POLITECNICO DI TORINO

DEPARTMENT OF CONTROL AND COMPUTER ENGINEERING

Master of Science in Data Science and Engineering

Master Degree Thesis

Detection of Anomalous Contracts of the Italian Public Administration

Supervisors
Prof. Antonio Vetrò
Dr. Davide Allavena

Candidato
Giulio Zabotto

Academic Year 2021-2022

Summary

ContrattiPubblici is a search engine and business intelligence tool for procurement contracts issued by the Italian public administration. Few contracts contain typographical errors, where their amount or duration are abnormal, either excessive or insufficient for the supply object of the contract.

Contracts contaminated with errors spoil the quality of the information displayed by ContrattiPubblici. Even if rare, any information derived by aggregating contracts will eventually be biased by such errors. Here arises the need of identifying and removing contracts having typographical errors. A collateral objective to find contracts showing errors is to score how reliable the information displayed by the contracts is.

The task is to develop an unsupervised anomaly detector able to indentify which contracts display wrong information with regards to their amount and duration.

After the removal of missing values, the **data set** consists of a table of 755660 rows. Each row is a contract issued by a contracting authority in the Veneto region from 2016 to 2018. The data set is not annotated, hence the task is unsupervised. The features characterize the contracts identifying the contracting authority that issued the contract, the firm to whom the contract is awarded, the so-called *business entity*, the value of the supply, the object of the supply, the start date of the contract and its termination, the type of awarding procedures, i.e. how to contractor chooses the supplier.

The lack of an annotated data set impedes the evaluation of the effectiveness of outlier detectors. Thus we opted for developing a baseline model with which compare more advanced ones.

The **baseline model** is an ensemble of a set of heuristics developed with the aid of a domain expert, named *rule model* and a statistical model that detect which contracts lie in the tails of their variable probabilistic distributions, thus the name *tail model*.

The *rule model* heuristics find anomalous contracts that 1. have amount values bigger than both the annual expenditure of the contracting authority

and the annual specific revenue of the supplier firm; 2. are direct assignments or assignment under framework agreement and their contract duration lasts longer than ten years; 3. have an amount value bigger than twenty-five times the annual specific revenue of the business entity.

The *tail model* embodies the idea that anomalous contracts reside in the upper or lower tails of a distribution of contracts grouped by contracting authority or business entity. The distributions of the amount variable and the duration variable are not known. For this reason, the model applies the Chebyshev's inequality that does not assume any distribution of the data. The contracts flagged as outliers by the *rule model* and the *tail model* comprise the ground truth with which compare the next models.

The **data preparation** step consisted in removing of all contracting authorities and business entities having less than ten contracts. Then, Contracts are grouped by contracting party and to each group is applied the following pipeline: normalization, Box Cox transformation, normalization. Afterwards, each contract is enriched with the mean, variance and skewness computed on the group the contract belongs to.

The advanced models are a kernel density estimator (KDE), a Gaussian mixtures model (GMM) and a one-class support vector machine (OCSVM). The multivariate nature of these models overcome the limitation of the baseline model of searching outlier considering one feature at a time. On the one hand, KDE and GMM assume that typographical errors seldom appear. If the contract probability density is know, then contracts have a lower probability. On the other hand, OCSVM assumes that anomalous contracts are well separated in the feature space the model projects the contracts onto, distinguishing them by means of euclidean distance. The reason behind the choice of shallow learners in place deep learners resides in necessity of having an outlier detector that does not requires a vast statistical background to interpret their results and it is easy to implement in a production setting.

The KDE parameter that affects the most the resulting probability density estimate is the bandwidth. The **optimization** of the KDE's bandwidth paratemer is carried out by minimizing the *mean integrated squared error* (MISE). Its optimization is carried out by minimizing the *mean integrated squared error* (MISE). In the literature there are three alternative ways. The first assumes that the data distribution is normal. The second minimizes the *asymptotic mean integral squared error* with an iterative process that would require an infinite number of steps unless the assumption of normality of the data distribution "plugged in" at a given step; in our experiments is the second step. The last bandwidth selector uses cross validation to approximate

the MISE. The implementation of the aforementioned bandwidth selectors originates three different models: KDE *normal scale* (NS), *plug in* (PI) and *cross validation* (CV).

The GMM's parameters are optimized minimizing the Bayesian information.

The OCSVM's *contamination* parameter is set to match the ratio of outliers found by the baseline model. The kernel used is a Gaussian radial basis function with gamma defined as the inverse of the product of the number of features and the average variance of data set columns.

Results.The training test split is 70-30 percent of the contracts whose awarding procedure is open. On the one hand, the discriminative approach embodied by OCSVM leads to poor results. It performs slightly better than a random guesser, as it average AUC is 0.510. The results suggest that the outlier are not well separated in the feature space.

On the other hand, the generative approach yields better results, all the KDE models and the GMM AUC averages are above the 0.800 threshold, with the best one being the GMM at 0.840. The better performance of the generative models is due to the presence of the outliers defined by the *tail model*.

The best model is KDE NS. The ROC curve shows that the model does not assign high probabilities to any outlier. Considering as the outlier class as the "positive class", the threshold that maximizes the specificity at .9895 flags 1% of the total number of cases, while sensitivity is 0.0556. Conversely, the threshold that maximizes the sensitivity at 1.00 flags 42% of the total number of contracts. Comparing the outliers flagged by the baseline with those of the advanced is a flawed by the fact that not all the anomalous contracts are found by the baseline model. Hence, a manual check of the contracts classified as outliers by the KDE model is needed. The research shows that all contracts flagged as outliers lie in tail of either the duration or amount feature, thus exceeding the ability of the *tail model*. Among these contracts, the almost ten percent of them are typographical errors. Yet, there exists typographical errors that are not flagged by the model.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 The necessity of an anomaly detection tool

The web portal *ContrattiPubblici.org* provides a set of tools for private companies and public administration entities to improve their knowledge of the public procurement market from a business intelligence perspective. The portal may show users the overall dimension of the market they are interested in, where the public entities that have already bought that good or service are located, what was the quantity exchanged, the type of award procedure, or other related information.

The granularity of the user's research is limited to a single contract. Indeed, the contracts are the atomic elements of the database that constitutes the query tool of the portal. This database collects those contracts issued by the general government, hospitals, universities, public authorities, and private companies, even partially own by the government. From now all these entities will be called *public entities*.

Every public entity is bound by Italian law [1] to provide the general public with a file that summarizes the content of the contract. Then, this file is processed and eventually cleaned up before reaching the main database where it will be incorporated by users' queries.

To provide customers with the most accurate and reliable information, the data must not contain outliers or errors. Among the features that characterize a public procurement contract, the most sensitive is the amount, that is

[1] L. 190/2012

the value of the contract. Outliers regarding the value of the contract may lead users to overestimate or underestimate the market or, more in general, to retrieve misleading, when not utterly deceitful, information. Hence, outliers must be recognized, eventually flagged, or deleted from the main data set.

This thesis aims at detecting and analyzing the outlier contracts, supplying a tool to further clean the data set and improve the quality of the service provided to the customers of *ContrattiPubblici.org*.

1.2 Legislative framework

In 2012 the Anti-Corruption law established the National Anti-Corruption authority, ANAC. Its main purpose is to watch, prevent, and contrast corruption and illegality in public administration [2]; among its duties, the administrative authority supervises public procurement.

Every six months the public administrations are requested to issue and update the list of provisions regarding the authorization or concession; the choice of the tender for the committed works, supplies, or services [3], specifying the roles assigned to each tender and the amount spent, if any. Moreover, public administrations have to issue contracts signed with private subjects or other public administrations [2].

To fully achieve the transparency the Anti-Corruption law aims at, information regarding the contracts signed by public entities must be made accessible to everyone. The Freedom of Information Act (FOIA) grants access to information held by the public administration to the general public. The act allows the citizen watch over the public administration's activities, which was previously limited to the public authority ANAC.

The Freedom of Information act translates into the possibility for citizens or private companies to download from the internet the list of the provisions public entities have to deliver to the ANAC authority in a machine-readable format.

1.2.1 The public procurement contract

The entries in the data set object under analysis are the contract lots for public procurement. The law regulating the matter is the Italian Public Contract Code. According to article 1, *this Code establishes rules on public procurement contracts and concession contacts by contracting authorities*

and contracting entities of services, supplies, works as well as on designs contests. [4]. The parties of the contract are the *contracting entities* and the *economic operator*. With *contracting entity*, the code refers to State public administrations; local public authorities, such as municipalities; the bodies governed by public law, such as hospitals or the Italian National Institute for Social Security (*it.* INPS); the *central purchasing bodies*, i.e. consortia of usually small contracting entities to improve their bargaining power; public-owned companies. (art.3 letter e) With *economic operator*, the code refers to any business entity, that is, any legal entity whose purpose is to gain profits, that offers *the execution of works and/or, the supply of products or the provision of services on the market;* (art.3 letter p).

Among the many procedures regulated by the code - there are almost forty types - the most common and the most valuable are the *open procedure*, the *negotiated procedure*, the *direct assignment* and *assignment based on a Framework Agreement*.

The open procedure requires a public call of tenders to deliver an offer for satisfying the needs of the public call. Once the time for bids is elapsed, the contracting authority selects the best offer, according to their criteria. Eventually, the authority announces the winner of the contest and generally explains why that tender won. [4]

The negotiated procedure applies in those cases in which the matter of the order is sensitive. For this reason, the call is preemptively limited to selected economic operators satisfying the needs of the call.

The assignment, which we will call *direct assignment* to distinguish it from a generic assignment, applies whenever the object of the procurement is relatively small and not complicated in needs. The assignment may be used for the purchase of writing materials for the staff of a small municipality or to carry out out repair work on a boiler. This procedure does not require an open call; the contracting entity directly assigns the work, supply, and service to a business entity.

A peculiar type of direct assignment is one based on a framework agreement. The framework agreement is not an award procedure, but rather a contractual tool [5] that, on a whole, specifies the prices, the minimum quantities, the maximum delivery time, and all the requirements that the contracting entity needs for the given order. This distinction is relevant because the input data set presents it. Yet, the laws that apply are the same for direct assignment.

1.2.2 Award procedures thresholds

In order to determine the proper award procedure apt to the order, the contract value and the type of sector are the main drivers of the decision. The legislation is time-dependent; this is particularly true for the award procedures' thresholds as they act as upper limits for the use of the different procedures. As lower threshold procedures leave the contracting entity a higher degree of discretion and the execution of the order is significantly quicker, the thresholds are a matter of political interest and they change frequently. The data set in use collects the contracts from 1st January 2016 to 31st December 2018; in this time span, the law regarding the threshold is uniform. The tables 1.2, 1.1 show the thresholds valid at the time.

Works sector	
Award Procedure	Thresholds [€]
Direct assignment	lot value $<$ 150,000
Negotiated procedure	150,000 $\leq$ lot value $\leq$ 1,000,000
Open procedure	lot value $\geq$ 1,000,000

Table 1.1: Value thresholds of award procedures for the works sector according to the 2018 legislation

Supplies and services sectors	
Award Procedure	Thresholds [€]
Direct assignment	lot value $<$ 40,000
Negotiated procedure	40,000 $\leq$ lot value $\leq$ 214,000
Open procedure	lot value $\geq$ 214,000

Table 1.2: Value thresholds of award procedures for the supplies and services sectors according to the 2018 legislation

1.2.3 Common Procurement Vocabulary

The Common Procurement Vocabulary is a system designed for the classification of goods, services, and works mandatory for public procurement in the European Union. The Italian Public Contract Code implements the European directive [6] that mandates it. The entries of the *vocabulary* are nine-digit codes. As the *Information about European Procurement* web page reports, the codes incorporate a tree classification system where

- The first two digits identify the divisions (XX000000-Y);
- The first three digits identify the groups (XXX00000-Y);
- The first four digits identify the classes (XXXX0000-Y);
- The first five digits identify the categories (XXXXX000-Y);

Each of the last three digits gives a greater degree of precision within each category.

A ninth digit serves to verify the previous digits. ([7])

Whilst the law mandates the classification of public procurement contracts according to this system, only 7 percent of the contracts report the CPV code.

1.3 Structure of the thesis

The thesis consists of two parts. The first one describes the input data set and aims at delving into the its features to develop knowledge and insights. The second part reports the development of a series of methods to detect outliers within the data set in the light of the discoveries made in the previous part.

The output of the fist part will be forwarded to the methods of the second; it can be considered as a pre-process step to the second part.

Part I

Data Exploration

Chapter 2

Materials

The data set collects the contract lots issued by contracting entities located in the Veneto region spanning from 1st January 2016 to 31st December 2018.

The choice of time and space is not random. In these three years, the legislation regarding public procurement is uniform. Hence, contract award procedures are uniquely dependent on the lot amount, not on the starting date of the contract.

Contracts are confined to a single region because the data set is smaller than the one collecting the contracts issued by the contracting authority of the whole country; consequently, it can be handled with ease.

In addition, the regional confinement of the data set allows the possibility to check whether the model built on the data of a single region is enough to model the entire country — even if the peculiarities of the Veneto region must be taken into account for the test to be reliable and sound.

Moreover, even if the data set collected the contracts issued by all Italian contracting entities, it would still miss the assumption of catching all the value produced by the economic operators. It is not uncommon to have an economic operator that works abroad as well. Indeed, a large portion of public procurement contracts issued by hospitals for the provision of drugs or vaccines is purchased from international pharmaceutical companies such as GlaxoSmithKlein or Pfizer. One cannot reasonably assume that the revenues of these companies are limited to the Italian market.

The data set does not picture the whole public procurement contracts for the region, but only those issued by the contracting entities *located* in the Veneto region. For this reason, the data set does not actually represent the entire region's public expenditure. For example, suppose that a contract for hiring nurses is issued by the State government in place of the single hospital

as it usually is; then, the expenditure would not appear in the data set as the head office of the State administration is in Rome, although a portion of the nurses hired would actually work in Veneto.

The rows of the relational database are the contracts' *lots*, as already mentioned. A single public contract may cover several sub-calls; these are called *lots* as they are partitions of the same contract. Each one of them may be won by an economic operator. The same firm can win multiple lots of the same contract.

Moreover, a single contract lot may be assigned to an ensemble of firms. The ensemble of firms has a leading firm and an array of ancillary firms. A typical example of this kind of call is the construction of a building. The leading company manages the subordinate firms that handle the specialized works that the building requires, such as the thermal plant, the electrical wiring, or the sewage system. Generally, the leading company receives money from the public contract and distributes it according to the agreement between the ensemble of firms.

The amounts of public money received by each firm are not written in the data set entry regarding their contract lot, hence, in the cases where a lot is won by more than one firm the sum of money will be equally distributed among all the winners. In a Bayesian perspective, this is the assumption of an uninformative prior.

The actual data set consists of two *comma separated value* files: one describes the public procurement contracts, and the other assigns a winner to each contract. The former will be referred to as the *lot data set*, while the latter as to the *winner data set*.

The two data sets will be joined together in order to form a single data set. Each row will have the tuple (*contracting authority issuing the lot, lot amount, economic operator winning the lot*). The semantics of the tuple lies in the nature of the contract that is defined by the contracting parties and the object to be exchanged between them.

2.1 The *lot* data set features

For each entry of this paragraph, the bold text refers to the name of the features, the italics to the content type of the features, while the text written with an unmodified font describes the content of the feature.

Lot id *integer*
: The contract lot identifier. The primary key of the database

Pa id *integer*
The identifier of the contracting authority issuing the contract lot. *Pa* stands for *public authority* or *public administration*, that is a calque from the italian *amministrazione pubblica*.

Object *string*
A brief textual description of the object of the contract lot

Award procedure id *integer*
The identifier of the contract award procedure

Amount *float*
The amount spent by the contracting authority

Auction start price *float*
Whenever the award procedure calls for a dutch auction, the field describes the starting price for the work, supply, or service commissioned.

Amount paid *float*
The amount paid by the contracting authority.

Start date *date*
The effective date of the contract.

End date *date*
The termination date of the contract.

Inferred date *date*
The contract start date is extracted by a machine-learning algorithm from the attached documents.

sector id *integer*
The code identifies the sector of the commission. The sectors are works, services, or supplies.

Legal form *integer*
The identifier of the legal form of the contracting entity.

Uber legal form *integer*
The identifier of the classification of the contracting authorities into macro-categories.

CPV *integer*
: The first two digits of the Common Procurement Vocabulary code as inferred by Synapta's algorithm.

original CPV *integer*
: The complete nine-digit Common Procurement Vocabulary code as written in the contract's original XML file, if found.

The *uber legal form* specifies a category for contracting entities and business entities. The categories are the following:

- Environment and habitats
- State government *(it. Amministrazione dello Stato)*
- Regional and local government *(it. Amministrazione regionale e locale)*
- Culture and tourism
- Economics and labor
- Private entity
- Public economic entity
- Public non-economic entity
- Justice
- Education and research
- Public defense
- Healthcare
- Private company or joint venture

2.2 The *winner* data set features

As for the previous paragraph, for each entry of this paragraph, the bold text refers to the name of the features, the italics to the content type of the features, while the text written with an unmodified font describes the content of the feature.

Lot id *integer*
The contract lot identifier. The primary key of the data set. The first member of the composite key of the table.

Business entity id *integer*
The identifier of the business entity that won the contract. The second member of the composite key of the table.

Legal form *integer*
The legal form of the business entity winning the contract.

Uber legal form *integer*
The identifier result in a classification of the business entities into macro-categories. they are the same defined for the contracting authorities

Chapter 3

Methods

3.1 Data preparation pipeline

The input data undergoes the processing:

1. handle missing values
2. join the *lot* set and *winner* set by contract winner
3. features engineering

Finally the data is visualized according to the

3.2 Missing values handling

The majority of the models for machine learning cannot handle missing values. As the right model is not known a priori, the missing entries are to be removed.

The *winner* data set does not present any missing values; table 3.1 shows the percentages of missing values for each feature of the *lot* data set.

To reduce the number of missing values, those values missing a column may be replaced by those belonging to another column but having similar content.

The cases where the *start date* is missing are replaced with *inferred date*.

Albeit the *amount paid* may replace the *amount* — the amount paid by the contracting authority should be comparable when not similar to the amount written in the contract — in practice, it is not a viable substitution. Indeed, the *amount paid* column accounts only for the sum the contracting authority

Feature	N/A percentage
Lot id	0.000000
Pa id	0.000000
object	0.012878
Award procedure id	0.096727
Amount	0.396962
Auction start price	98.109166
Amount paid	30.442027
Start date	31.450964
End date	34.544839
Inferred date	0.033930
Sector id	2.858028
Legal form id	0.000000
Uber legal form	0.001302
CPV	12.254250
CPV true	93.756977

Table 3.1: Percentages of *Not Available* entries for each feature of the *lot* data set.

paid at the time the XML file was sent. The *auction start price* may replace the *amount*; the auction start price is an upper limit to the amount that will be assigned in the procedure. The substitution leads to unwanted outcomes, as it will be shown later; hence, it is discarded.

When possible, the *CPV true* replaces the *CPV* column. Since the *CPV true* reports eight digits, only the first two are kept to maintain uniformity with the *CPV* column.

