on the interpretability of the methodology than on the accuracy as it is the end-user who determines the acceptance and deployment of an analytical piece of work [2]. Interpretability is crucial as it enables a decision to be justified which is important for explaining why a transaction has been tagged for further investigation.

The computational effort is significantly reduced when the effort on surfacing the underlying data structures is focused on data manipulation rather than the algorithm for processing the data. For instance, standardization, transposition and extraction of metadata can be performed ahead of a transaction and stored in look-up tables to reduce the computational effort when a transaction is assessed. In training and preparation, this may have minimal impact upon systems. However, in a production environment, this can mean the difference between an application running in real-time or batch mode. This is the difference between detection and prevention.

For a fraudster to profit from illegitimate activity there needs to be sufficient activity: firstly to justify the effort and secondly to offset the risk of capture. This activity is often greater than historical behavior. This observed change in behavior can be used to protect the customer and financial services provider by using legitimate behavior as a second form of transaction validation. The first form of validation stems from authentication via the use of a Personal Identification Number (PIN), password or signature for example. Approaches such as these are vital in the fight against fraud, including identity theft.

To detect process, system or individual behavioral change requires a measure of behavior (feature) to be identified. This feature must then be cleaned to enable meaningful comparisons with historical or other entity behavior (isolation). This is achieved with standardization. Standardization of behavior provides greater versatility in creating cohorts for modeling and calibration. With fraud, it is changes in behavior that are often of interest for detection purposes.

Understanding the data and underlying process that generates the data is crucial before any analysis can begin. For this reason, students of statistics are directed to

examine the data first. This includes examining the distribution of each variable via histograms and examining data quality issues. The context of the data, process and problem provide the framework for the solution. Understanding these attributes determines the data manipulation required to solve the problem. This process is summarized generically in the next section and then applied to three case studies.

The first case study demonstrates the value of data manipulation by detecting a specific type of credit card fraud known as a BIN (Bank Identification Number) attack. The composition of the credit card number and anonymous access to online merchants enables hundreds of new card numbers to be generated. In this instance, examining the collective behavior of cards that share the first 12 digits of a credit card enables the fraud to be identified.

The second case study requires manual intervention in the pre-processing stage to correctly flag credit card applications for identities that have been fabricated or stolen and modified. This enables algorithms to be trained to more efficiently classify between the apparently linked cases which are fraud and those that are not. Importantly, this enables an automated, scalable solution for application fraud prevention to be developed.

Finally, the third case study uses data manipulation to create a flag for identifying clients whose payments to an organization have been compromised due to internal fraud. Construction of a single customer view enables all entities to be tracked throughout the system. Pre-processing is then used to identify departures from the pattern of historical client payment behavior. An abstraction of this pattern is taken to make the process scalable and suitable for on-going fraud prevention.

2 A Process for Feature Isolation

Identification of fraud is not easy. Fraud detection is much simpler when there are historical instances of fraud which enable a supervised learning model to be constructed to identify future events. There are unsupervised learning methods which allow fraud to be identified, such as principal component analysis; however, these are reliant on user interpretation. In this section, a generic process for isolating features of interest is outlined. This will flag the behavior of interest enabling supervised learning to be used for on-going fraud detection. Whilst this process is similar to the *CR*oss *I*ndustry *S*tandard *P*rocess for *D*ata *M*ining (CRISP-DM), it specially focuses on the construction of a suitable dataset and variables for fraud detection thereby enabling standard analytical approaches to be deployed. The process is summarized by four key steps, which are explored below:

1. Find entry point within data based on problem context.
2. Create short-fat view of the entity of interest using the entry point to define the perspective.
3. Quantify behavior for entity of interest with respect to problem context.
4. Review unmatched entities or statistically different behaviors at any stage.

