NOVA IMS Information Management School

MGI

Mestrado em Gestão de Informação
Master Program in Information Management

DECISION TREES FOR LOSS PREDICTION IN RETAIL

Case of Pingo Doce

Mariana Bonito Henriques

Project Work presented as the partial requirement for obtaining a Master's degree in Information Management

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

NOVA Information Management School

Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa

DECISION TREES FOR LOSS PREDICTION IN RETAIL

Case of Pingo Doce

Mariana Bonito Henriques

Project Work presented as the partial requirement for obtaining a Master's degree in Information Management with specialization in Knowledge Management and Business Intelligence

Advisor: Professor Doutor Roberto André Pereira Henriques

August 2019

ABSTRACT

The use of data mining as a way of solving problems from the widest range of areas with the main purpose of gaining competitive advantage is rising, specially in retail, an extremely competitive sector that requires an even bigger advantage. Additionally, food loss, beyond representing a huge waste of resources, can also be considered a major issue to the retail sector due to the financial losses originated from it. Thus, I proposed to help Pingo Doce, a well-known Portuguese retail company, to solve their food loss issue which, despite being the major cause of a huge drop in the company's profits, has never been solved till this day.

Therefore, this project focuses on the development of a classification algorithm that will allow to predict future significant losses in several fruits sold in certain Pingo Doce stores. To do so I applied a Decision Tree algorithm that, due to its representation in the form of if-then rules, will help to identify the main features that lead to a higher number of losses, namely the period of the year and the category to which each fruit belongs, among others.

The dataset provided by the company contains variables that measure the quantity and value of sales, stocks, identified and unidentified losses, over a one-year period, and regarding 81 different fruits and 20 stores from all over the country. Additionally, I created new variables such as the criminality rate of the municipality and the climate class of each store, as well as the seasons and the day of the week in which each observation occurred. All these variables allowed me to create four different datasets that originated four different Classification Trees.

The results show that, using a dataset with no information regarding stocks and sales, containing only variables that describe the characteristics of the stores, products and periods of time, as well as the value of product sold per unit of measurement, i.e. the price per unit of measurement of each fruit, it is possible to create a Decision Tree that reaches an accuracy of 74% and correctly predicts 82% of the observations that represent significant losses.

The algorithm obtained allowed to identify the variables that are more prone to originate significant losses, namely: the day of the week, the fruit's category, the season of the year, the position of that week in the respective month and the price at which the product is being sold.

KEYWORDS

Machine Learning; Retail; Loss; Classification Algorithm; Classification Tree; Decision Tree

INDEX

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

DM	Data Mining
ML	Machine Learning
DT	Decision Tree
CT	Classification Tree
TP	True Positives
TN	True Negatives
FP	False Positives
FN	False Negatives

1. INTRODUCTION

1.1. CONTEXT

According to (Jagdev, 2018), retailers are now mining customer data to increase profits and growth, as well as to be in competition, allowing them to reach an increase in sales in more than 70%. Therefore, the exponential growth of data in the retail sector (Jagdev, 2018) presents unique opportunities to its holders to create information, and therefore knowledge, with the purpose of obtaining advantage towards their competitors (Davison & Weiss, 2010). Although originally the process of turning data into knowledge relied on manual analysis (Fayyad, Piatetsky-Shapiro, & Smyth, 1996)**,** nowadays we are capable of "*discovering useful patterns and trends in large data sets*", by the appliance of several technologies and techniques (Larose & Larose, 2014, Chapter 1, para. 4).

That said, retailers should rely on Data Mining to access and analyse their data, not only to achieve better results, but also to obtain a better understanding of both their customers and their operations (Jagdev, 2018).

1.2. PROBLEM

According to (Daryanto & Sahara, 2016), food loss, which occurs both at wholesale and retail markets, represents a problem since it not only contributes to financial losses but also to the waste of natural resources. Although there is no agreement on the quantity of currently lost global food production, fresh fruits and vegetables are among the products with a higher percentage of food waste (Parfitt, Barthel, & MacNaughton, 2010). Despite the difficulty to measure this proportion it is estimated that "worldwide about one third of all fruits and vegetables produced are never consumed by humans" (Kader, 2005).

Since a high number of losses leads to a strong decrease in the net income attributable to the company, in comparison with the value that would be expected given the number of stored products in each store, and once this is a recurrent problem in all Pingo Doce stores, it results in a significant decrease in the revenue of the company. Additionally, given the competitive nature of retail, this unnecessary decrease in the revenue must be eliminated or at least reduced so that it won't be a disadvantage to this company.

Given the fact that food loss in Pingo Doce is more common in fresh products such as fruits and vegetables, the company decided that it is a priority to start by analysing the losses that occur in those products. Consequently, in this project data mining techniques will be used to develop a model that will be able to predict if a significant loss will occur, more specifically in the fruit's category, in several Pingo Doce stores.

1.3. MOTIVATION

As previously referred, losses induce a decrease in the net income of the company together with an increase in the food waste. Due to both reasons it is in the interest of everyone involved to find a solution to this problem. Solving this problem obviously results in a notable reduction in the number of significant losses, what would derive in a decrease in the quantity of wasted products, as well as in the value lost by the company. This would obviously be beneficial to the company since it would

increase their profit. Finally, solving this problem using Machine Learning algorithms would demonstrate how important this field is to retail companies, though most of them are not yet ready to take advantage of this useful techniques.

1.4. PROPOSAL

The proposal to solve this problem is to rely on Data Mining techniques by applying a supervised Machine Learning algorithm. This approach consists in performing not only data analysis but also discovery algorithms that create patterns or models over the data, allowing to predict whether certain article will suffer a significant loss or not, in a certain future time period, depending on the values observed in the variables available in the provided dataset.

Although machine learning algorithms are widely used and relatively simple to create and implement, and even though this is a huge company with thousands of employees with a lot of expertise, this process has never been used with the purpose of solving the losses issue. Additionally, as previously said, this is a huge concern to the company, which has already tried more than once to find a solution to this question, however always unsuccessfully. Therefore, the success of this project would be unprecedented, bringing great benefits to the company and consequently to the Jerónimo Martins Group.

1.5. OBJECTIVES

The major objective of this work is to show the applicability of Machine Learning and Data Mining in the study of the high number of losses that occur in this company, and probably in all retail companies. Besides this major objective, I can also point out several smaller objectives such as: pre-processing the data that was provided, creating and applying several models to the pre-processed data, evaluating all models to figure out which presents the best results and present it to the company as the model that will help solving their problem.

