Analysis of insurance claims data based on networks

Manuel Alejandro Moreno Vásquez
M.Sc.(c) Statistics

Universidad Nacional de Colombia
Facultad de Ciencias
Departamento de Estadística
Bogotá, D.C.
July 2020

Analysis of insurance claims data based on networks

Manuel Alejandro Moreno Vásquez
M.Sc.(c) Statistics

A dissertation submitted in partial fulfillment for the degree in
Master of Science in Statistics

Advisor
Martha Patricia Bohorquez, Ph.D.

Coadvisor
Rafael Renteria Ramos, Ph.D.

Universidad Nacional de Colombia
Facultad de Ciencias
Departamento de Estadística
Bogotá, D.C.
July 2020

Title

Analysis of insurance claims data based on networks

Abstract: This work proposes a statistical methodology for learning relational encodings of influential high dimensional variables for supervised binary classification. The encoding ranks the categories according to its relative importance for obtaining the outcome of interest in the training data using a personalized PageRank algorithm for bipartite networks. For obtaining the scores, a dyadic analysis of the bipartite networks constructed on the relationships among the categories under study is made, enriching the knowledge and interpretability of the intrinsic structures of the target variable in the training process.

Binary classification tasks account for a high percentage of applications of predictive modelling in industries such as insurance, banking, telecommunications, etc. The hardship that the curse of dimensionality carries in widespread statistical learning algorithms makes it necessary to explore encoding alternatives to dummy and other ad hoc methods in the literature. The proposed methodology brings a statistically driven and structure oriented representation of categorical variables that can be fed into supervised learning binary classification models.

An application of the proposed methodology is supervised classification for fraud detection. Fraud is a social phenomena with several impacts in which active research is made from the statistical and network community. Insurance companies are highly exposed to fraudulent claims and the nature of the data required for its analysis is mostly qualitative. An experimental case study is conducted with an automobile insurance fraud detection scenario for comparing the performance of the proposed methodology for bipartite encoding and the popular target encoding (Micci-Barreca, 2001). The empirical results show that the bipartite networks encoding can help random forest models to lower the false positive rate. This encoding also highlights relations among categorical variables, making it more interpretable than some of the popular methods in the statistical learning community.

Keywords: Statistical network analysis, Bipartite networks, PageRank, Moran's I, Representation learning, Supervised learning, Fraud detection

Contents

List of Tables

List of Figures

Introduction

The adoption of statistical learning models to predict and classify phenomena for many aspects of modern society makes the challenges in the field to have a multifaceted responsibility. As the influence of the algorithms that perform predictive decision making grows, the accountability for their performance and integrity is important in the role they have in our society. Binary classification may be the most simple form of decision logic, but it accounts for the most useful in many fields. Would a client like a product or not? will he/she buy it or not? if he/she is in debt, will he/she pay it or not?. Its simplicity reflects the need for a clear and concise distinction in the outcome of an event. Far from being a perfect methodology, it is an active research field given its predominance and wide range of applications in industry. Currently, studies are being held on accuracy improvement for a wide variety of purposes and scenarios. Interpretability of deep learning approaches, bias, and discrimination reduction, among others. Each challenge requires its thorough analysis.

Traditionally, when representing individuals on a dataset, the most frequent depiction is a vectorized representation in a p-dimensional space. This representation has several good characteristics that make it suitable to analyze and interpret. However, this is not the only representation that can be created from the records nor is the one that captures the most structural information from it in the presence of categorical data. The use of a relational representation of entities trough networks can unveil information about the phenomenon under study. As described by Getoor and Taskar (Koller et al., 2007) in their brief history of relational learning, the intuition towards the use of relational logic inside statistical learning models was to use the dependence structure of entities, given that it exists on the data. This field grew with a rich stochastic structure given its related nature. Never the less, as described by the same authors, the sparse and complex representation of data slowed its widespread adoption.

Fienberg (2012) dates the adoption of network models into the statistical literature since the late seventies. The rise in research from statistical physics and computer science areas came with the new millennium and the widespread use of the internet and social networks. In this fast-growing field, the focus of the developments has been, among others, community detection, node classification, link prediction, topology, etc. Nowadays network data is abundant and the methodologies developed for its analysis have a wide range of applications.

In the last decade, there has been an increased effort to integrate the relational structure of network analysis and the statistical learning process. In Bengio et al. (2013) the authors hypothesized that different representations can showcase different explanatory factors of variation in the data. The usage of probabilistic models, stochastic processes,

auto-encoders, manifolds, and deep networks were among the available options for feature learning and capture the geometry of the data. The community built calls this field *Representation Learning*, and in recent years have been able to formalize and achieve empirical and theoretical breakthroughs (Hamilton et al., 2017; Schlichtkrull et al., 2018). Networks are without a doubt complex representations given its topological structures, no fixed ordering or reference points, and can display many features. The most relevant goal is then to capture the important information outlined in a network and transform it into a vector-based representation. This representation is called network embedding and can be learned using encoder-decoder functions (Sabokrou et al., 2019), factorization based methods, random walks (such as *DeepWalk* and *node2vec*) and others that have roots in the text mining and computer vision literature (Hamilton et al., 2017; Backstrom & Leskovec, 2011).

Although most of the focus on the field has been on the embedding of unipartite networks, there are some recent works with the same goal for bipartite networks. Bipartite networks arise naturally in many contexts like recommender systems, ecology, marketing, social networks, etc. Methods for their analysis add value to the nodes information by acknowledging that there are two classes of nodes interacting. Gao et al. (2018) use biased random walks to generate vertex sequences out of the bipartite networks, then, applying methods for capturing the explicit and implicit relations in the data produce a low dimensional representation that captures the information of the nodes. These were used in link prediction and recommender classification tasks and achieved the state of the art results at the time. Other works for bipartite networks have used similar approaches to rank and give vector features to the nodes (He et al., 2016), and others to capture higher-order structures in the networks (Sybrandt & Safro, 2019). Recently, algorithms on multipartite network embedding have also been proposed (Rajan et al., 2019), and new papers are coming out on the subject which indicates it is a currently active research field.

From the statistical point of view, there is a wide range of methodologies in the literature to tackle the representation of high cardinality categorical variables and they evolve according to waves of research in novel topics. Recent compilations address the handling of this type of data for popular methods like neural networks (Potdar et al., 2017). A common topic in the research literature is the criticism of the dummy representation of data for statistical learning models, and alternatives obtained trough factorization (Cerda & Varoquaux, 2020), decision trees (Lucena, 2020) and density-based representations (Chen et al., 2013) are proposed. However, most of these methods do not reach mainstream usage yet due to domain knowledge requirements. The methods widely used in the industry usually are the easiest to implement in machine learning pipelines, such as target encoding (Micci-Barreca, 2001). Then, an effective framework for categorical variables encoding should consider a software implementation and scalability.

This work aims to contribute to the literature of preprocessing techniques for categorical variables making use of networks. The main objective is to propose a statistical methodology for learning relational encodings of influential high cardinality variables for supervised binary classification. The reason why this approach is taken, what is the current state of research on the topic and a critical view on them, is the topic of this chapter.

A bipartite network representation of categorical variables is proposed and the information captured from them is used to improve the classification accuracy of the binary target variables. This approach is partially based on the empirical and theoretical work made by Van Vlasselaer et al. (2016). In the literature, the most similar approaches to

network analysis of categorical variables are the ones by Moeyersoms & Martens (2015) and Zhang et al. (2015). The former by acknowledging that there are relationships in categorical data that can be exploited to get structural sense out of them trough bipartite networks, but not establishing use and characterization of them. And the latter, on the contrary, arguments why their methodological proposal generalizes even to a more general representation of the categorical variables, in a p-partite network. This last approach is not considered in this work given its abstraction increases the complexity of the representation and the more recent and tested algorithms are in the field of bipartite networks. A follow up of the work made by the mentioned authors is made and to the best of the knowledge, there are no new developments on those techniques at the moment. The reference paper followed in this work by Van Vlasselaer et al. (2016), was able to successfully implement and obtain an increase in the accuracy of the binary classification of fraudulent nodes in network data (Vlasselaer et al., 2015; Vlasselaer et al., 2013). In the fraud detection community, as noted in the most recent literature review of the use of network algorithms in the field by Pourhabibi et al. (2020), there are two main approaches to fraud detection with network data, either by considering one-mode networks or considering bipartite networks.

Usually, bipartite networks are used to study connections between users and products or services to detect suspicious behaviors and anomalies. In this work, a different approach is made as categorical data is non-networked. Bipartite networks will be used to study the relationships among categories of pairs of variables. The bipartite networks will display information about the target variables in the training data set of in the binary classification process, and a score of importance will be used to summarize the structural impact of the categories for the classification purpose. A thorough analysis of the characteristics of the networks is made to display the hidden information they can bring on the analysis of the binary outcomes.

In chapter 1, the theoretical reference framework for the methodology proposed is set, using statistical network analysis and statistical learning. In chapter 2 a description of the bipartite network methodology for categorical variables encoding is made, alongside its analysis, role in a binary classification process and software implementation. In chapter 3 a study case is developed with the proposed bipartite network encoding methodology to test its performance in a binary classification process. The case is set in the insurance industry, to classify fraudulent claims. This study case is selected by its industry relevance and the close relation of the algorithms in the proposed methodology for the study of this type of events. In Figure 3.4.4, the discussion of the results and a critical evaluation of the methodology is made.

CHAPTER 1

Theoretical Framework

This chapter provides the conceptual framework in which the methodology is established. By starting with some theory of network analysis and ending with theory on supervised statistical learning it is intended to display the same order that these concepts will be used in the proposal methodology in chapter 2.

1.1 Statistical Network Analysis

This field of knowledge uses graph theory to extract useful statistics from network data. Formally, a network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ consists of a set of vertices or nodes $\mathcal{V}$, where $|\mathcal{V}| = N$, and a set $\mathcal{E}$ of edges or links. A node $\mathfrak{v} \in \mathcal{V}$ can represent any entity. An edge represents a relationship between the entities it connects (Kolaczyk, 2009). Attributes represent characteristics of each of the nodes.

A network can be represented by an adjacency matrix $A_{N\times N}$, where

$$[a_{ij}] = \begin{cases} 1 & \text{if nodes i and j are linked} \\ 0 & \text{if node i and j are not linked} \end{cases} . \tag{1.1}$$

The adjacency matrix is often a sparse matrix and it is fundamental in many mathematical techniques for describing the network. A network where the edges express the strength of relationships between the nodes is a weighted network. The matrix representation of a weighted network is the matrix W, which is defined as in (1.1), replacing the indicator by a strength w_{ij}.

Then, a network can represent a complex set of interactions among entities and characterize the relations that arise. If there is a direction in the edges, then a network is said to be directed and its correspondent matrix W can be asymmetrical. This type of networks store information about the propagation of a feature among nodes. If the network is undirected, the matrix W is symmetrical. This type of networks are rich in structural information about the node's dynamics. In this work all of the considerations are made for undirected networks. If a network has no distinction on the type of nodes that can interact in it, is said to be a *unipartite* network. If there are two types of nodes that can interact exclusively with nodes of the other type, the network is said to be *bipartite*. This type of

networks are the ones used in the methodology and its properties will be highlighted in section 2.1.

1.1.1 Network Representation of Non-Network Data

As can be seen from the definition in Equation 1.1, the mathematical representation of a network is a numerical matrix. As such, categorical variables can not be directly represented as networks. However, in the literature there are techniques to construct network representations of non-network data. This subsection contains some of the methods described in Silva & Zhao (2016, chapter 4).

