Using Bayesian Networks to Identify and Prevent Split Purchases in Brazil

Rommel N. Carvalho,[*] Leonardo J. Sales, Henrique A. da Rocha, and Gilson L. Mendes
Department of Research and Strategic Information (DIE)
Brazil's Office of the Comptroller General (CGU)
SAS, Quadra 01, Bloco A, Edifício Darcy Ribeiro
Brasília, DF, Brazil

Abstract

To cope with society's demand for transparency and corruption prevention, the Brazilian Office of the Comptroller General (CGU) has carried out a number of actions, including: awareness campaigns aimed at the private sector; campaigns to educate the public; research initiatives; and regular inspections and audits of municipalities and states. Although CGU has collected information from various different sources – Revenue Agency, Federal Police, and others –, going through all the data in order to find suspicious transactions has proven to be really challenging. In this paper, we present a Data Mining study applied on real data – government purchases – for finding transactions that might become irregular before they are considered as such in order to act proactively. Moreover, we compare the performance of various Bayesian Network (BN) learning algorithms with different parameters in order to fine tune the learned models and improve their performance. The best result was obtained using the Tree Augmented Network (TAN) algorithm and oversampling the minority class in order to balance the data set. Using a 10-fold cross-validation, the model correctly classified all split purchases, it obtained a ROC area of .999, and its accuracy was 99.197%.

1 Introduction

The Brazilian Office of the Comptroller General (CGU) is the Brazilian central body of the internal control system of the federal executive branch. It has, among its responsibilities, the task of inspecting and auditing the Brazilian Government projects and programs with respect to their legality, results, efficacy and efficiency.

A primary responsibility of CGU is to prevent and detect government corruption. To carry out this mission, CGU must gather information from a variety of sources and combine it to evaluate whether further action, such as an investigation, is required. One of the most difficult challenges is the information explosion. Auditors must fuse vast quantities of information from a variety of sources in a way that highlights its relevance to decision makers and helps them focus their efforts on the most critical cases. This is no trivial duty. The Growing Acceleration Program (PAC) alone has a budget greater than 250 billion dollars with more than one thousand projects only in the state of Sao Paulo [1]. All of these have to be audited and inspected by CGU – and, in spite of having only three thousand employees. Therefore, CGU must optimize its processes in order to carry out its mission.

In Brazil, all contracts with the private sector must be in accordance with the Law N° 8,666/93, also known as the national Procurement Law. According to [15] procurement is the administrative procedure by which the Public Administration selects the most advantageous proposal for a contract in its interest. From the former definition, the conclusion is that the public interest must always be the objective of the procedure. In terms of purchasing with the use of public money, this means that not only must the winner of the procurement process be the best supplier in terms of the price of the good or service supplied, but also in terms of other objectives of the procurement process.

Corruption can happen in many ways in Public Procurements [16]. The public agent may favor a specific supplier that she happens to know. She may receive, from the bidder, a financial compensation for awarding a contract to that firm. Bidders may collude as to set the results of the procurement. The whole process is susceptible to many forms of corruption, from within and outside the public administration.

The government purchases large quantities of goods and services. It is also the sole purchaser for some goods, such

[*] Department of Computer Science (CIC), University of Brasília (UnB), Brasília, DF, Brazil

[1] http://www.pac.gov.br/

as hydraulic turbines for large damns. The government spends large quantities of money in the market and is a guaranteed payer. Hence, many firms have a strong interest in negotiating with the public administration. There is a temptation for many suppliers to cheat in the procurement process to find means of being awarded a lucrative government contract.

The Brazilian Procurement Law has as one of its main objectives the curbing of corruption in public purchasing and contracting. Concern over corruption is evident in Brazil's daily press. There are frequent accusations of public administrators who did not abide by the procurement rules, and are accused of favoring a certain supplier or worse, receiving a payment in the process.

When writing the law, legislators included many articles that established penalties for firms or/and public legislators caught in corruption activities. There are two types of penalties stated in Law N° 8,666/93 dealing with this subject. They are administrative actions and penal actions.

