U.PORTO

FEP FACULDADE DE ECONOMIA
UNIVERSIDADE DO PORTO

FRAUD DETECTION AND PREVENTION USING NETWORK MINING

Maria Inês Rodrigues Ferreira

Dissertation

Master in Modelling, Data Analysis and Decision Support Systems

Supervised by

Doutor Pedro José Ramos Moreira de Campos

Doutor Fábio Hernâni dos Santos Costa Pinto

2020

Biographic Note

Inês Ferreira was born in Santo Tirso in 1993, where she studied until high school.

In 2011 she joined Faculdade de Economia at Universidade do Porto and finished her bachelor in economics in 2016. In this year, she had her first professional experience at Banco Popular, as credit analyst.

Currently Inês is in the process of completing her master in Modelling, Data Analysis and Decision Support Systems and she works at BNP Paribas Lisbon as Controller, since 2018.

Acknowledgements

I would like to acknowledge everyone who played a role in this academic accomplishment.

To my parents, for the unconditional support, for being an example of commitment and hard work and for always wanting the best for me.

To my brother and my uncle Rui, who inspire me every day, being an example of ambition.

To my grandparents, for being an example of dedication.

To my friends for all the support and understanding my absence during these times.

A special thank you to Professor Pedro Campos and Fábio Pinto, my supervisor and co-supervisor, for always believing in me, for all the support, guidance and encouragement.

Abstract

Every year, with the advance of technology and the changing of the way that human being lives, a huge number of new concepts appears and the global markets get more complex. In the other hand, due in large part to the same advances in technology, especially with the continued growth of the online marketplace, businesses have to constantly fight fraud in order to protect their own interests as well as their customers' privacy.

Fraud is one of the oldest phenomena of the human activity and detecting it is a challenge.

The main goal of this study is to analyse the impact of network measures in fraud detection. As so, it is intended to present an effective way of detecting fraud, bearing in mind the network relationship of the customers and, eventually, preventing it.

For that purpose, three classifiers were studied using a dataset extracted by *PaySim* simulator, not only with the initially provided information, but also with the addition of some features, some network measures and evaluated its predictive capacity.

At the end, it is possible to conclude that the impact of network science in fraud detection is biggerr in logistic regression, when compared to decision tree and SVM.

Resumo

Todos os anos, com o avanço da tecnologia e dada a mudança dos estilos de vida do ser humano, um grande número de novos conceitos emerge e, globalmente, o mercado torna-se mais complexo. Por outro lado, devido aos mesmos avanços da tecnologia, especialmente com o crescimento contínuo do mercado online, as empresas têm que lutar constantemente contra a fraude para proteger seus próprios interesses, bem como a privacidade de seus clientes.

O conceito de fraude é um dos mais antigos fenómenos da atividade humana e detetá-lo é um desafio.

O objetivo deste estudo é analisar o impacto das medidas de rede na deteção de fraude. Assim, é pretendido apresentar uma forma eficaz de detetar fraude, tendo por base a interligação dos clientes em rede e, eventualmente, prevenir a sua ocorrência.

Para isso, foram estudados três classificadores usando dados do simulador PaySim, não só com a informação inicialmente providenciada, mas também com o acréscimo de alguns atributos bem como medidas de rede e foi avaliada a sua capacidade preditiva.

No fim, é possível concluir que o impacto das redes na deteção de fraude é maior na regressão logística quando comparado com a árvore de decisão e SVM.

Index

Index of Tables

1. Introduction

Recently, disruptive technologies such as smartphones, mobile payments, cloud computing and contactless payments have emerged very quickly and almost simultaneously and, at the same time, it was possible to detect large-scale data breaches. (Nick et al, 2018)

With this, innovative and sophisticated techniques associated to fraud are emerging on a regular basis and hence, it is urgent "to develop improved and dynamic techniques capable of adapting to rapidly evolving fraudulent patterns" (Adewumi and Akinyelu, 2017).

It is important to mention that the impacts of fraud do not only affect companies. It may lead to inexplicable changes at a global scale: currencies, innovative activities, interest rates, exchange rates and money demand change and the monetary instability increases. The impact of fraud is huge and has direct implications in both companies and economy (Lavion, 2018).

Along the years, statistics and machine learning provided effective technologies for fraud detection (Bolton and Hand, 2002), mainly focusing on credit card fraud (Sherly and Nedunchezhian (2010), Malekian and Hashemi (2013), Chandola et al. (2009)), telecommunications (Farvaresh and Sepehri, (2011) and Sanver and Karahoca, (2009)) and healthcare (Francis et al. (2011), Tsai et al. (2014) and Lu and Boritz (2005))

However, despite the recent events just mentioned, fraud is a world-wide phenomenon, so old as the humanity itself and can be performed by almost an unlimited number of ways. (Bolton and Hand, 2002)

As so, finding methodologies for fraud detection is essential.

In this study, it is going to be analysed the impact of network measures in fraud detection. Answering the question *Does network science improve the predictive capacity of algorithms in what regards to fraud detection?* becomes the central part of this study.

For that, three classifiers (Decision Tree, Logistic Regression and Support Vector Machine (SVM)) are going to be analysed and compared using three metrics (Precision, Recall and Area Under Curve). To obtain the results, R programming is going to be used.

The dataset used was obtained through *PaySim* simulator[1]. *PaySim* is a sample of the records of a mobile money service implemented in an African country. These records were provided by the multinational company Ericsson (ericsson.com). (Lopez, 2016)

The results were obtained for four different scenarios: 1) with the information provided by *PaySim*; 2) added to initially provided information the cumulative sum of the transactions (for both origin and destination users), the weight of the user, based on the proportion between the amount associated to fraudulent transactions of that user and the amount of transactions done by the user so far and the type of the user (for both origin and destination), given by the first letter of the users' ID; 3) added two network measures (betweenness and density) to the initially provided information; 4) combined the scenarios 1), 2) and 3).

This thesis is organized as follows: in chapter 2., Literature Review, we reviewed some of the previous work related to fraud and some concepts were presented. It was also introduced some of the most common metrics used to evaluate classifiers and networks and focused on fraud detection algorithms.

In chapter 3., Problem Description and Methodology, it is introduced the problem to be studied and the analysis to be performed. The details of the dataset to be tested are also presented.

The main analysis was performed in chapter 4., Results, where three classifiers were applied before and after using different sets of features and network measures.

Finally, in chapter 5., Conclusions, the main conclusions of this study were presented.

[1] Data available at: https://www.kaggle.com/ntnu-testimon/paysim1

2. Literature Review

2.1. Economic Fraud

According to Lavion (2018), "49% of organizations globally said they've been a victim of fraud and economic crime, 64% of respondents said losses due directly to their most disruptive fraud could reach US$1million, 52% of all frauds are perpetrated by people inside the organisations and 31% of respondents that suffered fraud indicated they experienced cybercrime."

These values are growing every year and reached an historical maximum. However, according to McDowell and Novis (2001) the side effects of fraud and economic crime affects not only the segment where it occurs, but it comes with a destructive effect to the whole economy: it affects business decisions, it damages the companies and state's reputation, it increases the risk of bank collapse, it decreases the efficiency of the governing bodies and decreases the well-being of a society. With this, we can say that fraud and economic crime are international matter and, as so, deserve an international attention.

According to the Cambridge dictionary, fraud is defined as "the crime of obtaining money or property by deceiving people". But it is important to distinguish some kinds of fraud. There is internal and external fraud. Internal fraud happens when an employee commits fraud against the organization where he/she works on (Phua et al. 2005). This one can be divided into two categories: the high level fraud, that occurs when the fraudulent individual belongs to the management body and the low level fraud when he/she is not from management body (Chen and Gangopadhyay, 2013). In other hand, external fraud happens when the fraudster does not belong to the organism that suffered the attack.

However, the upsurge of financial scandals in recent years raised awareness of deep-seated fraudulent activities (Kerr and Murthy, 2013), such as:

Business rates fraud is characterized by avoiding paying the charge that some business has to pay for the local services. It can be performed by several ways but the most common are not declaring the precise location of some business and declare that some building is not being used when it is.

Charitable fraud consists in of misleading people in order to obtain money from them, by giving the false belief that they are contributing to charity funds (Oana, 2018). This particular type of fraud happens quite commonly mostly because of the lax laws that regards to charity.

Corruption, according to OECD Observer, "is talked about in most countries these days and few countries deny they suffer from it. It is defined by the abuse of public office for private gain."

Credit card fraud occurs when fraudsters obtain someone's else credit card for their personal use. According to Maes et al (2002) it evolves several techniques, such as the fraudsters holds the secret pin of the credit card or when the seller charges more money to the costumer without their awareness.

Identity theft is a kind of fraud that had been growing mainly due to the technologies' advance. It is characterized by an actor having access to key pieces of someone's else personal information and using it to impersonate (commit theft and/or misuse) the person that was stolen (Lacey et al, 2016).

Insurance fraud can be performed in many ways. According to Chen (2020), it involves any misuse of insurance policies in order to illegally benefit from it. The majority of the cases involve exaggerated or false claims.

Money laundering has as main goal to hide the illegal source of money, such as drug trafficking or terrorist activity and it is process of including it in the economy, through legitimate sources (Masciandaro, 1999).

Mortgage fraud is the white-collar crime that is growing in the most rapid way and it is performed by falsification of mortgage documents (Carswell and Bachtel, 2009). This rapid increasing is due to the fact that in short periods of time, people can have access to large amount of money.

Payroll fraud is more common in small businesses because there is a more familiar environment and usually, less controlled due to the lack of resources (Smith et al., 2013). Here, the fraudulent actor theft from an organization taking advantage of the payroll processing system. There are several acts considered payroll fraud such as the employee claim of hours that he/she did not work.