The *object* column is replaced by the *CPV* column. This column contains plenty of useful information regarding the supply, service, or work object of the procurement. However, the extraction overly complicates the outlier detection analysis, which would require a natural language processing model. The latter should be validated on the *CPV* provided by the original contracts. The training and validation of such a model have been the object of a previous master of science thesis [8]. The results of Amato's work lie in the *CPV* feature. The *object* text first undergoes a classification according to a set of regular expressions. Those rules the text do not cover, yielding a void CPV code, is further classified with a Random Forest model. As the accuracy score on the test is a satisfactory 86 percent the *CPV* column replaces the

object column.

The biggest problem the model is going to face is that a CPV limited to the first two digits of the actual CPV (the so-called CPV *division*) covers an excessively broad spectrum of items. For instance, division 33 represents the *medical equipment, pharmaceuticals, and personal care products* that includes, among the others, ophthalmology equipment and dental mirrors. For the ophthalmology equipment, the prices range from the third order of magnitude to the fifth, while a single dental mirror average about ten euro.

After the substitutions so far mentioned, the columns dropped are *object*, *auction start price*, *amount paid*, *inferred date*, *sector id*, *CPV true*.

The *sector id* is dropped as it is redundant with the *CPV* column. Indeed, the CPV codes whose division is less than 45 are supplies, if their division is equal to 45, they are works, otherwise, they are services.

At last, all the rows with at least one *N/A* are dropped. Any solution to input the missing values is discarded: the yielded table is large enough — it counts 755660 rows — to compensate for the number current number of features.

3.3 Data sets join

To form the tuple (contracting entity, lot value, business entity) the *lot* data set and the *winner* data set are to be joined together. The joining key is the lot identifier. The *lot* data set has 1,382,247 rows; the *winner* data set has 917,316 rows. As it is evident, not all lots have a winner. Moreover, not all contracts in the *winner* table occur in the *lot* data set. The need of information from both the tables reduces the join options to only the inner-join, an alias for the θ-join in terms of relational algebra.

Because a contract lot may be assigned to an ensemble of firms, there may be more than one business entities resulting as the contract winner in the *winner* data set. This entails the fact that the value of the lots assigned to more than one firm is multiplied across the data set by the number of winning firms. To reduce the error, the lot amount is equally divided among all the lot winners, implicitly assuming an uninformative prior. Actually, the lot amount is shared among the ensemble of firms by their leader according to agreements between them, but the information is not known.

3.4 feature engineering

The tabular representation of the public procurement contract consists of the following features:

exp *float*
: the sum of all the lot amounts issued by the contracting entity along the year of the issue date of the contract.

rev *float*
: the sum of all the lot amount won business entity along the year of the issue date of the contract. Then, it is taken the median over the three years.

amount *float*
: the lot value

cpv *integer*
: the Common Procurement Vocabulary division of the contract

proc *integer*
: the type of public contract procedure

duration *integer*
: the number of days result of the difference between the contract termination date and start date

n winner *integer*
: The number of winners of the same contract lot

pa mean *float*
: sample mean computed over the distribution of contracts that are issued by contracting authority of this contract

pa std *float*
: sample standard deviation computed over the distribution of contracts that are issued by contracting authority of this contract

pa skewness *float*
: sample skewness computed over the distribution of contracts that are issued by contracting authority of this contract

pa kurtosis *float*
sample kurtosis computed over the distribution of contracts that are issued by contracting authority of this contract

be mean *float*
sample mean computed over the distribution of contracts won by the same business entity of this contract

be std *float*
sample standard deviation computed over the distribution of contracts won by the same business entity of this contract

be skewness *float*
sample skewness computed over the distribution of contracts won by the same business entity of this contract

be kurtosis *float*
sample kurtosis computed over the distribution of contracts won by the same business entity of this contract

The purpose of this thesis is to determine which contract lots are outliers. Hence, we have to build a model of the public procurement contract. Ideally, any contract requires at least two parties, and the definitions of the obligations of each party to be due to the other. In the public procurement contract case, the parties are the contracting entity and the business entity, while the obligations are the provision of work, service, or supply for the business entity and the exchange of money for the contracting entity. A first version of the model for a public contract is given by the tuple

$$\text{contract} = (\text{contracting entity}, \text{business entity}, \text{provision object}, \text{date}). \tag{3.1}$$

This model is rather abstract. Let's make it more tangible. The contracting entity and the business entity are represented by their ids in the data set. This representation is not satisfactory. The number of business entities is 71,455 while that of contracting entities is 1,752. A standard feature encoding technique like the dummy variables would cast the curse of dimensionality on the features: a sparse data set *where all cows are black*.

A simple solution is possible: the contracting authority may be replaced by its total annual expenditure, and the business entity by its gross revenue. On the one hand, this solution provides an estimate of the dimension of the contracting parties. **The underlying hypothesis is that the higher**

the annual expenditure or the revenue, the bigger the lot value; at least, this dimension should be an upper limit for the value of the contract provision. On the other hand, reducing the peculiarities of a contracting authority or a business entity to a single number may lead to an oversimplification of the entities. While the contracting entity's annual expenditure can be accurately accounted for by the sum of all the lot amounts issued by the same entity, the same is not true for the business entities. The sum of all the lot amounts won by the same business entity is only a portion of the gross revenue of that firm, unless the model assumes that the business entities work only for the public sector; yet the hypothesis hardly holds the test of reality. Nonetheless, with the given data set, the business entity's annual lots sum and the contracting authority's annual expenditure are among the few features one can easily extract. The other way to compensate the oversimplification is to add the *uber legal form* feature.

The second to last dimension of the model 3.1 is the *provision object*. The feature accounts for the object of the procurement; this may be the execution of a work, such as the building of a new facility, a service, such as the hiring of a new staff member, or the supply of a new item, such as medical instruments. The only features given by the data set are the lot value v, the CPV division CPV, and the type of procedure. All the characteristics of the possible provisions mentioned earlier are not given, hence the feature cannot be further complicated at this stage.

Last but not least, the time dimension. Every contract has a date a issue, and may have termination date. The difference between them yields the *duration* of the contract. The duration is a relevant feature in predicting the amount of the contract: **a longer duration should imply a higher amount**; this hypothesis will be verified in the next chapter. The *duration* is expressed in days. More precision is not needed in this context.

It is common knowledge that prices are inherently a time-dependent. For instance, prices increase according to the inflation. Yet, in the context of engineering the features of a contract, the time dimension is hard to employ. Even if the contract specifies the object of the provision, the quantities are usually not contained in the XML files that makes this data set. The usual object for a time series analysis of prices is the unit amount. The contract usually displays the total amount to be bought; seldom the price per unit. Additionally, even the object of the provision may not be clear. As stated above, inferring the object of the provision from the *object* field is out of the thesis' objectives, hence, the object is replaced by the CPV division which is a classification so broad it cannot reasonably be used to determine the object

of the provision. Thus, the *start date* and the *end date* of the contract are discarded from the model.

A natural dimension for the contract is the year. Several contracts have a duration expressed in years as the public procurement contracts are tied to budgets that are defined by the yearly Italian budget laws. for this reason, the inherited way to circumscribe the *contract* model is the year: the annual contracting entity expenditure, the annual business entity specific revenue. In order to provide for each contract a single value to distinguish the contracting entity and one for the business entity, a choice must be made about the aggregating function. A common choice would be the average among the three years the data set spans. Yet, the mean is biased in the presence of outliers. A more robust option is the median.

The underlying assumption behind the choice of the median is that the contract distributions is sufficient to describe the dimension of the contracting parties. **The hypothesis can be partially verified by testing whether the distributions over the three years differs.** The Kullback-Leibler divergence measure how much two distributions differs. These computations are displayed in the next chapter.

To give models more information about the distribution of the contract lot amount of each business entity and contracting authority, we computed the variance, the skewness and the kurtosis such distributions.

The skewness is defined as the third standardized moment of a random variable $E[(X-\mu)^3] = \tilde{\mu}_3$, while the kurtosis as the forth standardized moment $E[(X-\mu)^4] = \tilde{\mu}_4$.

Each contracting party has a minimum of ten awarded contracts over the three years, hence we cannot assume to have the whole distribution as the definitions of skewness and kurtosis require. For this reason, we employ the *sample moment* $m_r = \frac{1}{n}\sum(x_i - \bar{x})^r$ and we derive the sample skewness $g_1 = m_3/m_2^{3/2}$ and sample (excess) kurtosis $g_2 = m_4/m_2^2 - 3$. Unfavorably, such estimators are biased. Finally, to overcome their biases, we deploy the definition of sample skewness and kurtosis computed as ratios of *biased cumulant estimates* [9].

Unbiased skewness:

$$G_1 = \frac{\sqrt{n(n-1)}}{n-2} g_1$$

Unbiased kurtosis:

$$G_2 = \frac{(n-1)[(n+1)g_2 + 6]}{(n-2)(n-3)}$$

3.5 probabilistic distribution visualization

For the sake of a simple visualization of the data set distribution, the the uni-dimensional histogram is the statistical tool of choice. The histogram offers a discrete visualization of the theoretical distribution that generates the data. Indeed, it can be even devised as an outlier detector.

The most important histogram parameter is the bin width. The choice of such parameter determines whether the distribution visualization actually represents the theoretical true variable distribution, either warps its perception.

The question of which bin width determines a histogram that better describes the actual distribution has been posed before. In the literature there are two commonly used rules to choose a good bin width: that developed by Stuges in 1926 [10] and that developed by Freedman and Draconis in 1981 [11].

Sturges' rule that assumes that a good approximation of a continuous random variable is a symmetric binomial distribution $\mathcal{X} \sim \text{Bin}(n, 0.5)$ where n is the number of independent and identically distributed variables sampled from the theoretical distribution.

Sturges' rule works well when the original distribution is normal, thus its inevitable shortcomings when this condition is not met. Draconis and Freedman showed that is a bin width w given by

$$w = 2\frac{IQR(x)}{\sqrt[3]{n}} \tag{3.2}$$

minimizes the mean squared error δ^2 between the original probability density function $f(x)$ and the bars height $H(x)$ for the given continuous random variable X:

$$\delta^2 = E\left[\int (H(x) - f(x))^2\right]. \tag{3.3}$$

All the histograms' bins width are computed according to the Freedman Draconis rule.

Chapter 4

Visualizations

4.1 Business entity specific distribution by specific revenue and number of awarded lots

Several business entities appear only once in the data set. It is not statically sound to infer anything about the business entities which there are not enough data of; especially if their dimension is inferred by the median annual specific revenue, where *specific revenue* is the sum of all the public contracts awarded to that business entity.

As a consequence, the contracts awarded to business entities with a lower prevalence must be cut off: all the contract lots awarded to business entities that in any year won less than n lots are discarded.

Yet, this cut off is too strict for our purposes. The new cut off threshold is the number of contracts awarded to a single business entity over the three years; if such number is less than 10 units, then the business entity and all its awarded contracts are removed from the data set.

The data set rows narrows from 755,660 to 599,177 contract lots.

Figure 4.1 shows how much each threshold for n would impact the distribution of business entities dimension. The x-axis counts the number of contracts awarded to a business entity in a year. The y-axis counts the number of business entities having at least x contracts awarded in a year. The color of the bar accounts for the business entity dimension; each color is assigned to a specific revenue step. Each step is by order of magnitude. The absence of a color is due to low prevalence of that specific revenue order of magnitude in the data set. The graph shows that as the number of contracts

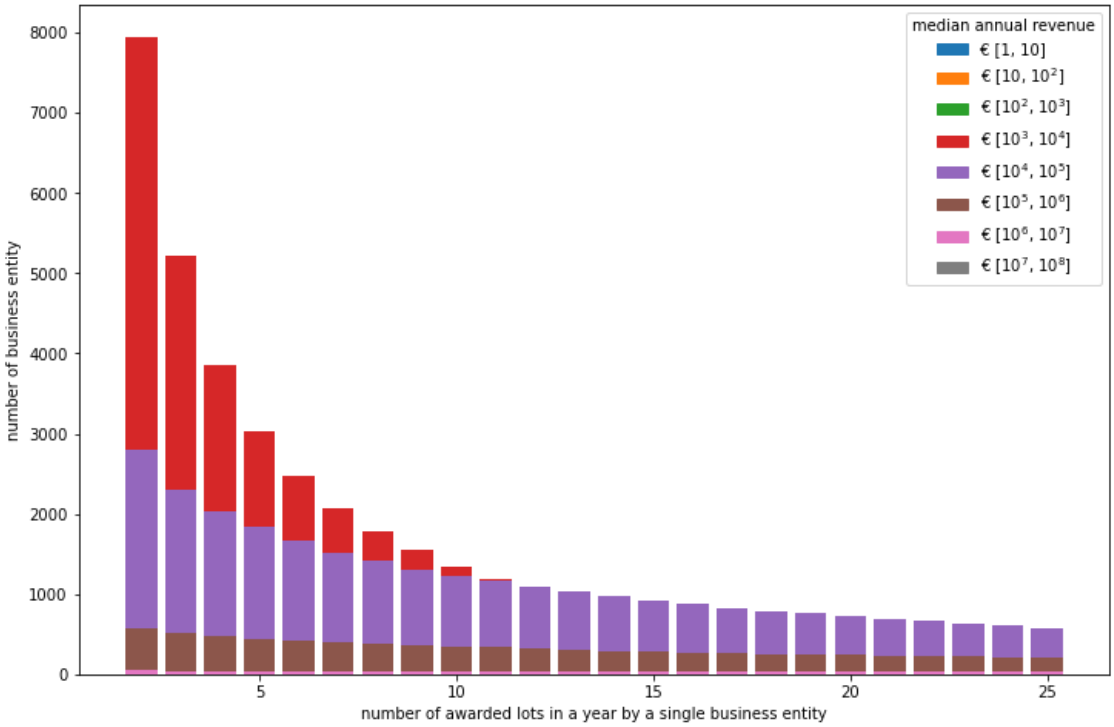


Figure 4.1: The x-axis counts the number of contracts awarded to a business entity in a year. The y-axis counts the number of business entities having at least x contracts awarded in a year. The color of the bar accounts for the business entity dimension; each color is assigned to a specific revenue step. Each step is by order of magnitude. The absence of a color is due to low prevalence of that specific revenue order of magnitude in the data set. The graph shows that as the number of contracts awarded to a business entity grows, the business entity with a lower specific revenue shrinks

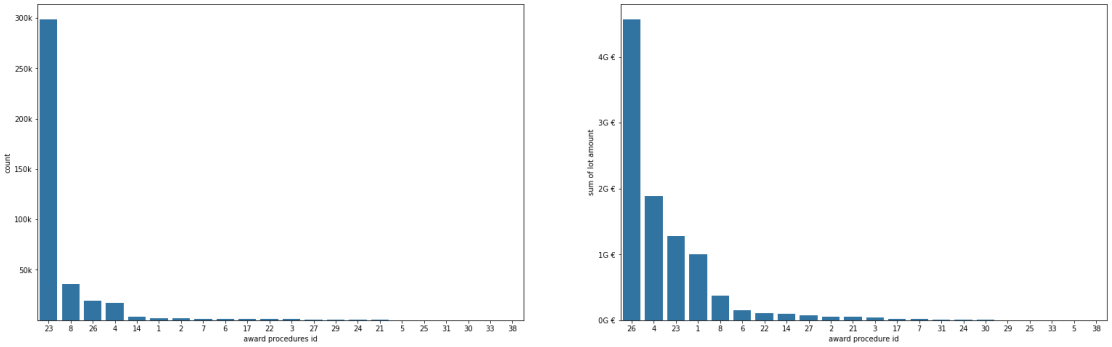
awarded to a business entity grows, the business entity with a lower specific revenue shrinks. With a threshold of $n = 10$ minimum yearly number of awarded contracts per business entity, the business entities with a median annual specific revenue within one thousand and ten thousands euro almost disappear.

4.2 Probabilistic distributions of features

4.2.1 Award procedure feature

The award procedures describe how the contract lots are going to be assigned for the given public call. Figure 4.2 shows that considering only the open procedure, the negotiated procedure, the assignment under framework agreement, and the direct assignment procedure, the number of rows narrows to the 87.11% of the total, while, from a business perspective, the sum amount shrinks to the 89.52% of the total expenditure for the public procurement.

For the purpose of reducing the complexity of the classification, this analysis will encompass only procedures stated above, that are, the open procedure, the negotiated procedure, the assignment under framework agreement, and the direct assignment procedure.



(a) Number of lots per procedure (b) Sum of lot amount per procedure

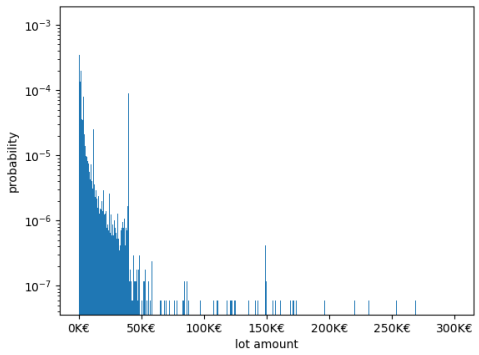
Figure 4.2: Number of lots and sum of the lot amount per award procedures. The first four award procedure codes identify the assignment under framework agreement (26), the piecework assignment (8), the direct assignment (23), the negotiated procedure (4), and the open procedure (1).

A few remarks on the graphs. Histograms in 4.3 suggest that the lot amount variable is log-normal, given that each procedure its own mean and variance. Figure 4.3a presents the peculiarity of two spikes at 40k euros and 100k euros. That are the Italian law thresholds for the public procurement contracts. The contracts lots that exceed the law threshold of 100 thousands euro should be illegal, yet as we consider them lawfully exceptions.

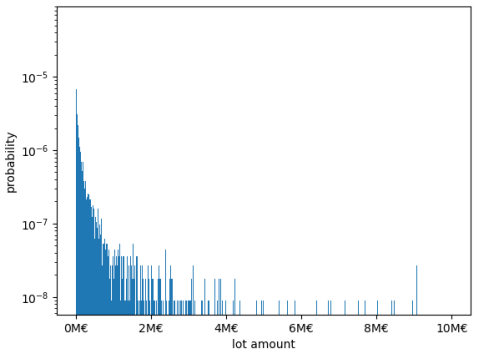
On the one hand, the business entity median annual specific revenue histograms 4.4 and the contracting authority median share the fact that there

are few entities with a large share of the market. On the other hand, the business entities histograms look more log-normal, while the contracting entities one look uniform, with few entities of the right tail. Figure 4.4a shows few bars because the bar width is so thin they do not appear in the figure. The contract lot whose won by a business entity with a median annual specific revenue over 300K euro are 1179.

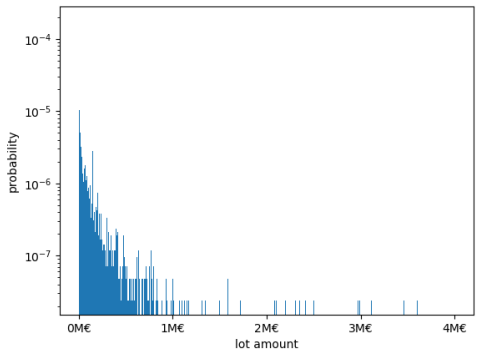
The duration histograms 4.6 look like the original distribution is log-normal. Each procedure type presents its peculiarities: the direct assignment has spikes around the year and 2 years thresholds; for in the negotiated procedures these spikes are strongly marked at every year; the open procedure and assignment under framework agreement have a higher kurtosis.