Since enforcing the law is difficult [16], legislators will have to find another manner to prevent corruption in public procurement. The question is one of preventing corruption practices, against one of punishing the ones that have already happened.

Therefore, the main objective of this project is to apply Data Mining methods to create models, more specifically to learn Bayesian Network models, that will aid the experts in identifying procurement frauds and, if possible, identify suspicious transactions as soon as possible in order to prevent the fraud from happening.

The structure of this paper is as follows: Section 2 presents the CRoss Industry Standard Process for Data Mining (CRISP-DM) methodology used in this work. Section 3 presents the data used and its underlying semantics. Section 4 presets the models learned as well as their evaluation. Section 5 presents ideas on how the learned model for identifying split purchases will be deployed. Finally, Section 6 presents the conclusion and future work.

2 Methodology

The CRoss Industry Standard Process for Data Mining (CRISP-DM) project developed an industry and tool-neutral data mining process model. Starting from the embryonic knowledge discovery processes used in early data mining projects and responding directly to user requirements, this project defined and validated a data mining process that is applicable in diverse industry sectors. This methodology makes large data mining projects faster, cheaper, more reliable, and more manageable. Even small-scale data mining investigations benefit from using CRISP-DM.

The process model provided by the consortium CRISP-DM can be summarized through the life cycle of the data mining process presented in Figure 1, reproduced from [5]. The outer circle symbolizes the cyclical nature of the process of data mining. Data mining does not end when a solution is achieved. In fact, the lessons learned during a process can be useful in subsequent processes.

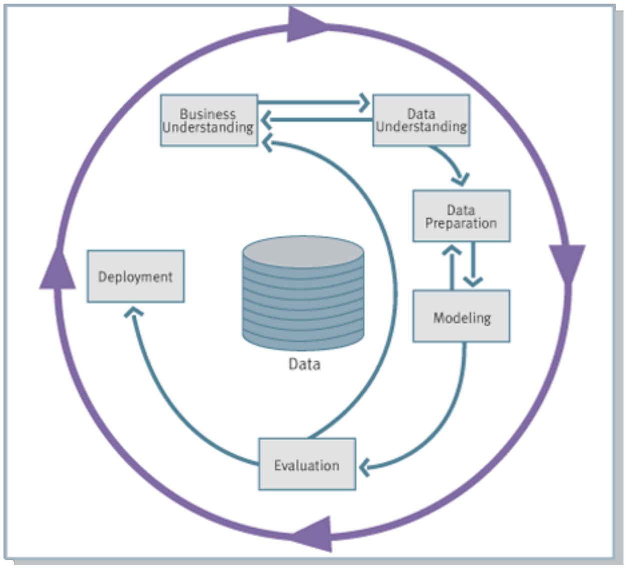

Figure 1: Phases of the CRISP-DM Process Model

The life cycle of a data mining project consists of six phases. The sequence of the phases is not strict. Moving back and forth between different phases is often required. The arrows indicate the most important and frequent dependencies between phases. The phases are [19]:

Business Understanding: This initial phase focuses on understanding the project objectives and requirements from a business perspective, and then converting this knowledge into a data mining problem definition, and a preliminary project plan designed to achieve the objectives.

Data Understanding: The data understanding phase starts with an initial data collection and proceeds with activities in order to get familiar with the data, to identify data quality problems, to discover first insights into the data, or to detect interesting subsets to form hypotheses for hidden information. There is a close link between Business Understanding and Data Understanding. The formulation of the data mining problem and the project plan require at least some understanding of the available data.

Data Preparation: The data preparation phase covers all activities to construct the final data set (data that will be fed into the modeling tool(s)) from the initial raw data. Data preparation tasks are likely to be performed multiple times, and not in any prescribed order. Tasks include table, record, and attribute selection, data cleaning, construction of new attributes, and transformation of data for modeling tools.