1.6. INTRODUCING THE COMPANY

Jerónimo Martins is an International Group with its headquarters in Portugal, founded in 1792 by a namesake young man from Galicia that decided to open a store in the downtown of Lisbon, soon becoming the main supplier of the portuguese capital. The business thrives until 1920, when on the verge of bankruptcy, the company is bought by a set of merchants from Porto called “Grandes Armazéns Reunidos”, who agreed to keep the name as “Estabelecimento Jerónimo Martins & Filho”. Among the five associates that formed the company, three eventually abandoned it, remaining only: Elísio Pereira do Vale and Francisco Manuel dos Santos. In 1945, Elísio Alexandre dos Santos, nephew of Francisco Manuel dos Santos, becames director of Jerónimo Martins, and after his death in 1968, the family business is assumed by his son Alexandre Soares dos Santos. Under the direction of Alexandre Soares dos Santos, the supermarket chain Pingo Doce is founded in 1978, with the opening of the first Pingo Doce store in 1980. Fifteen years later, in 1995, the group started its internationalization, firstly with the expansion to Poland and finally in the year of 2013 with the opening of the first Ara stores in Colombia. Therefore, this is a group with 227 years of experience as a Food Specialist, that nowadays is present in three geographies, namely Portugal, Colombia and Poland, in two different continents, counting with more than 104 000 employees and over 3 600 stores.

Currently, Jerónimo Martins' Group structure is divided in two major areas: Food Distribution and Specialized Retail. Examples of the first mentioned area are: Pingo Doce and Recheio in Portugal, Biedronka in Poland and Ara in Colombia. Regarding the Specialized Retail area there are examples like Jeronymo and Hussel in Portugal and Hebe, a chain of specialized Health and Beauty stores, in Poland. In Portugal, Jerónimo Martins´ businesses in Food Distribution have two main strands: Pingo Doce, as a food retail company, and Recheio Cash & Carry, as wholesale food company.

This project was developed in partnership with Pingo Doce, since among all the portuguese companies belonging to the Jerónimo Martins Group this is the one that presents the biggest number of losses.

The remaining of this document follows the according structure: the next chapter contains the literature review and the subsequent chapter presents the methodology used in this project. Thereafter, chapter 4 is assigned to the presentation of results and respective discussion, chapter 5 introduces the conclusion of the project and lastly, in chapter 6, the project limitations are mentioned together with several recommendations for future works.

2. LITERATURE REVIEW

2.1. DATA MINING

The amount of data created worldwide is growing exponentially through the years (Davison & Weiss, 2010). According to the graphic presented below (Figure 2.1), which reflects the increase in the amount of data created worldwide from 2010 to 2025 ("Data created worldwide 2010-2025 | Statista", 2019), in 2010, 2 zetabytes of data were created worldwide, while in 2015 that number grew to more than 15 zetabytes. Moreover, the number of worldwide created data in 2020 and 2025 were predicted using forecasting methods, expecting to reach 175 zetabytes in 2025, which represents 87,5 times more data than the one created in 2010.

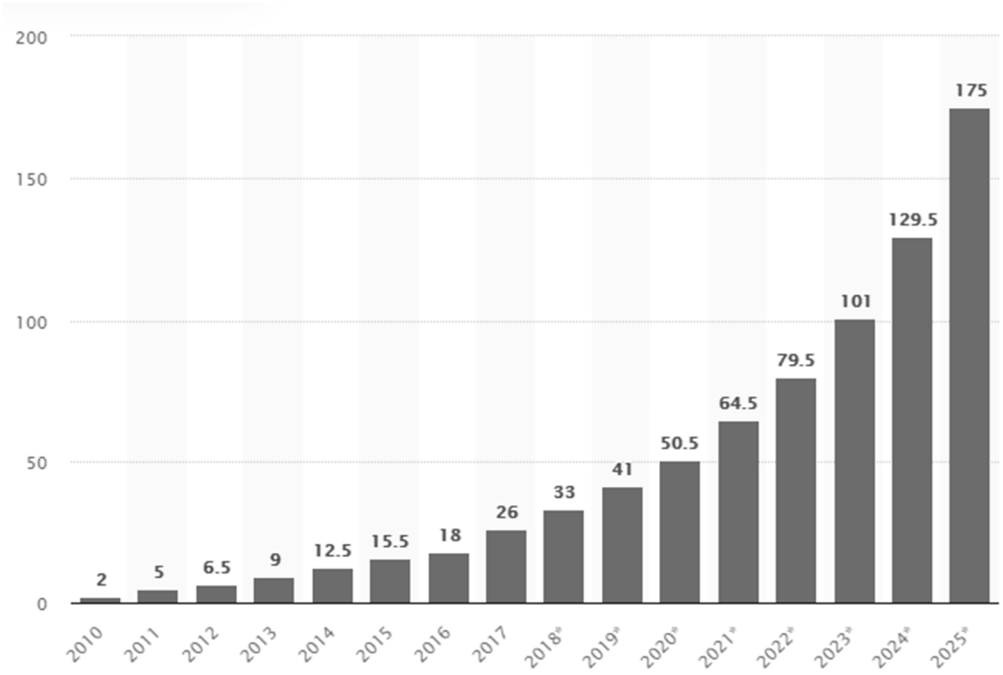

Figure 2.1: Volume of data created worldwide from 2010 to 2025, in zetabytes ("Data created worldwide 2010-2025 | Statista", 2019)

The tremendous growth in the data available, caused by the advances in computer technology, the emergence of high-speed networks and the diminishing of disk costs, represents a challenge, since traditional statistical techniques are not capable of handling such large amounts of data (Davison & Weiss, 2010). Additionally, the non-traditional nature of data together with its lack of structure makes it impossible to be analysed with the established data analysis techniques (Tan, Steinbach & Kumar, 2005). Due to all these reasons, manual data analysis became no longer sustainable and several new methods, tools and technologies were required to support humans in extracting information from all the accessible data (Fayyad et al., 1996)**,** resulting in the creation and development of the field of Data Mining (Tan, Steinbach & Kumar, 2005). Consequently, data mining appeared as a *"technology that blends traditional data analysis methods with sophisticated algorithms for processing large volumes of data"* (Tan, Steinbach & Kumar, 2005, p. 1), allowing to automatically recognize useful patterns, and therefore valuable information among data.

Therefore, DM tasks can be divided into two main categories (Davison & Weiss, 2010), namely: description, that summarizes the data in some way, perhaps by segmenting data based on similarities and differences observed (Chen et al., 2012), and prediction, that allows to predict the value of a future observation by finding patterns in data that allow to estimate the future behaviour of some entities (Fayyad et al., 1996). Inside the prediction task we can distinguish two of the most commonly used DM tasks: classification and regression, bearing in mind that both tasks entail the creation of a model that predicts a target variable from a set of explanatory variables (Davison & Weiss, 2010). While in classification a function that maps a data item into one of several predefined classes is learned, in regression the applied function maps data items to a real-valued prediction variable (Fayyad et al., 1996).