The first requirement is a notion of similarity among the entities that are intended to be transformed into networks. Using the dummy representation of categorical data, a notion of similarity between two individuals x_i and x_j can be described by the number of categories in common as in the contingency table 1.1, in it the M_{kl} represents the number of matches for the respective field.

TABLE 1.1. Contingency table of shared features (Silva & Zhao, 2016).

		x_j	
		Present	Absent
x_i	Present	M_{00}	M_{01}
	Absent	M_{10}	M_{11}

This simple notion can help us to create a similarity metric among entities x. For example, the Jaccard similarity is deffined as

$$J(x_i, x_j) = \frac{M_{11}}{M_{11} + M_{01} + M_{10}}. \tag{1.2}$$

The Sorensen similarity is defined as

$$S(x_i, x_j) = \frac{2M_{11}}{2M_{11} + M_{01} + M_{10}}. \tag{1.3}$$

The simple matching similarity is defined as

$$SM(x_i, x_j) = \frac{M_{11} + M_{00}}{M_{11} + M_{00} + M_{01} + M_{10}}. \tag{1.4}$$

And many more, each giving more relevance to different factors. Now, that a definition of distance has been made on the individuals according to their categorical data, this can be displayed in a matrix form and clustering procedures can be applied. The most straightforward is the nearest neighbors and ϵ-radius techniques. Then a link is made between the set of individuals in the nearest neighbour set. This is just one of the many techniques that can be used to define networks on individuals by categorical data.

As stated by (Silva & Zhao, 2016), the best choice out of the previous building techniques options has received little attention in literature, so heuristics plays a big role in their use. Because of this, and the fact that the objective of the methodology proposed in this work is to represent the variables as networks, not the individuals, that in section 2.1 a different approach that suits categorical variables is proposed.

1.1.2 Network Topology Measures

Topology measures are used to summarize relevant aspects of the structure of a network. In the literature there is a coarse amount of methods to analyze and describe networks (Kolaczyk, 2009; Aggarwal, 2011; Nisbet et al., 2009). In this section, a description of the most relevant network measures for unipartite networks is made. In section 2.3.2 a description of the relevant measures for bipartite networks is detailed with their interpretation in the context of the proposed methodology.

1.1.2.1 Global Description of the Network

The *size* of a network as the cardinality of the set $\mathcal{E}$ and the *order* as the cardinality of set $\mathcal{V}$. This concepts describe the dimensions of the network. In the following paragraphs, a definition of more insightful global measures of the network is made, as they will complement the proposal methodology.

Distance Metrics

Distance is an abstract concept in a networks, given that they live in non-euclidean space. A common notion of *distance* between nodes on a network is defined as the length of the shortest path(s) between them (Kolaczyk, 2009). In the context of a weighted network, *distance* can be the sum of the weights in the shortest path between them. Then the *diameter* is the length of the longest path that can be travelled between two different nodes in the network. The *average distance* is the average of the pairwise distances of all nodes in the network (Silva & Zhao, 2016).

Interconnectedness Measures

The *degree* of a node refers to the number of edges in which the node takes part (Aggarwal, 2011). The *average degree* of a network refers to the average of all node degrees, then the higher this value the more connected the network is (Kolaczyk, 2009). The *density* is a measure of cohesion defined as

$$density = \frac{size}{\binom{N}{2}} \tag{1.5}$$

where $\binom{N}{2}$ are all the possible edges in the network (Silva & Zhao, 2016).

Assortativity

Captures the preference of attachment between nodes based on similarity of degree, so it can be understood as a measure of degree correlation among vertices (Silva & Zhao, 2016). It is defined as the Pearson correlation of the degree between all pair of linked vertices. Hence, positive values indicate a high relationship between nodes of similar degree, and the contrary respectively.

Centrality Measures

Different measures try to assess the dominance of nodes in a network. That is, measure if a node is central in the structure of the network. One of them is *betweeness*, that measures the number of appearances a node makes in the paths among all other pairs of nodes in

the network (Nisbet et al., 2009). A measure of *closeness* , for each node in the network, is

$$closeness(u) = \frac{1}{\sum_v distance(u,v)} \tag{1.6}$$

which indicates that at lower the distance to all other nodes, the higher is the closeness (Kolaczyk, 2009). Bonachich's *eigenvector centrality* computes centrality using eigenvectors of the adjacency matrix of G (Kolaczyk, 2009). Then, solving $\lambda \cdot s = A \cdot s$ and taking the eigenvector associated to the first eigenvalue, will give the centrality score associated.

Some of the previous local measures can be globalized trough the *centralization* technique. Let $c(v)$ be any local centrality measure previously described on node v and let $c^* = max(c(v))$ among all nodes v. Then, the global centralization index of the measure c is defined as

$$centralization = \frac{\sum_v c^* - c(v)}{max \sum_v c^* - c(v)} \tag{1.7}$$

where max in the denominator makes reference to the theoretical maximum over all possible networks of order N (Kolaczyk, 2009). When this upper bound is known, this measure can indicate the dominance that nodes can have on the overall network.

Clustering Coefficient

Local *clustering coe cient* quantifies the extent at which a given node makes part of a highly connected subgraph of the network. The definition of this coefficient for node v is

$$CC_v = \frac{2|\mathcal{E}_v|}{degree(v)(degree(v)-1)} \tag{1.8}$$

where $|\mathcal{E}_v|$ is the number of edges that connect v to other nodes. The global measures for the network is (Kolaczyk, 2009)

$$CC = \frac{1}{|\mathcal{V}|} \sum_v CC_v \tag{1.9}$$

1.1.2.2 PageRank Algorithm

It is the famous algorithm that powered the early stages of the Google search engine (Page et al., 1999). The original paper uses matrix algebra with the purpose of ordering web pages by its importance in a complex directed network of millions of nodes. The importance score vector $\vec{\psi}$ is a measure of the amount of links a node has, and the importance score of the nodes those links connect with.

The rank vector $\vec{\psi}_{N\times 1}$ is calculated under the constraint that $\sum_{i=1}^{N} \psi_i = 1$ and requires to use the stochastic adjacency matrix $M_{N\times N}$, where

$$[m_{ij}] = \begin{cases} \frac{1}{d_j} & \text{if nodes i and j are linked, with } d_j \text{ the degree of node } j \\ 0 & \text{if node i and j are not linked} \end{cases} . \tag{1.10}$$

As can be seen, M is the column normalized version of the adjacency matrix A. Now, to gain intuition on how the importance score $\vec{\psi}$ is calculated, let $\vec{\psi}$ be the solution of the equations system

$$\vec{\psi} = M \cdot \vec{\psi}. \tag{1.11}$$

To understand it from a stochastic process point of view, imagine a discrete-time Markov chain where the state space are nodes $1, ..., N$. Let $\vec{p}(t)$ be a probability distribution vector such that $p_i(t)$ is the probability that at time $t \in \{0, 1, ...\}$, the chain is in node i. Then the transition probability $p(t+1) = M \cdot p(t)$. The chain above described is known as a random walk on the nodes of the network (Backstrom & Leskovec, 2011; Castañeda et al., 2012). If the random walk reaches a state where $p(t+1) = M \cdot p(t) = p(t)$, then p(t) is the stationary distribution of the process. Then, in (1.11), $\vec{\psi}$ is the stationary distribution of a long random walk defined over the network through the matrix M.

If the matrix M would not show signs of disconnected, cyclical or death-end paths in the network, then (1.11) would be enough to proceed to solve the system. If the random walk would get stuck in some subset of nodes in the network, then the importance score for that subset would be overestimated. Over the ever existing possibility of facing that circumstance, Brin & Page (1998) proposed as a solution the matrix O such that

$$O = \alpha M + (1 - \alpha)\vec{r}, \text{ where } \vec{r} = \left[\frac{1}{N}, ..., \frac{1}{N}\right]^T. \tag{1.12}$$

The intuition behind this matrix is that with probability $1 - \alpha$ the random walk will be capable of restart at a random node. The parameter α is an hyperparameter that in practice is set between 0.8 and 0.9.

Then, the PageRank equation for scoring node importance in the network is

$$\vec{\psi} = O \cdot \vec{\psi} \tag{1.13}$$

which, from linear algebra, can be seen as the eigen-vector related to eigen-value 1 of the matrix O. For scalability, solving this system is done iteratively trough the following algorithm, called the power iteration method:

1. Initialize: $\vec{\psi}^{(0)} = [\frac{1}{N}, ..., \frac{1}{N}]^T$
2. Iterate: $\vec{\psi}^{(t+1)} = O \cdot \vec{\psi}^{(t)}$
3. Stop when: $|\vec{\psi}^{(t+1)} - \vec{\psi}^{(t)}|_1 < \epsilon$ where $|\vec{x}|_1 = \sum_{1 \leq i \leq N} |x_i|$, the L_1 norm of $\vec{x}$, although Euclidian or other metrics can be used.

For computational purposes, (1.13) can be rearranged so that power method can be more efficient, such that

$$\vec{\psi} = \alpha \cdot M \, \vec{\psi} + (1 - \alpha) \cdot \vec{r}. \tag{1.14}$$

A slight modification of the original algorithm in which the restart vector $\vec{r}$ is modified in order to assign preference to nodes of interest is called Personalized PageRank. Also a modification in which the restart vector is set such that only one node has a value different to zero, is called a *random walk with restart*.

1.1.3 Node Attributes Autocorrelation

In this section, a description of the methods used in this work to explore correlation between node's attribute values at different locations of the network is made. It is indicated as autocorrelation, given the similarities that measuring correlations in networks keeps with the spatial statistics literature (Okabe & Sugihara, 2012, chapter 7). It is worth to mention the difference between network autocorrelation and abstract network autocorrelation. The former refers to real or tangible networks such as roads and the latter refers to intangible networks such as the ones arising in social contexts. In this work, abstract network correlations are the ones being studied.

The relational nature of networks allows to effortlessly introduce spatial statistics techniques into its analysis. The first law of geography states that everything is related to everything else, but near things are more related than distant things (Tobler, 1970). Given a notion of distance and a notion of randomness between nodes in the networks this law can be examined.

Complete Attribute Randomness (CAR) on networks can be defined in two ways, trough permutations or trough random samples from a population. For the former, the nodes attributes $z_1, ..., z_N$ are assumed to be fixed, their location in the network are assumed to be random and CAR would be that the probability of every permutation of the attributes is equal. In the latter, the attribute values $z_1, ..., z_N$ are assigned to nodes in a independent and identically generated manner from a probability distribution.

In the following chapters the use of standard spatial statistics techniques will be used to examine autocorrelation at a global and local level.

- **Network Local Moran's I statistic**
  Using the concept of Local Indicator of Spatial Autocorrelation (LISA) (Anselin, 1995), in the network context, it must (i) measure the degree of attribute autocorrelation around neighbouring nodes, (ii) be calculated for each node attribute in the network, (iii) add up to a proportional global measure of autocorrelation (Schabenberger & Gotway, 2004). A local, node version, of the Moran I statistic that satisfies those requirements is

$$I_i = \frac{N(z_i - \overline{z})\sum_{j=1}^{N} w_{ij}(z_j - \overline{z})}{\sum_{j=1}^{N}(z_j - \overline{z})^2}. \tag{1.15}$$

  where w_{ij} are the elements of interaction weights matrix W which is discussed in section 2.2.
  The expected value under the null hypothesis of permutation randomness is the equivalent to the standard spatial case, that is,

$$E(I_i) = \frac{\sum_{j=1}^{N} w_{ij}}{N-1}. \tag{1.16}$$

In order to test the null hypothesis of permutation randomness, the distribution of the statistic I_i can be approximated trough Monte Carlo simulations. This will allow us to make inference on the node attributes of the network. More details on the considerations related to inference are given in section 2.3.