Modeling: In this phase, various modeling techniques are selected and applied, and their parameters are calibrated to optimal values. Typically, there are several techniques for the same data mining problem type. Some techniques require specific data formats.

Evaluation: At this stage in the project you have built one or more models that appear to have high quality, from a data analysis perspective. Before proceeding to final deployment of the model, it is important to more thoroughly evaluate the model, and review the steps executed to construct the model, to be certain it properly achieves the business objectives. A key objective is to determine if there is some important business issue that has not been sufficiently considered. At the end of this phase, a decision on the use of the data mining results should be reached.

Deployment: Creation of the model is generally not the end of the project. Usually, the knowledge gained will need to be organized and presented in a way that the customer can use it. Depending on the requirements, the deployment phase can be as simple as generating a report or as complex as implementing a repeatable data mining process. In many cases it will be the user, not the data analyst, who will carry out the deployment steps. In any case, it is important to understand up front what actions will need to be carried out in order to actually make use of the created models.

The Business Understanding phase was already covered in the introduction section. The Data Understanding phase will not be covered in detail since the data and its structure is confidential. The following sections will cover the Data Understanding, Data Preparation, Modeling, Evaluation, and Deployment phases.

3 Data Understanding and Preparation

The data set used in this work is related to procurements and contracts of IT services in the Brazilian Federal Government from 2005 to 2010. This data set is actually merged data from different databases (DB) available in different agencies in Brazil.

CGU has been working closely with subject matter experts (SMEs) in order to identify certain fraud topologies, such as:

1. Identify if owners from different companies are actually partners. In the public procurement we expect competition, but if the only two enterprises participating in the procurement have owners that are partners, then it is obvious that there is no competition at all, so this should be prevented/identified.

2. In Brazil if the contract is small enough (8 thousand reais, or roughly around 4 thousand dollars), then there is no need to actually go through the whole procurement process. However, sometimes, they break down a big contract in a lot of small ones. This is called split purchase[2] and it is not allowable by law and should be prevented/identified. *E.g.*, identify a lot of contracts that sums up to more than the threshold with the same objective (buying computers, for instance) in a week or a month.

There are actually more than 20 typologies like these two described by the SMEs that we would like to identify/predict using the data set we have. However, this project focuses on the second one described above.

First the data set was loaded in Excel, then we removed some characters (comma, double quotes, and single quotes), because they were being a problem when loading the file in Weka (see [13] for details about Weka). Then we replaced values like NA, -9, -8, and 0 by "?" to represent a missing value in Weka.

The initial data set had 42 attributes and 70,365 transactions. From Excel we were able to bring that number down to 26. Most of the attributes removed were identification of descriptions that were also available as a different attribute. We decided to keep the description because it is easier to understand what it means. For instance, we kept the name of the state instead of its ID.

The next step was to save the data as a CSV file and load it in Weka. Then we changed the year to nominal and removed rows with missing values in the final price attribute, since we will need a value later on to identify the class.

Before running any algorithm we started looking at the data trying to understand it better. The first thing that we noticed was the range of the values. There were values from cents to hundreds of trillions of dollars. Besides that, there were cases where the proposed unit price was 4 thousand

[2] For more information about split purchases and case examples see `http://guide.iacrc.org/potential-scheme-split-purchases/`.

whereas the final actual price was a dollar. Those are typical cases of data that should not be trusted or considered incorrect.

The number of cases with a really high value is small so they should be carefully analyzed before being thrown away, just to make sure they are not outliers, but actually inconsistent data. For that, we got in contact with the SMEs at CGU and asked them which ones should be ignored. With their consent we removed those transactions that were considered noise. The transactions that could be outliers are being analyzed by the experts in more detail.

Since we are interested in transactions that sum to 8,000 in a given period of time (see typology 2 above), we computed using R[3] the transactions that involved the same institutions on the same month and year that added up to more than 8,000, then we flagged them as irregular transactions[4] (*i.e.*, split purchases).