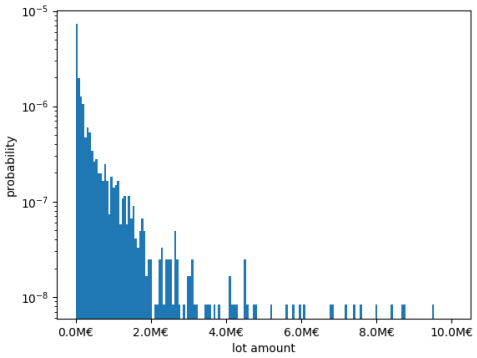
(a) Direct assignment



(b) Framework agreement assignment



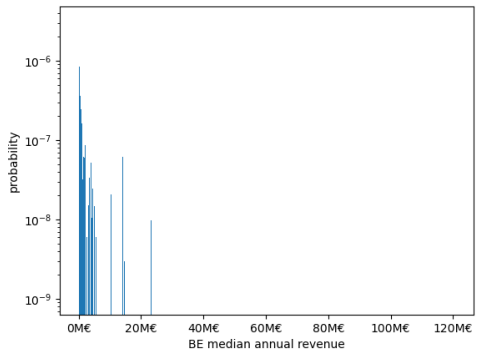
(c) Negotiated procedure



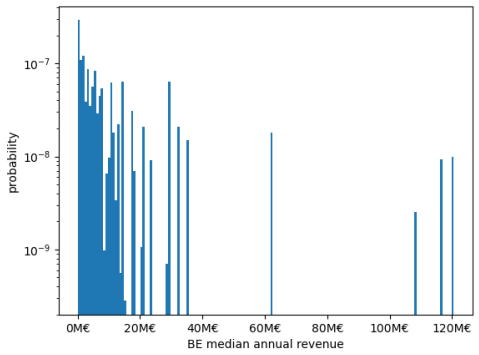
(d) Open procedure

Figure 4.3: Histograms of the lot amount variable by procedure type

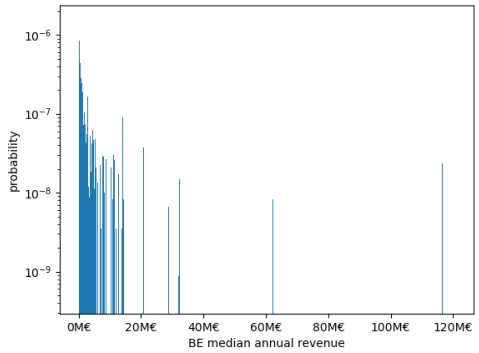
Figure 4.7 shows the award procedure clusters. In order to have a better view, the assignment under framework agreement and the direct assignment,



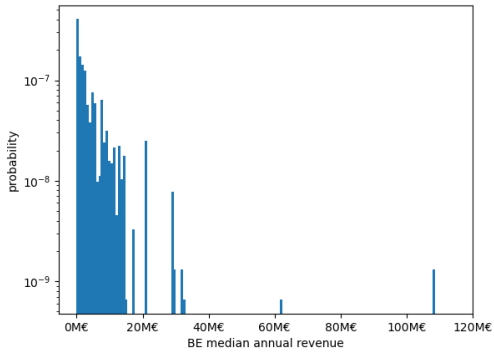
(a) Direct assignment



(b) Framework agreement assignment



(c) Negotiated procedure



(d) Open procedure

Figure 4.4: Histograms of the business entity median annual specific revenue variable by procedure type

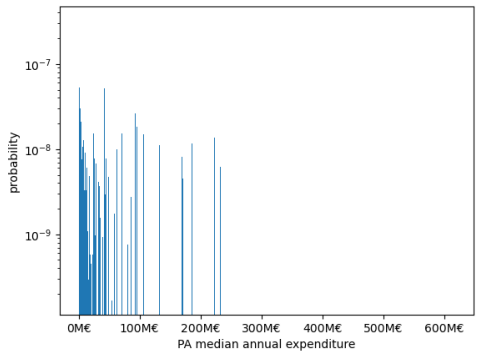
and the negotiated procedure are down-sampled to match that of the open procedure.

Observing the entire scatter plot, the the cluster is mostly globular with some regions with different densities. This peculiarity suggest the use of statistical models that take advantage of this structure, such as the Gaussian mixture models.

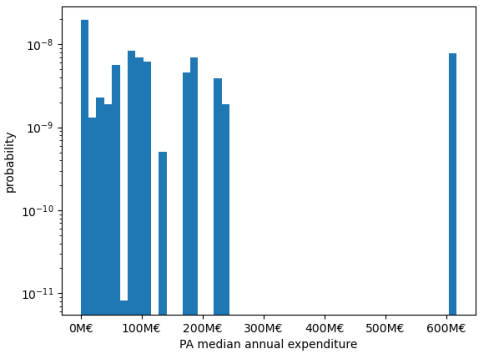
4.2.2 Common procurement vocabulary feature

The common procurement vocabulary division codes classify the object of the procurement. They are required by the European Regulation No. 2013/2008.

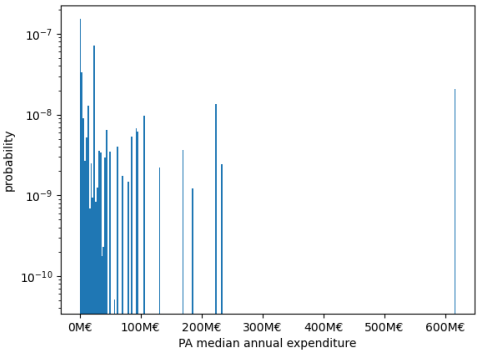
Figure 4.8 shows that narrowing the CPV divisions to medical equipments



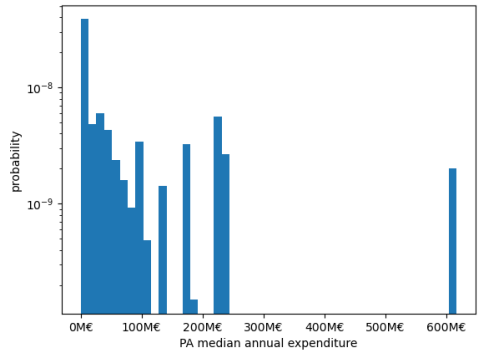
(a) Direct assignment



(b) Framework agreement assignment



(c) Negotiated procedure



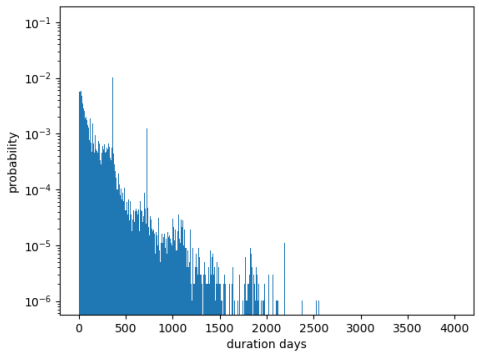
(d) Open procedure

Figure 4.5: Histograms of the contracting entity median annual expenditure variable by procedure type

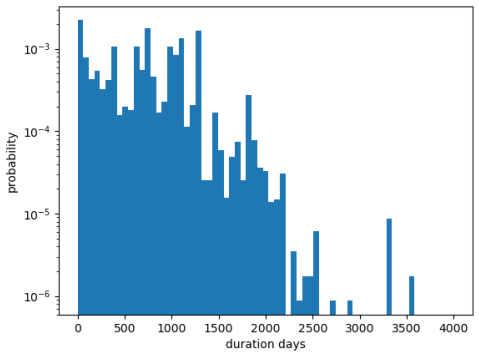
(33), construction works (45), public utilities (65), and health related services (85), the number of rows shrinks to 20.28% while the sum of the lot amount accounts for the 66.62% of the total.

To simplify the inspection, the CPV divisions considered are the 33, 45, 65, and 85.

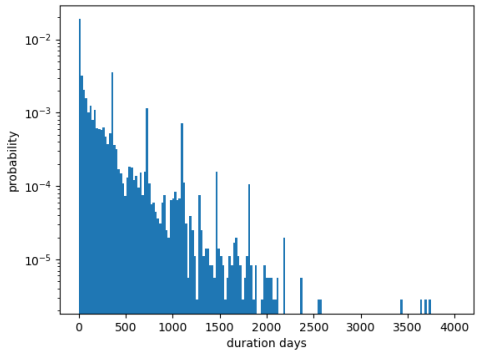
Most of the procurement contracts are issued by the regional local health firms such as Azienda ULSS 3 Serenissima, Azienda ULSS 2 Marca Trevigiana, by Azienda Zero, a public entity for purchasing health realated goods for the entire Veneto Region, and by the local government.



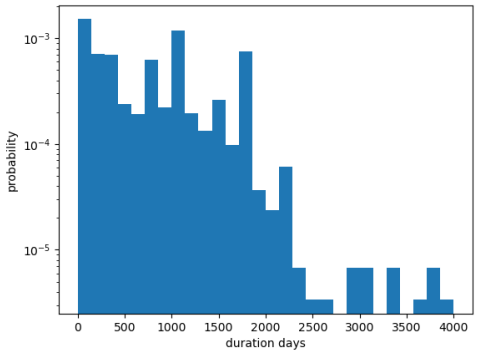
(a) Direct assignment



(b) Framework agreement assignment



(c) Negotiated procedure



(d) Open procedure

Figure 4.6: Histograms of the contract duration variable by procedure type

	CPV division			
Award procedure	33	45	65	85
open proc.	0.0066	0.0039	0.0001	0.0014
restricted proc.	0.0363	0.0183	0.001	0.0022
direct ass.	0.6234	0.0749	0.0261	0.025
framework ass.	0.1733	0.0012	0.0056	0.0008

Table 4.1: CPV and award procedure joint distribution

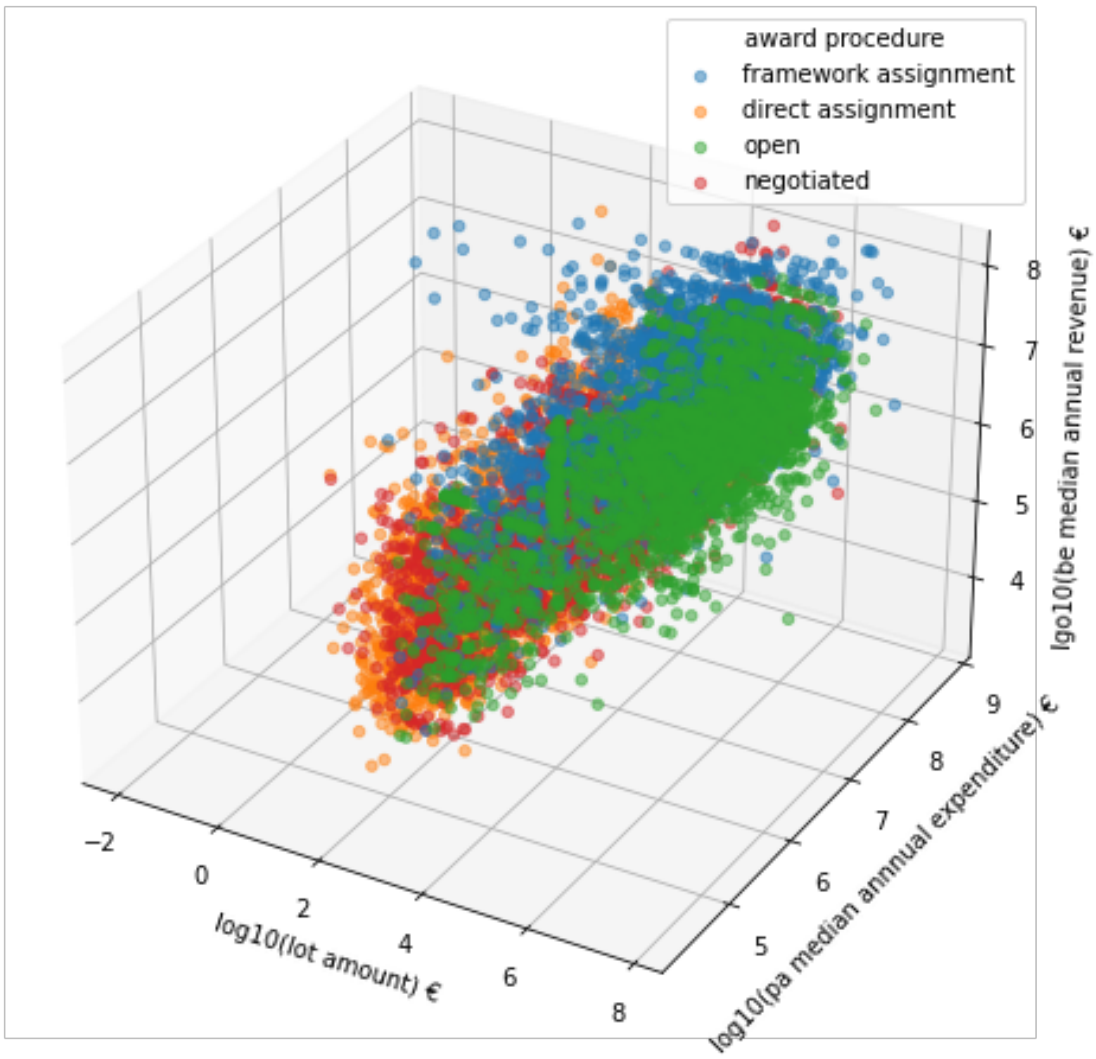
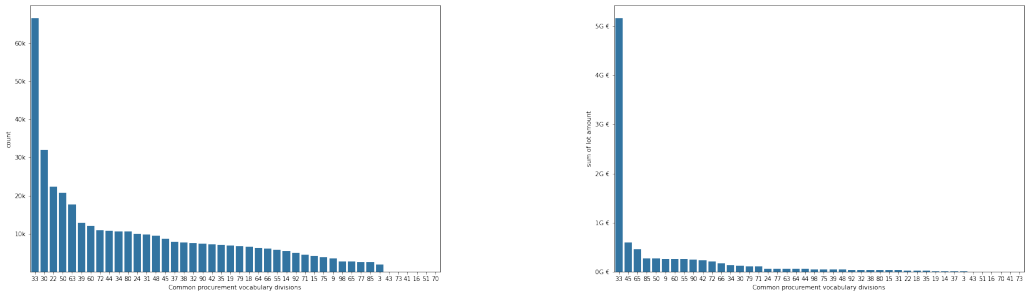


Figure 4.7: the award procedure clusters. The axis scale is log10. Due to their cardinalities, the assignment under framework agreement, the direct assignment, and the negotiated procedure are down-sampled to match that of the open procedure.

4.3 Analysis of the time variable

4.3.1 Problems of defining the object of the time series

When speculating about the public contract features, it comes naturally to theorize the contract as time dependent. The *amount* feature is the price of a work, service, or goods; prices are time-dependent; then, the contract's



(a) Number of lots per CPV division (b) Sum of lot amount per CPV division

Figure 4.8: Number of lots and sum of the lot amount per CPV division. The first CPV divisions by sum of lot amounts identify *medical equipments, pharmaceuticals and personal care products* (33), *Construction work* (45), *Public utilities* (65), *Health and social work services* (85).

feature *amount* is time-dependent.

The main problem with the approach is that with the given contract information are not enough to accurately determine the exact object of the procurement and the quantity bought. The only information available is the Common Procurement Vocabulary division, which is too broad of a definition to hold the assumption of *temporal continuity* [12], which states that data patters should not change abruptly.

4.3.2 Contracts clustered by year

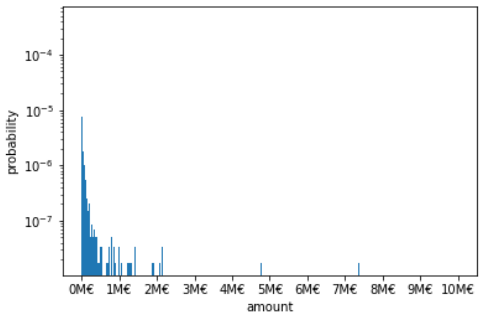
The model assumes that the contracting entities and the business entities can be represented by their median annual expenditure and their median annual specific revenue. Even though this hypothesis cannot be tested for the single entity, a test can be set up to measure whether the whole distribution changes over the years.

A rather empirical way to have a first impression of the phenomenon is to plot a point for each contract lot. Figure 4.14 shows the results. It is clear that the cluster overlap.

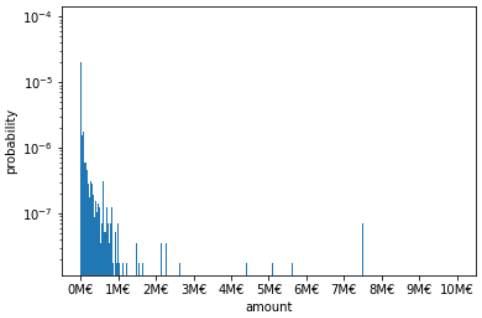
A common method to measure the separation between the cluster consists of computing the distance between the centroids of the clusters [13].

$$separation(C_i, C_j) = proximity(c_i, c_j) \tag{4.1}$$

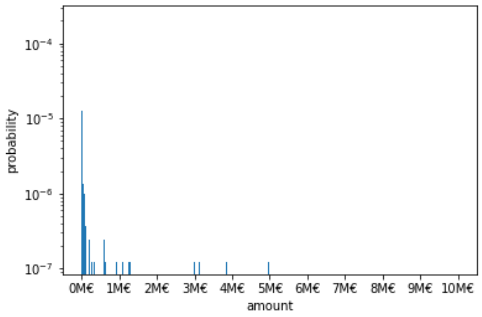
where a measure of proximity can be any distance metric. In our analysis,



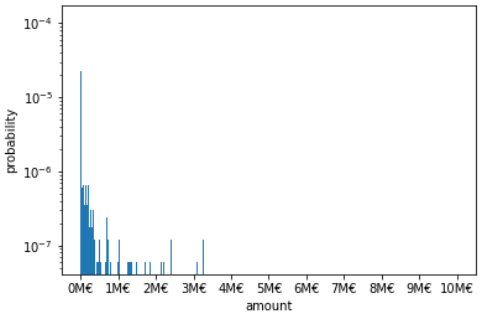
(a) Medical equipments (33)



(b) Construction work (45)



(c) Public utilities (65)



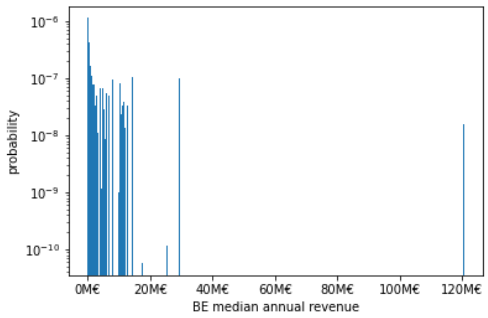
(d) Health and social work services (85)

Figure 4.9: Histograms of the *amount* feature grouped by CPV

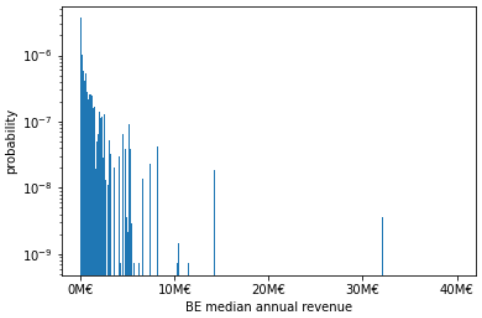
this measure is the euclidean distance. This definition of separation applies especially if the cluster are *prototype-based*, as Tan et al. call them, that is, the cluster can be represented by their centroids and they are globular.