Social care fraud is a type of fraud that has been growing in the recent years. It occurs when some individual that receives a social care service is dishonest about his/her financial situation (Newham London, 2019). As most part of fraud types, this one specifically can also be performed by several ways such as keeping the money which purpose was to pay a care, being dishonest about the financial situation to ask for social care support and taking advantage of their own position to take funds from a person with actual needs.

Telecommunication fraud has as main goal to get access to services that the owner of the contract/mobile phone had and/or use it to illegal purposes. This type of fraud can be associated to communications fraud, where we can find up to 200 fraud variants (Tsung et al, 2007).

People say that these crimes, when compared to street crimes, are victimless. However, there is considerable evidence indicating that fraud victimization continues to increase, and that losses experienced by victims of fraud greatly exceed those of street crime (Holtfreter, 2014). These crimes are largely pre-planned and, actually, the victims are communities, societies and even states and economies.

In recent years, technology was subject of huge investments, mainly in banks and financial institutions (Pouramirarsalani et al. 2017). This investment allowed to provide more products and services and more sophisticated and updated ones.

The financial innovation, according to Wolf (2009) also allowed the increase of the connected intervenients in a network, facilitating fraudulent acts. As a matter of fact, crimes that regards to technologies have twice more possibility to happen that the other economic crimes. At the same time, we can say that fraud and economic crimes became more complex due to the globalization and technologies' advance associated to the financial systems. Transactions and mobility around all the world are now allowed and the possibility of transferring large amounts of money with a simple click on our mobile phone or computer are simple examples of actions that we take every day that might turn us into a fraud victim.

As so, we can say that, in this topic, technology is a double-edged sword (Rana, 2019) : many people are getting more sophisticated in their goals and in their methods and use it

to commit fraud and, as so, is a threat and, in other hand, it is, indeed, used to solve many of our day to day problems but, at the same time and it can be a powerful protector if we use it to build more robust systems.

However, despite what was mentioned above, we can say that a set of conditions and motivations can lead to fraud. Donald Cressey (1953) presented three factors that we can find in every fraudulent act (*Pressure*, *Rationalization* and *Opportunity*) and named it the *Fraud Triangle.*

Pressure represents opportunity to carry out the fraud, or attitude/rationalization to justify fraudulent action (Lou and Wang, 2009). It might be originated by multiple reasons such as high pressure for achievements, risk of losing the job, decreasing of salaries but it is always related to the need to have money. The social and economic status nowadays are components with high importance and, for some people, the fear of losing it sometimes dictate the act of committing fraud. At the same time, when times are hard and people do not have strong ethical standards, people are more willing to take risks and the probability of take a fraudulent act increases.

Rationalization is the mindset that the fraudulent actor uses to justify the action of committing the fraud. In other words, it is the justification or excuse that conduct to fraud (Abdullahi and Mansor, 2015). The actor might claim that his/her company is robust and will not be affected by his/her action, that they can claim financial difficulties, the impossibility to reach the goals proposed or even revenge.

Opportunity is the situation that allows the occurrence of the fraudulent act. Usually the fraud occurs when the controls are weak (Abdullahi and Mansor, 2015) and in stress periods, those times increase.

In this analysis, it is important to mention that the most critical factor is the human behaviour. That happens because opportunities to commit fraud can be influenced and managed. However, the main challenge is in managing the rationalization and the pressure since those are inherent to our mindset. But, to break the possibility of committing fraud, it is just needed to remove one of the triangle's components and, for that, the companies' board can contribute a lot.

However, as companies grow, there is a separation between the ownership and management bodies but the duty to protect the companies' interest instead of the owner's remains and, for that, the board must avoid conflict of interest as well as put at first the companies' interest. Poor ethical leadership, lack of personal integrity, mismanagement, fraud, corruption and violation of corporate governance rules are the main contributors towards bankruptcy and financial failures in large organizations (Lutui and Ahokovi, 2017). Many times, compliance and ethics are treated by different bodies from the remaining functions. Anyway, the link between ownership/governance and management is strong but the ethics and/or social responsibility must be always present. Actually, when the financial costs of fraud take large proportions, it is natural that some explanations are required from the management bodies and that is just the beginning.

According to Lavion (2018), until some time ago, the prevention and detection of fraud used to be a second action plan but, today the companies are reinforcing it, leading it to their first line of defence. To do that, companies are increasing the use of powerful technology to fight fraud, what is, actually, a worldwide phenomenon.

More and more organizations invest in new and more sophisticated technologies so they can protect themselves from fraud as well as monitoring, analysing, predict and learn from it (Lavion, 2018).

2.2. Network Science

Every year, with the advance of technology and the changing of the way that human being lives, a huge number of new practices and concepts appears, and the global markets get more complex.

However, the spread of these new practices and concepts through a society, accordingly to Namatame and Chen (2016), depends to a large extent of the fact that people influence each other. As so, social diffusion started to have impact in the real world and with the new technologies, that diffusion is also verified via networks – social networks. These networks are receiving more attention and nowadays its investigation is crucial.

Every day, everyone deals with networks (Internet, or social media such as Facebook, or LinkedIn, for example) and here, a link between the users is created. There are several types of links, depending on the type of network evolved. We can find links of knowledge transfer, collaboration, exchange information, among others. Those links, as well as the network structure, will have impact on the diffusion process and that is why the investigation of the influence mechanisms becomes so important.

2.2.1 Network Measures

In order to make a proper analysis of a network, the first step that need to be taken into account is the analysis of its characteristics. However, we should always have in consideration that these are not absolute measures since the values obtained does not lead us to a concrete conclusion. In other words, we need to make a comparison to another network to compare the measures and make those values meaningful.

Assortativity coefficient

Also known as Pearson's correlation coefficient, it allows us to analyse the tendency of connection between the individuals with the same magnitude.

We can calculate it through the expression

$$r = \frac{\sum_{jk} jk\left(e_{j,k} - q_j q_k\right)}{\sigma_q^2}$$

(Newman, 2002), where $e_{j,k}$ represents the joint probability distribution of the excess degrees of the two intervenients, j and k, $q_k$is the distribution of the remaining degree and σ_q represents the standard deviation of the distribution q_k.

In this analysis, we can obtain a range of values from -1 to +1. When assortativity coefficient takes negative values, it means that large degree internenients will tend to be linked to low degree intervenient or, if we prefer, an intervenient will link to other with different degree. On the other hand, if this coefficient takes positive values, this indicates

that an intervenient tends to connect to others with similar degree. But if $r = 0$, it means that the connection will be made in a random way. In the extreme cases, where the coefficient is -1 (perfect disassortativity) or +1 (perfect assortativity), it means that all internenients will connect only with others with a different degree or with the same degree, respectively.

Centrality measures

One of the most fundamental network measures for nodes are those that try to capture the node's centrality (Buckley and Harary, 1989). In centrality measures, instead of assortativity that analyses the network as a whole, we are able to focus on individual internenients.

Degree centrality

It is the simplest centrality measure and it aims to record the number of connections of each intervenient (Zhou et al., 2017).

Betweenness centrality

Betweenness centrality quantifies the number of times a node acts as a bridge along the shortest path between two other nodes. It was introduced as a measure for quantifying the control of a human on the communication between other humans in a social network (Freeman, L. 1977)

The betweenness centrality of a node k can be calculated as follows (Costa et al., 2017):

$$BC(k) = \sum_{i \neq k \neq j} \frac{\sigma_{ij}(k)}{\sigma_{ij}}$$

where σ_{ij} is the total number of shortest paths from node i to node j and $\sigma_{ij}(k)$is the number of those paths that effectively pass through k. The information $i \neq k \neq j$, indicates that betweenness is not influenced by the direct links between the nodes.

Betweenness centrality can take values from 0 to 1.

Closeness centrality

It measures how close a node is to all the other nodes in a network (Okamoto et al., 2008). In practical terms, the more central node is, the closer it is to all other ones.

The following formula allows to obtain the closeness centrality (Opsahl et al., 2010):

$$CC(i) = \left[\sum_{j}^{N} d(i,j)\right]^{-1}$$

where $d(i,j)$ represents the distance between nodes i and j.

Density

Density represents the proportion of observed connections in a certain network to the maximum number of possible connections (Faust, 2006).

Diameter

This parameter evaluates the compactness of a network through its size and degree of interconnectedness. According to Luke (2015), a path is the series of steps required to go from node A to node B in a network. The shortest path is the shortest number of steps required. The diameter then for an entire network is the longest of the shortest paths across all pairs of nodes.

With this measure, we can evaluate the network efficiency because the diameter reflects the worst case possible for sending the information across the network.

2.2.2 Evolution and dynamic

With the statistics described before, we are able understand the structure of a network. However, in order to evaluate the contribution of a specific intervenient in a network, it becomes important to analyse the interplay between its structure and dynamic. Moreover, the most part of current networks are not static, but dynamic. The latter are modelled taking into account continuous states due to the changes of its topology over times. Besides that, its intervenients may come and go.

The impact of individual agents in the global performance of the systems inevitably depends on both the network topology and the specificities of the dynamics. (Namatame and Chen, 2016).

Namatame and Chen (2016) considered a model where it was assumed that some intervenient was influenced directly only by a small group of intervenients to whom he/she is connected.

In a binary model, where an individual is active ($x_i = 1$) or inactive ($x_i = 0$), considering that in the first time considered he/she was inactive, from one time to another, he/she changes their state based only on the adjacent agents who were active in the first time considered: if at the first time considered, the fraction of active neighbours is higher than the intervenient's threshold (ϕ), the inactive intervenient will become active.

The state of the next time (time $(t+1)$) can be achieved by the following:

$$x_i(t+1) = \begin{cases} 1\,, & \frac{\sum_{j \in N_i} x_j(t)}{k_i} \geq \phi \\ 0, & \frac{\sum_{j \in N_i} x_j(t)}{k_i} < \phi \end{cases},$$

where N_i represents the set of connected intervenients to i, k_i is the size of N_i and x_j is the state od an adjacent agent.