- **Network Global Moran's I statistic**
  This global measure can be written in terms of the local Moran statistic such that

$$I = \frac{\sum_{i=1}^{N} I_i}{\sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij}}. \tag{1.17}$$

  Then it is seen that this statistic is proportional to the average of the local Moran's statistics. The expected value under the null hypothesis of permutation randomness is the equivalent to the standard spatial case, that is,

$$E(I_i) = -\frac{1}{N-1}. \tag{1.18}$$

In a variety of network studies Moran's I statistic has been used to assess correlation of node's measured attributes (Shiode, 2008; Yamada & Thill, 2010).

1.2 Supervised Statistical Learning for Binary Outcomes

When having a dataset with known labels for the outcome variable, the goal is to divide the input space, defined by the predictors, into labeled sub-regions. According to the algorithm of classification, the decision boundary is smooth or rough (Friedman et al., 2001). In the following sections a description of the methods that will be used in this work is made.

1.2.1 Linear Methods

This type of methods are the ones such as linear discriminant analysis (LDA), perceptrons and logistic regression. Although similar in structure, logistic regression is preferred over LDA given that it makes less assumptions on the data. Its simplicity does not compromise its performance, and it stands out for its interpretability and its importance for classification purposes in many industry applications.

Logistic Regression

The method comes from the family of generalized linear models. It assumes that the grouped target variable $Y_i \sim binomial(1, \pi_i)$ to define the proportion of successes as $P_i = Y_i/n_i$ and estimate the expected probability $E(P_i) = E(\frac{Y_i}{n_i}) = \pi_i$ as a function of the covariates $\vec{x_i}^T$ trough the model

$$logit(\pi_i) = log\left(\frac{\pi_i}{1-\pi_i}\right) = \vec{x_i}^T\beta, \tag{1.19}$$

the estimates for parameter vector β are found optimizing the log-likelihood function, which leads to the $p+1$ nonlinear *score* equations in β. To solve this equations the use of the iterative Newton-Raphson algorithm is performed till convergence. (Dobson & Barnett, 2008). As can be seen, when the number of categories for nominal variables is high, the number of parameters increases.

The popularity of this model is due to the interpretability of its parameters, statistical amenability like tests for goodness of fit and significance of parameters, and a range of criteria for variables selection. Depending on the objectives for which the modelling is made, further methods and strategies are performed to reduce variance and effects of collinearity. Some of them, like *elastic net* regularization, will be used in the study case chapter.

1.2.2 Tree-Based Methods

The method works by performing partitions of the feature spaces and adjusting simple models in each one of them. They have been prove powerful in many insurance and other industries applications for prediction and classification (Li et al., 2018; Lin et al., 2017).

Random forest

Decision trees are simple enough to be able to produce good results in training stages. They are able to accommodate to the data to well, breaking the feature space in such a way that it emulates the observations on training. This leads to instability and a high variance of predictions when testing. This model also struggles to deal with categorical variables. As described by Friedman et al. (2001), the efficient way to make partitions for categorical variables is to order the predictors according to the proportion of outcomes of interest in it. This relates to the target encoding strategy for categorical variables, and as such, the overfitting that this produces makes it better just not to include high cardinality categorical variables in this model.

However, their low-bias, high-variance characteristics makes them work well with bootstrap aggregation (Friedman et al., 2001). The name of this procedure for decision trees is called random forest (Breiman, 2001). The algorithm starts by drawing a bootstrap sample from the training data, selecting $m \leq p$ variables randomly, picking the best split to make out of the m variables and growing the tree. This procedure is performed by user defined criteria with a selected measure of purity among classes. All of the classifications made by the trees go trough a majority vote procedure to pick the final category. Many variants of the original algorithm have been proposed in previous years.

There are many more hyperparameters to take into account for this model than logistic regression, usually a grid search is required to a select model that performs the best on the metrics established. It also provides insights on the features via a variable importance, measured as the associated accumulated improvement in the split criterion. Overall, its performance depends on the complexities that the weak algorithms can capture (Breiman, 2001).

In this section a review of some of the most relevant techniques in network theory and statistical learning were made. The proposed use of networks for dealing with high cardinality categorical variables is the topic of the following chapter.

CHAPTER 2

Statistical Methodology

In the previous chapters, an examination of the literature approaches and the reference framework to address the problem of categorical variables encoding was performed. This chapter it is dedicated to the description of the heuristics proposed to obtain a statistical driven and structure-oriented representation of categorical variables for binary classification to improve the accuracy metrics. In this methodology, it is assumed that sample problems and missing values have already been dealt with. An assessment of the performance of the methodology for a study case is made in chapter 3.

From a general perspective, when dealing with categorical variables in supervised classification tasks there are important questions that must be answered to continue with the analysis.

- What is the cardinality of the variables I'm working with?
- What is the frequency distribution of the categories in these variables?
- What is the distribution of the target variable in each category?
- Are there differences between continuous variables in the groups formed by the categories?
- Are there relationships among the categories of the different variables?

Thanks to descriptive metrics of analysis, the visualization of multiple correspondence analysis, the chi-squared tests, ANOVA tests, and more techniques, a broader picture of the complexity of the categorical data is taken, and a correct assessment of the indicated preprocessing scheme can be taken. It is important because the partition of the data that categorical variables perform can explain the intricate behavior of the target variable of classification.

The curse of dimensionality (Bellman, 2015) has many manifestations in the fitting of statistical learning models. Important explanatory variables can be nominal with high cardinality. Popular and easy to implement representations of these variables for model fitting are one-hot, target(Micci-Barreca, 2001), and weight of evidence encoding techniques. The former increases the dimensionality of the dataset with the most simple

representation and the later increase the probability of overfitting, without adding information on the relationships between variables and their categories.

With the rise of mainstream use of learning models to approach classification tasks, there has been a compromise of the quality of the representation of high cardinality categorical variables in exchange for efficiency. Non-structural approaches offer efficient representations yet simple and carry performance issues for the learning process. Treating categorical variables as networks, based on the current research in representation learning, it is possible to obtain a vectorial representation that can be used without having to jeopardize the performance of well-known classification models of mainstream use (Hamilton et al., 2017). It is with this hypothesis that the deductive process of this methodology it is developed.

Building a network representation of categorical data is the first step in this process. This is an approach that takes into account many factors about the features that want to be highlighted in the representation. For this methodology, the importance of the categories concerning the outcome of interest in the target variable is what wants to be captured. A description in subsection 1.1.1 was made on the different approaches to build networks out of non-network data, but as they create representations that are not elaborate enough to be acutely analyzed, they are not taken into account. A bipartite representation offers a more direct, easy to understand, and enriching structure. To the best of the knowledge available, it is a representation that has not been explored in literature for high cardinality categorical variables, and is described in the following section. A visual abstract of the methodology it is displayed in Figure 2.1.

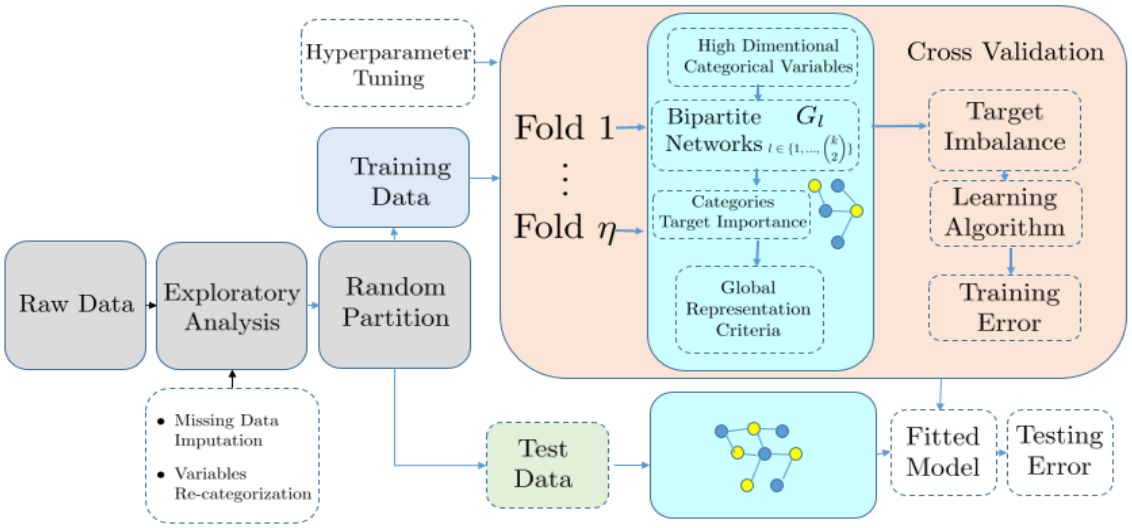

FIGURE 2.1. Visual abstract of the proposed methodology for categorical variables encoding

2.1 Bipartite Network Representation of Categorical Data

As described in section 1.1, networks are primarily composed of nodes and edges. Each of these components has its own set of characteristics. If, for example, a node in network G can be classified as type u or type v, then there are two types of nodes in network G.

Additionally, if G only has edges between nodes of type u and nodes of type v, then G is a bipartite network. In this work, only undirected bipartite networks are considered.

Formally, the notation used by Gao et al. (2018) for bipartite networks is $G = (U, V, E)$, where U and V denote the disjoint set of nodes of u and v and $E \subseteq U \times V$ denotes the set of edges between them. The nodes in set U are denoted by u_i with $i \in 1, ..., |U|$ and the nodes in V are denoted by v_j with $j \in 1, ..., |V|$. Each edge can have an associated weight with it, denoted $w_{ij} \in \mathbb{R}$, which describes the strength between node u_i and v_j. It is worth mentioning that if u_i and v_j are not connected then $w_{ij} = 0$, therefore an suitable representation for the edges weights is the matrix $W_{|U|\times|V|} = [w_{ij}]$.

Using this definition, if we have k categorical variables with high cardinality for the binary classification model, and take two out of them, say k_c and k_d $(c, d \in 1, ..., k)$ such that $c \neq d$. Then we can construct a bipartite network G_l between the categories of k_c and the categories of k_d. Formally, for the construction of G_l, the categories of k_c would form the nodes in set U, the categories of k_d would form the nodes in set V and the edges between them would be $E \subseteq U \times V$. In this sense, an individual who has category u of variable k_c and category v of variable k_d, would form an edge between nodes u and v with weight w. An example of this type of construction, for variables found in insurance claims datasets is displayed in Figure 2.2.

Given the previous definition of G_l, it can be seen that $l \in 1, ..., \binom{k}{2}$, that is, there are $\binom{k}{2}$ different bipartite networks to represent all the interactions between the k categorical variables in the dataset. Each variable k_c is represented in k networks, each one of them capturing different structural relationships with different variables k_d. Additionally, each node u_i is connected indirectly to another node of the same type when there is a n-steps path between them with all $w \neq 0$. The same occurs for nodes in variables v allowing for a diverse analysis of the information in the network. The methods described in the following sections are intended to detail the construction of the matrix of weights W, the featurization of the network (Baesens et al., 2015), and the use of its features for the classification task.