To compute the distances between each couple of centroids, each dimension has been projected into the log10 space, centered removing their median, and normalized with the interquartile range.

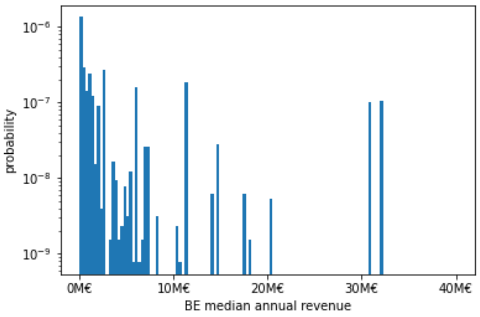
Table 4.2 shows the euclidean distances between each couple of centroids in the log-normalized space. The magnitude of the distances is not high, but it is not negligible. Figure 4.15a display that the difference in the amount value between 2016 and 2018 amounts to more the 20 thousand euro. The increment may be due to and increment in the total volume of goods exchanged; indeed, figures 4.15b, 4.15d support the hypothesis of increased volume as a higher amount is correlated with a bigger expenditure and a bigger specific revenue. Another explanation for the increment is the increased government public expenditure: during the from 2016 to 2018 there was a 3 percent



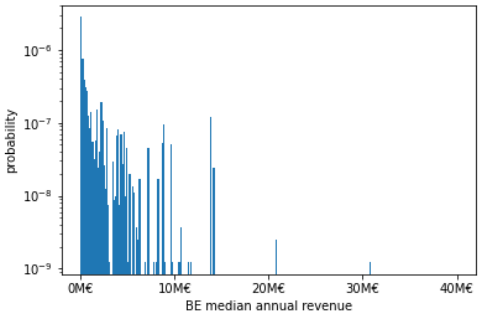
(a) Medical equipments (33)



(b) Construction work (45)



(c) Public utilities (65)



(d) Health and social work services (85)

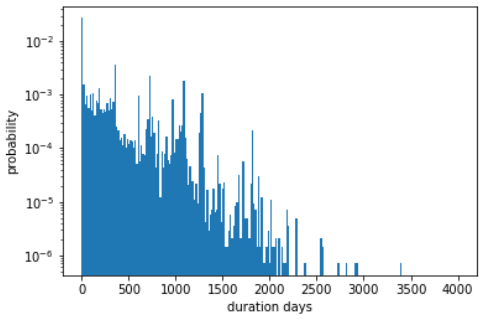
Figure 4.10: Histograms of the *business entity median annual specific revenue* feature grouped by CPV

	2016	2017	2018
2016	0	0.0041	0.0311
2017		0	0.0310
2018			0

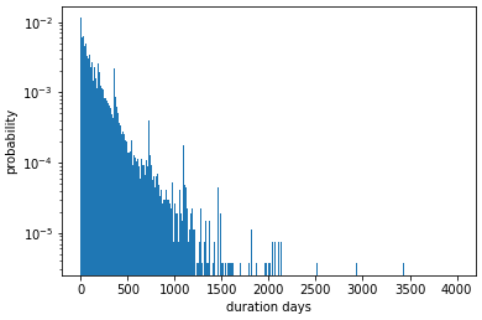
Table 4.2: Euclidean distances between each couple of centroids of the data clustered by year

increment.

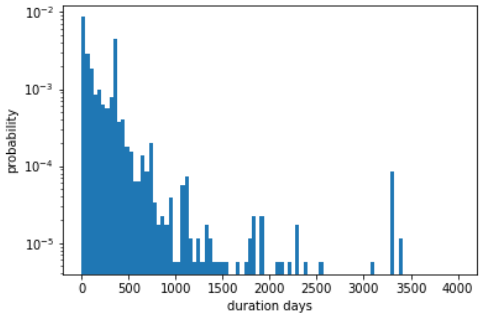
To sum up, is the data set actually independent from time? No, but this thesis assumes it is. The complexity that denying the time-independence assumption entails is out of the scope of this thesis. A simple and not exhaustive way to keep into consideration the time dimension is to add the year dimension as a set of dummy variables.



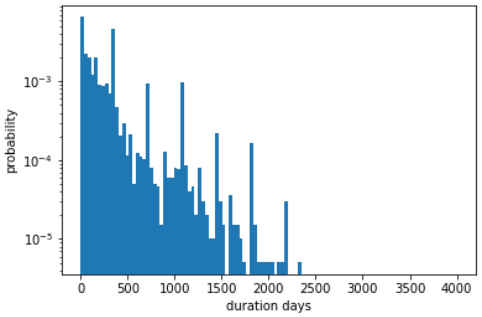
(a) Medical equipments (33)



(b) Construction work (45)



(c) Public utilities (65)



(d) Health and social work services (85)

Figure 4.11: Histograms of the *duration* feature grouped by CPV

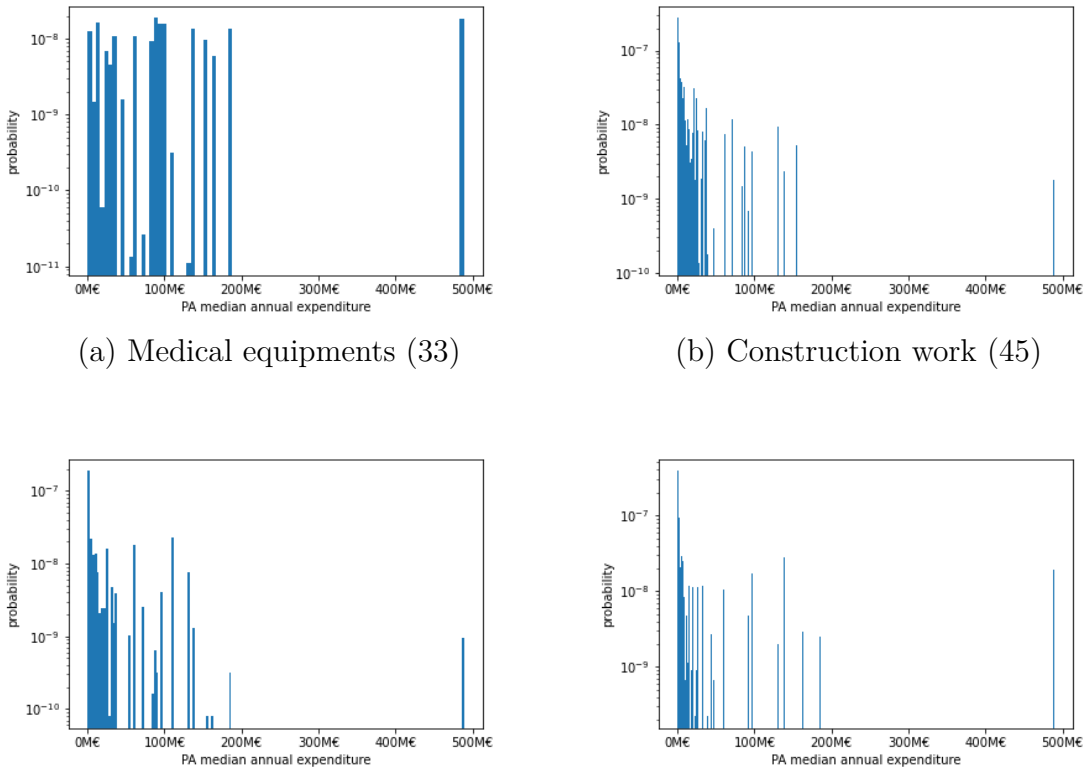
year	2016	2017	2018
PA expenditure	832,265	846,821	857,245

Table 4.3: Italian *Public Adminitration* (PA) expenditure. Values expressed in *millions* of euro. Source: ISTAT [1]

4.4 Pearson Correlation of the numerical features analysis

Figure 4.16 shows that Pearson correlation of the model continuous features. The highest correlations are:

- business entity's *amount* standard deviation and the business entity median annual specific revenue (0.56).



(a) Medical equipments (33)

(b) Construction work (45)

(c) Public utilities (65)

(d) Health and social work services (85)

Figure 4.12: Histograms of the *duration* feature grouped by CPV

- public administration *amount* standard deviation and public administration median annual expenditure (0.8);
- public administration *amount* skewness and public administration median annual expenditure (0.61);
- public administration *amount* kurtosis and public administration median annual expenditure (0.61);
- public administration *amount* skewness and public administration *amount* standard deviation (0.56);
- public administration *amount* kurtosis and public administration *amount* standard deviation (0.57);
- public administration *amount* kurtosis and public administration *amount* skewness (0.97);

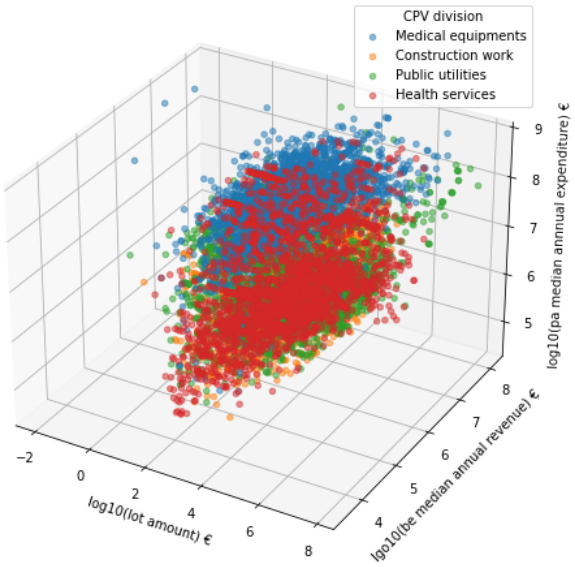


Figure 4.13: CPV clusters. The axis scale is log10. To make clusters more visible, the medical equipment, construction work, and public utilities are down-sampled to match that of the health services.

The correlation coefficient suggest that

- business entities that have higher annual revenue tend to have a higher degree of variation in contract prices;
- the same applies for contracting authorities: higher expenditures correlates with a higher variance of contract prices;
- the *amount* distribution of contracting entities is skewer and wider for those having a substantial annual expenditure;
- the *amount* distribution of contracting entities that have a higher skewness highly correlate with a high kurtosis distribution.

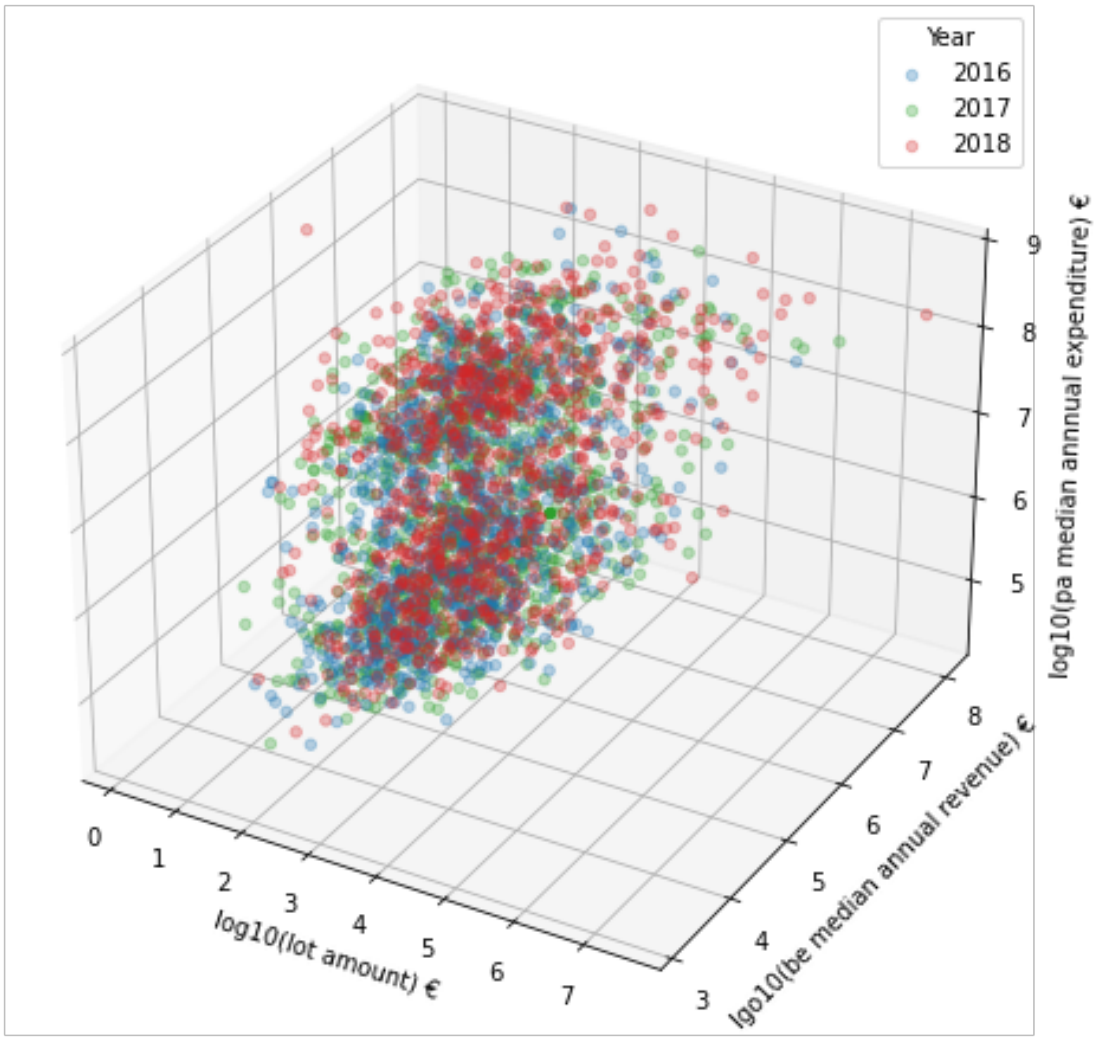


Figure 4.14: Scatter plot of the contract lots grouped by year.

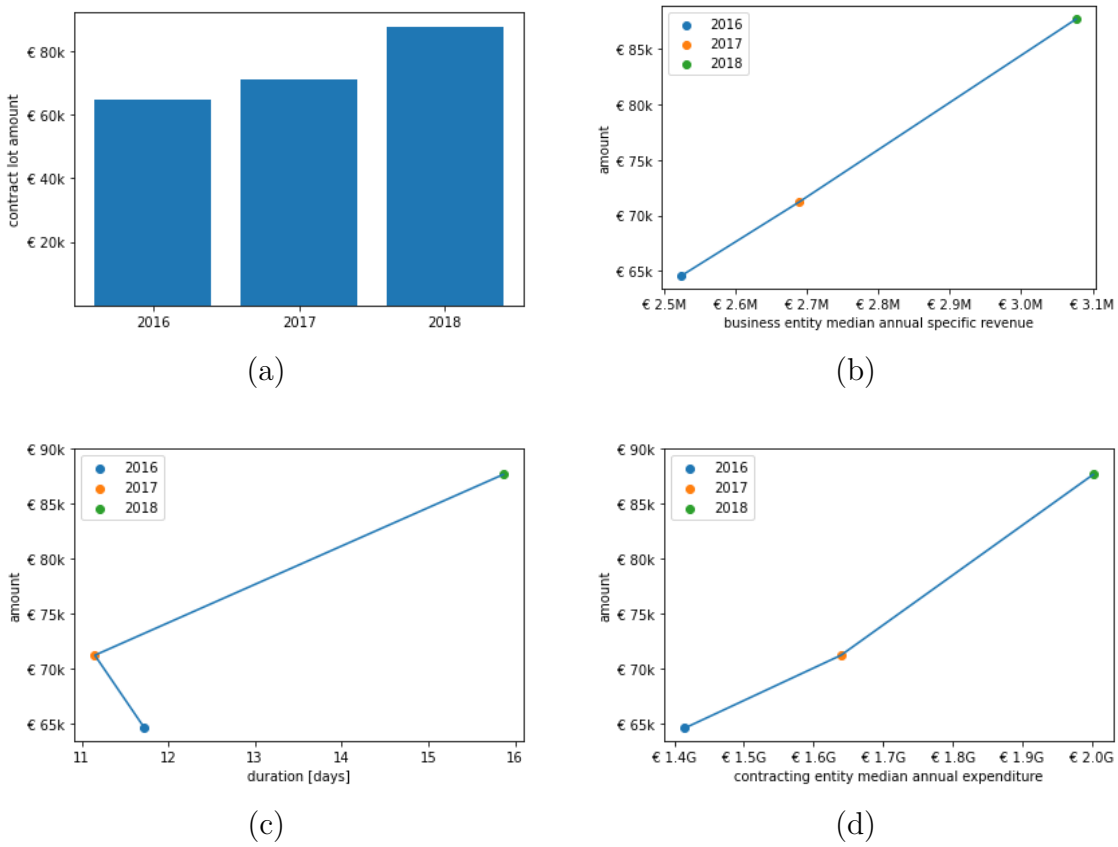


Figure 4.15: Centroids' amount feature plotted against the other features.

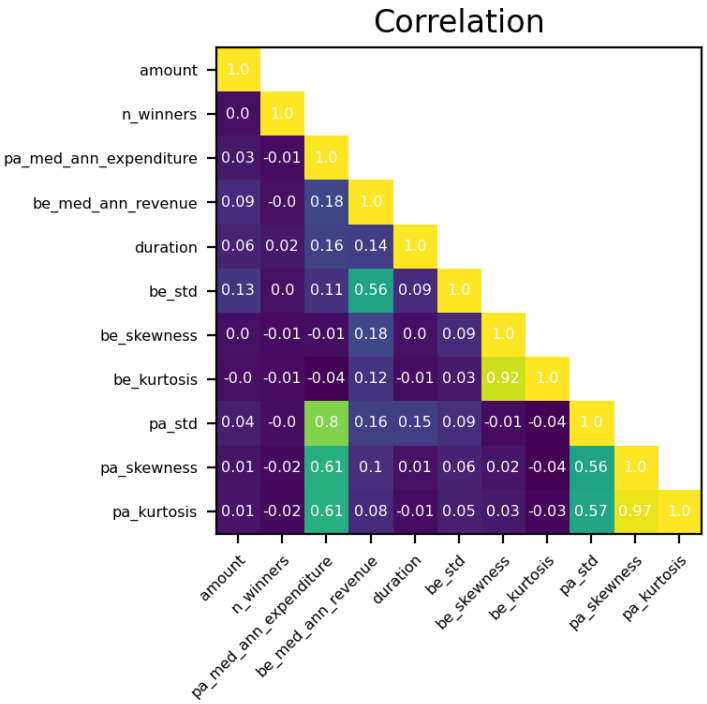


Figure 4.16: Pearson Correlation of continuous features

Part II

Outlier detection

Chapter 5

Background

5.1 Problem definition

The problem of anomaly detection is going to be defined in this section. We will pursue a probabilistic perspective. The structure of the definition follows that of [14].

Let $\mathcal{X} \subseteq \mathbb{R}^D$ be the data space yielded by an application; in our case, $\mathcal{X}$ coincides with the space of all possible contract lots. In order to have an anomaly detection, we need to assume that there exists a concept of normal behavior, by which we can tell if a contract lot is normal or deviates from the normal; let $\mathbb{P}^+$ be the distribution ground-truth of this normal behavior, and $p^+(x)$ is the respective probability density function. Then, we define the set of anomalies as

$$\mathcal{A} = \{\boldsymbol{x} \in \mathcal{X} \mid p^+(\boldsymbol{x}) \leq \tau\}, \quad \tau \geq 0, \tag{5.1}$$

where τ is some probability threshold.