In this project I will rely on classification since my objective is to predict if a certain product will suffer a significant loss, belonging to one of this two possible classes.

2.2. DATA MINING IN THE RETAIL SECTOR

The challenge of managing huge amounts of data and the capability of recognizing the relevant information from it, together with the need for immediate product and service development that allows to take advantage of freshly market opportunities are some of the factors that have contributed for the imperative need to adopt data mining tools for companies that want to differentiate themselves from their competition and gain a good position in today's market (Kleissner, 1998).

Therefore, data mining solutions are used to assist critical decision making in several areas such as banking, not only for fraud and money laundering detection (Raj, 2015) but also for loan credibility prediction trough the use of a Decision Tree that identifies the most significant customers' characteristics for credibility supporting banks in the process of decision making regarding loan requests (Sudhakar & Reddy, 2016); healthcare, trough the development of efficient heart attack prediction methods (K, M, & R, 2018); and transportation industries, allowing planners to better perceive public transport user behaviours, thus helping to improve their service (Agard, Morency, & Trépanier, 2006).

Lastly, the Retail area should be emphasized once it is the one being analysed in this project. Being defined as "*the activity of selling goods to the public, usually in shops"* ("RETAIL | Meaning, definition in English Dictionary", 2019), it allows and enhances the use of several DM mechanisms with the purpose of determining what kind of products a customer usually purchases at the same time or even to create targeted marketing campaigns to different clusters of customers, achieved through the definition of groups of customers that present similar characteristics (Kleissner, 1998), as presented in (Chen et al., 2012) where a Recency, Frequency and Monetary model was used with the purpose of dividing the customers of an UK-based online retailer into several meaningful groups through the use of the k-means clustering algorithm, clearly identifying the main characteristics of the consumers in each group. Therefore, many online retailers like Amazon, Walmart, Tesco and EasyJet are already applying data mining techniques with the purpose of supporting customer-centric marketing, allowing them to benefit from a competitive advantage (Chen et al., 2012).

Another example of the use of data mining in retail industry is the improvement of future stock estimate by the analysis of several datasets regarding sales with the objective of predicting alterations

in demands, as presented in (Manyika et al., 2011), where retailers rely on bar code systems data to automate stock replenishment, cutting down the number of incidents of stock delays.

Additionally, in (Zhang, Cheng, Liao, & Choudhary, 2012), customer reviews from Amazon's website were analysed trough the use of a model that returned each product ranking result by: filtering each review's content and removing unrelated comments, assigning weights to each review based on its helpfulness and time since posting date and calculating each product's ranking score by combining the previous weights to the polarity of each review, i.e. the difference between the number of positive and negative sentences that compose the review.

Finally, in (Karamshuk, Noulas, Scellato, Nicosia, & Mascolo, 2013) the extraction of knowledge from geographic data towards services improvements was studied. In the mentioned work, a dataset from Foursquare, a location-based social network, was analysed with the objective of identifying the most promising area to open a new store among a set of pre-defined ones. This study was focused on three retail chains, namely Starbucks, McDonald's and Dunkin' Donuts, and a square region around the centre of New York and allowed to identify which geographic and mobility features are the best indicators of popularity for each retailer brand.

2.3. MACHINE LEARNING

After introducing the concept of DM, I find it worth mentioning Machine Learning, since this project is based on predicting whether there will be a significant loss in certain products, through the application of ML algorithms. Therefore, considering the strong relationship between both concepts, making these definitions similar and therefore sometimes confusing, I find it imperative to present the concept of ML. As previously stated, DM can be defined as the *"nontrivial extraction of implicit, previously unknown, and potentially useful information from data"* (Sahu, Shrma, & Gondhalakar, 2008, p. 114) but the process of mining knowledge from data requires a technology that is able to provide tools that support it: Machine Learning (Witten, Frank, Hall, & Pal, 2016).

That said, and according to (Alpaydin, 2010, p. xxxi)**,** ML can be defined as the process of "*programming computers to optimize a performance criterion using example data or past experience".* According to (Shalev-Shwartz & Ben-David, 2013)**,** ML can be divided into several subfields, depending on the nature of the interaction that occurs during the learning task. Consequently, we can distinguish two major types of machine learning algorithms: supervised learning algorithms, in which the inputs are given with the corresponding outputs, and unsupervised learning algorithms, in which every instance in the dataset is untagged (Kotsiantis, 2007). This allows us to understand that supervised methods have the objective of understanding the relationship between the predictor variables and the target variable (Rokach & Maimon, 2014), what is exactly the objective of this project. That said, since all instances in our dataset are labelled as 0 or 1 regarding the *Significant Loss* variable, allowing the algorithm to rely on that label to create a function that maps inputs to expected outputs, being able to classify new instances regarding their target variable value, I will apply supervised learning algorithms.

Supervised models can be divided into Regression and Classification models, depending on the domain of the target variable, since in regression the target variable has a real-valued domain while in classification the input space is projected into previously defined classes (Rokach & Maimon, 2014). According to this distinction, since my objective is to assign unlabelled instances into one of the two

possible classes regarding the Significant Loss variable (0 or 1), I can easily conclude that I am dealing with a classification problem, where supervised learning is frequently used (Oladipupo, 2010).

Among the several supervised machine learning algorithms available for classification problems such as Support Vector Machine, Naïve Bayes Classifiers, Decision Trees, Neural Networks (Oladipupo, 2010), I decided to apply Decision Trees, since I found this the most suitable for the company's problem due to its simplicity and self-explanatory nature, making its use and interpretation very accessible, even for non-data mining experts (Rokach & Maimon, 2014). Furthermore, this is a method with a fast learning process, contrarily to Neural Networks for instance (Podgorelec, Kokol, Stiglic, & Rozman, 2002).

2.4. DECISION TREES

2.4.1. Definition

Decision Trees are tree-shaped structures that are capable of classifying observations trough the analysis of its variables (Kotsiantis, 2007), representing *"one of the most promising and popular approaches"* (Rokach & Maimon, 2014, p. ix) for the data mining process.

With roots from logic, management and statistics, DTs are a very accessible yet accurate tool to perform prediction, since it allows to easily describe and interpret the relationship between all features and the target variable values. This model can be used both for classification and regression, being respectively called a CT or RT (Rokach & Maimon, 2014). In this project, as already stated, I am dealing with a classification problem, which is why I will rely on a CT as the one presented below.

