Intl. Trans. in Op. Res. (2023)

INTERNATIONAL
TRANSACTIONS
IN OPERATIONAL
RESEARCH

Red flag algorithms for Brazilian electronic invoices: Outlier detection and price risk classification

Ruben Interian[a,*], Igor Carpanese[b], Bruno Mello[b], and Celso C. Ribeiro[c]

[a]*Institute of Mathematics and Computer Science, University of São Paulo, São Carlos, São Paulo 13566-590, Brazil.*
[b]*Aetos Tech, Rio de Janeiro, RJ 20010-010, Brazil.*
[c]*Institute of Computing, Universidade Federal Fluminense, Niterói, RJ 24210-346, Brazil.*
E-mail: ruben@icmc.usp.br [Interian]; carpanese@protonmail.com [Carpanese];
bruno@aetos.tech [Mello]; celso@ic.uff.br [Ribeiro]

Abstract

Brazilian electronic invoices (*Nota Fiscal eletrônica*, NFe) are digital documents that register the purchase and sale operations of goods and services. In this work, we developed and used data-cleaning procedures, clustering methods, and outlier detection algorithms to investigate the presence of two main risk patterns in the electronic invoices data of public entities, such as states or municipalities: abnormal product quantities and anomalous prices paid with public funds. As a case study, we analyzed nearly 3.5 million electronic invoices containing more than 11 million items from purchases made by public entities of the Brazilian state of Mato Grosso (MT) in 2016-2019. The results indicate that the total value of invoice items categorized as high-intensity price alerts amounted to more than R$ 560 million, i.e., approximately US$ 139 million at the rate of December 31st, 2019. Several cases of high-risk patterns are presented to illustrate these practical results.

Keywords: Fraud detection, electronic invoices, overpricing, risk patterns, drug purchases, fuel purchases, Farmácia Popular

1. Introduction

The Brazilian electronic invoice (*Nota Fiscal eletrônica*, NFe) is a digital document that registers the purchase and sale operations of goods or provisions of services that take place between two parties, whether they are public institutions or private companies. The electronic invoice formalizes the purchase of products and services in both physical and digital environments. It replaces a large number of manually issued records and documents. Electronic invoices are issued by the company that sells the product or provides the service. They are stored electronically in a specific system of the federal government [17], with the purpose of recording transactions related to commercial operations carried out in Brazil.

[*] Corresponding author.

Compared to other data sources that record public and private sector expenditures, electronic invoices offer a unique feature: they detail all the specific goods or services bought or sold, including descriptions, quantities, and prices of particular products. Bidding or contracting data from the public or private sectors rarely provide this detail. The degree of reliability of the product descriptions and the total values spent with each item in electronic invoices is very high. Electronic invoices are usually generated in XML (Extensible Markup Language) format.

In a previous work [24], we detailedly analyzed large public procurement datasets containing bidding process data and contracts with public institutions. We investigated several risk patterns for public contractors, such as collusion between companies participating in public bidding. The decision support system (DSS) described in the above work incorporates data mining and graph algorithms which led to the identification of a significant number of inconsistencies and even fraud cases by Prosecutor's Offices of Brazilian states. It also led to many effective operational results, resulting in potential improvements in the quality of public spending.

Fraud detection studies, including our previous work [24], generally analyze government bodies' bidding and contract data. This work proposes an alternative way to detect risk patterns by extracting red flags from non-government-generated data. In particular, we analyzed the dataset of invoices generated by companies that sell products and provide services to public bodies of the Brazilian state of Mato Grosso (MT). The estimated population of Mato Grosso is 3.6 million inhabitants (corresponding to the 16th largest out of 27 federative units), and its GDP was R$ 142.1 billion in 2019 (the 13th largest out of the 27 units), which allows us to conclude that it is a reasonably representative state of the Brazilian federative units. We developed and used data cleaning procedures, product description clustering methods, and specific outlier detection algorithms to detect the presence of two main risk patterns in the data: abnormal product quantities and anomalous prices used in purchases made by public entities of all municipalities in a given state. We focus mainly on product invoices since they are more uniform and consistent, but we also analyzed the hiring of some services.

In particular, we detected a substantial volume of purchases of medicines that are distributed free of charge by the national program *Farmácia Popular* for patients of some chronic diseases, so that purchases by municipalities should only occur in very limited quantities in specific cases. On average, the *Farmácia Popular* program's budget was R$ 2.9 billion in 2015-2019, attending about 25 million people annually. Several cases of abnormal volumes of gasoline and diesel purchases were also detected in several municipalities.

We remark that this work does not intend to reveal specific frauds or point to the occurrence of corruption cases. The goal here is to discover risk patterns in the data. The results may indicate inefficiency, lack of planning, or some urgent purchases of products in case of unexpected demands, like those that arose during the COVID-19 pandemic. Further investigations are always necessary for understanding and uncovering the occurrence of red flags that we might find. The decision-maker must verify each risk pattern to check for other alternative explanations.

This article is organized as follows. Section 2 offers a brief review of the literature on identifying fraud risk patterns in public procurement. Section 3 describes the data sources and the ETL process (Extraction, Transformation, Loading) carried out. The specific algorithms developed and used for detecting risk patterns are described in Section 4. Section 5 discloses the main results achieved. Concluding remarks are drawn in Section 6. The relational database model used in our approach is presented in the Appendix.

2. Background

With the progress of government data transparency policies, researchers have been studying how to recognize and reduce inefficiency and corruption cases by identifying waste and fraud red flags in public procurement. Institutional corruption can be substantially decreased by adopting a transparent system with good-quality datasets to reach sustainable development [1, 9]. Data analytics is widely used for assessing corruption and fraud risks, helping managers to identify the riskiest transactions [23]. A comprehensive guide of corruption techniques fraudsters use to take advantage of the public procurement process appeared in Ware et al. [25]. Modrušan et al. [19] reviewed emerging methods and models to detect suspicious expenditures in public procurement in different countries.

Ware et al. [25] and Ware and Noone [26] showed that corruption schemes are similar across the globe. Studies worldwide have proposed implementing anti-corruption techniques in different countries. The impact of development funds on corruption was explored in Fazekas and King [5] using data from over 100,000 public procurement contracts of the Czech Republic and Hungary from 2009 to 2012. A basic introduction to overcoming corruption in Indonesia, Malaysia, and Pakistan was presented by Kostyo [12]. An extensive set of corruption red flags in Brazil and how investigators can detect them was proposed in Santos and Souza [22]. A decision support system for fraud detection in Brazilian public procurement was developed by Velasco et al. [24], exploring more than 200 variables based on collusion risk patterns, political connections, incompatible company size, contract earliness, and a large number of company activities.

Most studies above apply data mining to identify common fraud and corruption red flags. As shown by Roselli and Almeida [21], this approach improves an entirely human-made decision-making process. However, network analysis and natural language processing allow more assertive results. Nicolás-Carlock and Luna-Pla [20] used a network approach to explore the relations between companies involved in documented corruption cases in Mexico. They used data about companies' shareholders, representatives, administrators, and commissioners to describe how these companies connect due to shared personnel with multiple roles. Lima et al. [14] applied natural language processing to detect collusion in public procurement data. Network analysis and graph databases were used by Carneiro et al. [2] and van Erven et al. [4] to identify hidden relations in public procurement processes for detecting fraud cases in Portugal and Brazil. An explainable scoring mechanism to prioritize the most critical expenditures was presented by Westerski et al. [27]. Luna-Pla and Nicolás-Carlock [15] used complex networks for studying corruption, identifying hundreds of "phantom" or shell companies, created under fake ownership. Understanding and implementing these and other possible red flags improve the quality of decisions administrators and authorities make.