5.1.1 Types of anomalies

There are three types of anomalies: *point anomalies*, *contextual anomalies*, and *group anomalies*. Point anomalies are individual observation that $\boldsymbol{x} \in \mathcal{A}$. Contextual anomalies are observation that are not anomalies per se, as their values are not inherently deviant, but they fall into the category when considering a conditional distribution $\mathbb{P}^+_{X|T}$ where T is the contextual variable. The usual instance for contextual anomalies is time series, where the textual variable T is time. The group anomalies are a set of data points bounded together by some relations. A group of anomalies may occur when

analyzing the signal from a sensor, such as a seismic sensor, where the group of anomalies are yielded by an earthquake signal.

Ruff *et al.* classify the instances of the anomaly set $\mathcal{A}$ as *anomalies* when abnormal observations are drawn from a distribution different from $\mathbb{P}^+$; the assumption is that the random process that output the anomaly differs from what it is expected to be normal. Let $\mathbb{P}^-$ be the ground-truth anomaly distribution. Then, the anomaly set follows $\mathbb{P}^-$. They classify instances as *outliers* when they are drawn from the same probability distribution $\mathbb{P}^+$, they come from the same process, bu their probability is low; They classify instances as *novelties* when they are drawn from a new region of an non-stationary $\mathbb{P}^+$. These differences are rather abstract and only some authors introduce these distinctions. Yet, they help up distinguish between the two types of anomalies may we encounter during this analysis. A contract lot is an *anomaly* when there is a typo in its XML file as the process deviates from the normal behavior of the process of writing the XML file; a contract lot is an outlier when all the entries of its XML file are correct, but the combination is infrequent. While the purpose of this thesis is to detect these anomalies, we are going to determine the outliers, as detecting this type of anomalies is complex in an unsupervised setting. From now on, we are going to use the terms *anomaly* as a super set of *outlier*.

5.1.2 Concentration assumption

A key assumption for outlier detection is that the normal data region can be bounded. This assumption is known as the *concentration assumption*:

$$\mathcal{X} \setminus \mathcal{A} = \{\boldsymbol{x} \in \mathcal{X} \mid p^+(\boldsymbol{x}) > \tau\}, \quad \tau \geq 0, \tag{5.2}$$

is non-empty and small. The assumption does not state that the normal distribution $\mathcal{P}^+$ must bu bounded, but that high-density subset is bounded.

Most of the times, we are unable to model the process yielding the distribution $\mathcal{P}^+$ as the process is too complex. One should be able to model the process of writing the XML files describing the contract lots. A viable option consists in estimating $\mathbb{P}^+$ with the data-generating distribution $\mathbb{P}$, with $p(\boldsymbol{x})$ its probability density function. The observation in the data set $\boldsymbol{x}_1, ..., \boldsymbol{x}_n \in \mathcal{X}$ are assumed to be drawn from independent and identically distributed random variables following $\mathbb{P}$.

We can now revise the objective of outlier detection as density level set estimation. Let C be the density level set of $\mathbb{P}$, given $\alpha \in [0, 1]$,

$$C_\alpha = \{\boldsymbol{x} \ \in \mathcal{X} \mid p(x) > \tau_\alpha\}. \tag{5.3}$$

Finally, the outlier detector $c_\alpha : \mathcal{X} \mapsto \{\pm 1\}$ can be defined as

$$c_\alpha(\boldsymbol{x} = \begin{cases} +1 & \text{if} \boldsymbol{x} \in C_\alpha, \\ -1 & \text{if} \boldsymbol{x} \notin C_\alpha, \end{cases} \tag{5.4}$$

5.1.3 Unsupervised setting

The setting under which we are going to carry out the outlier detection is unsupervised, as we have only unlabeled data while training the model:

$$\boldsymbol{x}_1, ..., \boldsymbol{x}_n \in \mathcal{X}. \tag{5.5}$$

Each sample is drawn from i.i.d. random variables following $\mathbb{P}$. For simplicity, we assume that the data-generating distribution coincides with the the normal law $\mathbb{P} \equiv \mathbb{P}^+$.

Noise. Another peculiarity of the data set is that its samples inherit randomness, $\boldsymbol{x}+\epsilon \sim \mathbb{P}$, while $\boldsymbol{x} \sim \mathbb{P}^+$. The source of noise is due to the errors the civil servant writing the contract lot XML file may do. Under the perspective of noise, the purpose of cleaning the data set from erroneous contract lots translates into determining which contract lots have been affected by noise, that is, isolating the noise from the original sample.

The last component describing the data set settings is *contamination.* The contamination accounts for undetected anomalies, which are inevitable in the unsupervised setting. In our context, the source of randomness is due to the manipulation of the raw data. Some data may be lost or corrupted during the data transmission between the servers; some data may be wrongly inferred by the Synapta's algorithm for inferring lost information.

As these error affect the whole data space $\mathcal{X}$, the contamination affects both the normal and the anomaly distribution. Thus, denoting $\eta \in (0, 1)$ the contamination rate, the data-generating distribution becomes

$$\mathbb{P} \equiv (1 - \eta)\mathbb{P}^+ + \eta\mathbb{P}^- \tag{5.6}$$

To sum up, a realistic assumption for the unsupervised setting is that the samples drawn from $\mathbb{P}$ have the form of $\boldsymbol{x} + \epsilon$ where $bmx \sim \mathbb{P} \equiv (1 - \eta)\mathbb{P}^+ + \eta\mathbb{P}^-$.

Chapter 6

Models

6.1 Baseline model

A first approach in building the baseline model consists in combing out the contracts that blatantly violate the law. This procedure is easy to implement as the Public Contract code is clear about the thresholds of contract amount for each type award procedures. These thresholds are reported in tables 1.1, 1.2. This method failed at determining the outliers from the inliers. Moreover, this procedure denies the assumption that contracts are lawfully, that is, contracts are unlawful by chance; for instance, the file reporting an apparently unlawful contract is corrupted, or was wrongly written by the civil servant.

The second approach to obtain a first set of outliers and inliers relied on the knowledge of Synapta's domain experts. The rules they came up with are:

1. if a contract lot has a value that exceeds the business entity median annual specific revenue and the contracting authority median annual expenditure, while the duration is at most one year, then it is an outlier, given that each business entity has at least ten contracts per year;

2. if a contract lot is a direct assignment and its duration is longer than ten years, then it is an outlier;

3. if a contract lot amount is 25 times bigger than the median annual specific revenue of the business entity that won the lot, then the contract lot is an outlier.

6.2 Probabilistic tails and Box Cox transformation

The Chebishev inequality offers a relation between the standard deviation of a univariate distribution and and the probability of encountering observations.

By the Chebishev Inequality [12], for a random variable X and a constant k,

$$\mathbb{P}(|X - \mu| > k\sigma) \leq \frac{1}{k^2}. \tag{6.1}$$

The weakness of the aforementioned expression is that we need $k \geq 10$ to achieve probabilities compatible with the the definition of outliers, as $\mathbb{P}(|X - \mu| > 10\sigma) \leq 0.01$. Its strength lies in the fact that the it applies no matter the distribution of the random variable.

The Chebishev inequality is a generalization of the so-called 68-95-99.7 rule that applies for the normally distributed random variables. For a comparison, with $k = 1, 2, 3$ the inequality states that any observation $x \in X$ lies outside the range of k times the standard deviation with a probability lesser or equal than 0, 25 and 11.1 percent. In the literature the Chebishev inequality is a considered a weak threshold to define outliers as the constant k must be high to achieve smaller probabilities of outliers.

The application of the Chebishev inequality follows from the definition of outlier within the Synapta context. The definition states that

> a contract lot is an outlier if, within the regional and time setting, either its amount lies in any tail of the business entity which is awarded to, either its amount lies in any tail end of the contracting entity that issued the contract, either its duration lies in the tail of its award procedure.

To apply the Chebishev inequality, we need to compute mean and standard deviation of the amount distribution grouped by business entity, mean and standard deviation of amount distributions grouped by contracting authority, lastly, mean and standard deviation of the duration distribution grouped by award procedure.

The Chebishev inequality assumes that the population mean and standard deviation are known. By the law of large numbers, the sample mean and the sample standard deviations statistics effectively replace their population counterparts when the number of sampling instances is big enough. It is usually assumed that this number is higher than thirty.

The number of contract lots grouped by business entity sometimes is less than thirty, but always more than ten, due to the data preparation step where the infrequent business entities have been removed. On the other hand, the number of contracts won by contracting authority and the number of contracts by award procedure are always big enough.

Another tool to ensure the normality of the distribution is the BoxCox transformation [15].

$$y(\lambda) = \begin{cases} \frac{y^\lambda - 1}{\lambda} & \text{if } \lambda \neq 0; \\ \log y & \text{if } \lambda = 0. \end{cases} \tag{6.2}$$

The equation applies to only non-negative values, such is the case into consideration.

Box and Cox obtain the preferred parameter λ as follows. They assume that the response vector should be Normal;

$$\mathbf{y}(\lambda) \sim N(\boldsymbol{X}\boldsymbol{\beta}, \sigma^2 \mathbf{I}_n) \tag{6.3}$$

where $\boldsymbol{y}$ is the response vector, $\boldsymbol{X}$ is the data matrix, and the model parameters are $(\lambda, \boldsymbol{\beta}, \sigma^2)$ The probability density function is

$$f(\boldsymbol{y}(\lambda)) = \frac{\exp -\frac{1}{2\sigma^2}(\boldsymbol{y}(\lambda) - \boldsymbol{X}\boldsymbol{\beta})'(\boldsymbol{y}(\lambda) - \boldsymbol{X}\boldsymbol{\beta}))}{2\pi\sigma^{2\frac{n}{n}}}. \tag{6.4}$$

Then, Box and Cox maximizes the likelihood to get the optimal λ.

$$L(\lambda, \boldsymbol{\beta}\sigma^2 | \boldsymbol{y}, \boldsymbol{X}) = f(\boldsymbol{y}) J(\lambda, \boldsymbol{y}) \tag{6.5}$$

where $J(\lambda, \boldsymbol{y})$ is Jacobian from $\boldsymbol{y}$ to $\boldsymbol{y}(\lambda)$. A complete explanation of the MLE computation for the λ parameter is beyond the scope of this work.

A note: not all distribution can be power-transformed to Normal [16].

6.3 Density estimation with kernel smoothing

The purpose of a kernel density estimation is to estimate the probability density function of a population from a given sample. Once the pdf is known, we can set probability thresholds to discriminate the outliers from the inliers.

The following is a summary of [17] chapters 2 and 3.

6.3.1 Univariate case

Histograms are a discrete estimation of a population probability density function. Given a sample of independent and identically distributed univariate random variables $X_i, X_2, ..., X_n$, a histogram is given by

$$\hat{f}(x;h) = \frac{1}{nh}\sum_{i=1}^{n}\frac{1}{2}\mathbb{1}_{\{-1<\frac{x-X_i}{h}<1\}} \tag{6.6}$$

where the subscript of the indicator function $\mathbb{1}$ states the subset where the variable x takes value one, zero otherwise; the h term is the width of the bars. The term $\frac{1}{2}\mathbb{1}_{\{-1<\frac{x-X_i}{h}<1\}}$ can be interpreted as a uniform distribution $\mathcal{U}(-1,1)$ centered at X_i and scaled by h. With the uniform distribution, we are giving equal value to each x. One can arbitrary set any other distribution in place of the uniform. If we choose a smooth function, such as the Gaussian $\frac{1}{\sqrt{2\pi h^2}}\exp\{-\frac{1}{2}(\frac{x-X_i}{h})^2\}$, the histogram yields a smooth estimate of the pdf. The general term for such window function is *kernel*.

A kernel density estimator for a univariate distribution is given by

$$\hat{f}(x;h) = \frac{1}{nh}\sum_{i=1}^{n}K(\frac{x-X_i}{h}). \tag{6.7}$$

Bandwidth selection The bandwidth parameter h is of paramount importance for a good estimate of the population pdf. The usual quantity for comparing and evaluating estimators is the mean squared error.

$$\text{MSE}[\hat{f}(x;h)] = \mathbb{E}\left[(\hat{f}(x;h) - f(x;h))^2 dx\right] \tag{6.8}$$

Yet, this quantity is defined for a given point x of the estimator, while we would like to have a quantity capturing the whole distribution. A possible solution consists in integrating the mean squared error, the so-called *Integrated Squared Error* (ISE)

$$\text{ISE}[\hat{f}(x;h)] \coloneqq \int\left[\hat{f}(x;h) - f(x;h)\right]^2 dx. \tag{6.9}$$

The ISE depends on the sample $X_1, ..., X_n$ as $\hat{f}$ is estimated on the sample. Instead, we want our error criterion to estimate the population integrated square error; for this reason, we consider the *Mean Integrated Squared Error*

(MISE):

$$\text{MISE}[\hat{f}(x;h)] \coloneqq \mathbb{E}\left[\int (\hat{f}(x;h) - f(x;h))^2 dx\right] \tag{6.10}$$

$$= \int \mathbb{E}\left[(\hat{f}(x;h) - f(x;h))^2 dx\right] dx \tag{6.11}$$

$$= \int \text{MSE}[\hat{f}(x;h)] dx \tag{6.12}$$

The assumption that holds up these equation are that the population density f is square integrable (i.e. the integration of f^2 is finite); for this reason, the Fubini's theorem holds and we can change the order of integration.

To find the optimal h, we minimize the MISE:

$$h^* = \arg\min_{h>0} \text{MISE}[\hat{f}(x;h)] \tag{6.13}$$

In an unsupervised setting, the probability density function of the population f is not given. We can approximate MISE by means of cross-validation.

$$\text{MISE}[\hat{f}(x;h)] = \mathbb{E}\left[\int (\hat{f}(x;h) - f(x;h))^2 dx\right] \tag{6.14}$$

$$= \mathbb{E}\left[\int \hat{f}(x;h)^2 dx\right] - 2\mathbb{E}\left[\int \hat{f}(x;h) f(x;h) dx\right] \tag{6.15}$$

$$+ \mathbb{E}\left[\int f(x;h)^2 dx\right]$$

The last addend of 6.15 is constant as f is square integrable by hypothesis. Then, to minimize the MISE$[\hat{f}(x;h)]$ is equal to minimize

$$\mathbb{E}\left[\int \hat{f}(x;h)^2 dx\right] - 2\mathbb{E}\left[\int \hat{f}(x;h) f(x;h) dx\right]. \tag{6.16}$$

It can be shown that the an unbiased estimator of MISE is given by *Lease Squares Cross validation* (LSCV) estimator (also known as *Unbiased Cross validation* (UCV) estimator)

$$\text{LSCV}(h) \coloneqq \int \hat{f}(x;h)^2 dx - 2n^{-1} \sum_{i=1}^{n} \hat{f}_{-i}(X_i;h), \tag{6.17}$$

where $\hat{f}_{-i}(X_i;h)$ is the *leave-one-out* kernel density estimated on the sample removed of X_i

$$\hat{f}_{-i}(X_i;h) = \frac{1}{n-1} \sum_{\substack{j=1 \\ j\neq i}} K\left(\frac{x - X_j}{h}\right). \tag{6.18}$$

6.3.2 Multivariate case

In a multivariate setting, we have that our sample and random vectors $\boldsymbol{X}_1, \boldsymbol{X}_2, ..., \boldsymbol{X}_n$, while $\boldsymbol{x} \in \mathbb{R}^d, d > 1$. The kde estimator is given by

$$\hat{f}(\boldsymbol{x}, \boldsymbol{H}) \coloneqq \frac{1}{n|\boldsymbol{H}|^{1/2}} \sum_{i=1}^{n} K\left(\boldsymbol{H}^{-1/2}(\boldsymbol{x} - \boldsymbol{X}_i)\right) \tag{6.19}$$

where the bandwidth matrix $\boldsymbol{H}$ is a $d \times d$ positive definite matrix, otherwise $|\boldsymbol{H}|^{1/2}$ and $\boldsymbol{H}^{-1/2}$ would not be well defined. The bandwidth matrix $\boldsymbol{H}$ is a real symmetric matrix with positive eigenvalues. If $\boldsymbol{H}$ is full, then we are implying correlation between the d variables. If $\boldsymbol{H}$ is diagonal, then we are assuming that the variables are independent. In our setting, we cannot assume that the variables are independent as the contracting authority annual expenditure and business entity specific revenue are computed as sum of the lot amount variable.

Bandwidth selection The considerations for computing the bandwdith matrix are an extension of the univariate case: we want to minimize the mean integrated squared error MISE$[\boldsymbol{x}; \boldsymbol{H}]$; as the MISE is unfeasible, we estimated it with its asymptotic version, the AMISE.

The actual formula for AMISE is rather complicated and cannot be computed in practice as they depend on the unknown f. For these reasons, we only provide the formula for $\boldsymbol{H}$ which assumes that $f = \phi_{\boldsymbol{\Sigma}}(\boldsymbol{x} - \boldsymbol{\mu})$. It is called the *normal scale bandwidth selector*

$$\boldsymbol{H}_{NS} = \left(\frac{4}{d+2}\right)^{\frac{2}{d+4}} \boldsymbol{\Sigma} n^{-2/(d+4)}. \tag{6.20}$$

When variance matrix $\boldsymbol{\Sigma}$ is replaced by the sample covariance matrix $\boldsymbol{S}$, we obtain the estimated $\hat{\boldsymbol{H}}_{\boldsymbol{NS}}$.

The problem is that we are back into the lands of parametric statistics.

There is an additional selector that tries not to assume the distribution f, as the normal scale selector does, until a condition is met. The idea is to estimate f a combination of its derivative up the ℓ order, where ℓ is a positive integer. As the process it iterative, this kind of selectors are called ℓ-stage plug-ins, in opposition to the *zero-stage* plug-ins, such as the normal scale bandwidth selector, where the f distribution is assumed at the beginning. This method was proposed by Sheather and Jones [18].

As for the univariate case, another way of estimating the bandwidth is by means of cross-validation.

$$\text{LSCV}(\boldsymbol{H}) = \int \hat{f}(\boldsymbol{x}; \boldsymbol{H})^2 d\boldsymbol{x} - \frac{2}{n} \sum_{i=1}^{n} \hat{f}_{-i}(\boldsymbol{X}_i; \boldsymbol{H}) \tag{6.21}$$

This estimator is unbiased. Then, the bandwidth matrix is given by

$$\hat{\boldsymbol{H}}_{LSCV} = \arg\min_{\boldsymbol{H}} \text{LSCV}(\boldsymbol{H}) \tag{6.22}$$

6.4 Gaussian Mixture model

The Gaussian mixture model aims at determining the probabilities of each point in the data set $\mathcal{D}$ by estimating data-generating distribution $\mathbb{P}$.

The model assumes that the data-generating distribution $\mathbb{P}$ is the combination of k distributions. Let us denote $\mathcal{G}_r$ the r-th component of the mixture distribution, and $f^r(x)$ its probability density function.

The probability that a data point x_j is generated by the mixture model $\mathcal{M}$ is given by

$$f(x_j|\mathcal{M}) = \sum_{i=1}^{k} \alpha_i \cdot f^i(x) \tag{6.23}$$

where $\alpha_i \in [0, 1]$, $i \in \{1, ..., k\}$ is the prior probability that captures the idea that a fraction of the data is generated by the component i.