To begin analyses an entry point is required. The entry point is a reason or purpose for performing the task and is necessary to ensure the activity gains traction within a business. There will be a variety of reasons for a fraud detection exercise

including: a belief within the organization that suspicious activity is occurring, whistle blowing, a known weakness in the system, or an industry trend. This defines the problem context and any solution must address this need. This also provides an analyst with insight from the business perspective into the process which can help shape the final solution.

Next, the data needs to be understood in the context of the environment. This is achieved via the construction of a single customer view. This enables elements in the system to be tracked throughout the process. In knitting together the single customer view, there are likely to be elements that do not match. This leads to exclusion analysis. The reasons for entities not matching within the process need to be explored as it may indicate flaws, data quality issues or fraud. Constructing the single customer view requires the data to be examined from a structural perspective. Exploring the data from a context perspective is the next part of the process. This includes examining distributions, metadata and apparent data quality issues.

The final stages involve quantifying behavior in process. This is achieved through the construction of ratings, benchmarking and outlier analysis. Abnormal behavior in the process is tagged. It is important to prioritize alerts to manage workflow. The best method is to sort alerts in descending order by risk and value.

At each stage, abnormal items require investigation. These abnormalities are the isolated features. Whilst not all of the features isolated will be due to fraudulent behavior, experience has taught us that there is an increased likelihood of it being fraud. This process is systematically applied in the following case studies.

3 Credit Card BIN Attack Case Study

3.1 Overview

Credit card fraud is unauthorized activity on a credit card account. Retailers have battled credit card fraud for years and, as a consequence, have developed methods, ranging in sophistication, to deter fraud. These merchants train their staff to perform simple checks to ensure the validity of the consumer's identity. Workers can compare the signature on the back of a credit card against the signature on the credit card receipt and may ask for photo identification. Merchants are further protected from the pitfalls of fraud by the nature of the card present environment. Card present means the credit card is physically present during the purchase. When a dispute occurs in this environment, and the merchant has properly collected the consumer's signature on the credit card receipt, the consumer's credit card company will most likely absorb the disputed amount based on the rules established by the credit card institutions.

However, in the rapidly expanding area of e-commerce, merchants do not have the advantages that physical world offers. First, it is impossible to collect a valid and acceptable signature of a consumer, let alone a photo ID. Without imposing costly technological solutions (such as biometric scanners) or increasing the number of steps in a transaction (such as two-factor authentication or use of tokens), it is difficult to perform any of the "physical world" checks to detect who is at the other end of the transaction and to conform to the protections in the card institution's rules. This

makes the Internet attractive to fraud perpetrators and is a huge problem for Internet merchants. Purchases made over the Internet are considered by financial institutions to be card-not-present transactions (CNP).

Fraudsters have many ways of stealing cards, including old-fashioned techniques such as counterfeiting (at point-of-sale, say) or simple theft, and new techniques such as phishing or hacking online databases. This case study explores a simple way of generating new card numbers from a single known-valid number, and describes an equally-simple way to combat the problem.

Credit cards have essentially five features: cardholder's name, card number, card expiry date, magnetic stripe and Card Verification Value or Card Verification Code (CVV2 or CVC2). The card number is broken into meaningful components. The first digit is the Major industry identifier (4, 5: traditional credit card). Digits 2–6 are the issuer identifier number. These first two components (digits 1-6) are the Bank Identification Number (BIN). Digits 7–15 are the cardholder's account number and digit 16 is the check digit.

Each issuing bank will have at least one BIN. They may only issue credit cards with numbers starting with one of their assigned BINs. The check digit is specified by the Luhn algorithm. The Luhn algorithm is a modified mod-10 checksum [3]. Consequently, a bank can issue at most 10^9 different credit cards for a given BIN.

There exist various supplemental measures for protecting consumers [4]. However, in order to use a credit card online, typically only need the card number and expiry date are required. The only security the credit card provides is the large space of potential numbers: for a particular BIN, a bank can issue 10^9 cards. If expiry dates take values within the next two years, this gives 24×10^9 (card number, expiry date) pairs; a number too big to effectively explore or guess from.