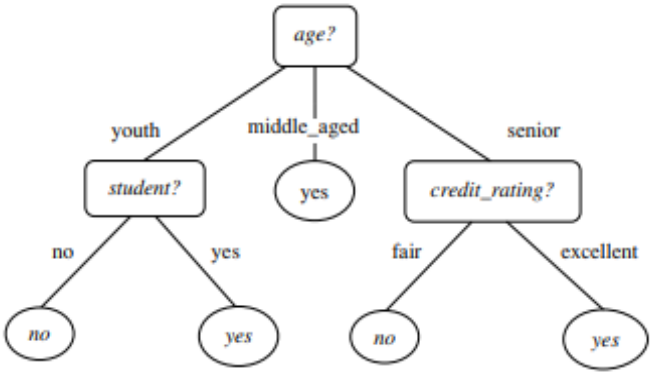

Figure 2.2: Example of a Classification Tree for the target variable *buys_computer* (Han, Kamber & Pei, 2012)

2.4.2. Advantages

The DT algorithm presents several advantages such as being simple and easy to interpret, due to its representation in the form of if-then rules, as well as its higher speed of classification, when compared to KNN and Naïve Bayes in classifying medium and large datasets (Jadhav & Channe, 2016). Additionally, despite being a straightforward method it can reach the same accuracy of the Back-Propagation Network when performing classification (Vafeiadis, Diamantaras, Sarigiannidis, & Chatzisavvas, 2015).

Considering the previously mentioned advantages I decided DT would be a good option to develop this project.

2.4.3. Method

In the DT algorithm the partitioning of data is usually achieved by successive partitions based on the values of the several explanatory variables, since each parent node will always be split and each child node will also become a parent node, unless it is a leaf (Moisen, 2008).

A tree always begins at the root node, being this the predictor variable that best splits the training data, according to the chosen method for finding this feature (Kotsiantis, 2007). The root node is the only with no incoming edges, while all the other nodes have exactly one incoming edge. All nodes that contain outgoing edges are called internal nodes, while the ones without outgoing edges are called terminal nodes or leaves (Rokach & Maimon, 2014). Each node corresponds to a variable of an instance, and each branch or edge represents a value or range of values that the variable can assume (Kotsiantis, 2007)**,** always taking into account that the range of values must be "*mutually exclusive and complete*" (Rokach & Maimon, 2014, p. 13), so that each instance of data can be mapped.

Each observation is classified starting at the root node and arranged depending on their predictor variables values (Kotsiantis, 2007) . That said, this algorithm starts by analysing the value of the feature that represents the root node, deciding to which branch this value belongs and moving on to the node that comes after the followed branch. This process is repeated until the algorithm reaches a leaf node (Rokach & Maimon, 2014)**.**

2.4.4. Splitting criteria

As previously mentioned, the root node as well as the remaining nodes are chosen by applying a method that identifies the variable that best splits the data**,** by minimizing each node impurity, which is the same as maximizing its homogeneity (Kotsiantis, 2007). In a classification problem, the two most common impurity-based methods are Information Gain and Gini Index. While the first one uses Entropy as the impurity measure, the second one measures the divergences between the probability distributions of the target variable values (Rokach & Maimon, 2014)**.**

According to the Information Gain method, the attribute with the highest value is chosen to split the node that is being analysed, since Information Gain represents the difference between the original information needed and the information needed if we split on a certain variable. This mentioned information represents the amount of information that is required to classify an instance in a certain partition and can also be named as entropy of that partition. By calculating the entropy of the original partition and the entropy after the partition using a specific attribute, we can calculate the Information Gain (Han, Kamber & Pei, 2012). That said, it is easy to conclude that the selected attribute will be the one that has the higher information gain that is obviously the one with the lowest entropy value.

In the Gini Index splitting rule the algorithm isn't measuring the information gain of the partition but rather its impurity. Since we want to minimize the impurity of each partition, the attribute that will be chosen to split a certain partition is the one that originates the maximum reduction in impurity, in other words: the one that has the minimum Gini Index. Using this splitting criteria means that the tree will be created by binary splits (Han, Kamber & Pei, 2012).

In the lack of a splitting criteria that has a better performance than the remaining ones, and taking into account that the method chosen won't make significant difference on the CT performance (Rokach & Maimon, 2014)**,** I decided to try both splitting rules in order to understand which would bring the best evaluation measures.

2.4.5. Overfitting

Since the nodes that result from the root are also divided into more nodes, as well as all the following ones, the tree can outgrow and reach the point where it reflects the training data, sometimes dividing one instance per leaf, resulting in overfitting (Moisen, 2008). Overfitting happens when the classifier is a perfect fit of the training data, not being able to generalize to instances that are not part of the training (Rokach & Maimon, 2014) and consequently returning poor predictions (Moisen, 2008)**.** According to (Kotsiantis, 2007, p. 252)**,** a DT overfits the training data if another tree exists with a larger error than the original one when tested on the training but a smaller error when tested in the whole dataset. Furthermore, the complexity of a DT has a huge influence on its accuracy and on its comprehensibility (Rokach & Maimon, 2014), meaning it is crucial to identify the most suitable tree size.

There are two ways of identifying optimal tree sizes and therefore preventing overfitting, namely: not splitting a node if a defined threshold for the reduction in the node impurity isn't reached or completely grow the tree and then prune unnecessary nodes until the it reaches an appropriate size (Moisen, 2008). While the first method represents a pre-pruning approach, the second one is called post-pruning. Although it is not possible to identify the best pruning method, pre-prune the DT by defining a stopping criteria to its growth can be recognized as the "*most straightforward way of tackling overfitting*" (Kotsiantis, 2007, p. 252).

2.4.6. Stopping criteria

A tree grows until a stopping criteria is reached. While rigid stopping criteria will lead to an underfitted DT, unconstrained ones create large and overfitted trees. Some of the most frequently used stopping rules are: the maximum tree depth is reached, the number of instances in a node is smaller than the minimum number of instances required for parent nodes, the split of a node makes the number of instances in one of the child nodes smaller than the minimum required number of instances per child node, and finally if the best splitting criteria doesn't reach the threshold for the reduction in a node's impurity value (Rokach & Maimon, 2014)**.**

2.4.7. Algorithms

According to (Rokach & Maimon, 2014), there are several DT algorithms, being ID3, C4.5 and CART the most common ones, all of them mostly differing in terms of splitting and stopping criteria. ID3 can be pointed out as the simplest of all above mentioned algorithms, relying on information gain as a splitting criterion and only stopping when all observations belong to a unique value of the target variable or the best information gain is equal or less than zero. This algorithm has several drawbacks, namely: not handling numeric features nor missing data and not using any pruning strategies, leading to overfitting. For all the previously specified disadvantages, and since the C4.5 algorithm can be considered an upgrade of the ID3, the C4.5 is preferred. This more evolved algorithm performs error-based pruning and handles both numeric and missing data. C4.5 uses gain ratio as a splitting criterion and only stops when the number of observations to split is below a defined threshold. The last of the three mentioned

algorithms, CART (Classification and Regression Trees) is known for creating binary trees, since each node has exactly two outgoing branches, being also able to generate regression trees since it supports numerical target variables. This algorithm uses Twoing, or prediction squared error in case of the regression trees, as splitting rules and performs Cost-Complexity pruning. From all available algorithms to create DT, C4.5 is considered the most popular in all literature (Kotsiantis, 2007). Although the algorithm available in scikit-learn is an improved version of the CART it can't deal with categorical variables, at least for now, which implies that all non-numerical variables of the provided dataset will need to be transformed.