The parameters of each distribution $\mathcal{G}_r$ and the probabilities α_r are estimated from the data via log-likelihood maximization.

Assuming that each sample of the data set $\mathcal{D}$ is drawn from i.i.d. random variables, the data set's pdf is

$$f(\mathcal{D}|\mathcal{M}) = \prod_{j=1}^{N} f(x_j|\mathcal{M}). \tag{6.24}$$

where N is data set number of samples. Then, the log-likelihood is

$$\mathcal{L}(\mathcal{D}|\mathcal{M}) = \log \left[\prod_{j=1}^{N} f(x_j|\mathcal{M}) \right] = \sum_{j=1}^{N} \log \left[\sum_{i=1}^{k} \alpha_i \cdot f^i(x) \right] \tag{6.25}$$

6.4.1 Parameter Optimization

The Gaussian mixture model parameters are

number of components whose combination is the actual mixture model,

distribution parameters of each Gaussian component.

A Gaussian distribution is entirely determined by its mean and covariance matrix. The covariance matrix describes the the directions and lengths of the axes of its density contours. There are four types of covariance matrix

full each Gaussian component has a full covariance matrix $\Sigma \in \mathbb{R}^{\times M \times M}$; the model covariance matrix is a three dimensional matrix where its depth equal the number of components k.

tied each Guassian component has the same full covariance matrix $\Sigma_{GM} \in \mathbb{R}^{M \times M}$, thus they are *tied* together;

diagonal the contour axis are oriented along he coordinate axes, yet, each component eccentricities can vary along each axes; the covariance matrix of each component are diagonal, but each diagonal entry can differ from one another; $\Sigma_i = diag(\sigma_j), j \in \{1, ..., M\}, i \in \{1, ..., k\}$ while Σ_{GM} is three dimensional (k, M, M).

spherical each component have one standard deviation. The covariance matrix Σ is diagonal but all the diagonal entries are equal; $\Sigma_{GM} = diag(\sigma_i) i \in \{1, ..., M\}$.

The standard method to optimize the Gaussian mixture model parameter relies on the *Akaike information criterion* [19] and the *Bayesian information criterion* [20]. We are going to use the Bayesian information criterion.

Konishi and Kitagawa [21] provide a simple explanation of the Bayesian information criterion. We are not going to see the whole derivation of the information criterion; we will revise only their main ideas.

Let $M_i \in \{1, ..., r\}$ be a set of candidate models. Each model is entirely defined by a parametric distribution $f_i(x|\boldsymbol{\theta_i}$, $(\boldsymbol{\theta} \in \Theta_i \subset \mathbb{R}^{k_i})$, and the prior distribution $\pi_i(\boldsymbol{\theta}_i$. Given n observations $\boldsymbol{x}_n = \{x_1, ..., x_n\}$, the marginal distribution of $\boldsymbol{x}_n$ for the model M_i is

$$p_i(\boldsymbol{x}_n) = \int f_i(\boldsymbol{x}_i|\boldsymbol{\theta}_i)\pi_i(\boldsymbol{\theta}_i)d\boldsymbol{\theta}_i \tag{6.26}$$

In a Bayesian perspective, $p_i(\boldsymbol{x}_i)$ is the likelihood relative to the candidate model M_i, hence *marginal likelihood*.

By the Bayes' theorem, assuming that the prior probability of the i-th model is $\mathbb{P}(M_i)$, then its posterior probability is

$$\mathbb{P}(M_i|\boldsymbol{x}_n) = \frac{p_i(\boldsymbol{x}_i))\mathbb{P}(M_i)}{\sum_{j=1}^{r} p_j(\boldsymbol{x}_n)\mathbb{P}(M_i)}, \quad i \in \{1, ..., r\} \tag{6.27}$$

It is reasonable to assume that models' priors $\mathbb{P}(M_i)$, $i \in \{1, ..., r\}$ are uninformative, that is, they all have the same probability. As the denominator probability is shared across all the models, the model having the highest posterior probability coincides with that having the highest marginal likelihood. Now, the marginal likelihood is not easily obtainable. Schwarz demonstrated that applying the Laplace approximation the marginal likelihood becomes

$$p(\boldsymbol{x}_n) \approx \exp\{\ell(\hat{\boldsymbol{\theta}})\}\pi(\hat{\boldsymbol{\theta}})(2\pi)^{p/2}n^{-2p/2}|J(\hat{\boldsymbol{\theta}}|^{1/2} \tag{6.28}$$

where

$\ell(\hat{\boldsymbol{\theta}})$ is the log-likelihood function $\ell(\boldsymbol{\theta}) = \log f(\boldsymbol{x}_n|\boldsymbol{\theta})$

$\hat{\boldsymbol{\theta}}$ is the maximum likehood estimator for the parameter $\boldsymbol{\theta}$

p is the number of features

n is the number of samples

$J(\hat{\boldsymbol{\theta}}) = -\frac{1}{n}\frac{\partial^2\ell(\boldsymbol{\theta})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}^T}$ evaluated at $\boldsymbol{\theta} = \hat{\boldsymbol{\theta}}$

Taking the logarithm, and multiplying the approximated marginal likelihood by -2, we get

$$-2\log p(\boldsymbol{x}_n) \approx -2\ell(\hat{\boldsymbol{\theta}}) + p\log n + \log|J(\hat{\boldsymbol{\theta}})| - p\log(2\pi) - 2\log\pi(\hat{\boldsymbol{\theta}}) \tag{6.29}$$

If we consider only the terms with order greater then $O(1)$, we get the Bayesian information criterion (BIC):

$$BIC = -2\log f(\boldsymbol{x}_n|\hat{\boldsymbol{\theta}}) + p\log n \tag{6.30}$$

Minimizing the Bayesian information criterion yields to a trade-off between the number of samples n and their number of features p and the maximization of the model likelihood. The purpose is to balance the eventual over fitness of a model with the penalty term of the number of samples and features.

6.5 One-class Support Vector Machine

The one-class support vector machine is a modified version of the commonly used binary support vector machine classifiers. The algorithm was proposed by Schölkopf *et al.* in 1999 [22]. The main model assumption is that inliers lie in small neighborhood of the origin of the feature space F — assuming that data is standardized — onto which the model's kernel projects them.

Consider the training data $\boldsymbol{x_i}, ..., \boldsymbol{x_n} \in \mathcal{X}$, where $\mathcal{X} \subseteq \mathbb{R}^D$ is the data space. Define the feature map $\Phi : \mathcal{X} \mapsto F$ such that the dot product of two images $\Phi(\boldsymbol{x}), \Phi(\boldsymbol{y})$, $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}$ can be computed by evaluating a kernel

$$k(\boldsymbol{x}, \boldsymbol{y}) = (\Phi(\boldsymbol{x}) \cdot \Phi(\boldsymbol{y})). \tag{6.31}$$

To separate the outliers from the center of gravity of the, the authors developed the quadratic problem

$$\min_{\boldsymbol{w} \in F, \boldsymbol{\xi} \in \mathbb{R}^D, \rho \in \mathbb{R}} \quad \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{\nu n}\sum_{i=1}^{n}\xi_i - \rho \tag{6.32}$$

$$\text{subject to} \quad (\boldsymbol{w} \cdot \Phi(\boldsymbol{x}_i)) \geq \rho - \xi_i,\ \xi_i \geq 0 \tag{6.33}$$

where

$\boldsymbol{w}$ is the vector representing whose entries are separating-hyperplane coefficients to be estimated;

ρ is the hyperplane bias term;

ν is the *contamination* parameter;

$\boldsymbol{\xi}$ is slack variable vector.

Besides the choice of the kernel and its eventual parameters, the contamination parameter ν defines the model. It represents the fraction of outliers that are supposed to populated the data set.

The optimization problem is then solved by a quadratic programming routine.

Chapter 7

Related works

7.1 Literature review

The study of public procurement is not new. Economics scientists, management engineers, legal scholars have their interests in understanding the bidding process, how different legislation lead to different outcomes in more transparent and cost-effective public procurement, whether optimal bidding strategy exists and how to pursue them.

On the other hand, the actual data of the bidding process were not available until recently. The need of more efficient bureaucratic procedures now that the information infrastructure is mature suggest governments to digitalize processes formerly analog.

Among the outcomes of digitalization lies the construction of data set. For public procurement these databases are open and easily accessible for the purpose of public transparency.

The European Union imposes their State members to provide lists of their public calls for tenders if the object of public procurement if higher than a given threshold. Such information is contained in the *Tenders Electronic Daily* (TED). The public availability and the richness of the data set spawned scientific researches such as [23]. In the paper the authors shows a positive correlations between a higher quality of public procurement rules and a higher degree of competitiveness between bidders. The quality of the public procurement rules that given by the European Public Accountability Mechanism (EuroPAM), a European institution that judges European country legislation assigning scores according to the transparency it imposes in political financing, financial disclosure, conflict of interest, freedom of information, public procurement. The quality scores given by EuroPAM is integrated by the

Benchmarking Public procurement by the World Bank.

In [24], using the TED data set, the authors statistically describe the use of Voluntary Ex Ante Transparency notice (VEAT) in place of the more common Contract Award Notice (CAN). The authors find no conclusive evidence towards the use of VEATs over CANs to guarantee a competitive and transparent public procurement.

Coming to the price modeling of tenders there have been a few attempts. Among them, a game theoretic approach has been proposed in [25] to model the bidding process and determine the price. The authors analyze define a payoff function and describe the game equilibria and the case for two bidders, then extend the analysis to more than two bidder. In the latter case, the proposed model captures the intuitive ideas that the more bidders, the lower the tenders, and that the higher the cost of performing the contract, the higher the tender price.

Another study [26] forecast the tender prices by means of machine learning techniques. The authors' data set consists of the auction bids for a highway in Vermont. They claim their models, an ordinary least squares regressor, and their regularized versions Ridge and Lasso, outperforms their baseline model, that is a random forest regressor.

Finally, Garcia and Portugues (2022) [17] try to forecast the awarded tenders in Spanish public Procurement Announcements. The Spanish legislation forces contracting authorities to publish XML files containing information regarding calls and their tenders.

These files are the input of their analysis. Scraping the XML files, the authors build a tabular data set. Each row represents a tender to a given contract. As such, the table columns describe the firm that made the tender, the object of the tender and type of contract, and the firm that won the procurement contract ans its award price.

Among the tenders lies the tender that won the public contract. The authors refer to the former as *tender price*, the latter as *award price*. The correlation between them is 97 percent and it is the highest among the predictors.

Nonetheless, the authors argue that the *tender price* predictors is not enough to determine which bid will actually win the public contract. They train a random forest regressor having the *award price* as the response variable.

The authors shows that the prices forecast by their regressor are actually better than the raw *tender prices*. Indeed, the absolute percentage error (APE) between the forecast prices and the *award prices* grouped by CPV

division are less disperse than the APE between the raw *tender prices* and the *award prices* grouped by CPV.

Albeit the different objective of the research, the study conducted by Garcia and Portugues is interesting as their input data set is similar to that of this thesis. The main difference between ours and their data set is that ours does not have tender prices, but only award prices.

Moreover, they show that the correlations between their data set columns are within $[-0.38, 0.38]$ with the only exceptions of *tender price* and award price (0.97) and type of contract and sub-type of contract (0.74). Even the correlations in the 40 percent neighborhood are scarce as well. The only correlations are the procedure code and the award price (-0.36), the procedure code and the tender price (-0.38) and the CPV and contract duration (.34). All the others are in $[-.20, 20]$.

Their correlation analysis suggest which features may be useful to extract from our data set. In their correlations analysis, those are

1. tender price (.97)
2. procedure code (-0.36)
3. type of contract (0.23)
4. subtype of contract (0.21)
5. number of received offers (0.19)

Such features do not rank among the highest by Random Forest importance (i.e. for each feature, the average variance reduction gained by the node splits of each tree. Refer to the Scikit-learn documentation for a full explanation). The trained regressor places in top five predictors

1. tender price (.87)
2. received offers (0.035)
3. duration (0.017)
4. date (0.013)
5. identifier of the firm that made the tender (0.012)

The CPV, CPV division and procedure code importance score are 0.009, 0.005, and 0.003. Their regressor feature importance counters the intuition that the CPV and the procedure type defines the award procedure.

7.2 The novelty of the research

The data set from the study of Garcia and Portugues [17] is similar to that we are using for this thesis; yet, the atomic element of their data set tables is the tender of a given public procurement contract. In opposition, the atomic element of our data set is the contract itself, that is characterized by one or more winners and it is issued by a contracting authority. Moreover, we do have access to the raw XML files, while we have access to the information extracted by the feature extraction pipeline of Spazio Dati, the owner of the data set. Due to the absence of such files, we lack the information about the bid prices and the firms or other public authority that made them. We know only the winner of that tender. The main difference between our objective and that of Garcia and Portugues is that they developed a regressor to forecast which tender price is going to be win the public call, while ours is to determine which contract are anomalous. As far as we can tell, this is the first time a anomaly predictor is developed for public procurement contracts.

Chapter 8

Development and implementation

8.1 Evaluation method

The evaluation method arises from the necessity of measuring the performance of the outlier detecting models in an unsupervised setting.

It is standard practice [14] to have rather small labelled subset which which the metrics of choice can be calculated. Unfortunately, we lack a labelled set of contracts. A first attempt to overcome the absence of a labelled subset consists in manually checking a set of two hundreds contracts. The number of outlier contracts found in the subset is one. The encounter of one outlier out of two hundred contracts suggests that the percentage of outliers of 0.5 percent, which is higher than the subject matter experts of Synapta expected. Moreover, having a single outlier cannot represent the population of outliers, entailing an imperfect representation of the actual models' performances. Last but not least, checking contracts one by one by hand is time consuming and requires a deep understanding of civil law.

A partial solution to label the data set is to automate the process. The automation is carried out concatenating the output of the baseline model and the probabilistic tails model. Figure 8.1 shows the process. The baseline model is chosen because the it is the implementation of a set a set of rules results of the experience of *Subject Matter Experts* (SME). The probabilistic tails is chosen because it follows the probabilistic definition of outlier, the discovery of which is one the objective of this thesis. As the outliers found by the two models differs in nature, we assign them different names: *SME*

outliers and *probabilistic* outliers. The *SME* outliers and the *probabilistic* are concatenated so that each contract has a *SME* outlier label and a *probabilistic* outlier label. The now labeled data set is forwarded to the outlier detecting models and their performance metrics can be easily computed on the test set.

The outliers found by the automated labeling process 8.2 are not *all* the outliers of the data set. As a consequence, the metrics computed, such as the True Positive Rate and the even more relevant False Positive Rate are to be taken with a grain of salt. Indeed, the outlier detectors used by the automated labeling come with the limitation that they can only determine outlier with respect to one dimension (the *amount* and *duration* dimensions in our models); if a contract is a very rare as a combination of features, it will be never spotted by such models, thus, the need of a manual check of the outlier discovered by the other outlier detectors.

8.2 Data preparation and enrichment

The data set input to the outlier detecting models is that resulting of the analysis of the exploratory part. Let us call this data set *contract.csv*, as it is called in the implementation code.

Contracts.csv is a tabular set. Each row describes a contract lot and its features are the contract amount a, the contract duration d, the identifier of the contracting authority issuing the contract *id pa*, the identifier of the business entity selling the contract *id be*, the identifier of the contract award procedure *proc*, the number of winners of that contract lot and the date the contract has been issued.

In addition to standard routines such as the removing of missing values, what greatly characterizes the input data is that the number of contract lots per contracting entity (business entity and contracting authority) is greater than ten.

The input to each outlier detecting model undergoes the same preprocessing steps illustrated in figure 8.3.

The input data set is read from its file *contract.csv* and imported as *pandas DataFrame* [27]. All the manipulation it will undergo are performed by the methods the pandas library [28] offers.

First step. Once the data is read and loaded in memory as a *pandas DataFrame*, the the contracts are grouped by contracting party. For the purpose of clearness, let us explain the preprocessing with and example.

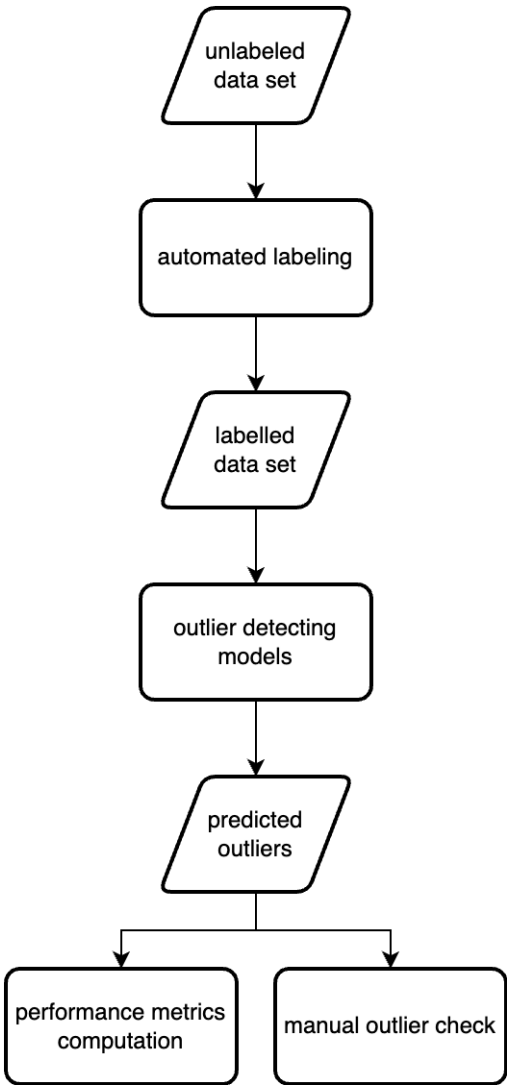


Figure 8.1: Models evaluation process

Suppose that the contracting entity are the business entity be_0. Then, the set of contracts sold by be_0 are $C_{be_0} = \{c_0, c_1, ..., c_n\}$.