However, if a bank has non-randomness in its card number allocation strategy, this introduces weaknesses that fraudsters can potentially exploit. Historically, banks have issued credit cards in segments, each card in a segment having the same expiry date. This has allowed fraudsters to attack cards in CNP situations by starting with a known-valid (card number, expiry date) pair and incrementing or decrementing the card number (modulo the Luhn algorithm). The fraudster will typically make small test transactions at a suitable merchant. If the transaction succeeds, they can then use the card for larger purchases, or sell the details on the black market [5]. This procedure is easy to automate, so the fraudster can test hundreds or thousands of cards in an hour, expecting many hits. This is known in the industry as a BIN attack.

3.2 Analysis

Most Australasian banks use a rule-based expert system to monitor credit card transactions. Because the initial tester transactions in a BIN attack are all small, they will typically not raise an alert in the expert system. Later transactions may, but the systems are not perfect so they may not, and by that stage the fraudsters know they have a valid number, and thus are spending large amounts. Furthermore, if the fraudster keeps their activity on the stolen card low, they may stay "below the radar" and remain undetected. A pattern exists (transactions on similar card numbers), but existing systems look at transactions in isolation and thus do not see the pattern.

To overcome this, we searched through daily transactions for clusters of transactions at the same merchant with the same 12-digit prefix. Intuitively, we hypothesized that such collisions would be rare amongst legitimate transactions, and thus their presence would signal a BIN attack in progress. This became our entry point. Our analysis later confirmed this hypothesis. We used transactional data from a six-month period containing known BIN attacks for our analysis.

We examined daily transactions across all merchants with the merchant category code for gambling. We mapped each transaction to its 12-digit prefix, and then counted occurrences of these prefixes per day at any gambling merchant. Knowing that credit card numbers are issued non-randomly, and the manner in which the fraudster harvests legitimate card details, provided the insight for structuring the data.

The frequency distribution of transactions was then examined. The 95% confidence interval for the average number of transactions under $30 at all gambling merchants grouped by 12-digit prefix for a typical day is (0.97, 1.09); whilst during a 24hour BIN attack the interval is (3.47, 4.14), a significant difference.

For approximately 99% of daily transactions under normal conditions, there was no second transaction on the same day with the same 12-digit prefix. In 0.03% of cases, the same card was used more than once; in all cases, this indicated a heavy user (the cardholders had used their card more than sixty times over the previous six months).

The usage was drastically different during the card validation phase of a BIN attack at one online gambling merchant over a three day period. More than 70% of transactions were for cards within a range that was used more than five times in a day. This highlights the simplicity of a BIN Attack.

We found that the fraudsters test each card number with a small transaction (typically less than $30) through an internet or phone merchant. If a test failed, it usually failed because the fraudster had the wrong expiry date, rather than because of a “hole” in the block. The fraudsters usually retested these cards using another expiry date if they succeeded with their original BIN attack. The fraudsters then make large purchases (generally $300–$3000) using the card numbers that worked during testing or attempt to sell the valid cards online.

Data manipulation (in this instance the simple truncating of a credit card number) enables standard analytical techniques to be used. A decision tree was found to provide explicit, robust answers. We achieved a 97% block rate through the addition of some business rules and proactive response strategies.

4 Application Fraud Case Study

4.1 Overview

Credit card fraud costs banks and merchants billions of dollars annually. The majority of credit card fraud comprises fraudulent use of legitimate accounts. However, a significant minority involves criminals fraudulently obtaining lines of credit, such as credit cards, from a bank. The fraudulent nature of this lies in that the fraudsters obtain credit under either a stolen identity, typically with some details changed, or a fully fabricated identity. When the bank attempts to recover the debt, it is written off

when the debtor cannot be found. In 2005, APACS, the United Kingdom's trade association for payments and payment services which claims to represent approximately 97% of the UK payments market by volume, noted that application fraud cost £10.9 million of £439.4 million of reported revenue lost to fraud. Account takeover cost an additional £19.8 million.