Namatame and Chen (2016) also presented an example where we can clearly see the importance of topology in a network's dynamic. It was considered a non-progressive cascade where, over time, an intervenient can change his/her stage to active or inactive, depending on the state that he/she is at the first time considered, based only on the states of their neighbours.

Given some *initiators* (initial set of active intervenients), the process of changing state is given in discrete steps: in step t, all intervenients that at step $t-1$ were active, remain active and it is activated any agent whose the fraction of neighbours is at least his/her threshold.

The network threshold (ϕ^*) corresponds to a phase of transition between the two processes described above and it highly related to the network topology.

2.3. Algorithms for Fraud Detection

Nowadays, more and more researchers try to extract knowledge out of data. That is why data mining is becoming so popular and had become one of the most powerful tools used for data analysis (Pouramirarsalani et al. 2017).

In this sub-chapter, we are going to study some algorithms used to detect and prevent fraud.

In what regards fraud detection, it is a set of operations and methods which the main goal is to detect the fraud that occurred and are still going to occur. Under a Data Mining perspective , regression and neural networks are used frequently. Fraud prevention aims to avoid the occurrence of fraud.

From the fraud types described in 2.2, all of them can be included in a specific area: credit card fraud, telecommunication fraud and healthcare insurance fraud.

In what regards to credit card fraud detection, Sherly and Nedunchezhian (2010), Malekian and Hashemi (2013) and Chandola et al. (2009) were some of the authors that studied this topic.

Sherly and Nedunchezhian (2010) built a system capable of detecting fraud by adapting it to the behaviour changes. This behaviour includes data regarding transaction amounts, category of the items, purchase adress, among others. For that, two steps are taken. In the first one, the system compares the incoming transaction to the consumer's history and, through BOAT algorithm, it is detected (or not) any anomaly. The goal of the second step,

is to reduce the anomalies marked as fraud. In this way, the fraud would be detected by analysing automatically the consumer's behaviour.

Malekian and Hashemi (2013), using a learning algorithm, studied the dynamic and non-stationary behaviour. In other words, it was introduced in the system a temporary profile and the algorithm is capable of, regardless the historical information, retain new concepts from the incoming data.

Another author that studied the fraud detection in credit card though profiling was Chandola et al. (2009). The authors used two approaches: owner approach and operation approach. The first one is characterized by comparing the incoming data with the credit card history and if both data do not match, the transaction is flagged as an anomaly. In other hand, operation approach is based on geographic location: if the location of the incoming transaction does not match with the profile, it is flagged as an anomaly.

Telecommunications fraud is another topic very used to study the fraud detection. Farvaresh and Sepehri (2011) used a CDR (Call Detail Record) method. This way it is possible to extract the profile of the user and detect any anomaly.

Sanver and Karahoca (2009) compared a set of different fraud detection techniques to find the best solution and came up with an approach by their own, using Adaptative Neuro Fuzzy Inference (ANFIS), that is a hybrid algorithm. The first step is, to have information regarding the mean and standard deviation of the membership functions (rule parameters) and, through ANFIS, these parameters are recursively updated until we reach an acceptable error. After this, the algorithm creates, for input variables, a membership function (Cost of the Call, Duration, …). A curve with all the information for each membership function is set and, if the data in analysis has an effect over the average or, in other words, is a deviation from the curve, it is an anomaly and will eventually be flagged as fraud. One of the advantages of this method is that it allows to reduce the complexity of the system and the efficiency of the training time is higher.

Because healthcare insurance is a very complex and confusing topic to many people, it is another topic that had been studied by many authors. There are many approaches, but the most recent studies focus fraud detection via Regression classification (Support vector machine and Rule Based) as well as statistical method.

Francis et al. (2011) focused their analysis in Support Vector Machines (SVM), where they analyse the efficiency of Quash, a bill processing system. With this study, it was possible to accelerate the time of detection fraud to the real time.

In a different approach, Tsai et al. (2014), detected an anomaly in the system and started their study from there. They claimed that the information is stored in heterogeneous databases. As so, they used knowledge of methodology CommonKADS, that develops a comprehension of the system. With this, the fraud detection will be easier and quicker and the labour costs will be reduced.

Using a statistical method, Lu and Boritz (2005) focus their analysis in Benford's Law. This law, according to Benford (1938), specifies the probabilistic distribution of digits for many commonly occurring phenomena, ideally when we have complete data of the phenomena. Lu andBoritz (2005), claim that the most common approaches are the ones that regards to supervised learning methods, that are characterized by training systems from a set of fraudulent data and then, from a test set, detect the patters studied before. The authors chose a different approach: the training process was done from a non-fraudulent set and then, it was compared to the test data. Any result that was not similar to the test data, could be a fraudulent case.

As known, fraud is considered an anomaly and, recently, as Tsai et al. (2014) did, lots of studies investigate the anomaly detection, mainly in graphs.

Jeh and Widom (2002), to detect anomalies, proposed an approach that is applicable to any object-to-object relationships domain. Basing on the object's relationship with other objects, the authors focused in measuring the similarity of the structural context where the object occurs. As the authors state, they compute a measure that says "two objects are similar if they are related to similar objects".

Another graph-based anomaly detection technique was proposed by Noble and Cook (2003), using two methods. The first one consists in detecting anomalous substructure analysing unusual substructures in a graph and, in the second one, the authors partitioned the graphs into several sets of subgraphs and tested each one of them against the others, looking for unusual patterns.

Xu et al. (2007) worked on Structural Clustering Algorithm for Networks (SCAN) in order to detect clusters and outliers in networks, using structural similarity measures. To detect clusters vertices they use common neighbours and two vertices are assigned to a cluster depending on their neighbours.

Akoglu et al (2010) showed how the discovery of some rules regarding density, weights, ranks and eigenvalues can help the anomaly detection. They also focused on questions such as “what features should we use to characterize a neighbourhood?” and “what does a “normal” neighbourhood look like?”. OddBall, the algorithm they created, detects anomalies in weighted, unlabelled graphs.

Gupta et al. (2012) studied anomaly detection in temporal datasets, trying to identify nodes that have different evolutionary behaviour when compared to other nodes. They focused on identifying evolutionary community outliers using two snapshots of a dataset, with the objective function to minimize community matching error where the outlier node weigh is low.

Another example of authors that studied anomaly detection was Yan & Han (2002), that developed an algorithm, gSpan, that is capable of discover frequent substructures. In other words, gSpan allows to detect frequent patterns of connectivity within a subgraph building a new lexicographic order and, in each graph it is associated to an unique Depth First Search code.

The studies regarding fraud detection in the financial system itself also had been more popular.

Among them, we can mention Brito et al. (2018), that built a model (BOND) capable of predict the connections between individuals. The goal of the authors is to, for a specific intervenient, detect the fraudulent transactions and, then, a using classification algorithm, find out if that intervenient is willing to commit fraud or not.

Pironet et al. (2009) developed a method capable of identify fraudulent patters in the social networks. The first step is to detect patterns in the social network and, gathering that information with the VAT declarations, they get rich information regarding the companies, however, it is still unbalanced. After balancing the data using boosting

algorithms the authors were able to improve the fraud detection classifiers for both singular individuals and companies.

Table 1 Algorithms for Fraud Detection

Author, year	Area	Topic	Goal	Methodology
Pouramir arsalani et al. (2017)	Geral	Fraud detection in E-banking by using the hybrid feature selection and evolutionary algorithms	To provide a new method to fraud detection in e-banking.	Used a hybrid feature selection and generic algorithm.
Jeh and Widom (2002)	Anomaly detection	SimRank: A Measure of Structural-Context Similarity	To measure the similarity of structural context of an object	Computing a measure, two objects are similar if they are related to similar objects
Noble and Cook (2003)		Graph-based anomaly detection	To detect unusual patterns in graph-based data.	The authors created a measure that calculates the regularity of a graph, using conditional entropy
Xu et al. (2007)		SCAN: A Structural Clustering Algorithm for Networks	To develop a method capable of detecting clusters based on individual's common neighbours	The individuals are assigned to the same cluster depending on how they share the neighbours
Akoglu et al (2010)		OddBall: Spotting anomalies in weighted graphs	To build an algorithm that is scalable and work for un-supervisioned data for anomaly detection	Through some discovered rules in density, weights, ranks and egonets, the authors show how to detect anomaly in graphs
Gupta et al. (2012)		Integrating community matching and outlier detection for mining evolutionary community outliers	To detect evolutionary community of outliers	Given two consecutive snapshots,, the authors focused in studying the evolving dataset
Yan and Han (2002)		gSpan: graph-based substructure pattern mining	To find frequent substructures within a graph	The authors adopted a DFS strategy to find patterns of connectivity in subgraphs
Brito et al. (2018)	Financial system	An Agent-Based Model for Fraud Detection in Economic Networks	Detect the fraudulent transactions and, then, a using classification algorithm, find out if that intervenient is willing to commit fraud or not.	Using machine learning algorithms, the authors aggregated the transactions per intervenient and through the BOND model were able to predict the transactions between them that allowed to find out if the intervenients were fraud prone.
Pironet et al. (2009)		Classification for Fraud Detection with Social Network Analysis	Develop a method capable of identify fraudulent patters in the social networks.	The first step is to detect patterns in the social network and, gathering that information with the VAT declarations it is possible improve the fraud detection classifiers.