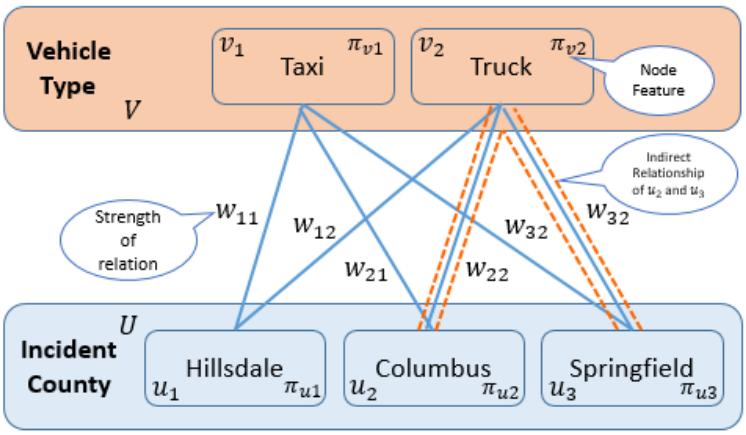

FIGURE 2.2. Example of bipartite network construction

2.2 Setting the Strength Among Categorical Variables

In a network structure, edge weights represent the strength of connectivity and are a measure of the sociality between the nodes (Baesens et al., 2015). There are two broad types of weights, *topological weights* which are defined in terms of the adjacency structure and *metric weights* which are defined in terms of distances between a representation feature. Examples of the former are weights established as a decreasing function of the shortest path distance, like $w_{ij} = \alpha \cdot exp\{-\beta \cdot d_s(u_i, v_j)\}$, or $w_{ij} = \alpha / d_s(u_i, v_j)^{\beta}$ (Okabe & Sugihara, 2012). And examples of the latter can be signed weights or a numerical representation of equal attributes between nodes u_i and v_j, also called the *common neighbor* weights. Normalized weights are often used in influence propagation studies, that is, weights that take values in the $[0, 1]$ range.

In the networks that concern the bipartite network encoding methodology, all the representations of the categories are relevant, but some to a higher degree given their frequency. Then, the weights in the network G are expected to showcase frequency of relations among categories of the variables. When there is a highly skewed distribution, methods like min-max and absolute scalers can contribute to mitigate discrepancies but they still would not be able to smooth outliers in the frequency of interactions. Scaling by dividing all of the frequencies by the interquartile range better deals with outliers by tightening the weight values. When there are no outliers the frequencies are better normalized by dividing them over the sum of all frequencies (Figure 2.3).

Another advantage of networks representations is that they can implement context features straightforward. When time is an important factor to take into account for the target variables, networks can display the passage of time by assigning recent links more weight on the network. For example, in an insurance company scenario, recent fraud cases should be more important than older ones. Using the exponential time decay function,$b_i = e^{-\gamma h}$, where γ is the decay constant and h is the time passed since the occurrence of the event(Van Vlasselaer et al., 2016), this can be represented.

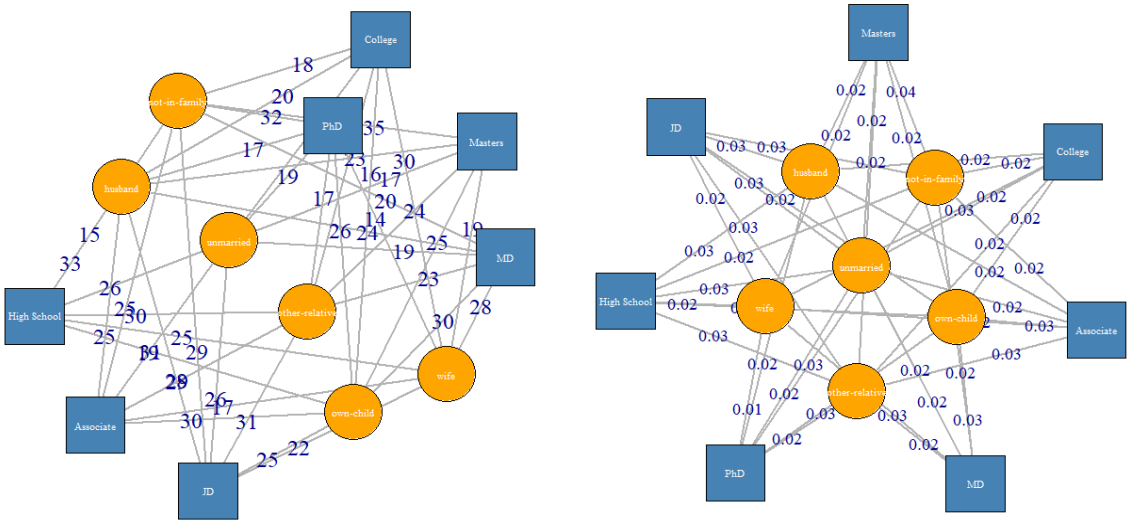

FIGURE 2.3. Example of weights normalization among categories.

2.3 Bipartite Network Featurization

Networks capture the strength of the relationship between the nodes trough the edges. The process of getting summary metrics for the nodes by the information captured trough the edges is called featurization of the network (Baesens et al., 2015) and it can provide a wide range of information depending on the desired objective of the analysis. In the setting of this work, the highlight is given to the nodes that have a higher proportion of the outcome of interest of the binary target variable. The proposed approach is an combination of a PageRank algorithm on bipartite networks along side nodes autocorrelation indicators of importance.

2.3.1 Local Measures of Outcome of Interest Importance

As described in subsection 1.1.2, there are several methods for analyzing and characterizing networks. In the following two segments a description of the use of specialized algorithms to assess the influence of the target variable in the network is detailed.

GOTCHA Score

GOTCHA is a scalable algorithm for network analysis that has been used to improve performance of traditional fraud detection models in social security context (Van Vlasselaer et al., 2016). It takes into account factors that Van Vlasselaer et al. (2016) have found relevant in their investigation of fraud events, i.e., they are uncommon, time-evolving, well considered and organized.

The algorithm makes uses of an importance score, that comes from the Personalized PageRank (Jeh & Widom, 2003) described in subsection 1.1.2, which has been used for many applications in recent years in the network data mining community (Jin et al., 2019). As the PageRank algorithm is defined for unipartite networks, there are some transformations required for matrix M in (1.14). Using the weighted adjacency matrix W of network G, create matrix Q, where

$$Q = \begin{pmatrix} 0_{|u|\times|u|} & W_{|u|\times|v|} \\ W^T_{|v|\times|u|} & 0_{|v|\times|v|} \end{pmatrix}. \tag{2.1}$$

This symmetric matrix is the extended weighted adjacency matrix of the bipartite network G. As in (1.14), Q must be column normalized, that is $\sum_i q_{ij} = 1, \ \forall \ j \in \{1, ..., |u| + |v|\}$. This normalized matrix will be called Q_{norm}.

The personalizing vector $\vec{r}$ defines nodes that are of interest inside the network. the following considerations are made :

- ✓ The focus must be to highlight the nodes that are more important for the outcome of interest of the target variable, then the vector $\vec{a}$ is defined such that a_i equals the percentage of outcomes of interest for category/node i in network G for in the training sample.
- ✓ In the PageRank algorithm nodes with high degree, spread their influence proportionally for each node, marginalizing its predominance. For the purpose of the

bipartite network encoding they should spread influence without this downsize. For tackling this situation, the vector $\vec{d}$ is defined such that it contains the degrees of all nodes in the network.

Taking into account the previous personalization helpers, we can define the personalization vector as $r = \vec{a} \odot \vec{d}$. Then the GOTCHA equations system is defined as

$$\vec{\xi} = \alpha \cdot Q_{norm}\, \vec{\xi} + (1 - \alpha) \cdot \vec{r}_{norm}. \tag{2.2}$$

Its solution is found with the power iteration algorithm described in section 1.1. This algorithm can be scaled to handle networks with millions of nodes, as was the case in the original paper (Van Vlasselaer et al., 2016).

The algorithm highlights direct and indirect network attributes that along side direct network features (node degree, betweenness, closeness, etc.) enrich the target variable analysis and can be used as special variables in learning algorithms (Van Vlasselaer et al., 2016).

Local Indicators of Importance Autocorrelation

As a result of the previous segment, there is an estimation of the importance for each node in the bipartite network G. Making use of the statistical devices developed in spatial statistics literature, the examination of the distribution of the importance score ξ in the network is made with local Moran's I (equation 1.15).

As can be seen in Figure 2.2, not only the direct relationship between the two set of nodes is of interest. The indirect relationship existing in a two step path tells information about how nodes of the same set relate. Making use of the second order adjacency weights matrix in the Moran's I calculation can indicate autocorrelation between same type of nodes regarding its importance score.

If the data presents an homogeneous variance across the network, making use of permutation test simulations, as proposed by Anselin (1995) is a way to approximate the distribution of the statistic I_i. In this, the $\xi_1, ..., \xi_{|U|+|V|}$ scores are considered fixed and permutations are randomly chosen. From these, the upper (I_i^{**}) and lower (I_i^{*}) critical values for each node of the random variable I_i are assumed, under the null hypothesis, with a significance level α. Then if an observed value of I_i is below (up) I_i^{*} (I_i^{**}), there is negative (positive) autocorrelation of ξ_i values in the neighborhood of nodes defined by the weighted adjacency matrix established (Okabe & Sugihara, 2012).

If the ξ scores are heterogeneous in mean and variance, as most times is expected in the research settings, performing a Monte Carlo test is preferred. A relevant issue on testing the local indicators of autocorrelation is the p-value threshold to properly reflect the type I error. For tackling this problem the use of the Bonferroni bound procedure is necessary, in which the threshold would is $\frac{\alpha}{N}$ (Caldas de Castro & Singer, 2006; Efron & Hastie, 2016).

Local Topology Measures

The following are measures that bring additional structural information about the relationship between categories of the variables. The theoretical maximum for applying the centralization procedure described in (1.7) is not yet established for bipartite networks.

Closeness

As defined in (1.6), the higher the value of this measure on a categories u for example, the more close it is to all other categories in the network. To normalize this value, we must take into account that the closest that u can be from any other category in v is $|V| + 2(|U| - 1)$ which is the distance from all other nodes in V and the distance from all other nodes in U. Then a normalized version of closeness for any category u and v is

$$closeness_u^* = \frac{|V| + 2 \cdot (|U| - 1)}{closeness(u)},\ closeness_v^* = \frac{|U| + 2 \cdot (|V| - 1)}{closeness(v)}. \tag{2.3}$$

Eigenvector Centrality

In this measure, as defined in subsection 1.1.2 the node's centrality score is proportional to the sum of the scores of its linked neighbors. That means that the centrality of category u depends of the centrality of all the categories that it is linked from variable V. This measure can not be obtained as for unipartite networks. For unipartite networks is obtained as the first eigenvector of the adjacency matrix of the network. For bipartite networks this matrix is of size $|U| \times |V|$. Then, in exchange, a singular value decomposition (SVD) of this matrix will provide the centrality scores for categories of variables U and V (Faust, 1997). The SVD theorem states that there exist orthogonal matrices $\widetilde{U}$ and $\widetilde{V}$ such that

$$A = \widetilde{U}_{|U|\times|U|} S_{|U|\times|V|} \widetilde{V}^T_{|V|\times|V|}, \tag{2.4}$$

and the scores are associated with the first column of the correspondent matrix $\widetilde{U}$ and $\widetilde{V}$ for U and V. This result is related with the principal eigenvectors of the matrices AA^T and A^TA which yields the same scores for U and V respectively. This scores have important properties and are a reliable description of the structural dominance that a category can have in network G.

2.3.2 Global Analysis of Network Measures

Alongside local measures, global measures allow a characterization of the general structure of the network. Their use depends on the characteristic that is wanted to be assessed. The measures described in subsection 1.1.2 and their interpretability for the bipartite networks created for the representation of categorical variables is discussed in this section.