4 Modeling and Evaluation

The main objective of this work is to try to classify, without looking at the sums, which transactions should be considered suspicious. The reason for that is to avoid waiting for the problem to occur (having several transactions that add to more than 8,000 reais) and to identify the problem as soon as possible.

Before being able to classify we had to discretize the value of the transactions, since we decide to work with learning algorithms that do not support continuous values. We used equal frequency to generate 11 different bins. The number of bins could be different. The only thing we kept in mind was that we did not want it to be binary, nor to have too many bins to the point that we would have just a few transactions in each bin.

Since Bayesian Networks (BNs) have been successfully used in classification problems (*e.g.*, see [4,7,10–12,17,18, 20]), we decided to experiment with different BN learning algorithms in order to classify the transactions as regular or not, in order to identify split purchases as soon as possible, without having to wait for them to actually be confirmed as such (before having several purchases summing up to more than 8 thousand reais). We ran all classifiers with 10-fold cross-validation.

At first we compared the results of Naïve Bayes versus Bayesian Network models. Since the data is unbalanced with 50,911 regular cases and 2,626 irregular cases, we decided to resample the data.

Table 1 presents the performance of the Naïve Bayes and Bayesian Network classifiers using 10-fold cross-validation. We learned models without resampling the data set, oversampling the minority class, undersampling the majority class, and both oversampling the minority class and undersampling the majority class. The metrics used to compare the models are regular false positive (FP) rate, irregular FP rate, accuracy, and Receiver Operating Characteristic (ROC) area.

The standard algorithm used in Weka for BN is K2. K2 uses a hill climbing algorithm restricted by an order on the variables. For more information on K2 see [8, 9]. The parameters used for learning the BN model were all the default values in Weka expect for the maximum number of parents allowed, which we changed from 1 to 2. The `estimator` parameter used, which is independent of which algorithm we use to learn the model, was the `SimpleEstimator`. `SimpleEstimator` is used for estimating the conditional probability tables of a BN once the structure has been learned. It estimates probabilities directly from data. The only configurable parameter for this estimator is `alpha`. Alpha is used for estimating the probability tables and can be interpreted as the initial count on each value. The default value used for `alpha` is .5. As explained in [3], the options for the K2 algorithm in Weka are:

`initAsNaiveBayes` – When set to true (default), the initial network used for structure learning is a Naïve Bayes Network, that is, a network with an arrow from the classifier node to each other node. When set to false, an empty network is used as initial network structure.

`markovBlanketClassifier` – When set to true (default is false), after a network structure is learned a Markov Blanket correction is applied to the network structure. This ensures that all nodes in the network are part of the Markov blanket of the classifier node.

`maxNrOfParents` – Set the maximum number of parents a node in the Bayes net can

[3] R is a free software environment for statistical computing and graphics. For more information see `http://www.r-project.org/`.

[4] Usually we also consider the type of service/product contracted/bought. However, since we limited the scope of our analysis to only computer purchases, all 10 different types of service/product are too broad and really similar (they all basically say that these are computer related purchases without any significant difference between each other). Therefore, we can safely ignore this field when defining split purchases, since they do not add any more useful information. To avoid comparing purchases of different nature, we use the contractor field, since a company that provides software development does not usually sell computer peripherals. Unfortunately, we do not have any other structured field that we could use to improve this proxy classification. Of course we might have cases that purchases identified as split purchases are, in fact, false positives. However, our current alert system would also identify those purchases as split purchases. What we are trying to do is to identify those ”split purchases” as soon as possible, instead of waiting for the whole month cycle.

have. When initialized as Naïve Bayes, setting this parameter to 1 results in a Naïve Bayes classifier. When set to 2, a Tree Augmented Bayes Network (TAN) is learned, and when set to greater than 2, a Bayes Net Augmented Bayes Network (BAN) is learned. By setting it to a value much larger than the number of nodes in the network (the default of 100000 pretty much guarantees this), no restriction on the number of parents is enforced.

randomOrder – When set to true, the order of the nodes in the network is random. Default random order is false and the order of the nodes in the data set is used. In any case, when the network was initialized as Naïve Bayes Network, the class variable is first in the ordering though.

scoreType – The score type determines the measure used to judge the quality of a network structure. It can be one of Bayes (default), BDeu, Minimum Description Length (MDL), Akaike Information Criterion (AIC), and Entropy.