Author, year	Area	Topic	Goal	Methodology
Sherly and Nedunchezhian (2010)	Credit card fraud detection	BOAT adaptive credit card fraud detection system.	To build a system capable of detecting fraud through behaviour analysis.	Compare income behaviour with historical information and using BOAT algorithm, anomalies are detected and identified (or not) as fraud. Also by combining classification and clustering techniques.
Malekian and Hashemi, (2013)		An Adaptive Profile based Fraud Detection Framework For Handling Concept Drift.	Study the dynamic and non-stationary behaviour.	It is introduced in the system a temporary profile and the algorithm is retain new concepts from the incoming data, regardless the historical information.
Chandola et al., (2009)		Anomaly detection: a survey.	Overview of the study of anomaly detection.	It was used a two approaches technique: owner and operation approach.
Farvaresh and Sepehri, (2011)	Telecommunication fraud detection	A data mining framework for detecting subscription fraud in telecommunication	Study and identify customers' subscription fraud.	Different data mining techniques were used. Also a hybrid approach was considered, using, for that, preprocessing, clustering, and classification.
Sanver and Karahoca, (2009)		Fraud Detection Using an Adaptive Neuro-Fuzzy Inference System in Mobile Telecommunication Networks	Compare different techniques to detect fraud in telecommunications sector and offer another efficient method.	Using Adaptive Neuro Fuzzy Inference it is calculated a set of parameters with the existent data that latter are going to be used to trace a curve and analyse if the income data is (or not) a deviation from it, being (or not) flagged as fraud.
Francis et al., (2011)	Healthcare insurance fraud	Using support vector machines to detect medical fraud and abuse	Analyse the efficiency of Quash, a bill processing system.	Using support vector machine, the authors accelerated the time of detection of fraud.
Tsai et al., (2014)		Using CommonKADS method to build prototype system in medical insurance fraud detection	Improve the inefficiency of the healthcare system.	Using a process which the main goal is to have a comprehension of the system (CommonKADS)
Lu and Boritz (2005)		Detecting Fraud in Health Insurance Data: Learning to Model Incomplete Benford's Law Distributions	To detect fraud using technique that uses an unsupervised learning approach to handle incomplete data.	With a digital analysis, the training was done from a non-fraudulent set and then, it was compared to the test data.

3. Problem Description and Methodology

As previously mentioned, the main goal of this study it to analyse the impact of network measures in fraud detection. As so, it is intended to present an effective way of detecting fraud, bearing in mind the network relationship of the customers and, eventually, preventing it.

In this study, the predictive attribute is the presence or absence of fraudulent behaviour associated to the nodes of a network.

As so, having into consideration the information provided in the data set (see description below), three classifiers are going to be analysed: Decision Tree, Logistic Regression and Support Vector Machine. The goal is to test if these classifiers correctly classify the transactions as fraudulent/non-fraudulent.

Additional features are going to be included, such as the cumulative sum of the transactions (for both origin and destination users), the weight of the user, based on the proportion between the amount associated to fraudulent transactions of that user and the amount of transactions done by the user so far and the type of the user (for both origin and destination), given by the first letter of the users' ID. Regarding network measures, it is also going to be calculated the betweenness and the density for each step, cumulatively and the predictive capacity of the classifiers are once again be tested and compared.

At each step, the number of nodes connected to fraudulent behaviour within the network will also be calculated. With this measure, we can study not only the number of fraudulent transactions, but the suspicious connections between the nodes.

To compare the performance of the different classifiers, the metrics used were Precision, Recall and Area Under Curve (AUC).

Precision

According to Powers (2011), precision is a measure of accuracy of predicted positive cases in contrast with the rate of discovery of real positives. As so, it can be obtained dividing the true positives by the sum of the true positives with false positives.

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

Precision is a measure of exactness (Gama, 2010) and when it is equal to 1, it means that all items labelled as belonging to a certain class, indeed belong to that class.

Recall

Recall (or sensitivity) is the proportion of real positive cases that are correctly predicted positive, accordingly to Powers (2011):

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

Recall, as Gama (2010) states, is a measure of completeness and when it takes the value 1, it means that all items that belong to certain class, were labelled as belonging to it.

Both precision and recall focus only on the positive examples and predictions and the best prediction would lead to 100% precision and 100% recall.

Area Under Curve (AUC)

The AUC, according to Cortes and Mohri (2003), is a criterion used in many applications to measure the quality of a classification algorithm.

In this study, it determines the inherent hability of the algorithm to discriminate between the fraudulent and non-fraudulent transactions (Hajian-Tilaki, 2013) and it is obtained using the Receiver Operating Characteristic (ROC), where it is possible to see the trade-off between sensitivity (True Positive Rate) and specificity (1- False Positive Rate).

AUC can take values between 0,5 and 1, and the closest to 1 the value is, the better, as the average AUC increases as a function of the classification accuracy.

To perform the tests, it was obtained, through *PaySim*[2] simulator, a synthetic dataset with information regarding money transactions for a period of one month.

[2] Data available at: https://www.kaggle.com/ntnu-testimon/paysim1

This dataset contains the following information regarding 6362620 observations:

- *step*: maps a unit of time in the real world. If the step is equal to 1, it means that the observation occurred in the first hour;
- *type:* cash-in, cash-out, debit, payment or transfer;
- *amount*: amount of the transaction in local currency;
- *nameOrig*: origin user of the transaction;
- *oldbalanceOrg*: balance before the transaction;
- *newbalanceOrig:* balance after the transaction;
- *nameDest:* customer who is the recipient of the transaction;
- *oldbalanceDest:* recipient's balance before the transaction (note: there is not information for customers that start with M (Merchants));
- *newbalanceDest:* recipient's balance after the transaction (note: there is not information for customers that start with M (Merchants);
- *isFraud:* indicates if the transaction was made by the fraudulent agents inside the simulation. There is fraudulent behaviour when the agent aims to empty the funds by transferring to another account and then cashing out of the system;
- *isFlaggedFraud:* an illegal attempt in this dataset is an attempt to transfer more than 200.000 in a single transaction.

Example:

step	type	amount	nameOrig	oldbalanceOrg	newbalanceOrig	nameDest	oldbalanceDest	newbalanceDest	isFraud	isFlaggedFraud
1	PAYMENT	9839.64	C1231006815	170136.0	160296.36	M1979787155	0.0	0.0	0	0

The transaction above represents a payment of 9839,64 monetary units done by the customer C1231006815 to the merchant M1979787155, at the first hour of timestamp analysed. In this particular case, it is possible to see the origins' balance before and after the transaction but those details, for this particular transaction, are not available for the destination.

In order to test the algorithms and produce results, the system used was R (R version 3.6.0 (2019-04-26)). However, due to computational restrictions[3], it was only possible to work with a sample of approximately 11% of the *PaySim* dataset.

[3] To produce results, it was used an ideapad 720S-14IKB, Intel® Core™ i7-7500U CPI with 8GB of RAM.

4. Results

In this chapter, three classifiers (Decision Tree, Logistic Regression and Support Vector Machine (SVM)) are going to be analysed and compared using three metrics (Precision, Recall and Area Under Curve).

The results are going to be obtained for different scenarios: 1) with the information provided by *PaySim*; 2) added to initially provided information the cumulative sum of the transactions (for both origin and destination users), the weight of the user, based on the proportion between the amount associated to fraudulent transactions of that user and the amount of transactions done by the user so far and the type of the user (for both origin and destination), given by the first letter of the users' ID; 3) added two network measures (betweenness and density) to the initially provided information; 4) combined the scenarios 1), 2) and 3).

As mentioned, the main goal of this study is to evaluate the impact of network measures in fraud detection.

In order to test the algorithms and produce results, the system used was R (R Core Team , 2019).

The information was stored as a CSV. The first step is to load the data and to order it from the oldest to the most recent transaction.

After, the dataset must be divided in order to obtain a subset for training and other for testing. The number of records per days and per hours are very different so, we chose to select, for training, the 70% of records represented by the oldest observations and the 30% more recent are for testing. In our data, the days are divided into hours and, according to the split just mentioned, an hour would be divided (step 323). As the most part of the information regarding this hour would be allocated to the training set, we chose to include all the information on it.

After this split, it is going to be presented two approaches: one based on classifiers and other one based on networks.

Due to computational restrictions, not all the information could be used to train and to test. As so, and because the variable we want to predict is very unbalanced (in the total

data set there are 8213 fraudulent transactions against 6354407 non fraudulent) the sample was selected after the data was divided into train and test, as follows:

- Training sample: all fraudulent transactions from training were selected and 11% of the non-fraudulent transactions were chosen randomly.
- Testing sample: 11% of each steps' transactions were chosen randomly.

4.1. Classifiers

With the sample selected, the first step is to predict the dependent variable (isFraud) using different algorithms. For that purpose, it was used Decision Tree, Logistic Regression and Support Vector Machine (SVM).

Decision Tree

To obtain a decision tree, the *rpart* package (Therneau and Atkinson, 2019) has to be used. After loading the package, the variable has to be converted into binary.

Here is important to mention that the information regarding the users' name was not considered because, as ID type, it does not contain relevant information to build a decision tree.

The next step is to evaluate the predictive capacity of the algorithm, using the testing set:

The predicted values obtained showed decimal numbers and, in order to obtain the confusion matrix, we need to convert them into binary.

In order to compare the performance of the different classifiers, the metrics to be analysed are Precision, Recall and AUC. The first two are extracted from confusion matrix using the fields *Pos Pred Value* and *Sensitivity* respectively.

```
Confusion Matrix and Statistics


predict_tree      0      1
           0 208352    166
           1      8    384


               Accuracy : 0.9992
                 95% CI : (0.999, 0.9993)
    No Information Rate : 0.9974
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.8149

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.698182
            Specificity : 0.999962
         Pos Pred Value : 0.979592
         Neg Pred Value : 0.999204
             Prevalence : 0.002633
         Detection Rate : 0.001838
   Detection Prevalence : 0.001876
      Balanced Accuracy : 0.849072

       'Positive' Class : 1
```

Figure 1: Confusion Matrix of Decision Tree with original information

Per the figure above, we can conclude that, for the Decision Tree classifier, using the information provided in the dataset, the Precision is 0,979592 and the Recall is 0,698182.

We can also see that the prediction of the Decision Tree classified correctly 208352 observations as non-fraudulent and 384 as fraud. However, the classifier pointed 8 transactions as fraudulent while they are not associated to fraud and 166 as not associated to fraud while they are fraudulent.