Most fraud detection studies analyze government bodies' public procurement and bidding datasets. However, an alternative way to detect inconsistencies and fraud cases involves extracting red flags from different, non-government-generated data. This study explores Brazilian electronic invoices issued by public contractors. Electronic invoices offer product-level data not contained in Brazilian procurement datasets, such as product descriptions, units of measurement, quantities, and expiration dates of specific items bought by public administration.

Year of issue	Number of electronic invoices	Number of items	Total spent (R$ million)
2015	18	32	0.92
2016	774,402	2,790,281	2,165.68
2017	824,757	2,773,907	2,248.32
2018	907,060	2,924,662	2,534.51
2019	961,864	3,062,825	2,991.15
2020	31	31	0.01

Table 1
Main statistics of electronic invoice data.

3. Electronic invoices and data treatment

This section briefly describes the electronic invoice database and the ETL process (Extract, Transform, and Load) carried out.

The data was received in XML, Brazil's most common electronic invoice format, with one XML file for each electronic invoice. The total volume of data was about 14 gigabytes, containing approximately 3,500,000 electronic invoices. Each invoice may contain one or more items (specific products or services). There were more than 11,000,000 items. A summary of the electronic invoice data we received is shown in Table 1. The total expenditures amounted to R$ 9.94 billion. The vast majority of NFEs were issued in the years 2016-2019.

Altogether, 929 state and municipal public entities of Mato Grosso contracted products and services from 31,700 different contractors: 31,108 companies (98%) and 592 individuals (2%). There are 141 municipalities in Mato Grosso.

The XML format is not the most appropriate for working with large volumes of data and identifying risk patterns. For this reason, we transformed the XML files into several tables, forming a relational database model.

There are four main tables in our model. Firstly, table *nota_fiscal* (electronic invoice) contains the total value of the products in the invoice, date of issue, status, and other general invoice attributes, one line for each invoice. Secondly, table *item_nota* (invoice item) contains each commercialized item of each invoice, one line for each invoice-item pair. Lastly, *emissor* and *destinatario* tables contain information about the issuers and recipients of the invoices, respectively. In our dataset, recipients always represented public entities.

After these main tables, several additional tables that expanded our model were created. These auxiliary tables are related to the risk patterns detection approach we proposed. Section 4 will further explain their content and function.

More detailed information about the relational model created for storing electronic invoice data can be found in the Appendix. It describes the relational database structure, the function, and the columns of each table in the model.

Several validation routines were implemented to ensure the correctness and completeness of the ETL process. The validity of the identification numbers of companies and individuals (CNPJ and CPF, respectively) were verified, ensuring the correctness of each identification number's check digits. The accuracy of the IBGE codes [8] of each municipality in issuer and recipient addresses was also verified since the correction of this data is essential for the implemented algorithms.

4. Identifying risk patterns in electronic invoices

Two main risk patterns were identified in electronic invoices from the Mato Grosso state. First, we detected abnormal product quantities bought by public entities. Second, we identified anomalous prices paid by administrators and authorities in those purchases.

One of the main challenges we faced was the need for more standardization of product descriptions. Several description variants were similar to each other for the same product due to nonstandard nomenclature or regional language characteristics, argot, or even spelling errors and typos in product descriptions. Next, we will explain how we deal with this problem.

4.1. Clustering product descriptions

Each product item in an electronic invoice already has a classification called NCM – *Nomenclatura Comum do Mercosul* [16] (Mercosur Common Nomenclature), a numerical code indicating product type assigned to each item. Unfortunately, this classification alone is unsuitable since, in many cases (e.g., medicines or fuels), several products with significantly different prices may fall into the same NCM. The main goal of the process of clustering product descriptions was to reduce the number of description-NCM pairs by identifying, within the same NCM, the descriptions that represent the same product or service. Another goal was to correct unconventional descriptions or even spelling errors and typos by assigning those descriptions to a cluster whose representative element is the correct product name.

To reduce the number of different description-NCM pairs, we carried out a cleaning procedure that removed noise from product descriptions. Then, we proposed and implemented a two-step clustering algorithm that groups similar descriptions into clusters. These three steps (the cleaning procedure, the fingerprint clustering, and the incremental clustering) are described below.

Since there were more than two million different product descriptions, we sought to use subquadratic-time clustering algorithms. We also avoided using algorithms that use distance matrices between the elements since they have quadratic space complexity.

4.1.1. Cleaning invoice item descriptions

We performed a cleaning procedure on electronic invoice item descriptions by removing noisy or useless patterns from description strings. A set of regular expressions representing common noise structures, such as long serial numbers, dates, and stop words, were used. We choose to keep the cleaning as conservative as possible, not removing any information that could compromise the analysis. Those numeric patterns that may represent medication dosage, model numbers, and electric equipment voltage, among other domain-specific product characteristics, were kept in the product descriptions.

For example, the product description string “*PARACETAMOL 500MG - 1998233 - Val.:31/05/2018*” was transformed into “*PARACETAMOL 500MG*”, removing the product validity date *31/05/2018* and some unrecognized identification number *1998233* of this product. This step is only partially replicable if applied to electronic invoices from other datasets since it is application-specific and there may be different systematic errors in product descriptions.

4.1.2. Fingerprint matching

The proposed clustering algorithm's first step involves using fingerprints to match several similar product descriptions inside each NCM. Our algorithm calculates the fingerprint of each description by generating a simple *1-gram* sequence [11] for each product description. In short, a 1-gram fingerprint is a simple alphabetic-ordered sequence of letters and numbers contained in the product description after putting the string in lowercase and removing punctuation. For example, the treated product description strings "*PARACETAMOL 500MG*", "*PARACETAMOL 500 MG*", and "*500MG PARACETAMOL*" produce the same fingerprint, and consequently will be placed in the same cluster.

If the 1-gram sequences of two descriptions inside the same NCM are equal, then the two will be placed in the same cluster in this first step of our clustering algorithm. The algorithm lowers the description string, removes punctuation, and then builds and sorts the 1-gram sequence of each description. Algorithm 1 presents the main implementation steps. It takes as input a set of product descriptions represented as a list and returns the 1-gram fingerprint of each product description. Line 1 initializes the fingerprint map that will assign a specific fingerprint for each product desciption. The loop in lines 2 to 9 considers each product description at-a-time. Line 3 lowers the description's alphabetical characters. Line 4 removes punctuation marks from the description. Line 5 generates all 1-grams from the product description and then removes any duplicates in line 6. Line 7 sorts the 1-grams in alphabetical order. Line 8 assigns the obtained 1-gram fingerprint to the current description.

Algorithm 1 *1-gram fingerprint*

```
    Input: list $descriptions$
    Output: $fingerprints$
1:  fingerprints $\leftarrow \emptyset$
2:  for description in descriptions do
3:      description $\leftarrow$ Lower(description)
4:      description $\leftarrow$ RemovePunctuation(description)
5:      ngrams $\leftarrow$ Get1Grams(description)
6:      RemoveDuplicates(ngrams)
7:      Sort(ngrams)
8:      fingerprints[description] $\leftarrow$ ngrams
9:  end for
```

At first glance, it might be expected that the algorithm may produce several collisions, identifying as the same different products whose descriptions are anagrams. However, the set of descriptions that are being clustered are already in the same NCM classification. Furthermore, the number of actually different products in the same NCM is small, and the number of different descriptions in the NCM set rarely reaches the four digits. Therefore, the likelihood of a collision or coincidence with the fingerprint of another different product is very remote. To confirm this claim, we manually validated a sample of 100 randomly chosen clusters inside several NCM codes, not detecting collisions.