The first fact that needs to be considered is that the fraudster needs to somehow receive the card for which they have applied. It would be a relatively straight-forward exercise to create an unrelated identity; but how is the fraudster to profit if the bank mails the card to an unrelated property in another city? How can the fraudster succeed if the bank calls to verify the identity, but a stranger's phone number has been supplied, and they deny asking for a card? Further consideration leads to a second fact: if a criminal succeeds in obtaining a credit card fraudulently, they will typically only get a few thousand dollars. If the fraudster relies on this illegal money, they must continually attempt to steal more. These facts underpin the abnormality of behavior; pattern matching tools rely on these facts to detect fraud.

In the credit card business model, the cardholder spends the bank's money, and then later repays the bank. If the bank does not know the cardholder, it will have no way of tracking this entity and forcing repayment. This idea underlies application fraud. In application fraud, the fraudster does one of two things: 1. He or she creates a fictitious identity and applies for a credit card in that name, or 2. He or she uses someone else's identity (possibly with some modification) to apply for a credit card.

It is suggested that this form of "synthetic identity fraud" is on the rise [6], facilitated by the fact that credit rating companies typically have dirty data and rely on fuzzy matching or sloppy standards to manage this [7]. Common tricks include:

- The fraudster applies for a card using an ex-partner or close friend's details.
- The fraudster applies for a card, then, after bank approval, but before the card is sent out, they phone the bank and change their delivery address.

Application fraud can be difficult to measure, because there are no chargebacks or victimised cardholders. If the bank approves a fraudulent application, the bank's collections department has to deal with the aftermath, and they may lack the ability to identify fraud, or they may fail to communicate with the fraud department. As such, application fraud detection rates vary significantly by bank. Some industry sources report that some banks do not acknowledge the existence of application fraud; they simply treat all instances of the problem as bad debtors. Furthermore, even if a bank has the systems to detect application fraud, they will not detect it until the bank has given up on trying to extract payment from the fraudster, which will typically be at least 3 to 4 months after the application. Finally, in the absence of regulatory requirements, banks have little incentive to publish fraud rates, or make this information available to researchers. Berkeley Centre for Law and Technology senior fellow Chris Hoofnagle ([6], [8]) discusses the difficulties of tracking this form of fraud in the United States, and the lack of public information on banks' performances [7]. On the other hand, because each card has only a short lifetime, application fraudsters are serial fraudsters by nature. As such, they often use similar details in their applications, and this makes them vulnerable to detection. Furthermore, they are often greedy, in that if they are accepted, they often apply again immediately. Finally, there is a third category of people who lie on their credit card applications: people whom the bank rejected, and who subsequently reapply, "exaggerating" their

salary, or otherwise making themselves look like a lesser credit risk. Some banks consider these people as fraudsters whilst others classify them as bad credit risks.

Application fraud differs from fraud on an existing card; in that there is no legitimate spend history, and no legitimate cardholder. The most common software package in the field of application fraud detection is Experion's *Hunter*. Hunter is a rules-based system that examines the preceding six months' worth of applications and looks for similarities with the current application. Some example rules:

- "Same date of birth and same surname as a known-fraudulent application."
- "Employer phone number used three times in the last thirty days."