When oversampling, we used the sample size of 200% to avoid removing known cases of the majority class, which is basically the oversampling of the minority class. The goal was to increase the irregular cases to match the number of regular ones.

We can see that we got much better results with Bayesian Network, when oversampling the minority class. Moreover, the false positive rate for the regular cases, which is our main concern since it represents the number of irregular transactions that were classified as regular, has dropped significantly after oversampling (from .363 to .140 for Naïve Bayes and from .452 to .005 for Bayesian Network). As a result, the FP rate for the irregular cases increased while the accuracy decreased for NB and slightly increased for BN. Nevertheless, since we are more interested in correctly classifying the irregular cases, the result with oversampling is better.

When undersampling, we decided to use the sample size of $(2,626/50,911) * 2 \approx 10\%$. Finally, we tried doing both oversampling and undersampling by keeping the sample size at 100%.

As it can be seen from Table 1, oversampling the minority class gave us the best results both in accuracy and in ROC Area. Besides that, we did have a significant decrease in regular FP rate, which is our main concern. The accuracy without resampling is higher than most of the other results with resampling due to the fact that we had an increase in irregular FP rate and since this is the majority class the number of correctly classified cases is much larger. However, as explained before, the main objective is to obtain low regular FP rate and not necessarily high accuracy, which could only mean that we are getting just the majority class right.

As BN gave the best result, we decided to try different search algorithms. The algorithms used were K2 (again using the default parameters), Hill Climber[5], Tabu Search[6], and Tree Augmented Network[7] (TAN).

The options for the Hill Climber algorithm in Weka are the same as K2, except that it does not have the parameter `randomOrder`. Besides, we have the `useArcReversal`, which when set to true (default is false), the arc reversal operation is used in the search. As in K2, we have used all default parameters, except for the maximum number of parents and the score metric (we used MDL instead of Bayes).

The Tabu Search algorithm in Weka has the same options as Hill Climber. Besides that, it also has `runs`, which sets the number of steps to be performed (default is 10), and `tabuList`, which sets the length of the tabu list (default is 5). As in both K2 and Hill Climber, we have used all default parameters, except for the maximum number of parents and the score metric (we used MDL instead of Bayes).

The TAN algorithm in Weka only has the `markovBlanketClassifier` and `scoreType` options with the same values as K2. We have used the default parameters. Note that the TAN algorithm does not allow the configuration in the maximum number of parents, since it is always 2.

Table 2 shows a summary of the BN classifiers performance using different search algorithms with oversampled data. It is worth noting that we were only able to use the default Bayes metric with the K2 and TAN algorithms. We had to use MDL metric in order to avoid out of memory error with the Hill Climbing and Tabu Search algorithms. This difference might be the reason why K2 and TAN outperforms both Hill Climbing and Tabu Search, since we have tried both K2 and TAN using MDL metric and we did obtain worse results. As it can be seen, TAN search algorithm had the best performance with a ROC area of .999.

Since oversampling resulted in very good results, we decided to also try Synthetic Minority Oversampling TEchnique (SMOTE) [6] to oversample the minority class. We used 1800% to get roughly the same number of transac-

[5] Hill Climber uses a hill climbing algorithm adding, deleting and reversing arcs. The search is not restricted by an order on the variables (unlike K2). For more details see `http://weka.sourceforge.net/doc.dev/weka/classifiers/bayes/net/search/global/HillClimber.html`.

[6] Tabu Search algorithm uses tabu search for finding a well scoring BN structure. For more details see [2].

[7] TAN algorithm determines the maximum weight spanning tree and returns a NB network augmented with a tree. For more details see [10].