Moran's I

A global assessment of Moran's I (Equation 1.17) is made on the ξ scores obtained from PageRank algorithm. This will allow to see in a general measure if there is an expected connectedness between categories with similar importance regarding the target variable. As stated by (Schabenberger & Gotway, 2004), under the assumption of a normal distribution with constant mean and variance on all the network for scores ξ, $E(I) = -\frac{1}{n-1}$.

When the assumption is not reasonable a correction of the mean trough least squares on the ξ scores is necessary so that the residuals can be used to calculate the Moran statistic.

Density

This measure of interconnectedness measures the strength of the diversity links between categories in network G. The higher the density, the higher is the variability of ties between variables U and V. Then, a low value of density indicates that the two variables are dependent, and a further chi-squared test between variables U and V could help in making this decision.

Eigenvector Centralization

Making use of the SVD of the adjacency matrix of G (2.4), the first eigenvectors of matrices $\widetilde{U}$ and $\widetilde{V}$ can be centralized so that a global measure of dominance can be used implemented. As stated when defining the centralization technique (1.7), the theoretical maximum needs to be defined. Eigenvector's centralization maximum for undirected networks is

$$\frac{(|i|-2)}{2^{\frac{1}{2}}}, \text{ where } i \in \{U, V\}. \tag{2.5}$$

Average Degree

For the bipartite network G, the degree of u would be the number of categories of variable V with which u appears at least one time in the dataset. The higher the degree, the more nodes in V it is related. Then, the global measure, average degree, will indicate how much variability exists between categories of the two variables. A low value of average degree for both variables indicate that the two variables are dependent, a further chi-squared test between variables U and V could help in making this decision (Borgatti & Halgin, 2011).

2.4 Networks Features for Supervised Classification

This section serves as a compilation of strategies to implement the bipartite network encoding of categorical variables described in the previous section. Up to this point, the analysis made with networks on categorical variables gave light on the importance of categories related to the outcome of interest in the target variable and the strength of associations between categorical variables. This information is then suitable to be used in the learning process, to enrich the model with vectorized network features of the categorial data, as performed in previous works done by Van Vlasselaer et al. (2016) and Baesens et al. (2015). This additional information can increase the accuracy of classification. However, a clear indication of this procedure is the topic of this section given that performing this technique in the incorrect step of the classification process would lead to unrealistic performance measures and overfitting.

In the statistical learning literature, there is a clear distinction between the use of training, validation, and test datasets. Training and validation datasets are used to adjust the hyperparameters that control the behavior of the statistical learning looking to estimate the bias and reduce variance. This is done through the estimation of the expected generalization error, and as such, it is quite optimistic related to what the real generalization error is. This is because improvements made based on the validation set will always

underestimate the error rate. To correctly estimate the generalization error the test set is classified using the model with less error in the validation dataset. Then, the estimated error in the test dataset is the best approximation to the expected generalization error, and no improvement on the model should be made with it (Friedman et al., 2001).

As can be seen, a clear difference between the three datasets is necessary, and no the training dataset should not contain any information about the test dataset. It is because of this, that the bipartite encoding of categorical variables should be made once the training, validation, and test samples are defined. As it is using the labels of the target variable, it should not take into account information from test and validation data in the training procedure. Once hyperparameters of the bipartite encoding are set such as the probability of restarting α in (2.2), the encoding is learned on training and is used to transform the categorical variables in validation and test set. This will lead to a realistic estimation of the expected generalization error rate on validation and test.

If a problem of class imbalance it is present in the target variable, procedures such as SMOTE-NC (Chawla et al., 2002), which is an adaptation of the *synthetic minority over-sampling technique* to handle both categorical and numerical variables, can be used on the original representation of the variables. The bipartite encoding of the categorical variables is made on the original distribution of the dataset and the result of the SMOTE-NC procedure, or any procedure implemented for under or over-sampling, it is encoded with the already learned bipartite encoding.

An effective method for estimating the expected classification error is cross-validation (Friedman et al., 2001). For each fold in the procedure, the bipartite encoding of categorical variables can be performed. The number of folds is open to the user's decision, although 5 or 10 empirically show stable approximations (James, Witten, Hastie & Tibshirani, 2013). Cross-Validation is a recommended procedure given that it can help to gain information on the correct setting of parameters for the network representation of categorical variables Figure 2.1.

In general, the challenges that the proposed bipartite encoding methodology faces are inherent to the nature of the network. The first challenge addresses the partitioning of the data set into training and test set. Important links and strengths may be lost in this procedure for the encoding. However, cross-validation can help to reduce the impact of this by averaging the error on encodings learned in different settings of the networks created, according to the folds defined. Also, the way the bipartite networks are constructed using categories as entities, makes them more likely to be dense, and less likely to have unconnected nodes. However, a second challenge is the presence of unconnected nodes in the bipartite networks. If there were at some point unconnected categories in the bipartite network, the PageRank algorithm would not inflate their importance given that it can restart its random walks at different categories of the network in the learning process, avoiding to get stuck.

As the representation that the dyadic encoding of categorical variables creates, say for categories of variable U, a matrix of size $|U| \times \binom{k}{2}$, that captures the importance score of the variables in all the bipartite networks it was represented. Then, the mean importance for each category is the network summary measure that can add value to the classification process.

Procedures for finding the best subset of variables for the performance metric desired can be applied as usual after the encoding. This can be performed with recursive stepwise

procedures, tree-based methods, and regularization, depending on the learning algorithm implemented and the metric used for the selection. It is important to note that the regularization procedure for *logistic regression* is not modified by the encoding of the categorical variables, and it can help to improve the model generalization error when a high variance is found when testing the fitted model.

As a final note, when using statistical learning methods with classification purposes, it helps to focus the efforts in optimizing a reduced set of performance metric. This can be the accuracy, the AUC, the F1 score, etc. Each of them can highlight an important metric for the real life scenario it is implemented.

2.5 R Implementation

The proposed approach for discovering relationships among the categories of the variables trough bipartite networks may seem an iterative approach that requires a big setup and time for applying it. Thanks to the experience gained in the development of this work a set of functions for the implementation of this methodology in R is available at `https://github.com/manumoreno-sl/bip_nets_analysis_binaryclassif`. It performs:

- the dyadic representation in bipartite networks of the user specified, high cardinality categorical variables.
- the GOTCHA importance score related to the target variable for the categories making use of the PageRank algorithm.
- the autocorrelation analysis with Moran's I statistic between the categories of the two variables U and V (first order), and among categories of the same variable (second order).
- structured data of the global metrics for the $\binom{k}{2}$ bipartite networks assessment

The time performance for the study case, that will be detailed in the following chapter, did not required of high computational resources. But the coded algorithms are expected to be scalable in datasets of higher dimensions. The main libraries used are igraph (Csardi & Nepusz, 2006), dplyr (Wickham et al., 2018) and assortnet (Farine, 2016).

CHAPTER 3

Case Study

In this chapter an statistical analysis is performed on a car insurance's claims dataset using the methodology detailed in chapter 2 for supervised fraud detection.

3.1 Insurance Problem Scenario

When an insurance company underwrites a policy it is making it self responsible for the losses the client may encounter due to unforeseen events. The claims rate it is an important factor in the risk assessment and pricing life cycle of an insurance company. Claims departments are in charge of receiving, documenting, assessing, and compensating the losses of the clients. It is in this process were a fraud evaluation is conducted. The evaluation is made to determine if the loss event was set up for the economical benefit of the insured. In practice, for car insurance departments the screening process consists of:

- An initial evaluation is performed by claims experts, who assess the cost of all damages. If the expert does not founds irregularities the case follows is payment process.
- If an irregularity is found, an expenses analysis is made. If the cost of paying the suspicious claim is less than the cost of the inquiry process, it is paid to the client.
- If the expense of an investigation is lower than the potential reward, an specialized third party entity is contracted to make it and rule the verdict.

Statistically, the labels for fraud cases under a binary supervised classification setting, are determined by the specialized third-party entity. Claims that don't go through step three can be noisy for the classification purpose. For example, in step one, internal fraud can cause an intentional payment of irregular claims and contaminate the sample. Even though external fraud can be committed by the third-party company the noise in this step is considerably smaller than the former case. Unfortunately taking claims trough step three is an expensive and time-consuming way for getting reliable labels for fraud detection. Then, in practice, the first characteristic of a dataset for supervised classification of claims is that it has a low sample size of labeled data.

In the screening process all available data is required, this includes the information of the client, the policy, and the insured vehicle. The second characteristic of datasets used for fraud detection is that most of the variables to take into account are nominal with high cardinality. A few ordinal and quantitative are also present. Depending on the case, the high cardinality of nominal variables can be addressed in several ways. Examples range from the empirical grouping of categories to supervised methods of encoding (Micci-Barreca, 2001; Kuhn & Johnson, 2019) which can have advantages for efficiency purposes at cost of interpretability and overfitting.

Besides of supervised algorithms for fraud detection (Artís et al., 2002), unsupervised approaches for car insurance have been developed focused on anomaly detection (Nian et al., 2016) and hybrid approaches (Lindholm, 2014). Still, experimentally these alternatives are surpassed in performance by supervised methods so in a scenario of extreme absence of fraud labels they would be better suited. The third characteristic of fraud datasets is that fraudulent claims have a low frequency related to the non-fraudulent labels. Over-sampling and under-sampling techniques can be implemented taking into account the mixed type of data present in the dataset (Chawla et al., 2002).

Ideally, with the appropriate information, client networks could be created to detect fraud communities and patterns (Šubelj et al., 2011a). However, this is not a real scenario for insurance companies. In practice, the imbalanced-class high-dimensional low-sample size datasets are usual. With this setting, client networks would be unconnected, sparse, and end up being uninformative. However, statistical network analysis can cope with these challenges and bring relational information into the classification analysis with the use of bipartite networks as described in the methodology.

3.2 Dataset Description

The confidentiality and data protection regulations prevent an open sharing of datasets from the insurance companies with the statistical learning community. Although this limits the amount of research on the specific case of automobile claims data, some researches have partnered with companies for their research (Bodaghi & Teimourpour, 2018).

The data used for this case study is, to the best of the knowledge, the best open access one that most resembles the structure of the typical dataset for fraud detection in automobile insurance claims. It is available on the Kaggle platform to the public. It contains anonymized information of 1000 automobile claims made to a United States-based insurance company in the year 2015 from January to March. For each claim, there is a label that indicates if the claim was detected fraudulent. From the exploratory analysis, this dataset has the characteristics of a stratified sample, where each day on average there are 17 claims reported, of which on average 25% are labeled as fraud daily. This being said, as mentioned in section 3.1, although the proportion of daily fraud cases is high relative to a real scenario, it resembles that not a high percentage of daily claims can be deemed as clean.

The dataset has information about the three relevant aspects of the insurance claims screening. That is the client, the policy, and the claim. From the statistical point of view, although it has a relatively small dataset, it contained less than 1% of missing data and much information for them. There are four groups of variables present in the dataset.

- **Dates**
  Incident date and policy bind date.
- **Numerical**
  Related to the policy, months as customer, annual premium, capital gains, capital loss, automobile year. Related to the client, age. Related to the claim, hour of the day, total claim amount, injury claim, property claim, vehicle claim.
- **Categorical of low cardinality** (6 categories or less)
  Related to the policy, state, umbrella limit. Related to the client, sex , civil state, education level. Related to the claim, incident type, collision type, severity, authorities contacted, vehicles involved, property damage, bodily injuries, witnesses, police report available, fraud reported.
- **Categorical of higher cardinality**
  Related to the policy, automobile maker (14), automobile version (39). Related to the client, occupation (14), hobbies (20). Related to the claim, incident county (34).