To obtain information regarding the AUC, it is going to be calculated the ROC curve, using the package ROSE (Kuhn, 2020). The AUC displayed is 0,849.

In the figure below we can see the ROC curve where the AUC was obtained from.

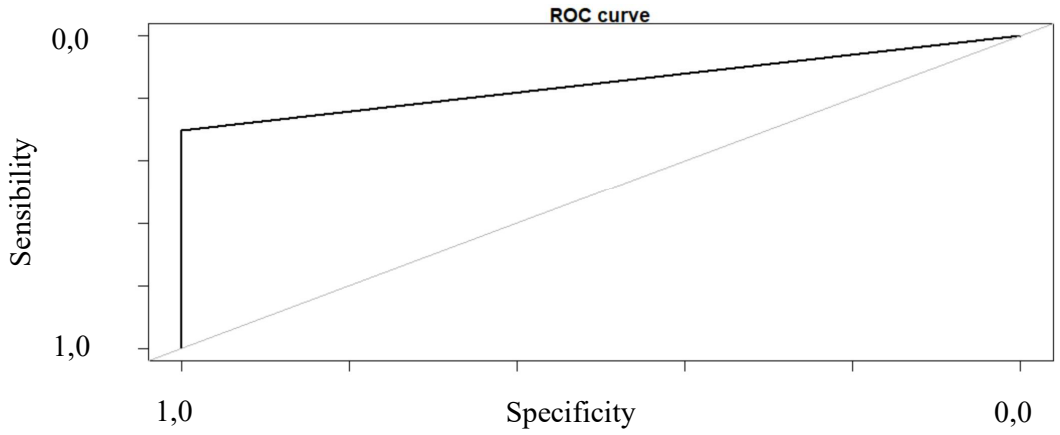

Figure 2: ROC Curve of Decision Tree with original information

Logistic Regression

The next classifier to be studied is logistic regression.

Similar to what was described to decision tree, ID variables were not considered to obtain the logistic regression.

After obtaining the regression and predict it for the testing sample, the values obtained were not in the range that the variable in analysis can take (0s and 1s). As so, the result has to be converted into matrix (to obtain the confusion matrix) and amend the values to the correct range. After, the information is ready to generate the confusion matrix and the Precision is 0,014945 and the Recall is 0,981818.

Also, the area under curve obtained was 0,905.

Support Vector Machine (SVM)

The last classifier to be used is SVM and, to train the algorithm, it is needed the *e1071* package (Meyer et al., 2019).

One more time, the information regarding IDs was removed and, after training the classifier, the next step is to test and evaluate its performance.

The results shown a Precision of 0,96831, a Recall of 0,5 and an AUC of 0,75.

After obtaining the results for the three classifiers with the sample of the initially provided information, additional numerical and categorical features were included in order to test the improvement of the predictive capacity of the algorithms. Those are: the cumulative sum of the transactions (for both origin and destination users), the weight of the user, based on the proportion between the amount associated to fraudulent transactions of that user and the amount of transactions done by the user so far and the type of the user (for both origin and destination), given by the first letter of the users' ID (C is regarding Clients and M stands for Merchants).

At this point, is important to mention that the additional numeric features (cumulative sum of the transactions and weight of the user) had to be calculated based on the past information due to the fact that we are dealing with temporal information: at each transaction, we don't have access to the information regarding the ones after, meaning that if we consider all the transactions, we could have an overfitting problem.

The addition of the new features was done in one dataset, where training and testing samples were merged. After the process was completed, the subsets were divided again maintaining the division initially defined in order to proceed with the analysis. For that, the package *dplyr* (Wickham, 2020) was used.

Once the features are included in the sample, the next step is to evaluate the classifiers studied before to check if its performance was improved.

After a quick analysis of the features created, we can see that the weight of the origin is always 0. This happens because there are 208900 unique origin users and from the 10 repeated, none is associated to fraud.

We can also see that all the origin types are "Cs".

As so, for the next analysis, these two features are not going to be considered as they do not contain useful information.

Decision Tree

The logic behind obtaining the decision tree is the same as before but now we include the new features.

To evaluate this classifier, the Precision and Recall were calculated and are 0,979592 and 0,698182 respectively. Also, an AUC of 0,849 was obtained.

Logistic Regression

Adding the new features as independent variables, we obtain a new logistic regression.

Using the same logic as before, it is possible to obtain a Precision of 0,016317, a Recall of 0,934545 and an AUC of 0,893.

Support Vector Machine (SVM)

The last classifier to be obtained is SVM and it was used the same logic.

After, a Precision of 0,968085, a Recall of 0,496364 and an AUC of 0,748 were obtained.

Below, we can find a summary of the analysed metrics regarding the three classifiers (with and without the additional features):

Table 2 Classifiers results with additional features

		Original information	Additional features
Decision Tree	Precision	0,979592	0,979592
	Recall	0,698182	0,698182
	AUC	0,849	0,849
Logistic Regression	Precision	0,014945	0,016317
	Recall	0,981818	0,934545
	AUC	0,905	0,893
SVM	Precision	0,96831	0,968085
	Recall	0,5	0,496364
	AUC	0,75	0,748

As we can see, the inclusion of additional features allowed to have improvements regarding the predictive capacity of logistic regression. It happens mainly because of the

increase of the precision: the classifier, in comparison to the one with no additional information, was able to correctly predict more positive cases out of the cases the classifier predicts positive.

In what regards to SVM, it is possible to see a generalized decrease of its predictive capacity.

Finally, in what regards to the decision tree, no improvement was detected, however, it is still the classifier with the best performance.

4.2. Network measures

After evaluating the performance of some classifiers, we focused on the network analysis.

The first approach was to include some network measures and evaluate the classifiers previously presented.

As mentioned before, we are dealing with temporal information and, because of that, to crate and evaluate the algorithms, it cannot be considered all the information at once. As so, we created cumulative networks, where each one includes the previous. For example, the network regarding step 3 includes the origin and destination nodes of step 1, 2 and 3.

At each step, the transactions with users that committed fraud are going to be flagged as contaminated and the betwenness and density of the network were calculated.

At the end, we obtain the number of contaminated transactions for each step, as well as the betweenness and density of it.

Per the results, we can see that the betweenness takes values between 0,14 and 0,34 and the density can vary between 0 and 0,02. Regarding the density, it is possible to say that, at the most connected network, only 0,2% of the total possible connections are effectively stablished.

In what regards the contaminated transactions, the logic is as follows: if a user committed fraud, all the transactions where that user participated are going to be contaminated. For example, in step 1, even though there are 326 observations, only 7 are fraudulent.

However, in that step, the transactions contaminated were 17, meaning that those 7 fraudulent transactions were somehow connected to a total of 17. As this process is done in a cumulative way, the number of contaminated transactions is always increasing and, at the end, the total number of contaminated transactions was 9212.

For each transaction of our sample, depending on the step that we are analysing, it is going to be retrieved the respective betwenneess and density.

The next step is to evaluate the predictive capacity of the model with the additional network features.

The logic behind obtaining the classifiers is the same as before but now, the independent variables to be considered are *step, type, amount, oldbalanceOrg, newbalanceOrig, oldbalanceDest, newbalanceDest, Bet* and *Dens*.

Below, is a summary table of the metrics obtained:

Table 3 Classifiers results with network measures

		Original information	Network measures
Decision Tree	Precision	0,979592	0,979592
	Recall	0,698182	0,698182
	AUC	0,849	0,849
Logistic Regression	Precision	0,014945	0,915152
	Recall	0,981818	0,549091
	AUC	0,905	0,774
SVM	Precision	0,96831	0,966667
	Recall	0,5	0,421818
	AUC	0,75	0,711

Similar to what happened before, the performance of the decision tree did not change and we also observe a slightly deterioration in all the metrics in what regards to SVM.

Here, the classifier that stands out is the logistic regression. Despite the fact of the deterioration of recall and AUC, the precision improved in a larger scale. Usually, between recall and precision there's a trade-off but the improvement of the precision is larger than the decrease of the recall.

To conclude, it is necessary to obtain the metrics with all the features previously proposed: the original features provided with the dataset, the features proposed on 3.1 and the network measures. That must be done after combining all the information and obtain the training and testing subsets.

The independent variables to be considered are *step, type, amount, oldbalanceOrg, newbalanceOrig, oldbalanceDest, newbalanceDest, amountOrig, amountDest, wDest, typeDest, Bet* and *Den,* where *amountOrig* and *amountDest* are the cumulative sum of the transactions for origin and destination respectively, *wDest* stands for weight of the destination, *typeDest* represents the type of the destination and *Bet* and *Dens* are the betweenness and density of the networks respectively.

Below is a summary of the results obtained:

Table 4 Classifiers results with additional features and network measures

		Original information	Additional features and network measures
Decision Tree	Precision	0,979592	0,979592
	Recall	0,698182	0,698182
	AUC	0,849	0,849
Logistic Regression	Precision	0,014945	0,915408
	Recall	0,981818	0,550909
	AUC	0,905	0,775
SVM	Precision	0,96831	0,970711
	Recall	0,5	0,421818
	AUC	0,75	0,711

Once again, the decision tree did not change its performance, but we can point some variations regarding SVM and logistic regression.

Comparing the metrics of SVM, despite the fact that the precision increased, the values of the recall and AUC decreased, leading to a global deterioration of its predictive capacity.

In what regards to logistic regression, the improvement of the precision more than compensates the deterioration of recall and AUC. As so, globally, we can say that the predictive capacity of the logistic regression improved with the inclusion of the features.