Each cluster's most common element (i.e., the one with the most observations) is considered its representative element. For example, if there are five description strings "bread" and one description "breda" (with a typographic transposition error – letters d and a are swapped), we considered "bread" as the correct one that will represent the entire cluster in the next clustering step. If, by coincidence, a transposition error creates another unrelated but valid word (a pretty common effect), the correct product description is chosen as the most common one.

This is a computationally efficient algorithm that can be easily parallelized. Each iteration of the loop in lines 2 to 9 can be implemented with time complexity $O(k \log k)$, where k is the maximum length of any description (which is typically a small value), corresponding to the sort computation in line 7. Its overall computational complexity is $O(nk \log k)$, where n is the number of descriptions.

4.1.3. Incremental clustering

The second and last step of the clustering procedure consists of grouping different product descriptions using an incremental clustering algorithm [7]. The algorithm can handle large datasets since it traverses the product descriptions only once. Algorithm 2 presents the pseudo-code of the adapted version of incremental clustering for the case of product descriptions.

Algorithm 2 *Incremental clustering*

```
    Input: list $descriptions$
    Output: $clusters$
 1: firstDescription $\leftarrow$ GetRandom(descriptions)
 2: clusters $\leftarrow$ { CreateCluster(firstDescription) }
 3: for description in $descriptions$ do
 4:     closestCluster $\leftarrow -1$
 5:     $d_{min} \leftarrow \infty$
 6:     for cluster in clusters do
 7:         if distance(description, cluster) $< d_{min}$ then
 8:             closestCluster $\leftarrow$ cluster
 9:             $d_{min} \leftarrow$ distance(description, cluster)
10:         end if
11:     end for
12:     if $d_{min} < \alpha$ then
13:         AssignCluster(description, closestCluster)
14:     else
15:         clusters $\leftarrow$ clusters $\cup$ { CreateCluster(description) }
16:     end if
17: end for
```

The algorithm starts by randomly choosing a description in line 1 and creating a cluster for it in line 2. The external for loop in lines 3 to 17 analyzes all descriptions, assigning a cluster to each. For each description, the algorithm identifies the closest among all existing clusters in lines 4 to 11. Line 4 sets the closest cluster to -1, while line 5 sets the minimum distance to some upper bound. The internal for loop in lines 6 to 11 considers each existing cluster. The distance between the current description and cluster is computed in line 7 and compared with the minimum distance. The closest cluster and the minimum distance are updated in lines 8 and 9. Finally, if the distance to the closest cluster is small enough (i.e., smaller than a certain threshold α), the current description is assigned to the closest cluster in lines 12–13. Otherwise, a new cluster is created in line 15, containing the current description exclusively.

Two relevant issues must be solved in order to apply properly this algorithm in practice.

The first one is the product description representation. As already mentioned, we have maintained a fairly conservative strategy, in the sense of only grouping descriptions that certainly represent the same product, with the aim of having greater confidence in the results of our approach. For this purpose, at this stage, we decided to separate the numerical patterns of the product descriptions (those that may represent medication dosage, model numbers, voltage), considering these patterns immutable. In the description "*PARACETAMOL 500MG*", the numerical pattern *500* is considered as fixed and immutable. We excluded these sequences of digits from product descriptions that participate in the second step of our clustering process. Then, we only grouped descriptions whose numerical patterns are exactly the same. For example, descriptions "*PARACETAMOL 500MG*" and "*PARACETAMOL 750MG*", close to each other in terms of string distance measures, must lie in different clusters. Its textual parts are equal ("*PARACETAMOL MG*"), but its numerical patterns are different.

After removing the numerical patterns, the textual part of each product description becomes represented by its set of 2-grams (or bigrams) [11]. Bigrams are any two-character sequences in some product description. They are used to compare strings. Two strings are similar if their bigram sets are similar to each other.

The second relevant issue is the definition of the distance function in Algorithm 2, which measures how far the description and the cluster are from each other. The Jaccard distance $d(A, B)$ between two strings represented by their 2-gram sets A and B is defined using the Jaccard similarity index; see Jaccard [10]:

$$d(A,B) = 1 - J(A,B) = 1 - \frac{|A \cap B|}{|A \cup B|}.$$

Note that by design, $0 \leq d(A,B) \leq 1$, since $0 \leq J(A,B) \leq 1$. If the two 2-gram sets are equal, i.e., $A = B$, then $J(A,B) = 1$ and $d(A,A) = 0$. If the two 2-gram sets have no common elements, i.e. $A \cap B = \emptyset$, then $J(A,B) = 0$ and $d(A,B) = 1$.

At the end of the second and last step of the clustering procedure, the representative element of each cluster was stored permanently in the database table of products.

4.2. Identification of abnormal product quantities

Based on a list of products of interest, one of the demands we faced was to identify abnormal product quantities purchased by municipal entities in Mato Grosso. In particular, we focus on finding purchases incompatible with those municipalities' populations, using historical purchase data from the entire state as a reference.

The general procedure for identifying abnormal quantities of a product purchased by municipalities or public bodies can be summarized as follows:

1. Identify the descriptions of the product to be analyzed.
2. Identify in which units of measurement this product is sold and choose one or several of them that are compatible or comparable.

3. Using the units of measurement chosen in the previous step, calculate how much of this product was purchased per inhabitant in the analyzed state in a relevant period (typically, in the last two or three years) to obtain a reference quantity.
4. Using the same units of measurement, identify how much of this specific product was purchased per inhabitant for each municipality (or public body).
5. The user (i.e., the analyst) determines, from all the municipalities (or public bodies) with the highest values purchased per inhabitant, those that should be investigated more in detail.

Steps 3 and 4 of this procedure can be fully automated. Most times, step 2 can be managed by choosing the most common unit of measurement for all electronic invoice products picked from step 1. Using this conservative strategy, if there is an abnormal product quantity in purchases with one single unit of measurement, the total quantity considering all units of measurement is also abnormal.

However, suppose there are several units of measurement, each representing a significant proportion of product purchases. In that case, the product quantities may have a different meaning for each unit, so they cannot be compared. To solve this problem, we implemented specific procedures that establish equivalences between the different units of measurement. More information about the work we performed with the units of measurement can be found in the Appendix (see the *sigla* table entry). Still, there were cases where it was not possible to identify specific units of measurement due to poor data quality. For example, if the unit of measurement is “box”, it is often impossible to know its size and how many products the box may have.

Step 1, on the other hand, is challenging to automate. For example, if the product is “gasoline”, it is unlikely that the clustering process described in Section 4.1 has unified all occurrences of “gasoline” into one single cluster. Instead, different descriptions containing “gasoline” may appear, including “regular gasoline”, “additivated gasoline”, or “gasoline pump” (unrelated to the product we are looking for). It will be up to the analyst to decide which of these descriptions corresponds to the same product. Typically, the analyst filters all the standardized descriptions using one or a few keywords and chooses those that, in his opinion, represent the same product. The clustering algorithm described in Section 4.1 enormously facilitates this selection process by the analyst since we use only one representative description for each cluster. After this choice, the rest of the processing is carried out automatically, showing the municipalities or public bodies with the highest quantities of the analyzed product per inhabitant.