3.3 Exploratory Analysis

In this key step for the implementation of the methodology an assessment of outliers is performed on the continuous variables and a re-categorization of the low frequency categories for categorical features. Additionally, the distribution of the target variable is assessed by the features in the data looking for drivers in the difference between fraudulent and clean claims. For example, in Figure 3.1 an assessment of the hour of occurrence of the claims by groups shows that, on average fraudulent cases tend to happen earlier than clean claims in the dataset.

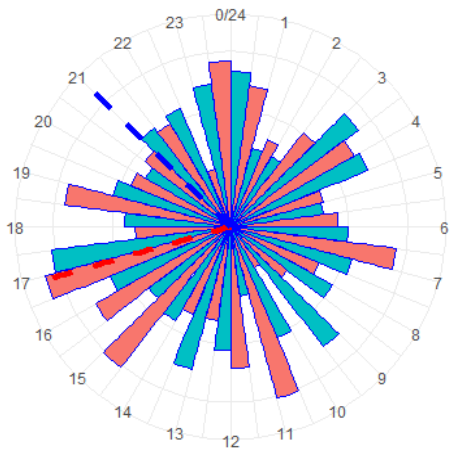

FIGURE 3.1. Distribution of fraud cases by hour of the incident. In red are the fraud claims, in blue the clean claims. The doted lines denote the mean for each group.

3.4 Fraud Classification

In this section, an experimental testing of the bipartite network encoding of high cardinality variables is performed. The considerations are the following:

- The data is separated in two subsets, the training data set (80%) and the test data set (20%). Each scenario will make use of the same training data set and will test on the same testing dataset.
- Three scenarios will be built and compared. First, a full use of the variables available in the data set in their original state. Second, use of the bipartite network encoding for the high cardinality categorical variables. Third, use of target encoding (Micci-Barreca, 2001) for the high cardinality categorical variables, given that is mainstream use in the machine learning community.
- Two methodologies will be used to perform the supervised classification in each scenario, they are *logistic regression* and *random forest*.
- In each scenario a 5-fold cross validation strategy for estimating the expected generalization error will be carried. The folds are all the same in the three scenarios.
- As seen in the exploratory analysis, there is a class imbalance of the fraud cases. This can be treated with over or under-sampling for mixed type data (Chawla et al., 2002), however in neither of the scenarios this will be carried for not adding noise for the comparison of the strategies. This imbalance inflates metrics such as area under the curve (AUC) of the receiver operating characteristic (ROC) functions because of the prevalence of true negatives. More realistic metrics on imbalanced classification scenarios are the AUC of the precision - recall function (AUC-PR), or the absolute Matthews Correlation Coefficient (MCC) (Sofaer et al., 2019). The former will be the performance metric of comparison.

In the following sections, a description of the findings of the experimental process is detailed.

3.4.1 Scenario 1: No Use of encoding

Logistic Regression

In the first trials of this model, the signs of overfitting were evident when the cross-validation training AUC-PR score was 15% higher than the testing AUC-PR score. A *elastic net* penalty strategy for regularization, that is $\lambda = 0.2$ and $\alpha = 0.5$, is performed to control the variance and handle problems of collinearity.

The testing AUC-PR score of this model is 0.649612, with a 8% estimated variance. In Table 3.1 the correspondent confusion matrix is showed.

TABLE 3.1. Test confusion matrix for logistic regression with non encoded variables.

	Predicted	**Class**	
True	clean	fraud	**Error**
clean	131	19	0.126667
fraud	4	45	0.081633
Totals	135	64	0.115578

Random Forest

The regularization approach taken to avoid overfitting and hyperparamter selection for this model was a grid search. The combination of parameters to evaluate is the number of trees, the number of variables to sample for each split, and the maximum depth of the trees.

The testing AUC-PR score of this model is 0.6429, with a 3% estimated variance.In Table 3.2 the correspondent confusion matrix is showed.

TABLE 3.2. Test confusion matrix for random forest with non encoded variables.

	Predicted	**Class**	
True	clean	fraud	**Error**
clean	132	18	0.120000
fraud	6	43	0.122449
Totals	138	61	0.120603

3.4.2 Scenario 2: Bipartite Encoding

The encoding is performed on the high cardinality variables mentioned in the data set description. This encoding is performed only with the training set in each fold of the cross validation. The encoded variables are transformed in the test data set according to the encoding learned on the training.

As a summary of the methodology, the categories of the variables are encoded according to a dyadic bipartite network analysis of importance, which is determined with the GOTCHA algorithm (Equation 2.2). In this, a hyperparameter needs to be set, the time decay constant. A depiction of how this parameter influences the importance of the categories is showed in Figure 3.2. For the purpose of this experiment, the parameter will be set to 0, as we want all fraud cases to be weighted equally, regardless of time passed. The R functions described in section 2.5 provide the importance of each category for each dyad of variables encoded. This importance score is mean aggregated for each category. This is the score used for the binary classification purpose.

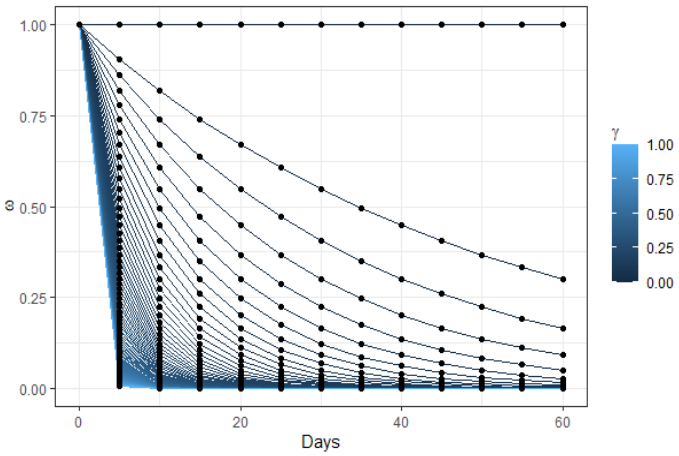

FIGURE 3.2. Time decay parameter calibration.

As stated by Van Vlasselaer et al. (2016) and Baesens et al. (2015), the use of the GOTCHA algorithm can improve the performance of fraud classification models by interacting alongside intrinsic features in the data. It is because of this that the encoding joins the categorical variables in the training process for this experiment. This can be done given that the high cardinality variables are not greater in order of magnitude, also an empirical removal of the categorical variables decreased the performance of the classifiers.

Logistic Regression

The testing AUC-PR score of this model is 0.6282, with an 8% estimated variance.In Table 3.3 the correspondent confusion matrix is showed.

TABLE 3.3. Test confusion matrix for logistic regression with bipartite encoding.

	Predicted	**Class**	
True	clean	fraud	**Error**
clean	127	23	0.153333
fraud	5	44	0.102041
Totals	132	67	0.140704

Random Forest

The testing AUC-PR score of this model is 0.6743241, with a 1% estimated variance.In Table 3.4 the correspondent confusion matrix is showed.

TABLE 3.4. Test confusion matrix for random forest with bipartite encoding.

	Predicted	**Class**	
True	clean	fraud	**Error**
clean	128	22	0.146667
fraud	4	45	0.081633
Totals	132	67	0.130653

Among the numerical variables, the encoded categorical ranked higher in importance for purity of the splitting process in the trees. In Table 3.5 a sample of the result of the bipartite encoding is showed. It allows for a notion of importance among categories.

In Table 3.7, the scores of importance are showed. One of the reasons for implementing bipartite encoding is that it can be analyzed with from a statistical and network point of view. In Table 3.8 can be seen the global measures defined in subsection 2.3.2 for this variable. The Moran autocorrelation measure indicates that the education levels do not behave in the same way as the relationships relative to fraud. The eigencentrality measure for education level (U), shows no strong dominance by any particular category, while variables like incident county (V) show that there is some. The weighted assortativity score shows that highly connected nodes tend to connect to less connected nodes. Measures like order, size, density, diameter and average distance allow a depiction of the topology of the network from which the importance was calculated.

TABLE 3.5. Standardized bipartite network encoding of auto Maker.

Auto maker	Importance
Dodge	1.08
Suburu	1.07
Saab	0.96
Chevrolet	0.93
Ford	0.52
BMW	0.36
Nissan	0.29
Audi	0.04
Mercedes	-0.17
Volkswagen	-0.29
Toyota	-0.63
Accura	-0.80
Jeep	-1.25
Honda	-2.10

TABLE 3.6. Standardized bipartite network encoding of occupation .

Occupation	Importance
machine-op-inspct	1.83
prof-specialty	0.99
exec-managerial	0.95
tech-support	0.74
sales	0.54
craft-repair	0.46
transport-moving	0.26
armed-forces	-0.24
priv-house-serv	-0.36
other-service	-0.37
protective-serv	-0.83
adm-clerical	-0.88
farming-fishing	-1.44
handlers-cleaners	-1.65

TABLE 3.7. Standardized bipartite network encoding of education level.

Education level	Importance
JD	1.38
High School	0.97
MD	0.23
Associate	0.05
Masters	-0.19
PhD	-1.09
College	-1.36

TABLE 3.8. Bipartite network's global measures for Education Level variable.

U	V	Moran I	Eig. U	Eig. V	Assort.	Order	Size	Diam.	Avg.Dist.	Dens.	Match
education level	auto year	0.97	0.08	0.04	-0.96	28	147	2.00	1.61	1.00	7
education level	auto model	0.97	0.07	0.11	-0.97	46	265	3.00	1.75	0.97	7
education level	incident hour	0.96	0.06	0.06	-0.96	31	167	3.00	1.64	0.99	7
education level	hobbies	0.96	0.07	0.07	-0.95	27	140	2.00	1.60	1.00	7
education level	auto make	0.94	0.07	0.04	-0.93	21	98	2.00	1.53	1.00	7
education level	occupation	0.91	0.06	0.09	-0.91	21	98	2.00	1.53	1.00	7
education level	incident county	0.78	0.05	0.27	-0.91	41	186	4.00	1.87	0.78	7
education level	relationship	0.46	0.07	0.06	-0.46	13	42	2.00	1.46	1.00	6

3.4.3 Scenario 3: Target Encoding

This highly popular encoding technique makes use of the observed ratio of the fraud cases at each category as a prior for is value. For this dataset, the performance decreased dramatically when the categories were replaced by the encoding. Then, for this reason they will join the categorical variables as in the case of the bipartite encoding in scenario 2.

Logistic Regression

The testing AUC-PR score of this model is 0.6440385, with a 4% estimated variance.In Table 3.9 the correspondent confusion matrix is showed.

TABLE 3.9. Test confusion matrix for logistic regression with target encodign variables.

	Predicted	**Class**	
True	clean	fraud	**Error**
clean	132	14	0.093333
fraud	12	37	0.244898
Totals	148	51	0.130653

Random Forest

The testing AUC-PR score of this model is 0.6317824, with a 5% estimated variance. In Table 3.10 the correspondent confusion matrix is showed.

TABLE 3.10. Test confusion matrix for random forest with target encoding variables.

	Predicted	**Class**	
True	clean	fraud	**Error**
clean	132	18	0.120000
fraud	5	44	0.102041
Totals	137	62	0.115578

3.4.4 Performance Comparison

In Table 3.11, can be seen that, by AUC-PR score the model with the best overall performance is the one that uses the bipartite encoding technique on a Random Forest. It also is the one with the less variance, that is, the one model who's training AUC-PR score was closer to the testing score. However the use of the bipartite encoding in the logistic regression model did not improve the performance metrics.

TABLE 3.11. Summary table of performance metrics.