Second step, transformation routine. For each set of contracts such as C_{be_0}, take the amount feature and the duration feature, $A_{be_0}\{= a_0, a_1, ..., a_n\}$ and $D_{be_0} = \{d_0, d_1, ..., d_n\}$ normalize each set of feature according to the function

$$norm(X) = \frac{x - min(X)}{max(X) - min(X)}.$$

The image of the normalization is to $[0, 1]$. This prepare each set of features to be transformed by the Box Cox function as implemented by the `boxcox`

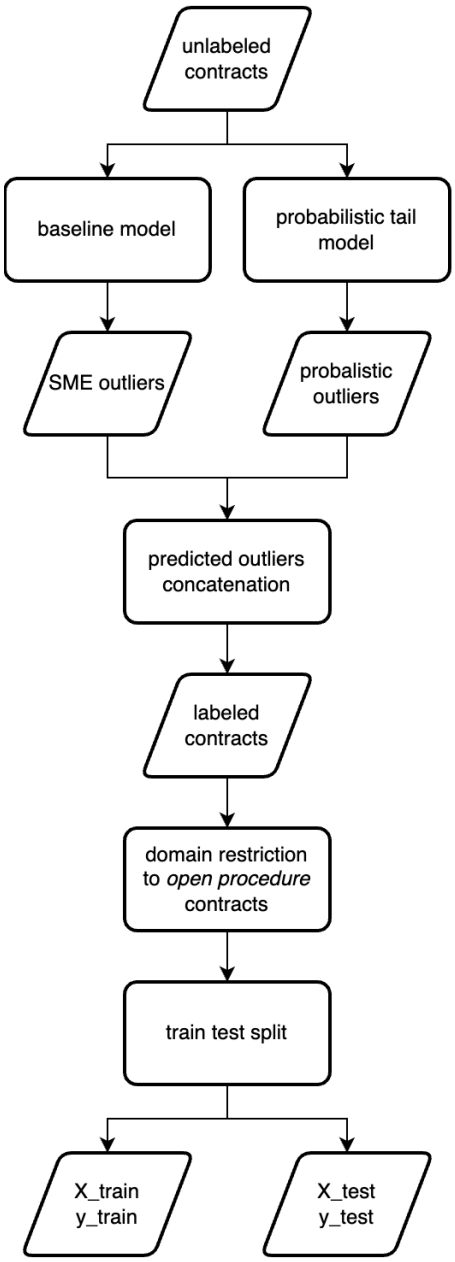


Figure 8.2: The input data are labeled by the baseline and probabilistic tails models. Next, the predicted SME and probablistic outliers are horizontally concatenated. The now-labelled data are restricted to only contracts awarded with an open procedure and split into train and test set, with their respective label sets.

form the *scipy stats* package [29]

$$y(\lambda) = \begin{cases} \frac{y^\lambda - 1}{\lambda} & \text{if } \lambda \neq 0; \\ \log y & \text{if } \lambda = 0. \end{cases} \tag{8.1}$$

where the optimization of the parameter λ is obtained by minimizing the maximizing the likelihood the the transformed data are normally distributed. The optimal λ is achieved by the function `optimize_scalar` of the *scipy optimize* package

The Box Cox transformation projects image set is $\mathbb{R}$. To prevent extreme value, we normalize the data again so that the feature values are restricted to $[0, 1]$.

Third step. Once the data is normalized, each contract is enriched with the mean, variance and skewness computed on each set of features, such as A_{be_0} and D_{be_0}.

Fourth step. The collection of (statistical) moments computed in the previous step is appended to each contract c_i. As a consequence, a new set of features is added to the input data set *contracts.csv* In addition to the older features, the new ones are

- amount mean, amount variance, amount skewness
- duration mean, duration variance, duration skewness

8.3 Baseline model

The baseline model implements the rules defined by the domain expert to determine which contracts are outliers and which are not. The rules are the following:

1. if a contract lot has a value that exceeds the business entity median annual specific revenue and the contracting authority median annual expenditure, while the duration is at most one year, then it is an outlier, given that each business entity has at least ten contracts per year;
2. if a contract lot is a direct assignment and its duration is longer than ten years, then it is an outlier;
3. if a contract lot amount is 25 times bigger than the median annual specific revenue of the business entity that won the lot, then the contract lot is an outlier.

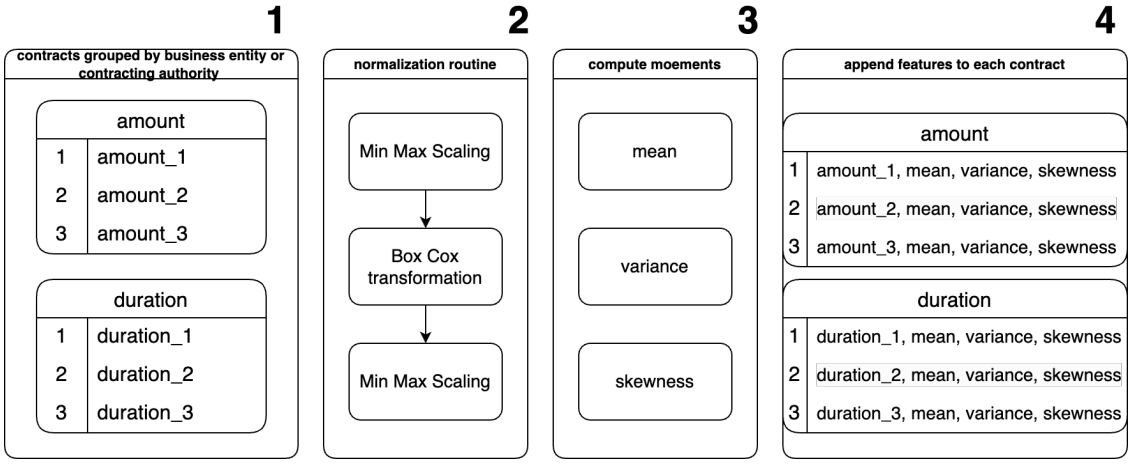


Figure 8.3: Data preparation and enrichment steps the input data set undergoes before being fed to the outlier detecting models. The output of each step is supposed to be fed to the next one in ascending order.

To enforce the rules, we first need to compute the median annual specific revenue and the median annual expenditure of each contract. To compute the former, we group the contracts by business entities and years, to compute the latter, the contracts are grouped by contracting authorities and year. The amount values are summed within each contracting party and for each year. finally, we take the median value of the years and we append this value to its contract row. Now, each contract has two new features: the median annual specific revenue of the business entity to whom the contract was awarded and the median annual expenditure of the contracting authority that issued that contract.

With the new features appended to each contract row, we can easily determine which contracts are outlier creating *pandas DataFrame masks* objects that flag the outliers as ones and the inliers as zeros.

```
def rule1(df: pd.DataFrame) -> pd.Series:
    mask = (df.amount > df.be_med_ann_revenue) & \
        (df.amount > df.pa_med_ann_revenue)
    return mask

def rule2(df: pd.DataFrame) -> pd.Series:
    n_years = 10
    mask23 = (df.award_procedure == 23) &  \
        (df.duration > n_years * 365)
```

```
    mask26 = (df.award_procedure == 26) &  \
11          (df.duration > n_years * 365)
    return mask23 | mask26
13

def rule3(df: pd.DataFrame) -> pd.Series:
15      k = 25
    return df.amount > k * df.be_med_ann_revenue
```

The output masks values are transformed to comply with standard notation for inliers and outliers:

inlier : 1

outlier : -1

Next, each output is recorded in a *comma separated value* file.

8.4 Probabilistic tails model

To apply the Chebishev inequality we first compute the Box Cox transformation to the contract features *amount* and *duration* grouped by contracting entity; that is, we first regroup the contracts by contracting party, then we apply the Box Cox transformation optimizing lambda to each subset of contracts. The results is a new set of features:

- business entity Box Cox transformed contract amount
- contracting authority Box Cox transformed contract amount
- business entity Box Cox transformed contract duration
- contracting authority Box Cox transformed contract amount

Then, for each subset, we compute its mean and standard deviation. Finally, we apply the Chebishev's inequality to determine which is an outlier and which is not.

The Chebishev's inequality requires to specify the value of k. We arbitrarily set $k = 10$, as from 6.1 the probability of finding an extreme value is less than one percent.

With the application of the Box Cox transformation, we aim at shaping our distribution as normal. The Shapiro-Wilk test [30] provides a handy tool to verify the hypothesis that the distribution are actually normal. The *scipy*

package [29] offers the *shapiro* function to compute the p-value under the null-hypothesis that the sample is normally distributed.

Table 8.1 shows the percentages of business entities and contracting authorities having an *amount* and *duration* normal distribution. There are 11037 business entities and 1562 contracting authorities. The threshold to pass the Shapiro-Wilk test is the standard 0.05.

	business entity	contracting authority
Box Box transformed *amount*	6.63%	0.06%
Box Box transformed *duration*	8.44%	1.60%

Table 8.1: Percentages of contracting parties having having a normal distribution with respect to the amount and duration feature

$k = 3$ in 6.1 for contracts following a normal distribution, as $\mathbb{P}(|X - \mu| \leq 3\sigma) = 0.9973$, which is comparable to 99 percent yielded by the Chebishev's inequality with $k = 20$.

The determination of the contracts in the tails is as described by the following Python pseudocode

```
for entity in ["business_entity",
               "contracting_authority"]:
    for feature in ["amount", "duration"]:
        for X in df.groupby(entity)[feature]:
            mu = mean(X)
            sigma = std(X)

            pvalue = shapiro_wilk_test(X)
            if pvalue > 0.05:
                k = 3
            else:
                k = 20

            outlier_flags = abs((X - mu)) > k * sigma
```

8.5 Kernel density estimation model

This model is implemented in the `R` language. The reason for the change resides in the lack of Python packages that implements the optimization

of the bandwidth parameter H in a multivariate case by means of cross validation.

There are three python packages that provide tools to perform the estimation of densities with kernel smoothing, namely the `scipy.stats` package, the `sklearn` package and the `KDEpy` package, yet none of them offers a bandwidth selector that works out-of-the-box.

On the other hand, the `R` package `ks` [31] provides the functions `ks::Hns`, which implements the *normal scale bandwidth selector* 6.20, the `ks::Hpi`, which implements a *2-stage plug-in* selector as described by [18], and the `ks::Hlscv`, which implements a least squares cross validation selector. The `ks` package is based on the work of Chacón and Duong [32].

The package comes with a few limitations. The maximum number of dimension of the input data is six. The computation of the density estimate with an interpolation algorithm is limited to four dimensions on a evaluation points on a evenly spaced grid. We want to use the interpolation algorithm to calculate the density estimate because the process is quicker than computing over the whole grid of each sample we want the prediction of.

For the aforementioned reason, the data set's features are restricted to four: $amount_{be}$, $amount_{pa}$, $duration_{be}$, $duration_{pa}$ where the subscripts specify the contracting party with which the data set has been grouped by in the preprocessing stage.

In our experiments, each density is estimated on a grid linear grid of $21 \times 21 \times 21 \times 21$ nodes.

The kernel density estimation process works as showed in the listing below. We are assuming the the train and the test sets are already loaded in memory as `X.train` and `X.test`.

```
# compute bandwidth matrix
Hns <- ks::Hns(x = X.train)

# compute the density function of the test set
fhat <- ks::kde(x = X.test, H = Hns)

# get probability prediction
preds <- ks::predict(fhat, x = X.test)
```

The bandwidth selector `ks::Hns` can be changed with the preferred `ks::Hpi` or `ks::Hlscv`.

8.6 Gaussian mixture model

The package that implements the Gaussian mixture model is provided by the Python Scikit-learn library [33].

The model is implemented by the `GaussianMixture` class from the `sklearn.mixture` package.

The computation of the density estimates works as showed by the next Python code listing. The code uses the *application programming interface* provided by scikit-learn. The `X_train` and `X_test` variables are supposed to be the training and testing set output of the preprocessing stage.

```
# instantiate a model
gmm = GaussianMixture(n_components,
                      covariance_type)

# fit the model on the training set
gmm.fit(X_train)

# compute the density estimates of the test set
preds = gmm.score_samples(X_test)
```

The class requires the selection of the number of components and the type of covariance matrix. Minimizing the *Bayesian information criterion* 6.30 is the selection method we opted for. The implementation of the criterion is provided as a method of the `GuassianMixture` class, namely the `bic` method, that for given data and parameter set computes the BIC value. The parameter optimization is carried out by the following python function. The X is supposed to be a `numpy` array matrix representation of the data set.

```
def optimize_paramters(, X:np.array):
    lowest_bic = np.infty
    bic_list = []
    best_gmm = None
    for cv_type in ["shperical", "tied", "diag", "full"]:
        for n in range(1, 60):

            # instantiate a model
            gmm = GaussianMixture(
                n_components = n,
                covariance_type = cv_type
                )
```

```
13
                # fit the current model
15              gmm.fit(X)

17              # compute the BIC value
                bic_value = gmm.bic(X)
19
                # record the BIC value
21              bic_list.append(bic_value)

23              if bic_list[-1] < lowest_bic:
                    lowest_bic = bic_list[-1]
25                  best_gmm = gmm

27      return bic_list, best_gmm
```

Figure 8.4 shows the *BIC* scores of the Gaussian Mixture models with number of components ranging from one to sixty and different types of covariance matrix.

The best model has a BIC score of -9522.4081, a diagonal covariance matrix, and 57 components. The diagonal nature of the best model covariance matrix suggest that the axis the generated model are orthogonal; from a statistical perspective this entails that the predictors of this generative model are independent.

8.7 One-class support vector machine

The `sklearn.svm` package provides the `OneClassSVM` class that implements the model proposed by [22].

The classification in inliers and outliers works as showed by the next Python code listing. The code uses the *application programming interface* provided by scikit-learn. The `X_train` and `X_test` variables are supposed to be the training and testing set output of the preprocessing stage.

```
1       # instantiate a model
        svm = OneClassSVM(nu)
3
        # fit the model on the training set
5       svm.fit(X_train)
```

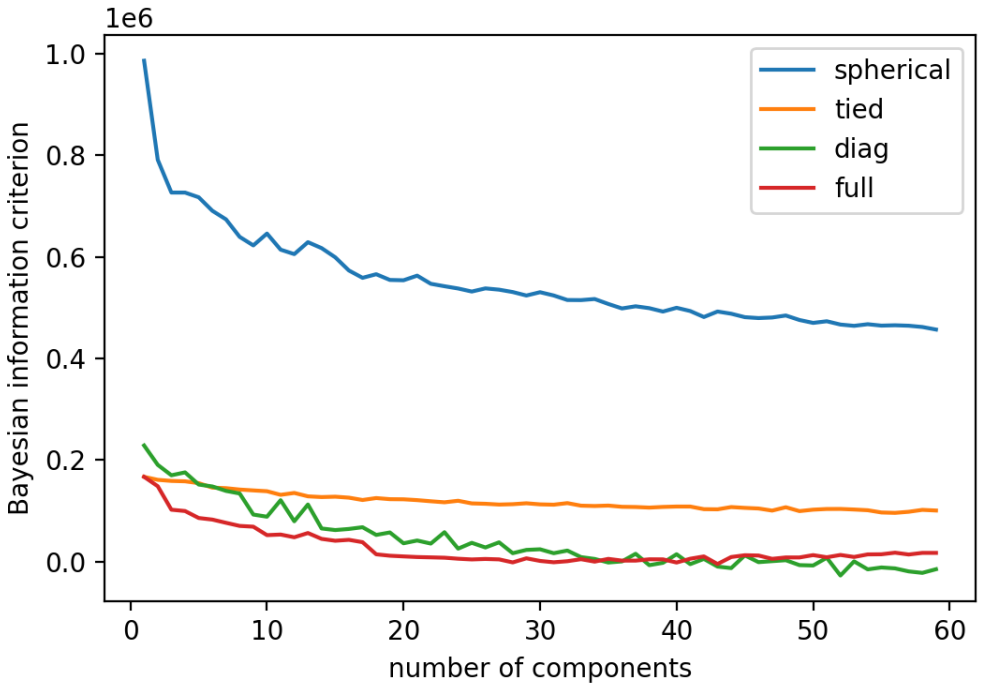


Figure 8.4: Bayesian Information criterion scores of Gaussian Mixture models by number of components and type of covariance matrix computed on the training set

```
7       # classify the test set
        labels = svm.predict(X_test)
9
        # compute the distances from the separating
11      # hyperplane
        distances = svm.decision_function(X_test)
```

The `nu` parameter is the ratio of outliers the model expects to find in the data set; in the equation describing the model 6.32, it is called *contamination* parameter ν. In the model implementation ν is estimated with `y_train.csv` set, that contains the labels output of classification carried out by the *baseline* and *probabilistic tails* models.

The `distances` variable are computed to sort the classified sample in order of descending distance from the separating hyper-plane. The assumption is that the greater the distance, the greater the anomaly.

Chapter 9

Results

9.1 SME and probabilistic outliers

The application of the rules defined by the subject matter expert in the baseline model yields the following outlier counts by rule 9.1.

feature	# SME outliers	percentage
amount	218	0.035%
duration	12	0.003%

Table 9.1: Count of SME outliers resulting from the rules defined by the subject matter expert

The main characteristic is that they generally do not reside in the tails of their awarded business entities amount and duration distributions nor of the contracting authorities.

Table 9.2 shows the count of outliers result of the application of the Chebishev inequality in the probabilistic tails model. As one would expect, the

feature	# probabilistic outliers	percentage
amount	187	0.031%
duration	152	0.025%

Table 9.2: Count of the outliers found by the probabilistic tails model grouped by feature

outliers lie in the upper tails of the amount and duration distribution of the contracts grouped by contracting party.

9.2 Models comparison

A common metric to evaluate the performance of binary classifiers is plotting *Receiver Operating Characteristic* curves (ROCs). It involves the computation of the *True Positive Rate* (TPR) (also called *hit rate*) and the *False Positive Rate* (FPR) (or *false alarm rate*) for a range of thresholds. Defined

TP true positive, the count of samples that are predicted as positive (inliers) by the model and they are positive in the ground truth

FP false positive, the count of samples that that are predicted as positive (inliers) by the model, but they are negative in the ground truth

TN true negative, the count of samples that that are predicted as negative (outliers) by the model and they are negative in the ground truth

FN False negative, the count of samples that that are predicted as negative (outliers) by the model, but they are positive in the ground truth

TPR and FPR are given by

$$TPR = \frac{TP}{TP + FN} \qquad FPR = \frac{FP}{FP + TN}$$

The ROC plots the TPR on the y-axis, the FPR on the x-axis.

The thresholds used to determine inliers from outliers are densities for the all the KDE model variations and GMM, the distance from the hyperplane for the OC-SVM model.

Another common measure the *Area Under the Curve* (AUC) that, as the name suggests, is value of area under the ROC curve. Yet, the measure is less expressive as the ROC plot because it cannot show the thresholds at which models mis-label the samples. Table 9.3 shows the AUC scores of each model.

On the other hand, the ROC curves can be used to determine the threshold apt to our purposes. Indeed, it is generally preferred to have outliers in the lower end of the decision function; it follows that ROC curves where the TPR is high and the FPR is low are preferred, as such curves are yielded by models where the outliers are in the lower end of the decision function spectrum.

Figures 9.1a, 9.1b 9.1c shows the ROC curves yielded by the KDE models with different bandwidth selectors, namely the normal scale selector (NS), the plug-in selector (PI), and the least squares cross validation selector (LSCV).

The best performance among the three is achieved by the normal scale selector as percentage of miss-classified samples is zero for almost the 60 percent of the sample. The performance of the KDE_{pi} and KDE_{lscv} models is comparable.

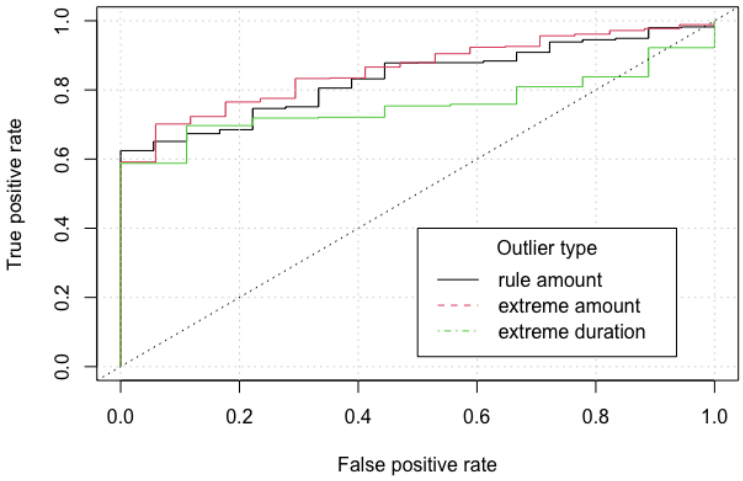
Figures 9.2 and 9.3 shows the ROC curves of the GMM and OC-SVM models. The GMM is exceptionally good in predicting the probabilistic outliers in the duration feature while the other ROC comparable with those of the KDE models. The OC-SVM performance is very poor. The curves resemble that of a random guesser.