While Hunter is the major player in banking, there exist other products that target similar areas. *NORA* (Non-Obvious Relationship Awareness) is a similar product recently purchased by *IBM*. It targets casino-goers, looking for cheats who have been banned, and are trying to return under a different (but typically similar) identity [9]. *id:analytics* offer a service to examine data and look for strange linkages. Unusual connections within customer data might indicate that a company has suffered a data loss and criminals are conducting identity theft against customers. *Identity Systems*' Identity Search Server seeks to match callers to their identity. It assumes that the caller is not seeking to deceive, but that operator or system errors or failures may make their identity hard to locate. In 2000, Wheeler and Aitken released a study on the use of case-based reasoning (CBR) to study application fraud [10]. They gained access to real data at an unnamed institution. This institution used a rules-based system to classify applications, and then humans processed the output of the system to provide definitive labels for the data. Wheeler and Aitken were given the applications that the rules-based system had classified as fraud (approximately 2100) together with the real labels (almost 2000 were not fraud; 175 were). They used this data to do their experiments. They ultimately claimed to a classifier that was 80% accurate on non-fraud and 52% accurate on fraud. However, they say that their results were somewhat "fragile" and very dependent on the precise values of certain thresholds. They do not appear to have published any follow-up research.

In 2005, Clifton Phua and the Data Mining in Identity Crime Prevention group at Monash University in Australia started working on a link analysis system for detecting application fraud. Link analysis is, in many ways, a logical extension of CBR. Given an application, they use a similar system to Wheeler and Aitken to determine other related applications by looking for common attributes, and then test this count against a threshold. They then use this to build a directed graph, where directed edges point from each case to similar cases (if any similar cases can be found). They were able to use this framework to support a black-list (of known fraud cases), a white-list (of reasons why a connection would not indicate fraud), and decaying suspicion score. Their work is ongoing ([11], [12]).

We are currently aware of no other academic research in the field of application fraud, although [13] and [14] both cite credit scoring (in the case of loan applications) as motivating examples. Bolton and Hand [15] and Phua et al [16] surveyed literature relating to fraud detection without noting other academic research relating to application fraud.

4.2 Analysis

The standard industry accepted approach to application fraud is that it can be best fought by cross-referencing new applications against historical applications, looking for anomalies. Experian's Hunter, the standard software package used by banks worldwide to combat this problem (banks, that is, which acknowledge application fraud occurs), implements this theory. However, there is no published research examining the accuracy of this theory. Industry workers observe partial success, but they miss some fraud, and typically deal with high false-positive rates from the software. Furthermore, Hunter stores only six months' worth of historical data. If the bank approves an application later found to be fraudulent, they may not discover their error until after the application has expired from the database. Finally, the industry workers typically lack the time or budget to conduct their own analysis.

In order to identify this type of fraud using appropriate algorithms, it is first necessary to have full false negative data (i.e. credit card applications that the bank approved, but where the cardholder did not make any payments, and the bank was unable to trace them for debt collection purposes). This allows a case-by-case analysis of false negatives, to determine if relationships existed between these applications and others that the system should have detected. This is important in determining the overall success of the strategy (for example, if it should turn out that a high proportion of false negatives have no obvious connection to other applications, then we would see the need to search for complementary techniques).

Business rules can be written to parse the data to tag what may be a fraudulent application. This is a form of data manipulation. However, to ensure that the deployed algorithm is successful (high capture and hit rates) requires human intervention, and further data manipulation. This involves manually assessing each application tagged as a match by the business rules and categorizing it as:

- the same person, with no apparent lying or intent to mislead
- either the same person but lying (typically about their age, or changing their name), or a suspicious connection to someone else (e.g. the same phone number, but no other connection)
- no connection at all

It is only from this foundation that suitable learning algorithms can be deployed as true indication of underlying behavior is now present in the data. This is crucial for a successful supervised learning exercise. This enables useful measures of rule and overall system effectiveness to be established.