Encoding	**Model**	**AUC-PR**	**Variance**	**Fraud Error**
No	LR	0.6496	0.08	0.081633
No	RF	0.6429	0.03	0.122449
Bipartite	LR	0.6282	0.08	0.102041
Bipartite	RF	0.6743	0.01	0.081633
Target	LR	0.6440	0.04	0.244898
Target	RF	0.6317	0.05	0.102041

It can be seen that the target encoding does not affect the performance of the logistic regression as it does for the random forest model. The AUC score indicates that the bipartite encoding in random forest is best at lowering the false positive rate than the other models. However, neither of the encoding techniques are able to increase the predictive power of fraud in testing, being the bipartite encoding on the random forest the best along side the logistic regression with no encoding. In Figure 3.3 can be seen that the Precission-Recall curve is better for the scenario 1 than for the others, while in the random

forest PR curves (Figure 3.4) the higher is for the bipartite encoding. In both figures, the straight line indicates a completely randomn classification.

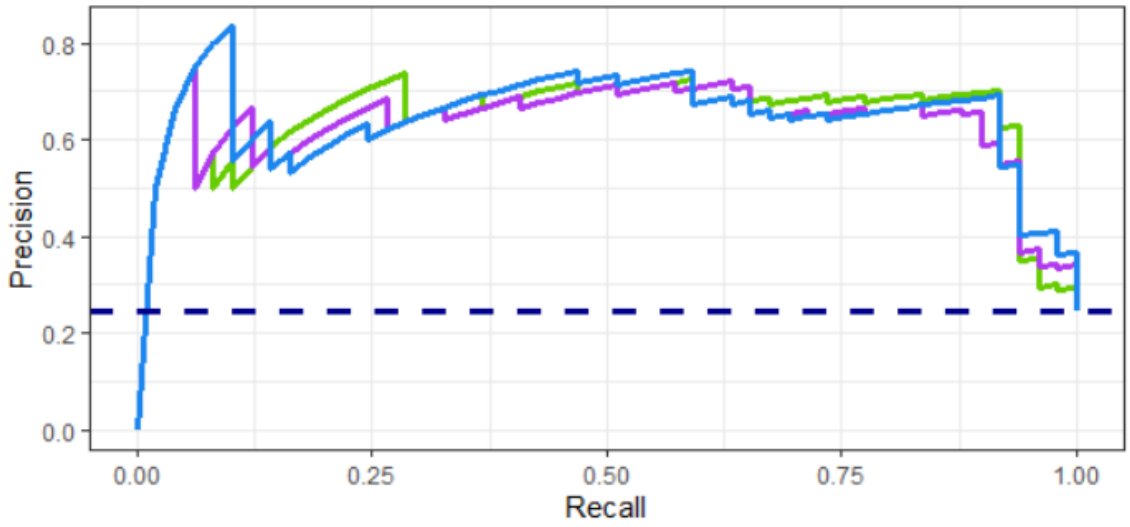

FIGURE 3.3. PR curves of the logistic models for scenario 1 (green), scenario 2 (purple), scenario 3 (blue).

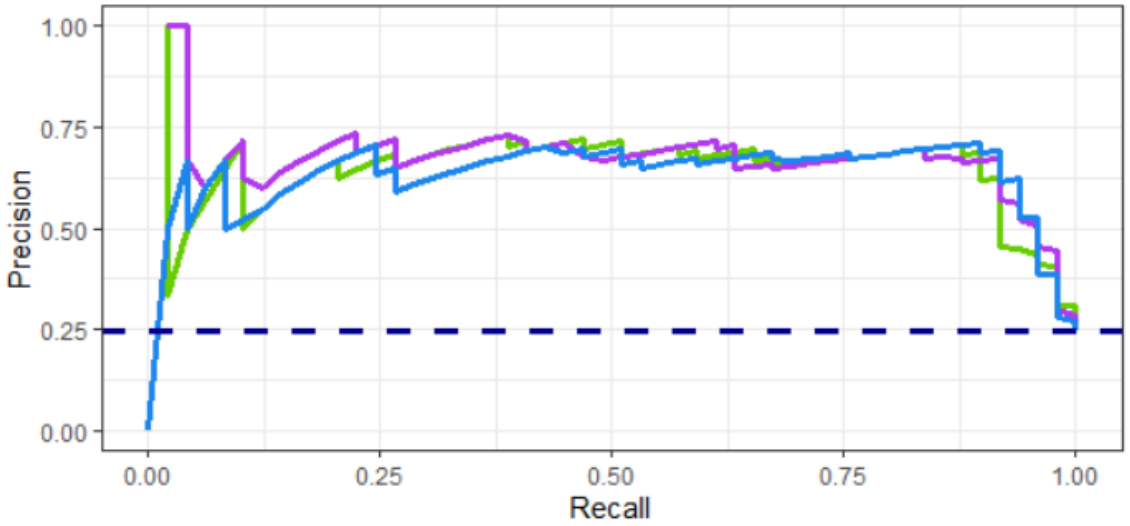

FIGURE 3.4. PR curves of the random forest models for scenario 1 (green), scenario 2 (purple), scenario 3 (blue).

Discussion

This work intends to propose a bipartite network-based representation to add more value from high cardinality variables into binary classification models. This inductive research was set to take advantage of the rich information structure that networks bring into its study. The use of bipartite networks is a key part of the methodology proposal. It is the most studied and proven worthy type of representation for multipartite networks, the amount of research that is made on the subject supports this claim. A k-mode network representation of the categorical features requires to extend the definition of the bipartite (two-mode) network to multipartite (k-mode) network along with its methods. Some works generalize some algorithms from bipartite networks (Lind et al., 2005) and others that develop special ones (Phillips, 2015), but still there is not enough literature to support that it improves the performance of classification, as in the case of the bipartite networks (Van Vlasselaer et al., 2016).

Regarding the use of bipartite networks, there are some voices in the community that claim that a better way to deal with naturally formed bipartite networks it is to project each of the types and analyzing the weighted unipartite representations that are created (Opsahl et al., 2010). This was considered in the early stages of the research but this faces more challenges than opportunities in the experimental work made. The first problem is the setting of the projection weights, there are plenty of ways in which this can be done, say by overlap counting, Jaccard's similarity, simple matching, Pearson's correlation, Yule's Q, etc. Each one of these will highlight in different degrees the structure of the original indirect relation of nodes. This adds more complexity in the correct use of the measure. A better approach, as addressed by Kitsak & Krioukov (2011) and as it was treated in this work is to treat the weighted bipartite network and study its features.

The proposed approach to analyze high cardinality categorical data has many advantages. Starting with its flexibility in the structural information to extract for the learning purpose. Compared to vector-based representations they can take advantage of many topological inherent structures (Silva & Zhao, 2016). It also can take into account, trough first-order interactions, its relations with other variables, and trough second-order interactions its relation with other categories of the same variable. The use of the fraud orientated GOTCHA algorithm serves the purpose of insurance fraud. However, many scoring procedures can be implemented trying to highlight different aspects of the relationships in the bipartite network. Regarding the use of deep neural networks for encoding the relationships, they are the current state of the art methods for encoding variables and finding structures trough convolutions on networks. In this work, they were not considered as it was preferred an interpretable approach that was not more complex than the objective binary classification goal.

It is worth mentioning as well that the global and local analysis that was performed made use of devices that are naturally entangled with network structures, such as Moran's I and the measures topological measures. It is recommended the use of this type of measures that make sense on network data than just apply statistical algorithms on the numerical attributes. For example, when dealing with inference on attributes of the network it would be necessary, to apply network inference procedures like conditional uniform graphs, quadratic assignment, or stochastic actor-oriented modeling, to name a few. Networks have a rich stochastic structure that can be well used for making the best out of the networked and non-networked data.

Regarding the improvement of the methodology proposed for the classification goal, a measure of noise introduced by the use of the features created is a natural next step. This would be a regularization procedure for the bipartite representation formation that can help to select the most relevant ones out of the available. Some work on this topic can be found on the representation learning literature, but its development in the context of bipartite networks it is still a topic of debate in the bipartite network case. Further experimentation is also required, with simulated data to discover and assess the properties and effects this encoding has on the performance of classification purposes.

Conclusions

1. It is possible to create bipartite network representations from non-networked data that can highlight the inner structure of the variables under study.

2. Interaction patterns based on the measures and properties of the bipartite networks were found. The use of numerical analysis to address behaviors of categorical data brings complementary information that bipartite networks can best help to obtain.

3. The adequate way of implementing network-based features, that use information from the target variable for classification purposes, is on the training data set. That is, avoiding using any data from the test set to reduce an under-estimation of the generalization error.

4. The diversity and the flexibility of measures that can be extracted from network data allows for more in-depth knowledge of the high cardinality categorical variables, which are sometimes overlooked by its complexity in statistical learning models.

5. The proposed methodology offers a representation of importance related to the target variable. The learning procedure uses random walks to assess the most relevant categories for the flow of information. There are more complex learning methods in the literature of representation learning embedding techniques that can offer alternatives for more complex learning of features of the categorical variables.

Future Work

An implementation of this methodology to a broader set of databases with different structures it is deemed to fully understand its performance. It is presumed that the algorithms implemented scale well to larger datasets, so addressing this problem when handling a larger amount of categories in the networks is something that needs to be assessed to consider the most favorable scenarios for this approach.

The methodology proposed keeps a close connection with spectral clustering methods on networks. Addressing how this clustering technique can be suited to bipartite networks and see their performance on benchmark datasets it is a natural next step for improvement of the methodology. Also consideration of the multipartite approach it is needed. This representation, although more complex and abstract could fully represent a set of k categorical features in one k-mode network, which surely would highlight missing insights in the bipartite approach. This being said, considering a network handling of both numerical and categorical data at the same time is a field yet to explore.

With the experience gained in this work, there are plenty of interesting topics to address in future research. The field of representation learning is asking challenging questions about what is the best way to learn from data. The developments made to extract relevant information from complex structures like networks is a promising topic with several applications. Its close connection with methods for text data, which in a general way, can be seen as linear networks, makes it compelling for a lot of purposes in industry and research.

Bibliography

Aggarwal, C. C. (2011). An introduction to social network data analytics, *Social network data analytics*, Springer, pp. 1–15.

Akoglu, L., McGlohon, M. & Faloutsos, C. (2010). oddball: Spotting anomalies in weighted graphs, *PAKDD*.

Akoglu, L., Tong, H. & Koutra, D. (2014). Graph based anomaly detection and description: a survey, *Data Mining and Knowledge Discovery* **29**: 626–688.

Alzahrani, T. & Horadam, K. J. (2016). Community detection in bipartite networks: Algorithms and case studies, *Complex systems and networks*, Springer, pp. 25–50.

Amelio, A. & Pizzuti, C. (2014). Community detection in multidimensional networks, *2014 IEEE 26th International Conference on Tools with Arti cial Intelligence*, IEEE, pp. 352–359.

Anselin, L. (1995). Local indicators of spatial association—lisa, *Geographical analysis* **27**(2): 93–115.

Artís, M., Ayuso, M. & Guillen, M. (2002). Detection of automobile insurance fraud with discrete choice models and misclassified claims.

Backstrom, L. & Leskovec, J. (2011). Supervised random walks: predicting and recommending links in social networks, *Proceedings of the fourth ACM international conference on Web search and data mining*, pp. 635–644.

Baesens, B., Van Vlasselaer, V. & Verbeke, W. (2015). *Fraud analytics using descriptive, predictive, and social network techniques: a guide to data science for fraud detection*, John Wiley & Sons.

Bellman, R. E. (2015). *Adaptive control processes: a guided tour*, Vol. 2045, Princeton university press.

Bengio, Y., Courville, A. & Vincent, P. (2013). Representation learning: A review and new perspectives, *IEEE transactions on pattern analysis and machine intelligence* **35**(8): 1798–1828.