2.5. MEASURING PERFORMANCE

After defining which algorithm I will rely on to create my CT, it is important to identify a set of measures that allow me to evaluate the predictive performance of the created classifiers, with the objective of determining which model is the best at predicting the class of the target variable. These evaluation measures are calculated not only using the training data but also the test data, since we want to measure the model's performance on new data, that was not used to create the classifier (Han, Kamber & Pei, 2012).

Among all the evaluation measures available to assess the predictive performance of CT, the most relevant are: accuracy, sensitivity, specificity, precision and f-measure (Rokach & Maimon, 2014). These evaluation measures are calculated using four simple variables, namely: True Positives, True Negatives, False Positives and False Negatives (Han, Kamber & Pei, 2012), whose meaning is presented in the table below (Table 2.1).

Measure	Meaning in this project
True Positives (TP)	Number of instances in which the loss was significant, and the Significant Loss variable was correctly labelled as 1.
True Negatives (TN)	Number of instances in which the loss was not significant, and the Significant Loss variable was correctly labelled as 0.
False Positives (FP)	Number of instances in which the loss was not significant, and the Significant Loss variable was incorrectly labelled as 1.
False Negatives (FN)	Number of instances in which the loss was significant, and the Significant Loss variable was incorrectly labelled as 0.

Table 2.1: Meaning of each measure

These four measures construct the Confusion Matrix (table 2.2), which helps in calculating all the initially referred evaluation measures that will be used to assess the model's performance.

		PREDICTED CLASS	
		Significant Loss = 1	Significant Loss = 0
ACTUAL CLASS	Significant Loss = 1	TP	FN
	Significant Loss = 0	FP	FN

Table 2.2: Confusion Matrix

2.5.1. Accuracy

The accuracy measure represents the percentage of instances that were correctly predicted by the model among all the observations (Kotsiantis, 2007), indicating how well the classifier identifies instances of both classes (Han, Kamber & Pei, 2012). Although usually model's evaluation is assessed mostly using the accuracy measure, it is not enough to perform a good evaluation, especially if the dataset has unbalanced classes' distribution (Rokach & Maimon, 2014).

2.5.2. Sensitivity and specificity

Since accuracy is a function of sensitivity and specificity (Han, Kamber & Pei, 2012), these last two measures can be used, instead of the first one, each time the dataset is unbalanced (Rokach & Maimon, 2014).

Sensitivity, also called TP rate, is the percentage of positive instances that are correctly classified (Han, Kamber & Pei, 2012). Therefore, this measure, that can also be named Recall, represents how good the model is in identifying instances that belong to the positive class, that is instances with a Significant Loss value equal to 1 (Rokach & Maimon, 2014). Consequently, in this project, sensitivity will return the number of instances correctly classified as a significant loss divided by all the actual observations that have a significant loss.

In its turn, Specificity or TN rate, returns the percentage of negative instances that are correctly classified (Han, Kamber & Pei, 2012), which represents how good the model is in identifying negative instances (Rokach & Maimon, 2014). Consequently, in this project, specificity represents the number of instances correctly classified as a not significant loss divided by all the actual observations that are not significant losses.

2.5.3. Precision and recall

Precision is an evaluation measure that returns the proportion of actual positive instances among all observation that were classified as positive (Rokach & Maimon, 2014)**,** being considered a measure of exactness (Han, Kamber & Pei, 2012). In this project it represents the number of instances correctly classified as a significant loss divided by all the observations classified as a significant loss.

Recall, as already stated, is the same as Sensitivity.

A score of 1 in the precision or recall measures is not necessarily a good indicator. Furthermore, since the relationship between both measures is inverse, allowing to increase one measure but at the cost of decreasing the other, a good approach to use these two measures is to combine them into a unique one which is called F-measure (Han, Kamber & Pei, 2012).

2.5.4. F-measure

The last evaluation measure I will use is called F-measure and represents the trade-off between Precision and Recall (Rokach & Maimon, 2014), since it returns the harmonic mean of these two measures, giving equal weight to both.

MEASURE	FORMULA
Accuracy	$\frac{TP + TN}{P + N}$
Sensitivity/Recall	$\frac{TP}{P}$
Specificity	$\frac{TN}{N}$
Precision	$\frac{TP}{TP + FP}$
F-Measure	$\frac{2 * precision * recall}{precision + recall}$

Table 2.3: Summary of each Evaluation Measure formula

2.6. CROSS-VALIDATION

To assess the evaluation measures of the classifiers I used 10-fold cross-validation. In this method the original data is randomly divided into 10 mutually exclusive sets with approximately the same number of instances and training and testing phases are carried out 10 times, using 9 of the 10 sets as training data and the remaining as test data for each one of the 10 iterations. By following this approach, we guarantee that each instance is used 9 times for training and 1 for testing, obtaining an accuracy that represents the number of correctly classified instances from the 10 iterations divided by the total number of instances (Han, Kamber & Pei, 2012).

3. METHODOLOGY AND TOOLS

Despite the lack of a standard framework to guide the deployment of DM projects (Wirth & Hipp, 2017), we can name more than one methodology for this purpose, such as SEMMA and CRISP-DM, which are considered the most popular since their common use in several publications regarding DM (Azevedo & Santos, 2008).

On the one hand, to present the core process of conducting a data mining project, we can mention SEMMA. According to the SAS Institute, responsible for the SEMMA development, the acronym stands for: Sample, Explore, Modify, Model and Assess, which represents the five main stages considered mandatory in a data mining project. On another hand, there is CRISP-DM, which stands for CRoss-Industry Standard Process for Data Mining, being a reference model for data mining which provides an overview of the life cycle of a data mining project (Wirth & Hipp, 2017), relying on six phases: Business Understanding, Data Understanding, Data Preparation, Modelling, Evaluation and Deployment (Azevedo & Santos, 2008).