As an example of flagged entries, 18 applications were made with one of two mobile phones, both of which passed through the same household at one point; this is clearly suspicious behavior. A further 12 applications for a credit card were received from the same man, varying spelling of his first name, varying spelling of his surname, and providing varying information. Potentially his Inland Revenue (IRD) number, or his mobile phone, or his date of birth could be used to link him to other applications; but there was no single theme. About 30 applications were clustered around a single employer (a restaurant). It was difficult to tell how many actual people were involved. The applications might be suspicious because they are different people with same contact details (all or most of home, work, mobile phone, email address, home address), because they have the same name but a different date

of birth, or a similar name and similar date of birth. The common theme is the employer, but this is not an effective way of identifying them programmatically because (1) they use variant spellings (including misspellings and whether or not they add "ltd" on the end), and (2) a rule like "same employer as a past fraudulent application" has a very high false-positive rate. Last, 26 people, mostly different, using the same loyalty program reference number. The only other common theme is that all the applicants lived in the same city (though most had names from a particular ethnic group). Upon investigation it was discovered that a car dealer was referring them to the bank for loans and giving them her loyalty program reference number; this is probably not fraud, but definitely worth watching each application carefully.

Commercial sensitivity prevents us from disclosing the performance of the techniques we have developed. However, our approach has saved one bank millions of dollars in the past two years. Furthermore, fraudsters will continue to evolve thus any process must be readily adaptable.

5 Internal Fraud Case Study

5.1 Overview

Any organization is vulnerable to internal fraud. This is an extremely sensitive issue and as a consequence details are brief to protect the identity of those involved. This case study examines fraud perpetrated by an employee who held the position of Accounts Clerk at a large corporation. The fraud was characterized by the theft of a large number of client cheques worth a substantial sum over an extended period of time. This was achieved via a simple and effective money laundering technique. Whilst the organization had protocols to prevent the theft of cheques, the fraudster was able to circumvent these. The nature of the organization's business meant that payments were not always expected. This meant that some invoices were not raised till after payment had been received. Additionally, sometimes a missed payment would be added to the next due payment. Consequently, the approach outlined below looks for the absence of an event per client.

5.2 Analysis

Data manipulation was crucial to detecting this fraud. Firstly, a single customer view was constructed enabling payments to be tracked throughout the system. Using the historical behavior of the clients, an approach was developed for identifying the characteristic of this fraud and detecting payments that were not present. Detection was reliant on the existence of sufficient repetitive behavior prior to the fraud event allowing a pattern to be established. Each series of transactions per client was examined. Where there appeared to be a departure from repetitive behavior, this was noted and the date and amount were estimated. An example of this is provided in the table below. The client made regular payments of approximately $1000 paid about every 6-7 months. It was estimated that a payment of $1037 (average payment) was scheduled for May 20XY given that 14 months had lapsed between payments. Later, it was found that the client had sent a cheque for $677 on 21 May 20XY.

Table 1. Example of apparent missingness in client cheque payment behaviour.

Date Stamp	Cheque Amount	Days Between Transactions
Sep-XW	$1,276	-
Mar-XX	$769	181
Oct-XX	$851	214
Dec-XY	$1,198	426

This process was performed manually for a sample of payments where the client had paid by cheque at least three times. The intent was to identify features which could be detected automatically using a parsing algorithm and thus become scalable. This crude parsing algorithm identified 25% of the fraud with a hit rate of 1 instance of fraud for every 12 transactions flagged. Importantly, this led the organization to the fraudster who then confessed.

Two main types of behavior were used to detect the fraud. The first behavior was the time between payments and the second type is the amount of the payments. Several variables were generated from these two types including the interaction between payment and time to account for clients who "doubled-up" when they missed a previous payment. Several steps followed, including standardization and converting the findings into a tool for fraud prevention [17].

6 Conclusion

This paper has demonstrated that data manipulation is an important pre-processing step enabling abnormal behaviors, often indicative of fraud, to be exposed. More importantly, it is vital that any analysis considers the practicalities of the situation. Understanding the process provides the insight necessary to structure the data in such a way that fraud can be detected. In the three case studies that were explored, the data manipulation was simple and successfully identified fraud. However, the simplicity of the pre-processing was gained from abstracting the core drivers of behavior to expose the latent features that were potentially indicative of fraudulent behavior. Consequently, it can be straight-forward for organisations to embark on a fraud detection program. Thus, perceived complexity should not be a barrier for an organization to examine their systems for fraud. From a prevention perspective, there is room for increasing complexity in order to minimize false positive rates. Most importantly, effective data manipulation enabled standard analytical techniques to be used leading to real-time deployment. In each of the case studies, the preliminary work enables the exercises to evolve from detection to the development of real-time processing for on-going fraud prevention.