We specifically prioritized the analysis of purchases of medicines distributed by the federal government program called *Farmácia Popular* [18] (“people’s pharmacy”) since this was one of the project’s main goals. Program *Farmácia Popular* was created in Brazil to give the population access at no cost to medicines that are essential for treating high-incidence chronic diseases such as diabetes, hypertension, and asthma. Since these medicines are distributed free of charge at the national level, their purchase by the municipalities should only occur in limited quantities in very specific cases, thus eliminating the possibility of any alternative explanation for their purchases in excessive amounts by public agencies. We also investigated the occurrence of abnormal quantities of other products with high frequency and weight in the states’ budgets, such as fuels (gasoline and diesel).

4.3. Identification of anomalous prices

We created two additional tables to identify anomalous prices that enrich the basic proposed relational model. First, table *preco_limite* (price limits) contains pre-calculated statistics describing the behavior of the product prices in each year. Second, table *classe_preco* (price category) classifies each electronic invoice item into one of five possible categories using the price's standard score [13]. The standard score (commonly called z-score) is the number of standard deviations by which an observed value is above or below the mean. An advantage of the z-score is that it allows price values to be placed on the same standard scale, normalizing the overpricing risk of different products. It enables the identification of high-risk prices used in a set of purchases.

4.3.1. Normalized prices

We identified normalized or standard prices for all analyzed products wherever possible.

Table *preco_limite* (price limits) contains a set of pre-calculated price statistics for each product sold using some unit of measurement in one specific year. This triad (formed by a standardized product description-NCM pair, a unit of measurement, and a year) will be referred to as a DUY (Description, Unit, Year). Each value assignment to a DUY triad defines a set of unit price observations x charged by the suppliers.

Typically, unit prices of products do not have a well-behaved normal distribution. Pricing data has many outliers. Products purchased in atypical circumstances, geographical differences, wholesale and retail sales presence, or even an input error can cause significant price variations. We also note that the outlier elimination process is carried out automatically due to many products, which prevents any human intervention in this case. To deal with this problem, we adopted an approach that eliminates atypical observations, using only the most reliable prices to estimate the central tendency measures.

First, for each set of price observations x associated with some DUY, a restricted set of unit prices x' is defined by taking only the prices in the interval $[\overline{x} - s(x), \overline{x} + s(x)]$, where $\overline{x}$ is the sample mean, and $s(x)$ is the sample standard deviation of the set of all available price observations. The prices in the restricted set x', which should no longer contain outliers, were used in the rest of our calculations. Then, we calculated the mean $\overline{x'}$, median, and standard deviation $s(x')$ of the restricted set of unit prices x'.

In short, there were two main steps for creating the *preco_limite* table:

1. Exclude from the original set of price observations x associated with some specific DUY triad the prices whose z-score [13] is greater than 1, or less than -1, creating a restricted set of unit prices x'.
2. Use the restricted set of unit prices x' for calculating the mean $\overline{x'}$, median, and standard deviation $s(x')$. These pre-calculated price statistics are stored in the *preco_limite* table.

Next, we present some details of our implementation that we consider relevant to other practitioners. As shown in the Appendix, we could not identify a small number of units of measurement used in product purchases. These observations do not belong to any DUY triad; consequently, we did not use them. Also, if the set of price observations x or the restricted set x' associated with some DUY contains only one element, we excluded it from *preco_limite* table since the standard deviation is undefined. In any other case, only one single line in the *preco_limite* table will be associated with each DUY triad.

4.3.2. Price classification and price alerts

Based on the statistics calculated in the *preco_limite* table, we implemented a price classification for electronic invoice items. It was stored in a new table called *classe_preco* (price category). The classification is established mainly according to the z-score of the invoice item price.

We recall that $\overline{x'}$ is the sample mean, and $s(x')$ is the sample standard deviation of the restricted set x' associated with some item's DUY triad. Each electronic invoice item was classified into one of five possible price categories: **R** (regular), **A** (low-intensity price alert), **B** (medium-intensity price alert), **C** (high-intensity price alert), and **X** (unclassified). The conditions that an electronic invoice item must meet to be included in each category are:

- Category **R**: An invoice item price that belongs to the interval $\left[\overline{x'} - s(x'), \overline{x'} + s(x')\right]$ is considered regular, i.e., its z-score is in the interval $[-1, 1]$.
- Category **A**: An invoice item price that does not fit into the previous category, but belongs to the interval $\left[\overline{x'} - 2s(x'), \overline{x'} + 2s(x')\right]$, i.e., its z-score is in the interval $[-2, 2]$, is considered to have a low intensity alert.
- Category **B**: An invoice item price that does not fit into the two previous categories, but belongs to the interval $\left[\overline{x'} - 3s(x'), \overline{x'} + 3s(x')\right]$, i.e., its z-score is in the interval $[-3, 3]$, is considered to have a medium intensity alert.
- Category **C**: An invoice item price that does not fit into the three previous categories, but is associated with a valid DUY triad, is considered to have a high intensity alert.
- Category **X**: An invoice item price not associated with a valid DUY triad (e.g., having a non-identified unit of measurement) is considered unclassified.

The following section describes the results obtained by this approach, including aggregate price alert statistics and some illustrative cases of risk patterns found in the electronic invoices of Mato Grosso.

5. Results and practical use

This section presents two sorts of results: general aggregate statistics and some specific cases identified as risk patterns. More detailed results can not be disclosed due to their sensitive nature.

The product description clustering process described in Section 4.1 analyzed more than eleven million electronic invoice items, with 2.3 million different product description-NCM pairs. Fingerprint matching, the first step of the clustering algorithm, is an automated, parameterless process. On the other hand, the second step, incremental clustering, is sensitive to the parameter α of Algorithm 2: the higher the value of $alpha$, the fewer the number of clusters in the final result. There is no "ideal" number of clusters to be obtained, so we manually checked samples of clustering results for each $\alpha = 0.1, 0.2, 0.3$, and so on. From $\alpha = 0.5$, the algorithm started to cluster unrelated product descriptions. Therefore, we used $\alpha = 0.4$ as the last parameter value that grouped descriptions that indeed represent the same product, intending to have the greatest confidence in the results of the proposed approach.

The number of different description-NCM pairs was reduced by roughly 50% after the two clustering steps. We recall that the main goal was to guarantee the correct grouping of the descriptions, eliminating errors in clusters with more than one product. No clusters with these characteristics were present after verification by sampling. The reduction rate of some description-NCM pairs was significantly higher for some specific NCM codes. Based on a list of products of interest and using specific keywords related to them, the human analyst performs the description selection and filtering process, choosing the description clusters that best represent the same product. The proposed clustering algorithm greatly facilitates this selection and filtering process by reducing the number of available options.

5.1. Purchases of abnormal product quantities in Mato Grosso

Several abnormal product quantities purchased by some municipalities and public bodies were identified. Typically, a single public entity made all purchases of a particular product within a specific municipality.