Since SEMMA defends a more SAS related approach and because I personally consider CRISP-DM a more complete and business-oriented approach to Data Mining, I chose to guide my project using the last-mentioned methodology.

3.1. BUSINESS UNDERSTANDING

As it is supposed in the first phase of CRISP-DM, I started this project by understanding the business needs and the objective of the project from the company's perspective. Since the company has as goal the decrease in the number of losses, I think it is necessary to start by understanding the most common causes of this problem. In conversation with the enterprise I understood that losses represent a loss in a product that won't be sold anymore, whether because it was damaged, lost or donated, for example. Losses can occur due to several factors, which allow to classify each one as an identified or unidentified loss. For instance, rotten fruits are immediately classified by employees as an identified loss as well as donations made by the company. Conversely, any loss that is only discovered while performing the weekly inventory are categorized as unidentified, being mostly caused by thefts or errors caused by the distribution centre (the wrong quantity of fruit was sent to the store, for example).

As previously mentioned, the goal of this project is to reduce the number of losses in certain fruits in Pingo Doce's stores. To achieve this, my project plan is to implement predictive machine learning algorithms that will allow to predict if there will be a significant loss or not, from a set of explanatory variables.

3.2. DATA UNDERSTANDING

After understanding the project requirements from a business perspective and defining a preliminary project plan based on a data mining approach that allows to find a solution to the company's problem, it is necessary to have some understanding of the available data. Therefore, a deep analysis on all variables was performed, starting on a descriptive analysis, where each variable meaning is defined, and ending with a variable analysis, where both numerical and non-numerical variables were studied with the purpose of generally analysing the descriptive statistics of all variables within our dataset.

3.2.1. Descriptive Analysis

The excel file provided to me by the company contained data related to sales and stock movements as well as losses measurements, regarding the whole year of 2018, in which we could identify the following variables:

Variable	Variable Meaning
Loja	Number of the store that is being analysed
Loja Nome	Name of the store that is being analysed
Categoria	Number of the category to which the product that is being analysed belongs
Categoria Nome	Name of the category to which the product that is being analysed belongs
Artigo	Number of the product that is being analysed
Artigo Nome	Name of the product that is being analysed
Ano	Year in which the transaction occurred
Mês	Month in which the transaction occurred
Semana	Week in which the transaction occurred
Data	Date (day, month and year) in which the transaction occurred
Venda Quantidade	Amount of product sold, in units of measurement
Venda Valor PV	Value of the product sold in Euros (€)

Venda Valor Unitário Médio	Value of the product sold per unit of measurement
Venda Valor PC	Value of the product sold, without inclusion of the profit, in Euros (€)
Stock Quantidade	Amount of product available in the store, in units of measurement
Stock Valor	Value of the product available in the store, in Euros (€)
Stock Cobertura	Length of time that the product available in the store will last if current usage continues, in days
Quebra Identificada Quantidade	Amount of product that was considered an identified loss (Rotten fruit, donations, …)
Quebra Identificada Valor	Value that resulted from a products' identified loss
Quebra Inventario Quantidade	Amount of product that was considered an unidentified loss (stolen or missing fruit, …)
Quebra Inventario Valor	Value that resulted from the products' unidentified loss

Table 3.1: Meaning of each variable of the original dataset

3.2.2. Variable Analysis

After the initial data collection phase some first insights on the available data need to be discovered with the main purpose of identifying potential data quality problems. This can be made using a brief statistical analysis on the several existent variables, both numerical and non-numerical. Using the *info* function, I performed a global analysis to all variables, with the objective of understanding how many observations are available in the file provided, what are the variables contained in that file and its respective type, and also the number of non-null entries per feature, that will allow me to calculate the number of missing values for each variable. By observing the results, I could conclude that the dataset consists of 51612 observations, 21 variables whose type is mostly float64, but also int64, object and datetime64[ns]. Regarding the missing values, I was able to conclude that there are no missing values in any variable in our dataset, what will slightly facilitate the Data Cleaning phase.

3.2.2.1. Numerical data

To take a deep dive into the numerical variables I used the *describe* function which returns the minimum, maximum, mean, standard deviation and all the quartiles values of those variables. This first

approach allowed me to understand the range of values of each variable, before analysing each one individually.

Venda Quantidade

Through the analysis of the *Venda Quantidade* variable histogram I concluded that, although the quantity of product sold ranges between -2 and approximately 737, as I observed whilst applying the describe function, the big majority of products sold varies from 0 to less than 300. Furthermore, more than 50% of the observations correspond to a quantity of products sold between 3 and 24 units of measurement.

Venda Valor PV

The *Venda Valor PV* histogram is quite similar to the previous one, which makes sense since we are comparing the quantity of products sold to the respective value obtained from that sale. Therefore, despite its minimum value being less than -4€ and its maximum value superior to 614€, we can conclude that most values range between 0€ and less than 200€. Moreover, more than half of the observations result in a value obtained from the sale somewhat between 6€ and 37€, with only 25% of the observations ranging between approximately 37€ and 615€.

Venda Valor Unitário Médio

Regarding the variable that reflects the average price per unit of product sold, I noticed that although this value ranges between 1 and 2 in more than 50% of the observations, in some cases it assumes values superior to 3 or even below 0. Regarding the first case I didn't consider it a problem since it is possible that some fruits have a price per unit of measurement higher than 3€, but regarding the second ones I will analyse it afterwards since in my perspective it might represent an incoherence.

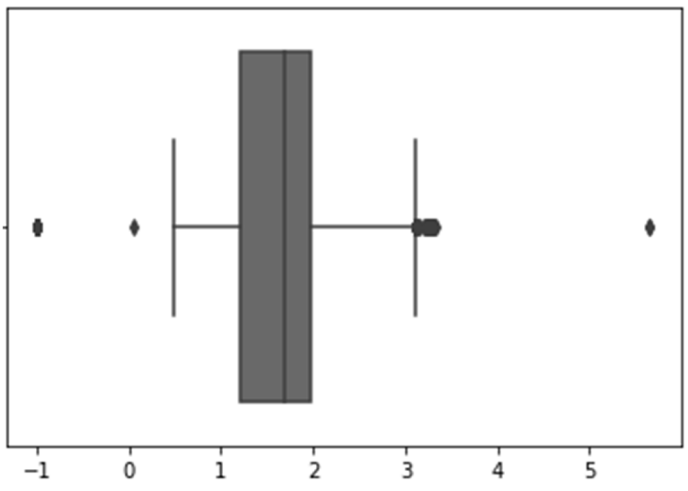

Figure 3.1: Boxplot of the *Venda Valor Unitário Médio* variable

Venda Valor PC

As expected, the *Venda Valor PC* histogram shape is close to the one of the *Venda Quantidade* variable and especially to the one of the *Venda Valor PV* variable, they are highly related. Therefore, by

observing the histogram I could conclude that the big majority of values ranges between 0€ and less than 200€, even though the minimum value for this variable being less than -2€ and its maximum value being higher than 580€. Additionally, it is relevant to point out that this variable's similar behaviour to *Venda Valor PV* from the fact that the first represents the value of products sold without inclusion of the profit (using the cost price of a product to calculate this variable) while the second one reflects the value of products sold with the profit (using the sale price of a product to obtain this variable).