Table 1: Evaluation of Naïve Bayes (NB) and Bayesian Network (BN) classifiers without resampling, with oversampling, with undersampling, and with both over and undersampling

Metric	NB	BN	NB - Over	BN - Over	NB - Under	BN - Under	NB - Both	BN - Both
Regular FP Rate	.363	.452	.140	**.005**	.169	.068	.146	.012
Irregular FP Rate	.028	**.003**	.048	.036	.082	.054	.054	.045
Accuracy	95.6%	97.5%	90.6%	**97.9%**	87.4%	90.6%	90.0%	97.1%
ROC Area	.931	.973	.975	**.997**	.945	.968	.971	.996

Table 2: BN classifiers performance with oversampling and different number of parents

Metric	BN - K2		BN - HC		BN - Tabu		BN - TAN
Parents	2	3	2	3	2	3	-
Regular FP Rate	.005	.018	.098	.066	.093	.093	**0**
Irregular FP Rate	.036	.038	.082	.067	.039	.039	**.02**
Accuracy	97.94%	97.23%	91.01%	93.35%	93.40%	93.40%	**98.98%**
ROC Area	.997	.996	.976	.985	.988	.988	**.999**

tions in both classes. After we oversampled the data, we discretized it as we did previously (11 bins with equal frequency). However, even though the SMOTE oversampling gave better results for most metrics, the standard oversampling method outperformed the SMOTE method on the regular FP rate, which is the one that we are most interested in. Therefore, we do not present the details of the results here.

Since TAN algorithm presented the best results, we decided to try different values for the alpha parameter in order to improve its performance. Table 3 presents performance for the alpha values .7, .5, .3, .1, and .05. On the one hand, when we increased the default value of .5 to .7, the performance slightly worsened. On the other hand, when we decreased the default value we kept getting slightly better results up to .05, when the performance started worsening again.

Table 4 presents the confusion matrix for the best model we obtained, which was learned using TAN with the value .1 for the `alpha` parameter. As it can be seen, every single split purchase (irregular) was correctly classified. Although the regular FP rate in Table 3 suggests that alpha values of .5, .3, and .05 also classified all irregular instances correctly, the model with .5 alpha missclassified 7 instances (the regular FP rate is shown as 0 due to rounding, since the value is actually .00013). Besides that, even though the irregular FP rate is the same for the models with .1 and .05 alpha, the model with the .05 alpha missclassified 5 instances more than the model with .1 alpha.

In order to try to better understand the relationship between the variables, we generated some association rules using Apriori algorithm [1, 14]. In the first run we used .1 as lower bound min support, 1 as upper bound min support, lift as the metric type, 1.1 as min metric, and 50 as number of rules. For more information on each of these parameters see `http://weka.sourceforge.net/doc.dev/weka/associations/Apriori.html`.

The results were related to variables that were not of interest and even though they had a high confidence (1) and high lift (1.82) they were not describing anything that we did not already know. So we decided to remove those variables from the data set and run the algorithm again to see the new rules it would generate.

We analyzed all 50 new rules generated and they still did not provide any insightful information. Besides that, these new rules were not as "strong" as the previous ones in the sense that the best confidence was .62 and the best lift was 1.11.

The reason for not presenting the rules is two-fold. The first is because, as previously explained, they did not provide any insightful information. The second is because they are confidential and we cannot discuss the contents of the rules. All that can be said is that they were not useful.

Finally, we analyzed the BN that had the best evaluation results (BN – TAN with .1 alpha – standard oversampled data set) to try to better understand the relations between the variables in order to comprehend why it is getting such good results. This could provide insightful information that we were not able to extract from association rules.