The case of purchases of a medicine called simvastatin (*"sinvastatina"*), a common lipid-lowering medication, in the municipality of Sinop, the fourth largest city in the state, is presented. In 2018, the public administration of this municipality acquired more than 1.1 million simvastatin tablets, corresponding to more than eight pills per inhabitant in that year. For comparison, the average per capita consumption in the entire state of Mato Grosso was 1.6 pills. It means that in Sinop, the consumption was more than five times higher than the state's average. Additionally, simvastatin is part of the *Farmácia Popular* program. It is distributed free of charge by the Federal Government. Therefore its purchase in any municipality or state must be made for very specific or exceptional needs, thus eliminating the possibility of many alternative explanations. The occurrence of exceptionally high demand is possible but unlikely to happen in the case of medicines used to treat high cholesterol. Further investigations are necessary to understand the occurrence of this specific fact. Determining anomalies and risks are objectives of the developed algorithms while investigating them or detailing the fraud is left to the analyst.

Other examples of abnormal quantities are related to purchases of fuels: gasoline and diesel. Araguainha, a small municipality of Mato Grosso state with about 1000 inhabitants, acquired more than 50,000 liters of gasoline in 2016, which is more than 50 liters per inhabitant, spending approximately R$ 200,000. This quantity is completely abnormal considering as a reference the average quantity of gasoline purchased per inhabitant in the entire state, which was only 2.7 liters per inhabitant.

Nevertheless, another small municipality of Mato Grosso state, Luciara, with about 2100 inhabitants, bought more than 290,000 liters of diesel in 2019, paying R$ 1.27 million. Over 1000 different purchase operations were carried out, with the average purchase value being R$ 1150. As a piece of anecdotal evidence, an investigation was carried out [6] in September 2016, detailing a possible buying of votes in exchange for fuel in four municipalities of Mato Grosso, including Luciara, showing that patterns similar to those found in this municipality have been already investigated in the past and corroborating the assertiveness of the approach proposed in this work.

5.2. Occurrences of high-risk prices in Mato Grosso

Regarding the price analysis, Figure 1 shows the proportion of unit price observations corresponding to each price category. We note that unit prices of products generally do not have a well-behaved normal distribution. However, the observed behavior is in line with what we expected, reducing the number of observations as the price alert intensity increases. Most observations were categorized as regular (**R**). A

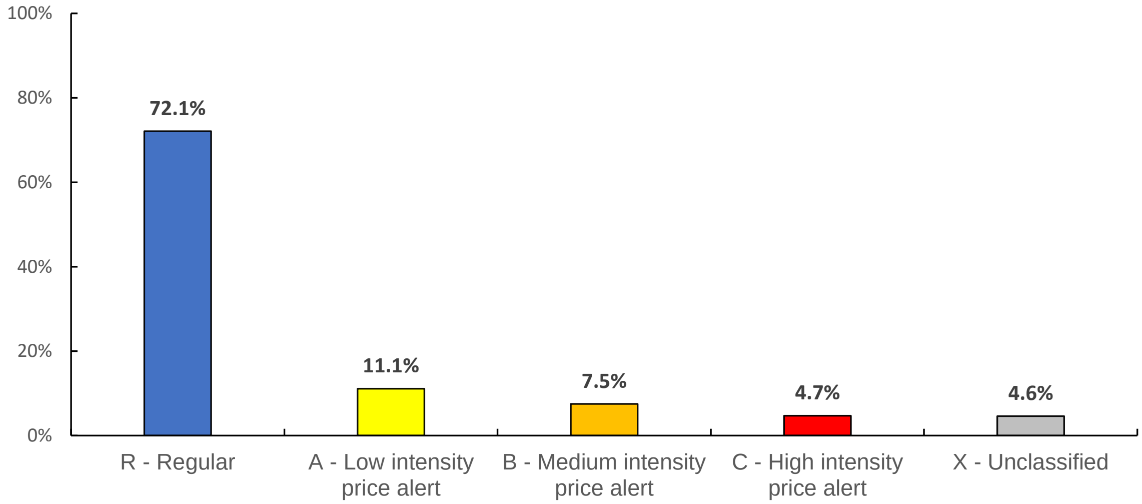

Fig. 1. Fraction in percent of unit price observations by category.

significant fraction of observations fell in categories **B** and **C**, corresponding to medium and high price alerts. We also remark that only 4.6% of invoice items were not classified due to the presence of, for example, a nonstandard unit of measurement or the occurrence of an atypical product description not present in any other invoice item.

The total value of invoice items categorized as high-intensity price alerts (category **C**) amounted to more than R$ 560 million. This value is significant considering the size of the state's economy and the total amounts spent in each year, as shown in Table 1.

As an example of an atypically high price, we take the purchases of frozen chicken in the municipality of Cuiabá in 2018. In that year, 1,538 purchases of this product were made in the state of Mato Grosso. The average price was R$ 7.67 per kilogram, with a standard deviation of just R$ 0.72, showing a high uniformity in the vast majority of price observations. However, 117 category **C** price alerts were associated with purchases made by two specific public entities located in Cuiabá. These 117 invoice items had the same unit price of R$ 15.14, close to the double of the average price, and the total spent value was R$ 5.2 million. All these 117 purchases were made to the same company, which could indicate public contract overpricing targeting one specific supplier.

In another case study revealing an unusual situation, we identified several high-price payments for aeromedical flights made by some state and local administration entities in Cuiabá. The flights were made by an Air Intensive Care Unit (Air ICUs). The total annual value spent on such flights has been growing from R$ 8.4 million in 2016 to R$ 16.7 million spent in 2019 alone, doubling in only three years. In each one of these two years, respectively, the public administration contracted between 500,000 and 900,000 flying kilometers. For comparison, the distance from the Earth to the Moon is 384,400 kilometers. Such long air travel distances contracted by the public administration are not easy to understand, at least at first sight. To find an alternative explanation, we found that aerial ICUs are often used to bring indigenous people from remote areas in case of medical emergencies. The Xingu Park is located at approximately 600 km from Cuiabá, the state capital of Mato Grosso. However, considering the air ICU mileage contracted by public entities only in 2019, and using a round-trip route from Cuiabá to Xingu Park, it would be possible to make 644 trips, approximately two per day in that year. The indigenous population of Mato Grosso is of approximately only 42,000 inhabitants, 56% of them living in urban areas. It doesn't seem easy to find a plausible explanation for this level of spending that totaled more than R$ 50 million in the analyzed period. As in the previous case, all payments were made to the same company.

6. Concluding remarks

Most state-level and federal Controller's Offices (*Controladorias*), Courts of Accounts (*Tribunais de Contas*), or NGOs (non-governmental organizations) in Brazil do not have the capability to carry out systematic risk assessments of public expenditures [24].

One possible solution to the problem of identifying the riskiest public contractors is the use of analytics and decision support systems to automatically provide government agencies with actionable information and specific risk patterns. In our approach, we identified two high-risk patterns inside the municipalities of one particular Brazilian state. First, we detected excessive, abnormal product quantities bought by public entities in each calendar year. Second, we identified anomalous prices paid by authorities and administrators of public entities. The results indicated that the proposed approach is capable of identifying various high-risk pattern occurrences in the electronic invoice data.

The total value of invoice items categorized as high-intensity price alerts (category **C**) amounted to more than R$ 560 million. We presented several examples of excessive, abnormal product quantities and atypically high prices paid by some municipalities and public bodies. The presented cases raise doubts as to whether the respective products or services were actually delivered to the municipalities and public bodies.