	Venda Quantidade	Venda Valor PV	Venda Valor Unitário Médio	Venda Valor PC
Mean	28.979559	34.331719	1.665598	28.579946
Std	56.047947	50.859884	0.586387	47.703146
Min	-2	-4.320755	-1	-2.8
25%	3.902	6.735850	1.216822	4.669375
50%	9.693	16.839622	1.688785	11.996950
75%	23.67	36.879719	1.972176	27.9485
Max	737.155	614.915093	5.651615	582.35245

Table 3.2: Summary of the statistical analysis of the four sales related variables

Stock Quantidade

According to the information previously obtained after applying the describe function, the amount of product available at a store varies from approximately -6894 to 1797 units of measurement, with more than 50% of the observations ranging between 19 and 90. An interesting conclusion that can be highlighted is that, as can be observed in the histogram, the range of negative values is wider than the range of positives, regarding the stock quantity of products.

Stock Valor

As expected, the histogram of the variable that reflects the value of the amount of product available at a store is very similar to the one that reflects the amount. The minimum of this variable is less than -13000€ and the maximum is higher than 1689€, with more than half of the observations assuming a value between 23€ and 107€.

Stock Cobertura

This variable is calculated using the variables that measure the stock quantity, the value of sales at a cost price and the number of days of each month. Although it ranges between approximately -1743 and 6142, more than 50% of its values are somewhere between 2 and 7, meaning that in over 50% of cases the product available in the store will last between 2 to 7 days, assuming the current usage continues.

$$STOCK\ Cobertura = (STOCK\ Quantidade \div VENDA\ Valor\ PC) \times number\ of\ days\ of\ the\ month$$

Figure 3.2: Stock Cobertura formula

	Stock Quantidade	Stock Valor	Stock Cobertura
Mean	84.590561	87.769304	6.574669
Std	135.244148	132.198597	31.781437
Min	-6894.095	-13098.78	-1742.693311
25%	19.22	23.76	2.151183
50%	42.14	53.19	3.777128
75%	89.7005	106.31	6.983156
Max	1797.273	1689.43	6142.307692

Table 3.3: Summary of the statistical analysis of the stock related variables

Quebra Identificada Quantidade

The variable that measures the quantity of identified losses assumes values between -278 and 90. Besides that, through the analysis of the histogram it is easy to understand that most values range between -50 and 50, with more precisely 50% of the values ranging between -2 and -0,25. At least 75% of the observations present a negative value (between -278 and -0,25) for this variable, meaning that at least three out of four observations represent an identified loss.

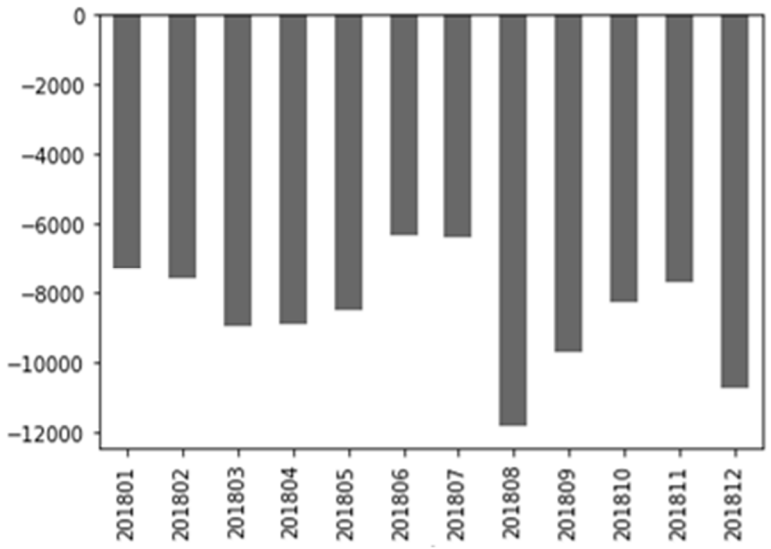

Figure 3.3: Sum of the quantity of identified losses, per month

Quebra Identificada Valor

Regarding the *Quebra Identificada Valor* variable, which measures the value originated by the quantity of identified losses, I can start by highlighting that it doesn't assume negative values since it has a minimum value of 0. Additionally, 75% of its values range between 0 and 2,25, with the remaining 25% varying between 2,25 and 26 280, representing the most expensive and therefore most significant identified losses.

Quebra Inventario Quantidade

While analysing the variable that measures the quantity of unidentified or inventory losses, I concluded that, although it ranges between less than -444 and more than 294, only less than 25% of the observations are below 0, representing an unidentified loss. Moreover, less than 25% of the observations range between 0 and approximately 295, representing moments in which the inventory was measured and there was more quantity of a product in the store that it was expected. The remaining more than 50% observations have a value of 0 for this variable, meaning that probably in more than half of the observations the inventory wasn't counted, which makes sense since it is supposed to be measures weekly.

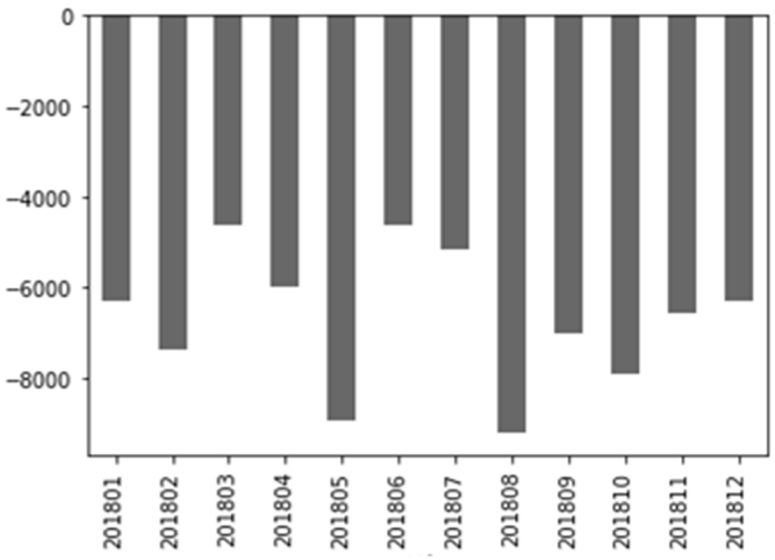

Figure 3.4: Sum of the quantity of unidentified losses, per month

Quebra Inventario Valor

The histogram of the *Quebra Inventário Valor* variable, as well as the information obtained from the use of the *describe* function, show that the value originated by unidentified losses ranges between 0 and aproximately 580, with at least 75% being equal to 0 and the remaining 25% assuming values between 0 and 579,6.