By analyzing the structure of the BN we were able to identify some interesting relations associated to the government

Table 3: TAN performance with different values for the alpha parameter

Metric	TAN - .7 alpha	TAN - .5 alpha	TAN - .3 alpha	TAN - .1 alpha	TAN - .05 alpha
Regular FP Rate	.001	**0**	**0**	**0**	**0**
Irregular FP Rate	.023	.020	.018	**.016**	**.016**
Accuracy	98.834%	98.984%	99.108%	**99.197%**	99.192%
ROC Area	**.999**	**.999**	**.999**	**.999**	**.999**

Table 4: Confusion matrix of the best model, which was learned using TAN with .1 alpha

	Predicted Regular	Predicted Irregular
Is Regular	52,670	860
Is Irregular	0	53,544

office responsible for the procurement and also to the winner of the procurements. However, due to the confidentiality nature of the problem we are not allowed to discuss these relations further. Nevertheless, we can comment on the fact that to better understand these relations a more detailed study is necessary by looking at the conditional probability tables on these relations, which was not done in this work due to time constraints.

5 Deployment

As it was explained before, the main goal is to identify suspicious transactions as soon as possible to avoid the occurrence of split purchases.

Split purchases in the data set we worked on happen when a procurement that should be done just once with a value higher than 8,000 is broken down in smaller ones to avoid the normal procurement procedure, allowing the office to hire an enterprise directly.

So, the idea is that we want to detect one or more of these procurements, but that still does not sum up to more than 8,000, as soon as they become suspicious (using the classifier) and act proactively. This will allow a faster response and prevent irregularities in the first place. This means that instead of waiting a whole month to find an irregular transaction, we will be finding suspicious ones (not necessarily irregular ones) and warn/educate/teach the people involved that they should be careful, since breaking down big procurements in small ones is not allowed.

Hopefully, this will make them think twice before continuing with these kinds of transactions, if they are really thinking about doing irregular transactions on purpose. And at the same time this will educate those that are not aware they cannot do these kinds of transactions by law.

Another thing to keep in mind when deploying this classifier is that as we warn/teach the different offices in Brazil about this type of fraud, it is expected that they will decrease the number of irregular transactions associated to this type of fraud. Therefore, it is important to keep track of these numbers in order to evaluate if this expected behavior is actually happening and at what rate.

Besides that, as people learn and change their behavior, it is also expected that our classifier will also need some modifications to cope with these changes. Therefore, it should be expected that every month or so a new classifier should be trained and evaluated to learn the new behavior and analyze its performance.

6 Conclusion

This project shows how Data Mining, BN learning more specifically, can be used to identify and prevent split purchases in Brazil using real data provided by the Brazil's Office of the Comptroller General (CGU).

At first, the idea was to allow the identification of a few different types of irregularities. However, as we started working with the data and creating the different models, it became clear that this is a great effort and after discussing with stakeholders at CGU, we decided to prioritize depth instead of breadth. In other words, it was decided that it was best to choose one type of fraud and do a thoroughly job to develop a model that is good enough to actually deploy it and analyze the results before looking at other types of frauds.

Fortunately, the results paid off in the sense that we were able to generate a model that correctly classified all split purchases and a really high ROC area (.999). The next step is to deploy this classifier in CGUs intelligence system and analyze its impact.

As future work, it is necessary to better understand why the model is capable of getting such good results. By analyzing the structure of the BNs generated, it is possible to understand relations between variables that were previously unknown. A brief analysis showed that some interesting relations were found, but a more detailed analysis is needed.

Even though we have used resampling for dealing with the rare event of split purchases and cross-validation to avoid overfitting, we might have not completely ruled out the possibility that this model might be slightly overfitted. Therefore, before deploying the model, we will gather new data from recent purchases in order to test the performance of our model with new unseen data. Due to time constraints, we have not been able to gather the new data and perform this test yet.

Besides that, there are a lot of different types of frauds that were not analyzed. So, a starting point for future work is to do the same kind of analysis done in this project with different types of fraud.

Acknowledgements

The authors would like to thank Prof. Daniel Barbará from George Mason University for providing useful insights for the development of this work. Finally, the authors would like to thank CGU for providing the resources necessary to work in this research, as well as for allowing its publication.