As we can see below, despite the fact that the global performance of the logistic regression increased, it still classifies 28 transactions as fraudulent while they are not and, at the same time, wrongly labels 247 as non-fraudulent.

```
Confusion Matrix and Statistics


predict_reg_features_G      0      1
                     0 208332    247
                     1     28    303


               Accuracy : 0.9987
                 95% CI : (0.9985, 0.9988)
    No Information Rate : 0.9974
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.6872

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.550909
            Specificity : 0.999866
         Pos Pred Value : 0.915408
         Neg Pred Value : 0.998816
             Prevalence : 0.002633
         Detection Rate : 0.001450
   Detection Prevalence : 0.001584
      Balanced Accuracy : 0.775387

       'Positive' Class : 1
```

Figure 3 Confusion Matrix of Logistic Regression with additional features and network measures.

5. Conclusions

As mentioned previously in this study, a huge number of new concepts appears every year, turning fraud a very dynamic topic. With this, is urgent to provide an effective way to detect fraudulent behaviour.

Overall, we can say that in recent years economic crime intensified but, at the same time, the tolerance to this matter decreased and the transparency and the rules applied are more critical because companies are aware that fraud hold back their competitive skills on a global scale and ignore the presence of fraudulent acts turned too costly (Lavion, 2018). Fraud became one of our biggest competitors we did not we had.

Along the years several fraud detection methods were presented, mainly regarding credit card fraud and telecommunication systems but, in this study, it is focused the fraudulent behaviour on daily transactions.

The main challenge of this study relied on the fact that fraud is a dynamic concept. As so, the process of including additional features had to be done in a cumulative way.

Looking at the information provided, we can say that when the user is a merchant, we have no information regarding its balance either before and after the transaction.

It is also possible to conclude that there is no fraudulent activity in what regards to payments, debits nor cash-in transactions. As so, we can say that fraudulent activity is only focused on cash-out transactions or transferences. Also, it is possible to observe that when there is a transference immediately followed by a cash-out transaction, fraudulent activity takes place.

In what regards to the amount of a transaction, it can be concluded that when it is equal to the old balance of the origin, the transaction is fraudulent.

Regarding the study performed to analyse the impact of network measures in fraud detection, three conclusions can be pointed:

- The first one is that along the process, the decision tree did not change its performance, meaning that the features and network measures, do not affect the predictive capacity of this classifier. However, this classifier is still the one with best performance;

- In what regards to SVM, the addition of the features and the network measures (both separately and combined) lead to the improvement of some metrics and the deterioration of others but, globally, we can see a deterioration of the predictive capacity of the algorithm;
- The biggest improvement was regarding logistic regression. Even though some metrics got deteriorated, others improved in a larger scale, leading to a global improvement of the performance of the classifier, where its best results were obtained when additional features and network measures were combined.

Having into consideration the results of the network measures, we can also conclude that these networks are not very dense, as the maximum value density takes is 0,2%, meaning that only 0,2% of the total possible connections are effectively stablished.

As any study, this also had some limitations and the main one was the computational capacity. As consequence, only 11% of the dataset was analysed, making the results not fully reliable. However, with the method chose to select the sample, we believe it was able to overcome that obstacle.

6. References

Abdallah, A., M. A. Maarof A. Zainal (2016), "Fraud Detection System: A Survey" in *Journal of Network and Computer Applications*, 68 pp. 90-113.

Abdullahi, R. and N. Mansor (2015) "Fraud Triangle Theory and Fraud Diamond Theory. Understanding the Convergent and Divergent For Future Research" in *International Journal of Academic Research in Accounting, Finance and Management Sciences*, Vol. 5 (4), October 2015.

Adewumi, A.O., A. A. Akinyelu, (2017), "A survey of machine-learning and nature-inspired based credit card fraud detection techniques" in *International Journal of System Assurance Engineering and Management* vol. 8, pp 937–953.

Akoglu, L., M. McGlohon and C. Faloutsos (2010) "OddBall: Spotting anomalies in weighted graphs", in *Proceedings of the 14th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD)*, Hyderabad, India, pp 410–421, 2010.

Barrenas, F., S. Chavali, P. Holme, R. Mobini and M. Benson (2009), "Network Properties of Complex Human Disease Genes Identified Through Genome-Wide Association Studies" in: *PLoS ONE*, 4(11), November 2009.

Benford, F. (1938), "The Law of Anomalous Numbers" in *Proceedings of the American Philosophical Society*, pp. 551-571.

Bolton, R. J. and D. J. Hand (2002), "Statistical fraud detection: A review" in *Statistical science*, Vol. 17, No. 3, 235–255.

Brito, J., P. Campos and R. Leite (2018), "An Agent-Based Model for Fraud Detection in Economic Networks" in *Highlights of Practical Applications of Agents, Multi-Agent Systems, and Complexity: The PAAMS Collection*, pp.105-115.

Buckley F. and F. Harary (1989), *Distances in graphs*, Addison–Wesley, Redwood City.

Carswell, A. and D. C. Bachtel (2009), "Mortgage fraud: A risk factor analysis of affected communities" in *Crime Law and Social Change*, Vol 52, pp 347-364.

Chandola, V., A. Banerjee and V. Kumar (2009), "Anomaly Detection: a Survey" in *ACM Computing Surveys*, Vol.41 (3), pp.1–58.

Chen, J. (2020), “Insurance Fraud”, https://www.investopedia.com/terms/i/insurance-fraud.asp, accessed in 30th August 2020.

Chen, S. and A. Gangopadhyay (2013), “A Novel Approach to Uncover Healthcare Frauds Through Spectral Analysis” in *Proceedings of the 2013 IEEE International Conference on Healthcare Informatics*, pp.499–504.

Cortes, C. and M. Mohri (2003), “AUC optimization vs. error rate minimization” in *Neural Information Processing Systems (NIPS)* 15, MIT Press.

Costa A., A. A. Petrenko, K. Guizien and A. M. Doglioli (2017), “On the calculation of betweenness centrality in marine connectivity studies using transfer probabilities” in *PLoS ONE* 12, available at https://doi.org/10.1371/journal.pone.0189021.

Cressey, D. R (1953), Other people's money; a study of the social psychology of embezzlement in *Free Press*.

Csardi, G., T. Nepusz, (2006), "The igraph software package for complex network research" in *InterJournal Complex Systems 1695*, available at http://igraph.org.

Disney, A. (2020), “Social network analysis 101: centrality measures explained”, https://cambridge-intelligence.com/keylines-faqs-social-network-analysis/ access in 23rd April 2020.

Farvaresh, H., M. M. Sepehri (2011), “A data mining framework for detecting subscription fraud in telecommunication” in *Engineering Applications of Artificial Intelligence*, Vol. 24 (1), pp. 182-194.

Faust, K. (2006), “Comparing social networks: size, density, and local structure” in *Metodolos˙ki zvezki*, Vol, No 3, pp 185-210.

Francis, C., N. Pepper and H. Strong (2011), “Using Support Vector Machines to Detect Medical Fraud and Abuse”, in *Proceedings of Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 8291–8294.

Fraud (n.d.), Online Cambridge Dictionary, https://dictionary.cambridge.org/dictionary/english/fraud, accessed in 19th December 2018.

Freeman, L. (1997), "A Set of Measures of Centrality Based On Betweenness", in: *Sociometry* 40 (1), Mach 1997, pp. 35-41.

Gama, J. (2010), *Knowledge Discovery from Data Streams*. Chapman & Hall/Crc Data Mining and Knowledge Discovery. Taylor & Francis Group.

Gupta, M., J. Gao, Y. Sun and J. Han (2012), "Integrating community matching and outlier detection for mining evolutionary community outliers" in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, August 2012.

Hajian-Tilaki, K. (2013), "Receiver operating characteristic (ROC) curve analysis for medical diagnostic test evaluation" in *Caspian journal of internal medicine*, Vol 4(2), pp 627–635.

Holtfreter K. (2014), "Fraud Victimization" in Bruinsma G., Weisburd D. (eds) Encyclopedia of Criminology and Criminal Justice. Springer, New York, NY. https://doi.org/10.1007/978-1-4614-5690-2_75.

Jeh, G., and J. Widom (2002), "SimRank: A Measure of Structural-Context Similarity" in *The Eighth ACM SIGKDD International Conference*, January 2002.

Kappel, M. (2017), "5 Types Of Fraud In Business That Could Put You In A Bind" in *Forbes*, 4th October 2017 (https://www.forbes.com/sites/mikekappel/2017/10/04/5-types-of-fraud-in-business-that-could-put-you-in-a-bind/#71cef09013e5).

Kerr, D. S., and U. S. Murthy (2013), "The Importance of the CobiT Frame- Work IT Processes for Effective Internal Control Over Financial Reporting in Organizations: an International Survey" in *Symposium on Information Systems Assurance*, October 11-13, 2007, University of Waterloo.

Knight, L. (2015), "Does Fraud Follow Economic Cycles?", in https://strategiccfo.com/does-fraud-follow-economic-cycles/, accessed in 15th December 2018.

Kolaczuk, E. D. (2009), "Statistical Analysis of Network Data" in *Springer*, pp. 245-281.

Kuhn, M. (2020), "caret: Classification and Regression Training", R package version 6.0-86, available at: https://CRAN.R-project.org/package=caret.

Lacey D., J. Zaiss and K. S. Barber (2016), "Understanding Victim-enabled Identity Theft, Perpetrator and Victim Perspectives" in *14th Annual Conference on Privacy, Security and Trust (PST)*.

Lavion, D. (2018), "Pulling Fraud Out of the Shadows: PwC's Global Economic Crime and Fraud Survey 2018" in https://www.pwc.com/gx/en/forensics/global-economic-crime-and-fraud-survey-2018.pdf, accessed in 7th January 2020.

Levi, M., Smith, R., 2011, "Fraud Vulnerabilities and the Global Financial Crisis" in *Trends & issues in crime and criminal justice*, no 422. Canberra: Australian Institute of Criminology.

Lopez-Rojas, E. A., A. Elmir, and S. Axelsson (2016), "PaySim: A financial mobile money simulator for fraud detection" in *The 28th European Modeling and Simulation Symposium-EMSS*, Larnaca, Cyprus.