We notice that the presence of some risk pattern does not necessarily indicate the existence of fraud or misdoing. The results may indicate inefficiency, lack of planning, or some urgent purchases of products in case of unexpected demands, like those that arose during the COVID-19 pandemic. The decision-maker must verify each risk pattern to check for other alternative explanations before making any complaint. In some specific (but rare) cases, the amounts indicated in the electronic invoices may not have been actually paid, because electronic invoices can be canceled after being issued and registered.

As a further improvement of this project, we consider the possibility of dynamically adding new invoices to an existing database without recalculating all the auxiliary tables.

Executive Order number 10.209, published on January 22, 2020 [3], determined the publication on the internet in open access of all electronic invoices related to Federal Government purchases in Brazil. The dissemination of these datasets will make it possible to expand the social control of public spending. It will provide greater detail on the items and services acquired by the public administration, allowing the use of approaches similar to the one presented in this work in a broader context.

This approach can be used by Brazilian and foreign practitioners and government agencies for identifying abnormal product quantities and anomalous prices in electronic invoices or any purchase data where the information of the purchased product and its price has a high degree of reliability.

Acknowledgments

The authors thank Leonardo Vilela Ribeiro for his assistance during the preparation of this work. The authors also thank Daniel Ortega Nieto, to whom they are extremely grateful for his unwavering support.

Ruben Interian was supported by research grant 2021/12456-5 of the São Paulo Research Foundation (FAPESP). The work of Celso C. Ribeiro was partially supported by CNPq research grants 309869/2020-0 and 425778/2016-9 and by FAPERJ (Fundação de Amparo à Pesquisa do Estado do Rio de Janeiro) research grant E-26/200.926/2021.

References

[1] Abbas, H.S.M., Qaisar, Z.H., Ali, G., Alturise, F., Alkhalifah, T., 2022. Impact of cybersecurity measures on improving institutional governance and digitalization for sustainable healthcare. *PLOS ONE* 17, 11, 1–13.

[2] Carneiro, D., Veloso, P., Ventura, A., Palumbo, G., Costa, J., 2020. Network analysis for fraud detection in Portuguese public procurement. In Analide, C., Novais, P., Camacho, D. and Yin, H. (eds), *Intelligent Data Engineering and Automated Learning – IDEAL 2020*, *Lecture Notes in Computer Science*. Vol. 12490. Springer, Cham, pp. 390–401.

[3] Controladoria-Geral da União, 2020. Decreto permite acesso da CGU a informações fiscais. online reference at `https://www.gov.br/cgu/pt-br/assuntos/noticias/2020/01/decreto-permite-acesso-da-cgu-a-informacoes-fiscais`, last access on April 29, 2023.

[4] van Erven, G.C.G., Holanda, M., Carvalho, R.N., 2017. Detecting evidence of fraud in the Brazilian government using graph databases. In Rocha, A., Correia, A.M., Adeli, H., Reis, L.P. and Costanzo, S. (eds), *Recent Advances in Information Systems and Technologies*, *Advances in Intelligent Systems and Computing*. Vol. 569. Springer, Cham, pp. 464–473.

[5] Fazekas, M., King, L.P., 2019. Perils of development funding? The tale of EU Funds and grand corruption in Central and Eastern Europe. *Regulation & Governance* 13, 405–430.

[6] G1, 2016. Polícia investiga suposta compra de voto em troca de combustível em MT. online reference at `http://g1.globo.com/mato-grosso/eleicoes/2016/noticia/2016/09/policia-investiga-suposta-compra-de-voto-em-troca-de-combustivel-em-mt.html`, last access on April 29, 2023.

[7] Gupta, N., Ujjwal, R.L., 2013. An efficient incremental clustering algorithm. *World of Computer Science & Information Technology* 3, 97–99.

[8] IBGE, 2021. Códigos dos municípios IBGE. online reference at `https://www.ibge.gov.br/explica/codigos-dos-municipios.php`, last access on April 29, 2023.

[9] Interian, R., Mendoza, I., Bernardini, F., Viterbo, J., 2022. Unified vocabulary in official gazettes: An exploratory study on procurement data. In *Proceedings of the 15th International Conference on Theory and Practice of Electronic Governance*, ACM, New York, NY, USA, p. 195–202.

[10] Jaccard, P., 1912. The distribution of the flora in the Alpine zone. *New Phytologist* 11, 2, 37–50.

[11] Kim, J.Y., Shawe-Taylor, J., 1992. An approximate string-matching algorithm. *Theoretical Computer Science* 92, 107–117.

[12] Kostyo, K., 2006. *Handbook for Curbing Corruption in Public Procurement*. Transparency International, Berlin. Online reference at `https://www.transparency.org/en/publications/handbook-for-curbing-corruption-in-public-procurement`, last access on April 29, 2023.

[13] Kreyszig, E., 1979. *Advanced Engineering Mathematics* (10th edn.). Wiley.

[14] Lima, M., Silva, R., Mendes, F.L.S., Carvalho, L.R., Araujo, A., Vidal, F.B., 2020. Inferring about fraudulent collusion risk on Brazilian public works contracts in official texts using a Bi-LSTM approach. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, Association for Computational Linguistics, pp. 1580–1588. Online reference at `https://aclanthology.org/2020.findings-emnlp.143`, last access on April 29, 2023.

[15] Luna-Pla, I., Nicolás-Carlock, J.R., 2020. Corruption and complexity: A scientific framework for the analysis of corruption networks. *Applied Network Science* 5, 13.

[16] Mercosur, 2023. Nomenclatura Común (NCM) y Arancel Externo Común (AEC). online reference at `https://www.mercosur.int/politica-comercial/ncm/`, last access on April 29, 2023.

[17] Ministério da Economia, 2023. Portal da Nota Fiscal Eletrônica. online reference at `https://www.nfe.fazenda.gov.br`, last access on April 29, 2023.

[18] Ministério da Saúde, 2023. Farmácia Popular. online reference at `https://www.gov.br/saude/pt-br/acesso-a-informacao/acoes-e-programas/farmacia-popular`, last access on April 29, 2023.

[19] Modrušan, N., Rabuzin, K., Mršic, L., 2021. Review of public procurement fraud detection techniques powered by emerging technologies. *International Journal of Advanced Computer Science and Applications* 12, 2.

[20] Nicolás-Carlock, J.R., Luna-Pla, I., 2021. Conspiracy of corporate networks in corruption scandals. *Frontiers in Physics* 9, 667471.

[21] Roselli, L.R.P., Almeida, A.T., 2023. The use of the success-based decision rule to support the holistic evaluation process in FITradeoff. *International Transactions in Operational Research* 30, 1299–1319.

[22] Santos, F.B., Souza, K.R., 2016. *Como Combater a Corrupção em Licitações*. Editora Forum.

[23] Secretary-General of the OECD, 2019. Analytics for Integrity: Data Driven Approach for Enhancing Corruption and Fraud Risk Assessments. Technical report, OECD.

[24] Velasco, R.B., Carpanese, I., Interian, R., Paulo Neto, O.C.G., Ribeiro, C.C., 2021. A decision support system for fraud detection in public procurement. *International Transactions in Operational Research* 28, 27–47.

[25] Ware, G.T., Moss, S., Campos, J.E., Noone, G.P., 2007. Corruption in public procurement: A perennial challenge. In Campos, J.E. and Pradhan, S. (eds), *The Many Faces of Corruption : Tracking Vulnerabilities at the Sector Level*. The International Bank for Reconstruction and Development – The World Bank, chapter 9, pp. 295–334.