Acknowledgments. Exerts of this paper are part of a larger body of work by John Fouhy, "*Investigating Credit Card Application Fraud Detection Using Causal Inference and Case-Based Reasoning*", a PhD Thesis funded by Offlode Ltd and a Technology in Industry Fellowship from Technology New Zealand. This research is supervised at Massey University by Dr. Chris Messom and Dr. Martin Johnson.

References

1. Bracewell, P.J. (2003). Monitoring Meaningful Rugby Ratings. *Journal of Sports Sciences*, 21 (8), pp. 611-620.
2. Bracewell, P.J. (2008). A framework for using analytics to make decisions. *Handbook of Research on Socio-Technical Design and Social Networking Systems*. pp. 250-263. B. Whitworth and A. de Moor eds. Information Science Reference: New York.
3. Luhn, H. P. (August 1960). Computer for verifying numbers, *U.S. Patent* 2950048.
4. Gee, G.L., Fouhy, J.C., Bracewell, P.J. & Zeiske, R. (2007). Identifying Credit Card BIN Attack Fraud Empirically. *IIMS Post Graduate Conference*. pp. 109-115. Massey University: Auckland.
5. Acohido, B. and Swartz, J. (2006). Cybercrime flourishes in online hacker forums, *USA Today*. [Online] http://www.usatoday.com/tech/news/computersecurity/infotheft/2006-10-11-cybercrime-hacker-forums x.htm
6. Hoofnagle, C.J. (2007). Identity theft: Making the known unknowns known. *Harvard Journal of Law and Technology*, 21(1).
7. Mcfadden, L. (16 May 2007). Synthetic identity theft on the rise. Available at : http://www.bankrate.com/brm/news/pf/identity_theft_20070516_a1.asp. Accessed: 26 August 2008.
8. Hoofnagle, C.J. (2008). Measuring identity theft at top banks. Law and Technology Scholarship (Selected by the Berkeley Center for Law & Technology),
9. Kushner, D. (April 2006). Vegas 911. *IEEE Spectrum.*
10. Wheeler, R. and Aitken, S. (2000). Multiple algorithms for fraud detection. *Knowledge Based Systems*, 13(2-3). pp. 93-99.
11. Phua, C., Gayler, R., Smith-Miles, K. and Lee, V. (2005). On the approximate communal fraud scoring of credit applications. *Proceedings of Credit Scoring and Credit Control IX*.
12. Phua, C., Gayler, R., Smith-Miles, K. and Lee, V. Lee (2006). Communal detection of implicit personal identity streams. *Proceedings of CDM06 Workshop on Mining Evolving and Streaming Data*.
13. Wilke, W. and Bergmann, R. (1996). Considering decision cost during learning of feature weights. *EWCBR*, pp. 460-472.
14. Ikizler, N. (2002). *Benefit maximizing classification using feature intervals*. Master's thesis, Department of Computer Engineering, Bilkent University, Ankara, Turkey.
15. Bolton, R. and Hand, D. (2002). Statistical fraud detection: A review. Statistical Science, 17(3). pp. 235-255.
16. Phua, C., Gayler, R., Smith-Miles, K. and Lee, V. (2006). Communal detection of implicit personal identity streams. *Proceedings of CDM06 Workshop on Mining Evolving and Streaming Data*.
17. Bracewell, P.J. & Palaci, F. (2008). Putting Cheques in Place to Identify Fraud. *SAS Forum*. Sydney. Available at: https://www.sasforum.com/anz/presentations Accessed: 26 August 2008.