Lopez-Rojas, E. A. (2016), "Applying Simulation to the Problem of Detecting Financial Fraud" (Doctoral dissertation, Blekinge Tekniska Högskola).

Lou, Y. and M. Wang (2009), " Fraud Risk Factor Of The Fraud Triangle Assessing The Likelihood Of Fraudulent Financial Reporting" in *Journal of Business & Economics Research*, Vol 7 (2), pp 61-78.

Lu, F., & Boritz, J. E. (2005, October). Detecting fraud in health insurance data: Learning to model incomplete Benford's law distributions. In European Conference on Machine Learning (pp. 633-640). Springer, Berlin, Heidelberg.

Luke, D. A. (2015), "A User's Guide to Network Analysis in R" in *Springer*, pp. 189-215.

Lunardon, N., G. Menardi and N. Torelli (2014), "ROSE: a Package for Binary Imbalanced Learning" in *R Journal*, Vol. 6(1), pp. 82-92.

Lutui, R. and T. Ahokovi (2017), "Financial fraud risk management and corporate Governance" in *Australian Information Security Management Conference*.

Malekian, D. and M. R. Hashemi (2013), “An Adaptive Profile Based Fraud Detection Framework For Handling Concept Drift” in *10th International ISC Conference on Information Security and Cryptology (ISCISC).*

Maes, S., K. Tuyls, B. Vanschoenwinkel and B. Manderick (2002), “Credit card fraud detection using Bayesian and neural networks” in *Proceedings of the 1st international naiso congress on neuro fuzzy technologies*, pp. 261-270.

Masciandaro, D. (1999), “Money Laundering: the Economics of Regulation” in *European Journal of Law and Economics*, no 7, pp 225–240, Kluwer Academic Publishers.

McDowell, J. and G. Novis (2001), “The Consequences of Money Laundering and Financial Crime” in *Economic Perspectives, Electronic Journal*, U.S. Department of State, Vol. 6 (2) .

Meyer, D., E. Dimitriadou, K. Hornik, A. Weingessel and F. Leisch (2019), "e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071)", TU Wien, R package version 1.7-3, available at https://CRAN.R-project.org/package=e1071.

Namatame, A. and S. Chen (2016), *Agent-Based Modelling and Network Dynamics,* Oxford University Press, pp. 162-252.

Newham London (n.d.) (2019), “Reporting fraud and how we deal with it”, retrieved 20 November 2019 from https://www.newham.gov.uk/advice-support-benefits/report-fraud/7.

Newman, M. E. (2002), “Assortative mixing in networks” in *Physical review letters*, vol 89(20), 208701.

Nick, F.R.T., P. Krauseb and W. Garnc (2018), “How Artificial Intelligence and machine learning research impacts payment card fraud detection: A survey and industry benchmark” in *Engineering Applications of Artificial Intelligence*, vol. 76, pp. 130 – 157.

Noble, C. and D. Cook (2003), “Graph-based anomaly detection” in *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Washington, DC, USA, August 24 - 27, 2003.

Oana, F. I. (2018), "Economic Fraud in International Business: Forms and Implications" in *"Ovidius" University Annals, Economic Sciences Series* Vol. 18 (1).

Okamoto K., Chen W. and Li X. Y (2008), "Ranking of closeness centrality for large-scale social networks" in *Frontiers in Algorithmics*, Preparata F. P., Wu X. and Yin J. (editors), Proceedings of Second International Workshop, Changsha, China. Springer, Berlin, pp 186–195.

Opsahl, T., F. Agneessens and J. Skvoretz (2010), "Node centrality in weighted networks: Generalizing degree and shortest paths" in *Social networks* vol 32 (3), pp 245-251.

Phua, C., Lee, V., Smith, K., Gayler, R., 2005, "A Comprehensive Survey of Data Mining-Based Fraud Detection Research"

Pironet, M., Antunes, C., Moura, P. and Gomes, J., 2009, "Classification for Fraud Detection with Social Network Analysis"

Pouramirarsalani, A., M. Khalilian and A. Nikravanshalmani (2017), "Fraud detection in E-banking by using the hybrid feature selection and evolutionary algorithms" in *IJCSNS, International Journal of Computer Science and Network Security*, Vol.17 No.8.

Powers, D.M.W. (2011), "Evaluation: from Precision, Recall and F-measure to ROC, Informedness, Markedness and Correlation" in *Journal of Machine Learning Technologies*, Vol 2(1) page 37-63.

Quiñones, E. (2000), "What is Corruption?", OECD Observer No 220 in http://oecdobserver.org/news/archivestory.php/aid/233/What_is_corruption_.html, accessed in 21st December 2018.

R Core Team (2019), "R: A language and environment for statistical computing". R Foundation for Statistical Computing, Vienna, Austria, available at https://www.R-project.org/.

Rana, A. S. (2019), "Is Artificial Intelligence the future of Anti-Financial Crime?", in https://corporatesocialresponsibilityblog.com/2019/10/28/artificial-intelligence-and-financial-crime/, access in 7th December 2019.

Sanver, M. and A. Karahoca (2009), “Fraud Detection Using an Adaptive Neuro- Fuzzy Inference System in Mobile Telecommunication Networks” in *Multiple-Valued Logic and Soft Computing*.

Sherly, K.K. and R. Nedunchezhian (2010), “Boat adaptive credit card fraud detection system” in *2010 IEEE International Conference on Computational Intelligence and Computing Research.*

Smith, S., T. Hrncir and S. Metts (2013), “Small business fraud and the trusted employee”, https://www.acfe.com/article.aspx?id=4294976289, access in 7th July 2020.

Thedchanamoorthy, G., M. Piraveenan, D. Kasthuriratna and U. Senanayake (2014), “Node Assortativity in Complex Networks: An Alternative Approach” in *14th International Conference on Computational Science*, Vol 29, pp. 2449-2461.

Therneau, T. and B.Atkinson (2019), "rpart: Recursive Partitioning and Regression Trees", R package version 4.1-15, available at: https://CRAN.R-project.org/package=rpart.

Tsai, Y. H., C. H, Ko and K. C. Lin (2014), “Using CommonKADS method to build prototype system in medical insurance fraud detection” in *Journal of Networks*, Vol 9(7).

Tsung, F., Z. Zhou and W. Jiang (2007), “Applying Manufacturing Batch Techniques to Fraud Detection With Incomplete Customer Information in *IIE Transactions Journal*, Vol. 39 (6), pp. 671–680.

Watts, D. J. (2002), “A simple model of global cascades on random networks” in *Proceedings of the National Academy of Sciences*, Vol. 99(9), pp. 5766-5771.

Wickham, H., R. François, L. Henry and K. Müller (2020), "dplyr: A Grammar of Data Manipulation", R package version 0.8.5, available at https://CRAN.R-project.org/package=dplyr.

Wolf, M. (2009), “FT Martin Wolf – Reform of Regulation and Incentives”, in *Financial Times*, June 23rd, 2009.

Xu, X., N. Yuruk, Z. Feng and T. Schweiger (2007), "SCAN: A Structural Clustering Algorithm for Networks" in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* San Jose, California, USA, August 12-15, 2007.

Yan, X. and J. Han (2002), "gSpan: Graph-Based Substructure Pattern Mining" in *Proceedings - IEEE International Conference on Data Mining*, pp 721- 724.

Zhou Q., F. Y. Womer, L. Kong, et al. (2017), "Trait-Related Cortical-Subcortical Dissociation in Bipolar Disorder: Analysis of Network Degree Centrality" in *The Journal of Clinical Psychiatry*, Vol 78, N 5, pp 584-591.