References

[1] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *20th International Conference on Very Large Data Bases*, pages 478–499. Morgan Kaufmann, Los Altos, CA, 1994.

[2] R. R. Bouckaert. *Bayesian Belief Networks: from Construction to Inference*. PhD thesis, University of Utrecht, Utrecht, Netherlands, 1995.

[3] Remco R. Bouckaert. Bayesian network classiers in weka for version 3-5-7. Technical report, May 2008.

[4] S. Ceccon, D.F. Garway-Heath, D.P. Crabb, and A. Tucker. Exploring early glaucoma and the visual field test: Classification and clustering using bayesian networks. *IEEE Journal of Biomedical and Health Informatics*, 18(3):1008–1014, May 2014.

[5] Pete Chapman, Julian Clinton, Randy Kerber, Thomas Khabaza, Thomas Reinartz, Colin Shearer, and Rudiger Wirth. CRISP-DM 1.0 step-by-step data mining guide. Technical report, The CRISP-DM consortium, August 2000.

[6] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321357, 2002.

[7] Jie Cheng and Russell Greiner. Comparing bayesian network classifiers. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, UAI'99, page 101108, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc.

[8] Gregory F. Cooper and Edward Herskovits. A bayesian method for constructing bayesian belief networks from databases. In *Proceedings of the Seventh Conference on Uncertainty in Artificial Intelligence*, UAI'91, page 8694, San Francisco, CA, USA, 1991. Morgan Kaufmann Publishers Inc.

[9] GregoryF. Cooper and Edward Herskovits. A bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 9(4):309–347, 1992.

[10] Nir Friedman, Dan Geiger, and Moises Goldszmidt. Bayesian network classifiers. *Machine Learning*, 29(2-3):131–163, November 1997.

[11] Nir Friedman and Moises Goldszmidt. Building classifiers using bayesian networks. In *Proceedings of the national conference on artificial intelligence*, page 12771284, 1996.

[12] Moises Goldszmidt, James J. Cochran, Louis A. Cox, Pinar Keskinocak, Jeffrey P. Kharoufeh, and J. Cole Smith. Bayesian network classifiers. In *Wiley Encyclopedia of Operations Research and Management Science*. John Wiley & Sons, Inc., 2010.

[13] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The WEKA data mining software: An update. *SIGKDD Explor. Newsl.*, 11(1):1018, November 2009.

[14] Bing Liu, Wynne Hsu, and Yiming Ma. Integrating classification and association rule mining. In *Fourth International Conference on Knowledge Discovery and Data Mining*, pages 80–86. AAAI Press, 1998.

[15] Hely Meirelles. *Licitacao e Contrato Administrativo*. Editora Revista dos Tribunais, Sao Paulo, Brazil, 7a. ed., atualizada pelo decreto-lei 2,300/86. edition, 1987.

[16] Andre Mueller. A critical study of the brazilian procurement law. Technical report, IBI - The Institute of Brazilian Business & Public Management Issues, Washington, 1998.

[17] Mehran Sahami, Susan Dumais, David Heckerman, and Eric Horvitz. A bayesian approach to filtering junk e-mail. In *Learning for Text Categorization: Papers from the 1998 workshop*, volume 62, page 98105, 1998.

[18] Wei Shi, Yao Wu Pei, Liang Sun, Jian Guo Wang, and Shao Qing Ren. The defect identification of LED chips based on bayesian classifier. *Applied Mechanics and Materials*, 333-335:1564–1568, July 2013.

[19] Rdiger Wirth. CRISP-DM: Towards a standard process model for data mining. In *Proceedings of the Fourth International Conference on the Practical Application of Knowledge Discovery and Data Mining*, page 2939, 2000.

[20] Ye Ye, Fuchiang (Rich) Tsui, Michael Wagner, Jeremy U. Espino, and Qi Li. Influenza detection from emergency department reports using natural language processing and bayesian network classifiers. *Journal of the American Medical Informatics Association*, pages amiajnl–2013–001934, January 2014.