[26] Ware, G.T., Noone, G.P., 2005. The anatomy of transnational corruption. *International Affairs Review* 14, 29–51.

[27] Westerski, A., Kanagasabai, R., Shaham, E., Narayanan, A., Wong, J., Singh, M., 2021. Explainable anomaly detection for procurement fraud identification – Lessons from practical deployments. *International Transactions in Operational Research* 28, 3276–3302.

A. Relational model for electronic invoices

We present in this appendix the relational database model used in our approach.

Figure 2 presents the initial relational database structure proposed for detecting risk patterns in the electronic invoices data obtained from XML files. We included the information recovered from the analyzed XML tags, the generating tables *nota_fiscal*, *item_nota*, *emissor*, *destinatario* and *frete*. We omitted some non-essential table columns to simplify the understanding of the schema.

Figure 3 presents the relational database structure after creating additional tables that enrich the basic model. In addition to the aforementioned, we included the tables *produto*, *sigla*, *preco_limite* and *classe_preco*.

Next, we present the descriptions and other relevant details about the generation of the tables in the created relational database model. We also describe the main columns of each table. More information about the original electronic invoice XML format used as a source can be found in the Taxpayer Manual [17], which details all possible invoice attributes.

Tables. *destinatario* (recipient), *emissor* (issuer)

Table *destinatario* contains information about public entities that purchase goods and services using public funds. Table *emissor* includes information on companies that provide goods and services and issue electronic invoices. Both tables have identical structures.

For creating those tables, tags <*dest*> (for *destinatario*) and <*emit*> (for *emissor*) must be used.

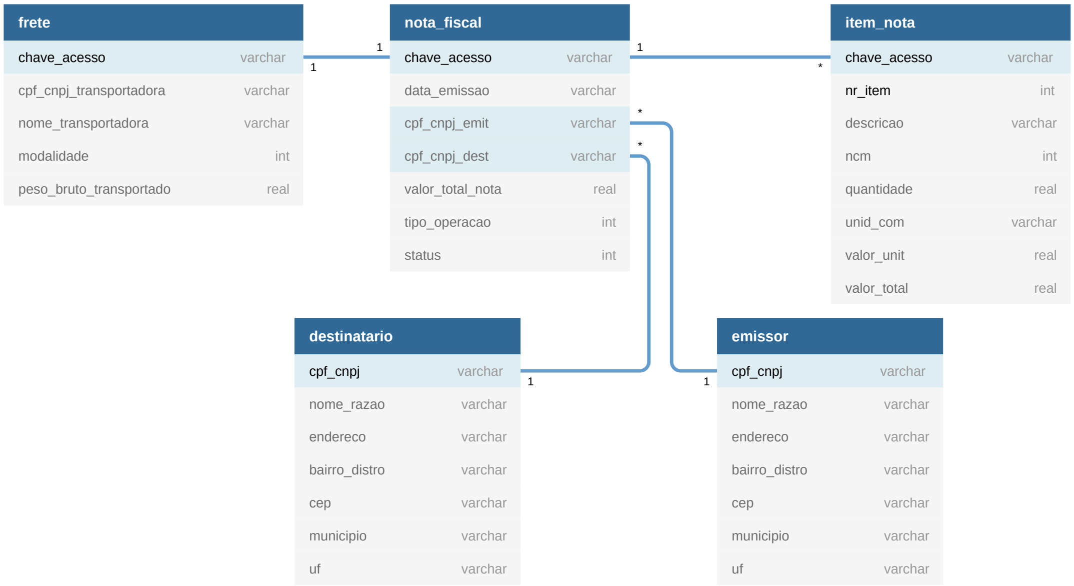

Fig. 2. Basic relational model composed of five tables obtained from XML invoice files.

The number of lines in these tables must equal the number of different issuer and recipient identification numbers in the dataset. If several electronic invoices had the same issuer's identification number (CNPJ), a single line must be created in table *emissor*.

To resolve any inconsistencies in the recipient or issuer data in our dataset (for example, in the case of different municipalities indicated in the address of some company), we take the mode (i.e., the most frequent value) of each variable related to this specific entity. For example, suppose that municipality A occurs in ten invoices of some company and municipality B in only one invoice. In that case, it is reasonable to assume that municipality B occurred due to some input error.

The main columns and their descriptions are:

1. **cpf_cnpj:** Company identification number (CNPJ) or physical person identification number (CPF) of recipient or issuer. XML tags: <CNPJ>, <CPF>. (**primary key**)
2. **nome_razao:** Company name or physical person's name. XML tag: <xNome>.
3. **endereco:** Address. XML tags: <xLgr>, <nro>.
4. **bairro_distro:** District. XML tag: <xBairro>.
5. **cep:** Brazilian postal code (*Código de Endereçamento Postal*, *CEP*). XML tag: <CEP>.
6. **municipio:** Municipality. XML tag: <xMun>.
7. **uf:** Federative unit. XML tag: <UF>.

Table. *frete* (shipping)

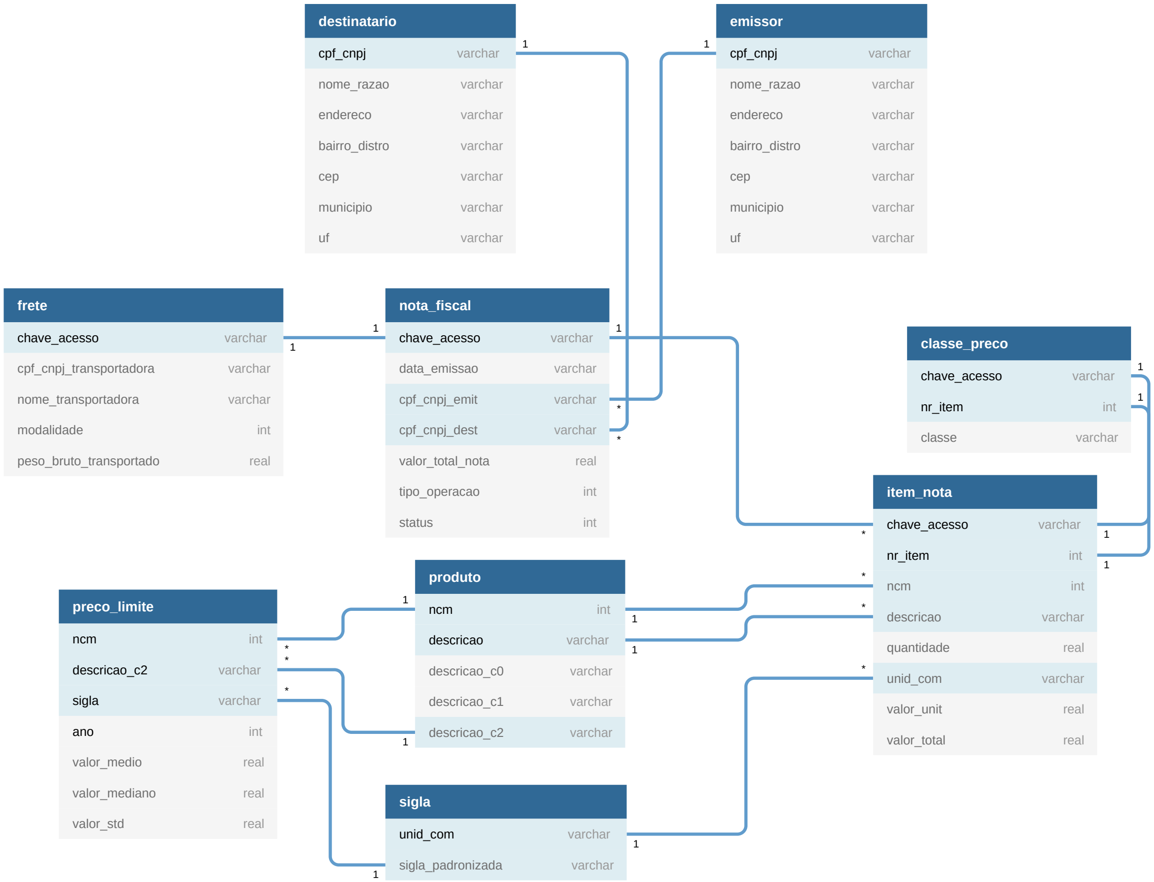

Fig. 3. Enriched relational model composed of nine tables. Four additional tables were created using the algorithms described in Section 4.