7. Attachments

```
> data<-read.csv("Dados.csv", header = TRUE, sep = ",")

> ordered.data<-data[order(data$step, decreasing = FALSE),]

> Train<-ordered.data[c(1:4463587),]

> Test<-ordered.data[c(4463588:6362620),]


> Train_F<-Train[Train$isFraud==1,]

> Train_NF<-Train[Train$isFraud==0,]

> Train_sample<-sample(1:nrow(Train_NF), 490594)

> train_sample<-Train_NF[Train_sample,]

> train<-rbind(Train_F, train_sample)

> train<-train[order(train$step, decreasing = FALSE),]


> library(dplyr)

> test_sample<-Test %>% group_by(step) %>% sample_frac(0.11)

> test_sample<-as.data.frame(test_sample)

> missing<-Test[!Test$step %in% test_sample$step,]

> add<-missing %>% group_by(step) %>% sample_n(1)

> add<-as.data.frame(add)

> test<-rbind(test_sample, add)

> test<-as.data.frame(test)

> test<-test[order(test$step, decreasing = FALSE),]
```

```
> train<-ordered.data[c(1:494237),]

> test<--ordered.data[c(494238:703147),]


> library(rpart)

> train$isFraud<-as.factor(train$isFraud)

> tree<-rpart(isFraud~step+type+amount+oldbalanceOrg+newbalanceOrig+
oldbalanceDest+ newbalanceDest, data=train)

> test$isFraud<-as.factor(test$isFraud)

> predict_tree<-predict(tree, test)

> predict_tree<-as.factor((predict_tree[,2]>0.5)*1)

> library(caret)

> confusionMatrix(table(predict_tree, test$isFraud), positive = "1")

> library(ROSE)

> roc.curve(test$isFraud, predict_tree)


> log_reg <- glm(isFraud ~ step+type+amount+oldbalanceOrg+newbalanceOrig+
oldbalanceDest+newbalanceDest,family=binomial,data=train)

> predict_reg<-predict(log_reg, test, type="response")

> predict_reg<-as.matrix(predict_reg)

> predict_reg[predict_reg < 0.5] <- 0

> predict_reg[predict_reg >= 0.5] <- 1

> confusionMatrix(table(predict_reg, test$isFraud), positive = "1")

> roc.curve(test$isFraud, predict_reg)
```

```
> library(e1071)

> SVM<-svm(formula = isFraud ~ step+type+amount+oldbalanceOrg+newbalanceOrig+
oldbalanceDest+newbalanceDest, data = train, type = 'C-classification', kernel = 'linear')

> predict_SVM<-predict(SVM, test)

> confusionMatrix(table(predict_SVM, test$isFraud), positive = "1")

> roc.curve(test$isFraud, predict_SVM)


> sample<-rbind(train, test)

> ordered.sample<-sample[order(sample$step, decreasing = FALSE),]

> ordered.sample$amount<-as.integer(ordered.sample$amount)

> ordered.sample$isFraud<-as.integer(ordered.sample$isFraud)

> data_features<- ordered.sample %>%

    group_by(nameOrig) %>%

    mutate(amountOrig=lag(cumsum(amount), default=0)) %>%

    mutate(FraudAmountOrig = isFraud*amount) %>%

    mutate(cumsumFraudAmountOrig=lag(cumsum(FraudAmountOrig),      default=0))
%>%

    mutate(wOrig = ifelse(cumsumFraudAmountOrig==0, 0, cumsumFraudAmountOrig/
amountOrig))

> data_features<-as.data.frame(data_features)

> data_features$typeOrig<-substr(ordered.sample$nameOrig, 1, 1)

.

> data_features$amount<-as.integer(data_features$amount)

> data_features$isFraud<-as.integer(data_features$isFraud)
```

```
> data_features<- data_features %>%
    group_by(nameDest) %>%
    mutate(amountDest=lag(cumsum(amount), default=0)) %>%
    mutate(FraudAmountDest = isFraud*amount) %>%
    mutate(cumsumFraudAmountDest=lag(cumsum(FraudAmountDest), default=0))
%>%
    mutate(wDest = ifelse(cumsumFraudAmountDest==0, 0, cumsumFraudAmountDest/
amountDest))
> data_features<-as.data.frame(data_features)
> data_features$typeDest<-substr(data_features$nameDest, 1, 1)
.
> train_features<-data_features[c(1:494237),]
> test_features<-data_features[c(494238:703147),]

> train_features$isFraud<-as.factor(train_features$isFraud)
>     tree_features<-rpart(isFraud~step+type+amount+oldbalanceOrg+newbalanceOrig+
oldbalanceDest+newbalanceDest+amountOrig+amountDest+wDest+typeDest,
data=train_features)
> test_features$isFraud<-as.factor(test_features$isFraud)
> predict_tree_features<-predict(tree_features, test_features)
> predict_tree_features<-as.factor((predict_tree_features[,2]>0.5)*1)
> confusionMatrix(table(predict_tree_features, test_features$isFraud), positive = "1")
> roc.curve(test_features$isFraud, predict_tree_features)
```

```
>log_reg_features <- glm(isFraud ~ step+type+amount+oldbalanceOrg+
newbalanceOrig+oldbalanceDest+newbalanceDest+amountOrig+amountDest+wDest+
typeDest,family=binomial,data=train_features)

> predict_reg_features<-predict(log_reg_features, test_features, type="response")

> predict_reg_features<-as.matrix(predict_reg_features)

> predict_reg_features[predict_reg_features < 0.5] <- 0

> predict_reg_features[predict_reg_features >= 0.5] <- 1

> confusionMatrix(table(predict_reg_features, test_features$isFraud), positive = "1")

> roc.curve(test_features$isFraud, predict_reg_features)


> library(e1071)

> SVM_features<-svm(formula = isFraud ~ step+type+amount+oldbalanceOrg+
newbalanceOrig+oldbalanceDest+newbalanceDest+amountOrig+amountDest+wDest+t
ypeDest, data = train_features, type = 'C-classification', kernel = 'linear')

> predict_SVM_features<-predict(SVM_features, test_features)

> confusionMatrix(table(predict_SVM_features, test_features$isFraud), positive = "1")

> roc.curve(test_features$isFraud, predict_SVM_features)


> library(igraph)

> sample<-ordered.sample[,c(“step”, “nameOrig”, “nameDest”, “amount”, “isFraud”)]

> isfraud<-subset(sample, sample$isFraud==1)

> A<-list()

> Fraud<-matrix(0, 743, 1)

> Bet<-vector()
```

```
> Dens<-vector()
> for (i in 1:743) {
 A[[i]]<-sample[sample$step<=i, 2:3]
 G<-graph_from_edgelist(as.matrix(A[[i]]), directed = FALSE)
 Bet[i]<-mean(betweenness(G))
 Dens[i]<-edge_density(G)
 for (j in 1:length(as.matrix(A[[i]])[,2])) {
   nr<-nrow(subset(isfraud, isfraud$destino==A[[i]][j,2]))
  if (nr>0) {Fraud[i]<-Fraud[i]+nr;print("fraud!")}
}
}
> Result<-cbind(Fraud, Bet, Dens)

> step<-(1:743)
> step<-as.data.frame(step)
> nobs<-ordered.sample %>% count(step)
> nobs<-as.data.frame(nobs)
> nobs<-nobs[,c("n")]
> new<-cbind(step, nobs, Result)
> ordered.sample$Bet <- new$Bet[match(ordered.sample$step, new$step)]
> ordered.sample$Dens <- new$Dens[match(ordered.sample$step, new$step)]

> train_G<-ordered.sample[c(1:494237),]
```

```
> test_G<-ordered.sample[c(494238:703147),]


> library(rpart)

> train_G$isFraud<-as.factor(train_G$isFraud)

>              tree_G<-rpart(isFraud~step+type+amount+oldbalanceOrg+newbalanceOrig+
oldbalanceDest+newbalanceDest+Bet+Dens, data=train_G)

> test_G$isFraud<-as.factor(test_G$isFraud)

> predict_tree_G<-predict(tree_G, test_G)

> predict_tree_G<-as.factor((predict_tree_G[,2]>0.5)*1)

> library(caret)

> confusionMatrix(table(predict_tree_G, test_G$isFraud), positive = "1")

> library(ROSE)

> roc.curve(test_G$isFraud, predict_tree_G)


> log_reg_G <- glm(isFraud ~ step+type+amount+oldbalanceOrg+newbalanceOrig+
oldbalanceDest+newbalanceDest+Bet+Dens,family=binomial,data=train_G)

> predict_reg_G<-predict(log_reg_G, test_G, type="response")

> predict_reg_G<-as.matrix(predict_reg_G)

> predict_reg_G[predict_reg_G < 0.5] <- 0

> predict_reg_G[predict_reg_G >= 0.5] <- 1

> confusionMatrix(table(predict_reg_G, test_G$isFraud), positive = "1")

> roc.curve(test_G$isFraud, predict_reg_G)


> library(e1071)
```

```
>SVM_G<-svm(formula = isFraud ~ step+type+amount+oldbalanceOrg+
newbalanceOrig+oldbalanceDest+newbalanceDest+Bet+Dens, data = train_G, type = 'C-
classification', kernel = 'linear')

> predict_SVM_G<-predict(SVM_G, test_G)

> confusionMatrix(table(predict_SVM_G, test_G$isFraud), positive = "1")

> roc.curve(test_G$isFraud, predict_SVM_G)


> data_features<- rbind(train_features, test_features)

> data_features<-as.data.frame(data_features)


> step<-(1:743)

> step<-as.data.frame(step)

> nobs<-ordered.sample %>% count(step)

> nobs<-as.data.frame(nobs)

> nobs<-nobs[,c("n")]

> new<-cbind(step, nobs, Result)

> data_features_G<-data_features

> data_features_G$Bet <- new$Bet[match(data_features_G$step, new$step)]

> data_features_G$Dens <- new$Dens[match(data_features_G$step, new$step)]


> train_features_G<-data_features_G[c(1:494237),]

> test_features_G<-data_features_G[c(494238:703147),]


> train_features_G$isFraud<-as.factor(train_features_G$isFraud)
```

```
> tree_features_G<-rpart(isFraud~step+type+amount+oldbalanceOrg+newbalanceOrig+
oldbalanceDest+newbalanceDest+amountOrig+amountDest+wDest+typeDest+Bet+De
ns, data=train_features_G)

> test_features_G$isFraud<-as.factor(test_features_G$isFraud)

> predict_tree_features_G<-predict(tree_features_G, test_features_G)

> predict_tree_features_G<-as.factor((predict_tree_features_G[,2]>0.5)*1)

> confusionMatrix(table(predict_tree_features_G, test_features_G$isFraud), positive =
"1")

> roc.curve(test_features_G$isFraud, predict_tree_features_G)


> log_reg_features_G <- glm(isFraud ~ step+type+amount+oldbalanceOrg+
newbalanceOrig+oldbalanceDest+newbalanceDest+amountOrig+amountDest+wDest+t
ypeDest+Bet+Dens,family=binomial,data=train_features_G)

> predict_reg_features_G<-predict(log_reg_features_G, test_features_G, type=
"response")

> predict_reg_features_G<-as.matrix(predict_reg_features_G)

> predict_reg_features_G[predict_reg_features_G < 0.5] <- 0

> predict_reg_features_G[predict_reg_features_G >= 0.5] <- 1

> confusionMatrix(table(predict_reg_features_G, test_features_G$isFraud), positive =
"1")

> roc.curve(test_features_G$isFraud, predict_reg_features_G)


> library(e1071)

> SVM_features_G<-svm(formula = isFraud ~ step+type+amount+oldbalanceOrg+
newbalanceOrig+oldbalanceDest+newbalanceDest+amountOrig+amountDest+wDest+
typeDest+Bet+Dens, data = train_features_G, type = 'C-classification', kernel = 'linear')
```

```
> predict_SVM_features_G<-predict(SVM_features_G, test_features_G)

> confusionMatrix(table(predict_SVM_features_G, test_features_G$isFraud), positive =
"1")

> roc.curve(test_features_G$isFraud, predict_SVM_features_G)
```