Bodaghi, A. & Teimourpour, B. (2018). The detection of professional fraud in automobile insurance using social network analysis, *arXiv preprint arXiv:1805.09741* .

Borgatti, S. P. & Halgin, D. S. (2011). Analyzing affiliation networks, *The Sage handbook of social network analysis* **1**: 417–433.

Bravo, C. & Óskarsdóttir, M. (2020). Evolution of credit risk using a personalized pagerank algorithm for multilayer networks, *arXiv preprint arXiv:2005.12418* .

Breiman, L. (2001). Random forests, *Machine learning* **45**(1): 5–32.

Brin, S. & Page, L. (1998). The anatomy of a large-scale hypertextual web search engine.

Caldas de Castro, M. & Singer, B. H. (2006). Controlling the false discovery rate: a new application to account for multiple and dependent tests in local statistics of spatial association, *Geographical Analysis* **38**(2): 180–208.

Castañeda, L. B., Arunachalam, V. & Dharmaraja, S. (2012). *Introduction to probability and stochastic processes with applications*, John Wiley & Sons.

Cerda, P. & Varoquaux, G. (2020). Encoding high-cardinality string categorical variables, *IEEE Transactions on Knowledge and Data Engineering* .

Chawla, N. V., Bowyer, K. W., Hall, L. O. & Kegelmeyer, W. P. (2002). Smote: synthetic minority over-sampling technique, *Journal of arti cial intelligence research* **16**: 321–357.

Chen, T., Tang, L.-A., Sun, Y., Chen, Z. & Zhang, K. (2016). Entity embedding-based anomaly detection for heterogeneous categorical events, *arXiv preprint arXiv:1608.07502* .

Chen, W., Chen, Y., Mao, Y. & Guo, B. (2013). Density-based logistic regression, *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 140–148.

Cheng, V., Li, C.-H., Kwok, J. T. & Li, C.-K. (2004). Dissimilarity learning for nominal data, *Pattern Recognition* **37**(7): 1471–1477.

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

Cunningham, D., Everton, S. & Murphy, P. (2016). *Understanding dark networks: A strategic framework for the use of social network analysis*, Rowman & Littlefield.

Dobson, A. J. & Barnett, A. (2008). *An introduction to generalized linear models*, CRC press.

Efron, B. & Hastie, T. (2016). *Computer age statistical inference*, Vol. 5, Cambridge University Press.

Farine, D. (2016). *assortnet: Calculate the Assortativity Coe cient of Weighted and Binary Networks*. R package version 0.7.6.
URL: *https://CRAN.R-project.org/package=assortnet*

Faust, K. (1997). Centrality in affiliation networks, *Social networks* **19**(2): 157–191.

Fienberg, S. E. (2012). A brief history of statistical models for network analysis and open challenges, *Journal of Computational and Graphical Statistics* **21**(4): 825–839.

Friedman, J., Hastie, T. & Tibshirani, R. (2001). *The elements of statistical learning*, Vol. 1, Springer series in statistics New York.

Gao, M., Chen, L., He, X. & Zhou, A. (2018). Bine: Bipartite network embedding, *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pp. 715–724.

Getoor, L. (2005). Link-based classification, *Advanced methods for knowledge discovery from complex data*, Springer, pp. 189–207.

Goodfellow, I., Bengio, Y. & Courville, A. (2016). *Deep learning*, MIT press.

Gyongyi, Z., Garcia-Molina, H. & Pedersen, J. (2004). Combating web spam with trustrank, *Proceedings of the 30th international conference on very large data bases (VLDB)*.

Hamilton, W. L., Ying, R. & Leskovec, J. (2017). Representation learning on graphs: Methods and applications, *arXiv preprint arXiv:1709.05584* .

He, X., Gao, M., Kan, M.-Y. & Wang, D. (2016). Birank: Towards ranking on bipartite graphs, *IEEE Transactions on Knowledge and Data Engineering* **29**(1): 57–71.

James, G., Witten, D., Hastie, T. & Tibshirani, R. (2013). *An Introduction to Statistical Learning: with Applications in R*, Springer Texts in Statistics, Springer New York.
URL: *https://books.google.cl/books?id=qcI_AAAAQBAJ*

Jeh, G. & Widom, J. (2003). Scaling personalized web search, *Proceedings of the 12th international conference on World Wide Web*, pp. 271–279.

Jin, W., Jung, J. & Kang, U. (2019). Supervised and extended restart in random walks for ranking and link prediction in networks, *PloS one* **14**(3): e0213857.

Kitsak, M. & Krioukov, D. (2011). Hidden variables in bipartite networks, *Physical Review E* **84**(2): 026114.

Kley, O., Klüppelberg, C. & Reinert, G. (2016). Risk in a large claims insurance market with bipartite graph structure, *Operations Research* **64**(5): 1159–1176.

Kolaczyk, E. D. (2009). *Statistical Analysis of Network Data [electronic Resource]: Methods and Models*, Springer.

Koller, D., Friedman, N., Džeroski, S., Sutton, C., McCallum, A., Pfeffer, A., Abbeel, P., Wong, M.-F., Heckerman, D., Meek, C. et al. (2007). *Introduction to statistical relational learning*, MIT press.

Kuhn, M. & Johnson, K. (2019). *Feature engineering and selection: A practical approach for predictive models*, CRC Press.

Li, Y., Yan, C., Liu, W. & Li, M. (2018). A principle component analysis-based random forest with the potential nearest neighbor method for automobile insurance fraud identification, *Applied Soft Computing* **70**: 1000–1009.

Lin, W., Wu, Z., Lin, L., Wen, A. & Li, J. (2017). An ensemble random forest algorithm for insurance big data analysis, *Ieee access* **5**: 16568–16575.

Lind, P. G., Gonzalez, M. C. & Herrmann, H. J. (2005). Cycles and clustering in bipartite networks, *Physical review E* **72**(5): 056127.

Lindholm, A. (2014). A study about fraud detection and the implementation of suspect - supervised and unsupervised erlang classifier tool.

Lucena, B. (2020). Exploiting categorical structure using tree-based methods, *arXiv preprint arXiv:2004.07383* .

Micci-Barreca, D. (2001). A preprocessing scheme for high-cardinality categorical attributes in classification and prediction problems, *ACM SIGKDD Explorations Newsletter* **3**(1): 27–32.

Moeyersoms, J. & Martens, D. (2015). Including high-cardinality attributes in predictive models: A case study in churn prediction in the energy sector, *Decision support systems* **72**: 72–81.

Nian, K., Zhang, H., Tayal, A., Coleman, T. & Li, Y. (2016). Auto insurance fraud detection using unsupervised spectral ranking for anomaly, *The Journal of Finance and Data Science* **2**(1): 58–75.

Nisbet, R., Elder, J. & Miner, G. (2009). *Handbook of statistical analysis and data mining applications*, Academic Press.

Okabe, A. & Sugihara, K. (2012). *Spatial analysis along networks: statistical and computational methods*, John Wiley & Sons.

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

Page, L., Brin, S., Motwani, R. & Winograd, T. (1999). The pagerank citation ranking: Bringing order to the web., *Technical Report 1999-66*, Stanford InfoLab. Previous number = SIDL-WP-1999-0120.
URL: *http://ilpubs.stanford.edu:8090/422/*

Pais, E. (2016). La crisis dispara los intentos de fraudes al seguro, https://elpais.com/economia/2016/01/18/actualidad/1453130872-610035.html.

Phillips, C. A. (2015). Multipartite graph algorithms for the analysis of heterogeneous data.

Potdar, K., Pardawala, T. S. & Pai, C. D. (2017). A comparative study of categorical variable encoding techniques for neural network classifiers, *International journal of computer applications* **175**(4): 7–9.

Pourhabibi, T., Ong, K.-L., Kam, B. H. & Boo, Y. L. (2020). Fraud detection: A systematic literature review of graph-based anomaly detection approaches, *Decision Support Systems* p. 113303.

Rajan, R. S., Shantrinal, A. A., Kumar, K. J., Rajalaxmi, T., Fan, J. & Fan, W. (2019). Embedding complete multi-partite graphs into cartesian product of paths and cycles, *arXiv preprint arXiv:1901.07717* .

Sabokrou, M., Khalooei, M. & Adeli, E. (2019). Self-supervised representation learning via neighborhood-relational encoding, *Proceedings of the IEEE International Conference on Computer Vision*, pp. 8010–8019.

Schabenberger, O. & Gotway, C. (2004). *Statistical Methods for Spatial Data Analysis*, Chapman & Hall/CRC Texts in Statistical Science, Taylor & Francis.
URL: *https://books.google.cl/books?id=iVJuVLArmZcC*

Schlichtkrull, M., Kipf, T. N., Bloem, P., Van Den Berg, R., Titov, I. & Welling, M. (2018). Modeling relational data with graph convolutional networks, *European Semantic Web Conference*, Springer, pp. 593–607.

Shiode, S. (2008). Analysis of a distribution of point events using the network-based quadrat method, *Geographical Analysis* **40**(4): 380–400.

Silva, T. C. & Zhao, L. (2016). *Machine Learning in Complex Networks*, 1st edn, Springer Publishing Company, Incorporated.

Sofaer, H. R., Hoeting, J. A. & Jarnevich, C. S. (2019). The area under the precision-recall curve as a performance metric for rare binary events, *Methods in Ecology and Evolution* **10**(4): 565–577.

Šubelj, L., Furlan, Š. & Bajec, M. (2011a). An expert system for detecting automobile insurance fraud using social network analysis, *Expert Systems with Applications* **38**(1): 1039–1052.

Subelj, L., Furlan, S. & Bajec, M. (2011b). An expert system for detecting automobile insurance fraud using social network analysis, *Expert Syst. Appl.* **38**: 1039–1052.

Sybrandt, J. & Safro, I. (2019). Fobe and hobe: First-and high-order bipartite embeddings, *arXiv preprint arXiv:1905.10953* .

Tobler, W. R. (1970). A computer movie simulating urban growth in the detroit region, *Economic geography* **46**(sup1): 234–240.

Van Vlasselaer, V., Eliassi-Rad, T., Akoglu, L., Snoeck, M. & Baesens, B. (2016). Gotcha! network-based fraud detection for social security fraud, *Management Science* **63**(9): 3090–3110.

Vlasselaer, V. V., Akoglu, L., Eliassi-Rad, T., Snoeck, M. & Baesens, B. (2015). Guilt-by-constellation: Fraud detection by suspicious clique memberships, *2015 48th Hawaii International Conference on System Sciences* pp. 918–927.

Vlasselaer, V. V., Meskens, J., Dromme, D. V. & Baesens, B. (2013). Using social network knowledge for detecting spider constructions in social security fraud, *2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2013)* pp. 813–820.

Wickham, H., François, R., Henry, L. & Müller, K. (2018). *dplyr: A Grammar of Data Manipulation*. R package version 0.7.6.
URL: *https://CRAN.R-project.org/package=dplyr*

Yamada, I. & Thill, J.-C. (2010). Local indicators of network-constrained clusters in spatial patterns represented by a link attribute, *Annals of the Association of American Geographers* **100**(2): 269–285.

Yang, K.-C., Aronson, B. & Ahn, Y.-Y. (2020). Birank: Fast and flexible ranking on bipartite networks with r and python, *Journal of Open Source Software* **5**(51): 2315.

Zhang, K., Wang, Q., Chen, Z., Marsic, I., Kumar, V., Jiang, G. & Zhang, J. (2015). From categorical to numerical: Multiple transitive distance learning and embedding, *Proceedings of the 2015 SIAM International Conference on Data Mining*, SIAM, pp. 46–54.