This table contains information about the shipping of the products. As shipping information has not been used in identifying the risk patterns presented in this article, we have omitted further details about this table.

Table. *nota_fiscal* (electronic invoice)

This table contains general information about each electronic invoice issued by companies that are suppliers of public entities. Two critical fields impact the processing and use of this table: *tipo_operacao* (operation type) and *status*.

The operation type allows differentiating income and outlay invoices. An outlay invoice (*nota de saída*, having *tipo_operacao*=1) registers a purchase of a product or service by a customer (in our case, a public entity). Usually, a product cannot leave the warehouse without generating an outlay invoice. An income invoice (*nota de entrada*, having *tipo_operacao*=0) is issued in some specific situations, like when the customer returns the product. The vast majority of electronic invoices we received, about 98%, were outlay invoices because they register sales made by private companies to public entities. In this study, we analyzed only outlay invoices. For discarding alternative explanations of specific risk patterns, like purchases canceled long after the issue, we checked the existence of an income invoice having the same value as some risky outlay invoice.

On the other hand, the status of the electronic invoice determines at which step of the issuing process the invoice is. The vast majority of electronic invoices, also about 98%, have a status value of 100, which indicates that the invoice was authorized. In our algorithms, we used only authorized invoices. If there is any doubt about whether some risky electronic invoice is still valid, the electronic invoices site [17] may be checked for the most up-to-date information.

For creating this table, we used attributes contained in the XML tag <*ide*>. The number of lines in this table must be equal to the number of electronic invoices being processed.

The main columns and their descriptions are:

1. **chave_acesso:** Access key, a unique identifier of each electronic invoice. (**primary key**)
2. **data_emissao:** Invoice's date of issue. XML tag: <dhEmi>.
3. **cpf_cnpj_emit:** Identification number of the issuer (**foreign key**, *emissor* table);
4. **cpf_cnpj_dest:** Identification number of the recipient (**foreign key**, *destinatario* table).
5. **valor_total_nota:** Total value of invoice products.
6. **tipo_operacao:** Operation type: 0 – Income, 1 – Outlay. XML tag: <tpNF>.
7. **status:** A code that indicates invoice status, allowing to identify if the note is active of was canceled. XML tag: <cStat>.

Table. *item_nota* (invoice item)

This table contains information about each electronic invoice item. Items can represent products or services. The number of lines in this table must equal the total number of items across all electronic invoices being processed.

Main columns and their descriptions:

1. **chave_acesso:** Access key, a unique identifier of each electronic invoice (**primary key**);
2. **nr_item:** Number of the item in the electronic invoice (**primary key**);
3. **ncm:** Product NCM, a numerical code indicating product type. XML tag: <NCM>.
4. **descricao:** Product or service description. XML tag: <xProd>. (**foreign key**, *produto* table);
5. **quantidade** Purchased quantity. XML tag: <qCom>.
6. **unid_com** Unit of measurement. XML tag: <uCom>. (**foreign key**, *sigla* table);
7. **valor_unit** Unit price. XML tag: <vUnCom>.
8. **valor_total:** Total gross value. XML tag: <vProd>.

Table. *produto* (product)

This table represents the products purchased by public entities. It is generated from different *descricao-ncm* value pairs extracted from the *item_nota* table. NCM is an acronym of *Nomenclatura Comum do Mercosul* [16] (Mercosur Common Nomenclature), a numerical code used for classifying product items. However, this classification alone is unsuitable since several products with significantly different prices may fall into the same NCM.

The name of a particular product can be written in many different ways. Product descriptions also may contain spelling errors and typos. Therefore, the process of clustering product descriptions needs to be carried out, adding to the existing columns (*ncm* and *descricao*) three additional columns that represent the description standardization process: *descricao_c0*, *descricao_c1*, and *descricao_c2*, as explained in Section 4.1.

The main columns and their descriptions are:

1. **ncm:** Product NCM code (**primary key**);
2. **descricao:** Original product description (**primary key**);
3. **descricao_c0:** Treated product description created after the cleaning procedure;
4. **descricao_c1:** Representative element of product descriptions' cluster after the first step of the clustering procedure;
5. **descricao_c2:** Representative element of product descriptions' cluster after the second and last step of the clustering procedure.

Table. *sigla* (unit of measurement)

This table contains standardized units of measurement used in electronic invoices. The primary key is obtained by extracting all different units of measurement, in uppercase, from the *unid_com* column in the *item_nota* table. The extracted units of measurement included strings like *KG*, *KLG*, *KG.*, *1KG* (kilogram); *UN*, *UND*, *UNID.* (unit). The table contains a few hundred different strings corresponding to all units of measurement used in items of products that appear in our database. Therefore it was analyzed manually to ensure highly reliable results.

We created a new column *sigla_padronizada* for mapping each unit of measurement to its standard representation. The analyst performs this task to identify different forms of writing the same unit. For example, "UNID." corresponds to the unit of measurement "UN" (one unit of some product), "1KG" corresponds to "KG". Due to poor data quality, we could not identify the meaning of a few units of measurement. For this reason, the item prices associated with these unclassified measurement units received the price category **X** - "unclassified" in the table *classe_preco*.

The main columns and their descriptions are:

1. **unid_com:** Unit of measurement (**primary key**);
2. **sigla_padronizada:** Standardized unit of measurement.

Table. *preco_limite* (price limits)

This table contains pre-calculated statistics that describe the behavior of the product prices sold each year. The generation of this table was described in more detail in Section 4.3.1.

The main columns and their descriptions are:

1. **ncm:** Product NCM code (**primary key**);

2. **descricao_c2:** Standardized product description obtained as described in Section 4.1 (**primary key**);
3. **sigla:** Standardized unit of measurement (**primary key**);
4. **ano:** Electronic invoice year (**primary key**);
5. **num_observacoes:** Number of observations used to generate the statistics;
6. **valor_medio:** Mean of product prices whose calculation is described in Section 4.3.1;
7. **valor_mediano:** Median of product prices whose calculation is described in Section 4.3.1.
8. **valor_std:** Standard deviation of product prices whose calculation is described in Section 4.3.1;

Table. classe_preco (price category)

This table classifies each electronic invoice item into one of five possible price categories. The generation of this table is described in more detail in Section 4.3.2.

Main columns and their descriptions:

1. **chave_acesso:** Electronic invoice access key (**primary key**);
2. **nr_item:** Number of the item in the electronic invoice (**primary key**);
3. **classe:** Item category as defined in Section 4.3.2.