The reason for OC-SVM performance lies in the nature of the classifier. The model clusters data according to the euclidean distance between samples. The performance suggests that on the current feature space, the euclidean distance is not well suited to distinguish inliers from outliers.

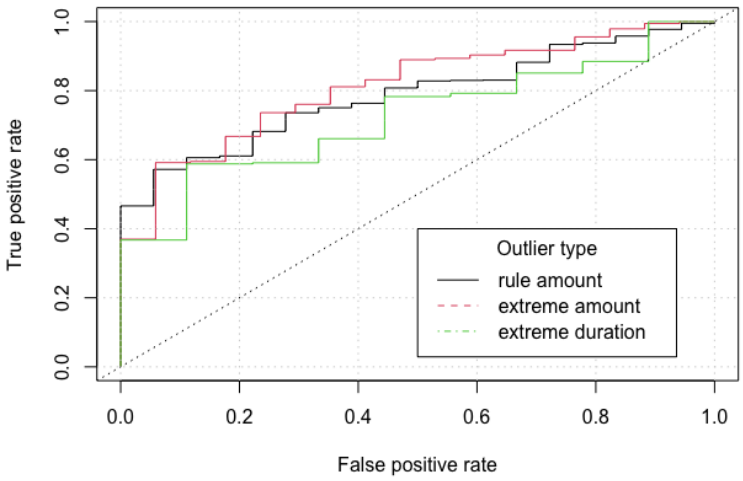
Figure 9.4 shows the OC-SVM decision function contours and the outlier location along with their type. To visualize the decision function contours in only two dimension, he data has been projected on the first two principal components of the singular value decomposition of the data matrix. The first two principal components explain 99.86 percent of the total variance. The OC-SVM model is trained on such projection. It is clear that almost all the outliers are classified as positive by the model. This confirms the inaccuracy of the model and the inadequacy of the euclidean distance as a means to differentiate the contracts.

	KDE_{ns}	KDE_{pi}	KDE_{lscv}	GMM	OC-SVM
SME amount	0.855	0.831	0.846	0.784	0.519
prob. amount	0.871	0.858	0.864	0.821	0.520
prob. duration	0.761	0.761	0.762	0.914	0.490
average	0.828	0.817	0.824	0.840	0.510

Table 9.3: models AUC values on the test set by type of outlier

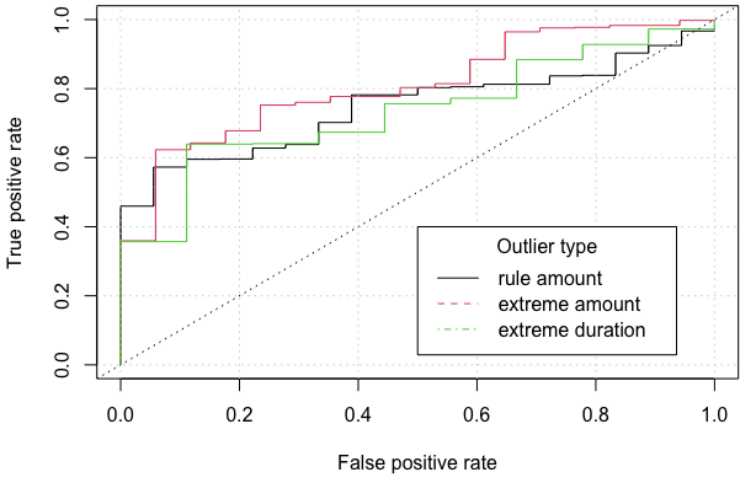


(a) $\hat{\boldsymbol{H}}_{NS}$



(b) $\hat{\boldsymbol{H}}_{PI}$

Figure 9.1: ROC curves of kde models with different bandwidth matrices.



(c) $\hat{\boldsymbol{H}}_{LSCV}$

Figure 9.1: ROC curves of kde models with different bandwidth matrices.

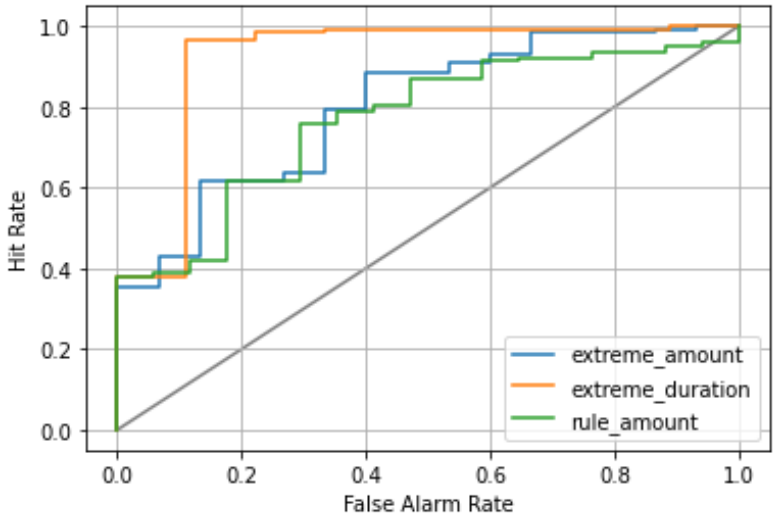


Figure 9.2: ROC curves of the GM model on the test set

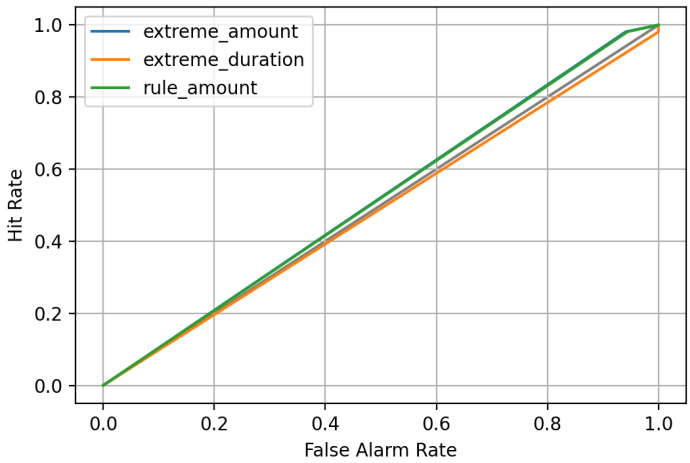


Figure 9.3: ROC curves of OC-SMV model on the test set

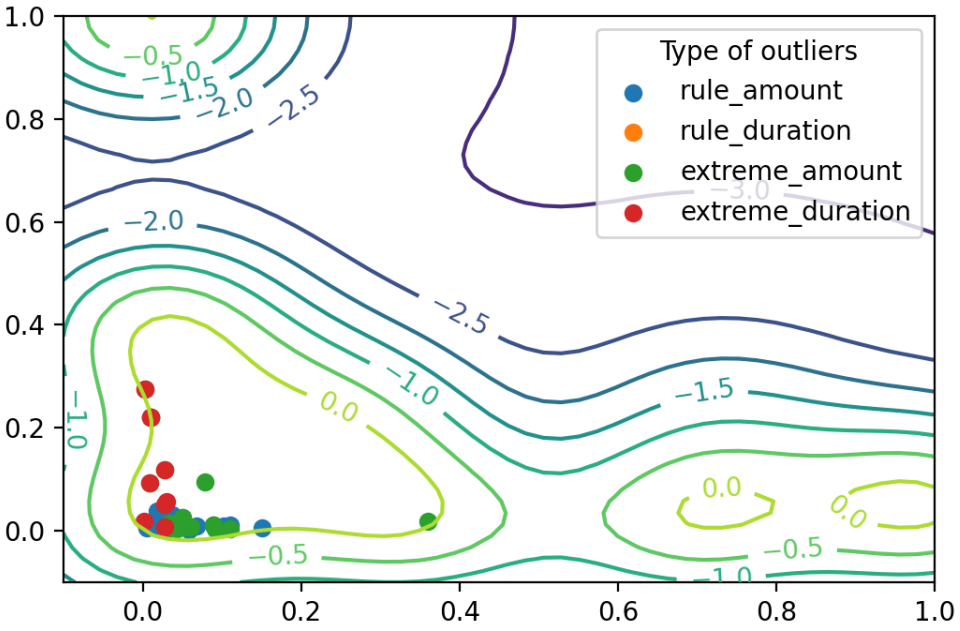


Figure 9.4: Contours of the decision function of OC-SMV model. The data are projected on the first two principal components of the singular value decomposition of the data matrix to visualize the decision function contours in two dimension.

Chapter 10

Conclusions and future works

The main purpose of this thesis is to develop an automated an process process to remove the public contracts containing errors. A secondary purposes is to develop a score of contract *normality* that should tell whether the single contract is aligned with its market.

The strategy used to achieve the objectives consisted in developing a model that could serve both purposes. The use of a probability as a decision variable seems the most intuitive means to achieve our goals, besides the fact that outliers are usually defined in a probabilistic context.

Thus, most of the models developed originates from the idea of determine the data probabilistic distribution. The *probabilistic tails* model tries to determine which samples lie in tails of each data features. The KDE and GMM tries to determine which contracts are rare in a multivariate manner, not only one feature at a time as the former. The only exception to the probabilistic perspective is the OC-SVM model that compares contracts by measuring the euclidean distance between them.

According to their performances, the most promising model is KDE with a normal scale bandwidth selector, a combination of non-parametric method (the kernel smoothing) and parametric statistics (the assumption of normality in the bandwidth selector).

The model main advantage is its simplicity. It uses only four dimensions, estimating the density of new samples is fast thanks to the interpolation method, and the interpretation of its output is straight-forward: it is the probability of encountering a contract with such characteristics in the amount and duration dimensions.

Its drawback is the lack of explainability, especially in the absence of a satisfactory visualization tool that could give hints as to why that contract has such a probability. A possible solution to the problem is the restriction of the number of dimensions to only two, so that the data density can be viewed in a three-dimensional space. Duong et al [34] developed a tool to determine which are the most significant features, so that the two most significant can be used to visualize the distribution. The main idea is that we can determine local maxima regions when the density curvature is non zero.

Another aspect in need of improvement is the feature space. The techniques used in the data preparation stage are not satisfactory. The idea that lead to the use of the Box Cox transformation is that business entities and contracting authorities should have a similar distribution with regards to the contracts amounts, once such spaces have been standardized. Even reducing the contracting authorities and business entities to those having at least ten contracts, the transformation is ineffective. If the contracting parties shared the amount distribution once standardized, the distribution of all the contracts in the amount features should be the mean of all of the single ones. Unfortunately, the amount distribution does not look so. The same applies for the duration feature.

The current state of the art in outlier detection are Auto-Encoders. They are a deep-learning architecture made of two parts. The first part, the so-called *encoder* learns to project the samples fed in a lower dimensional space, while the *decoder* part learns to reconstruct the samples from the lower dimensional space to the original one. The model learns by minimizing the reconstructing error from the original sample with the reconstructed sample. In the context of outlier detection, one can distinguish outliers from inliers by looking at the reconstruction error as outliers should have a greater reconstruction error. The most interesting auto-encoder architecture is that provided by [35]. The authors propose a variational auto encoder that should identify and repair typographical errors in tabular data. Given enough time, the implementation of the methods and architecture is what one should focus on.

Assuming a business point-of-view, the baseline and the probabilistic tails model are the most production-ready. Their simplicity is their strength: the reasons why the outliers they detect are considered are the models themselves. A high intelligible tool has a greater probability of being used by many, even users not fond of statistics. They both can be a helping hand for those responsible of filtering the wrong contracts out. Moreover, the result of such models can be used as a starting point for the developing of a set that

could change the developing of outlier detector from an unsupervised to a supervised task. This change would greatly benefit the quality of the predicted outliers as the model could autonomously learn how to project samples on a space where outliers are well separated from inliers.

Bibliography

[1] ISTAT. Conto consolidato delle amministrazioni pubbliche. http://dati.istat.it/viewhtml.aspx?il=blank&vh=0000&vf=0&vcq=1100&graph=0&view-metadata=1&lang=it&QueryId=18125&metadata=DCCN_FPA#. Accessed: 2023-02-10.

[2] P. Italiano, "Legge 6 novembre 2012, n. 190, art. 1," *Gazzetta Ufficiale*, 2012, art. 1. [Online]. Available: https://www.gazzettaufficiale.it/eli/id/2012/11/13/012G0213/sg

[3] ——, "Decreto legislativo 14 marzo 2013, n. 33," *Gazzetta Ufficiale*, 2013, art. 23. [Online]. Available: https://www.gazzettaufficiale.it/eli/id/2013/04/05/13G00076/sg

[4] ——, "Italian public contract code (legislative decree 50/2016, as modified by legislative decree n. 57/2017)," *Gazzetta Ufficiale*, 2016, art. 1. [Online]. Available: https://www.anticorruzione.it/portal/rest/jcr/repository/collaboration/Digital%20Assets/anacdocs/MenuServizio/English%20section/ITALIAN_PUBLIC_CONTRACT_CODE%2015%20giugno%202018_sito%20(2).pdf

[5] N. A.-C. Authority. (2022) Accordo quadro faq. [Online]. Available: https://www.anticorruzione.it/-/accordo-quadro

[6] E. Parliament and the Council of the European Union, "Directive 2014/24/eu of the european parliament and of the council of 26 february 2014 on public procurement and repealing directive 2004/18/ec," *Official Journal*, 2014. [Online]. Available: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:02014L0024-20180101

[7] SIMAP. [Online]. Available: https://simap.ted.europa.eu/web/simap/cpv

[8] D. F. Amato, "Approccio ibrido per la classificazione gerarchica automatica di oggetti di contratti della pubblica amministrazione italiana," Master's thesis, Politecnico di Torino, 2022.

[9] D. N. Joanes and C. A. Gill, "Comparing measures of sample skewness

and kurtosis," *Journal of the Royal Statistical Society. Series D (The Statistician)*, vol. 47, no. 1, pp. 183–189, 1998. [Online]. Available: http://www.jstor.org/stable/2988433

[10] H. A. Sturges, "The choice of a class interval," *Journal of the American Statistical Association*, vol. 21, no. 153, pp. 65–66, 1926. [Online]. Available: https://doi.org/10.1080/01621459.1926.10502161

[11] D. Freedman and P. Diaconis, "On the histogram as a density estimator:l 2 theory," *Z. Wahrscheinlichkeitstheorie verw Gebiete*, vol. 57, no. 4, pp. 453–476, Dec. 1981.

[12] C. C. Aggarwal, *Outlier Analysis*. Springer, 2013. [Online]. Available: https://doi.org/10.1007/978-1-4614-6396-2

[13] P.-N. Tan, M. Steinbach, A. Karpatne, and V. Kumar, *Introduction to Data Mining (2nd Edition)*, 2nd ed. Pearson, 2018.

[14] L. Ruff, J. Kauffmann, R. Vandermeulen, G. Montavon, W. Samek, M. Kloft, T. Dietterich, and K.-R. Müller, "A unifying review of deep and shallow anomaly detection," *Proceedings of the IEEE*, vol. PP, pp. 1–40, 02 2021.

[15] G. E. P. Box and D. R. Cox, "An analysis of transformations," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 26, no. 2, pp. pp. 211–252, 1964. [Online]. Available: http://www.jstor.org/stable/2984418

[16] N. R. Draper and D. R. Cox, "On distributions and their transformation to normality," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 31, no. 3, pp. 472–476, 1969. [Online]. Available: http://www.jstor.org/stable/2984350

[17] E. García-Portugués, *Notes for Nonparametric Statistics*, 2022, version 6.5.9. ISBN 978-84-09-29537-1. [Online]. Available: https://bookdown.org/egarpor/NP-UC3M/

[18] S. J. Sheather and M. C. Jones, "A reliable data-based bandwidth selection method for kernel density estimation," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 53, no. 3, pp. 683–690, 1991. [Online]. Available: http://www.jstor.org/stable/2345597

[19] H. Akaike, *Information Theory and an Extension of the Maximum Likelihood Principle*. New York, NY: Springer New York, 1998, pp. 199–213. [Online]. Available: https://doi.org/10.1007/978-1-4612-1694-0_15

[20] G. Schwarz, "Estimating the Dimension of a Model," *The Annals of Statistics*, vol. 6, no. 2, pp. 461 – 464, 1978. [Online]. Available: https://doi.org/10.1214/aos/1176344136

[21] S. Konishi and G. Kitagawa, *Information Criteria and Statistical Modeling*, 1st ed. Springer Publishing Company, Incorporated, 2007.

[22] B. Schölkopf, R. Williamson, A. Smola, J. Shawe-Taylor, and J. Platt, "Support vector method for novelty detection," in *Proceedings of the 12th International Conference on Neural Information Processing Systems*, ser. NIPS'99. Cambridge, MA, USA: MIT Press, 1999, p. 582–588.

[23] B. K. O. Tas, "Effect of public procurement regulation on competition and cost-effectiveness," *Journal of Regulatory Economics*, vol. 58, no. 1, pp. 59–77, Jun. 2020. [Online]. Available: https://doi.org/10.1007/s11149-020-09409-w

[24] E. Prier, C. McCue, and E. A. Boykin, "Assessing european union standardization: a descriptive analysis of voluntary ex ante transparency notices," *Journal of Public Procurement*, vol. 21, no. 1, pp. 1–18, Apr. 2021. [Online]. Available: https://doi.org/10.1108/jopp-12-2019-0086

[25] M. Schmidt, "Price determination in public procurement: A game theory approach," *European Financial and Accounting Journal*, vol. 10, no. 1, pp. 49–62, Mar. 2015. [Online]. Available: https://doi.org/10.18267/j.efaj.137

[26] J.-M. Kim and H. Jung, "Predicting bid prices by using machine learning methods," *Applied Economics*, vol. 51, no. 19, pp. 2011–2018, Oct. 2018. [Online]. Available: https://doi.org/10.1080/00036846.2018.1537477

[27] Wes McKinney, "Data Structures for Statistical Computing in Python," in *Proceedings of the 9th Python in Science Conference*, Stéfan van der Walt and Jarrod Millman, Eds., 2010, pp. 56 – 61.

[28] T. pandas development team, "pandas-dev/pandas: Pandas," Feb. 2020. [Online]. Available: https://doi.org/10.5281/zenodo.3509134

[29] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.

[30] S. S. Shapiro and M. B. Wilk, "An analysis of variance test for normality (complete samples)," *Biometrika*, vol. 52, no. 3/4, pp. 591–611, 1965. [Online]. Available: http://www.jstor.org/stable/2333709

[31] T. Duong, *ks: Kernel Smoothing*, 2020, r package version 1.11.7.

[Online]. Available: https://CRAN.R-project.org/package=ks

[32] D. T. Chacón J.E., *Multivariate Kernel Smoothing and Its Applications*, 1st ed. Chapman and Hall/CRC, 2018.

[33] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[34] T. Duong, A. Cowling, I. Koch, and M. Wand, "Feature significance for multivariate kernel density estimation," *Computational Statistics & Data Analysis*, vol. 52, pp. 4225–4242, 05 2008.

[35] S. Eduardo, A. Nazábal, C. K. I. Williams, and C. Sutton, "Robust variational autoencoders for outlier detection and repair of mixed-type data," 2019. [Online]. Available: https://arxiv.org/abs/1907.06671