	Quebra Identificada Quantidade	Quebra Identificada Valor	Quebra Inventário Quantidade	Quebra Inventário Valor
Mean	-1.976849	6.793631	-0.625015	2.9635
Std	4.910775	254.186453	11.274358	12.970371
Min	-278	0	-444.71	0
25%	-2	0.31	0	0
50%	-0.92	1	0	0
75%	-0.25	2.25	0	0
Max	90	26 280.79	294.99	579.6

Table 3.4: Summary of the statistical analysis of the four losses variables

3.2.2.2. Non-numerical data

In order to assess the non-numerical variables, I applied the describe function, that returned the number of observations and the number of different categories for each variable, as well as the most observed category per variable together with its frequency (Table 3.5).

	Loja Nome	Categoria Nome	Artigo Nome
Count	51612	51612	51612
Unique	20	5	81
Top	Santo António dos Cavaleiros	Maça	Banana Importada Cor 4
Freq	3520	20548	4700

Table 3.5: Summary of the analysis of the non-numerical variables

Loja Nome

The provided dataset contains 20 different stores which were chosen by the company in order to reach all existent types of Pingo Doce stores. They were chosen among the more than 400 Pingo Doce stores existent in Portugal, in order to ensure that I would analyse a representative sample of all types of stores, both in terms of store size and location, as well as the usual quantity of customers.

Categoria Nome

By analysing the *Categoria Nome* histogram I concluded that the dataset has five different types of fruits, namely: *citrinos, maça, uvas, pera* and *banana*. The category with more observations is maça, with more than 20 000 observations, followed by citrinos with approximately 10 000 instances and uvas, pera and banana, with more than 5 000.

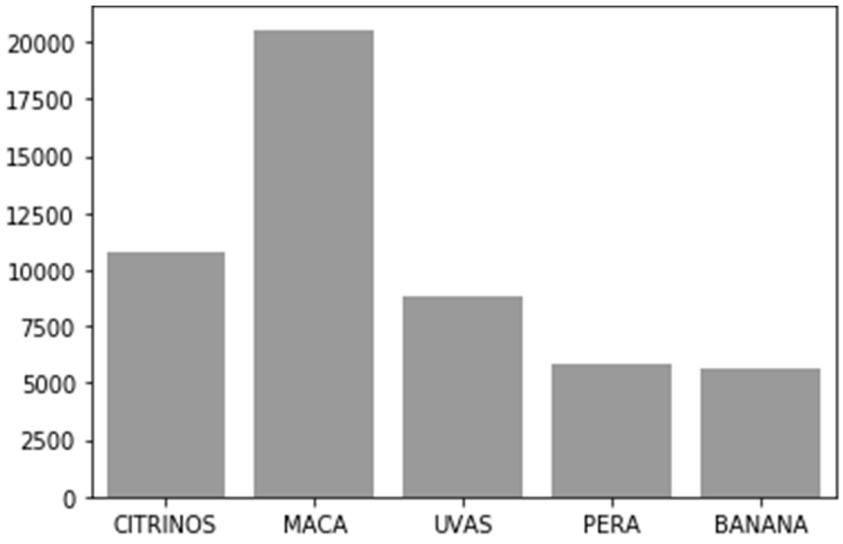

Figure 3.5: Number of observations per category of product

Artigo Nome

The provided dataset contains 81 different products, all distributed among the 5 categories. I analysed the number of observations per product name, obtaining the top 10 of the most observed products (Table 3.6).

Product Name	Count of Observations
Banana Importada Cor 4	4700
Laranja Cal 3/4/5	3809
Uva Red Globe	2791
Limão Cal 3/4	2464
Maçã Royal Gala 70/75	2423
Pera Rocha 65/70	2315
Maçã Fuji	2194
Maçã Golden Alpes Itália	2124
Maçã Reineta Parda 75/80	1923
Uva Branca	1482

Table 3.6: Top 10 of the most mentioned articles

Ano

As expected, the variable that represents the year in which the transaction being analysed occurred only assumes one value, namely “2018”, since I only received data regarding the sales, stocks and losses movements that happened in the year of 2018.

Mês

By analysing the histogram of this variable, I could measure the number of observations that exist per month and conclude that December is the month in which more transactions occur, whether they are related to sales or losses measurements, followed by March and April. July was the month with less movements, followed by the remaining summer months (June, August and September).

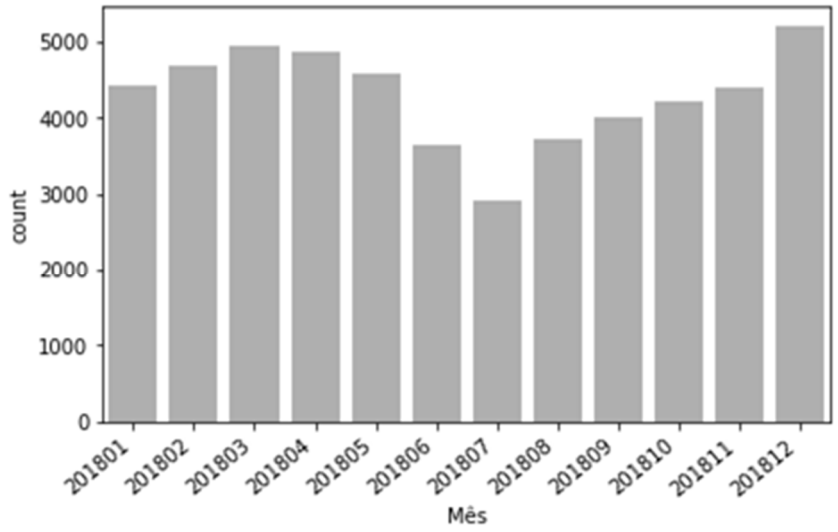

Figure 3.6: Number of observations per month

Semana

The variable whose histogram reflects the transactions per week shows that it varies greatly depending on the time of the month and the time of the year. For instance, I can clearly notice a pattern in which approximately one week per month has a lot more observations than the three weeks that precede and succeed the referred week. Also, as in the month variable histogram, I can recognize a decrease in the number of movements in the summer months.

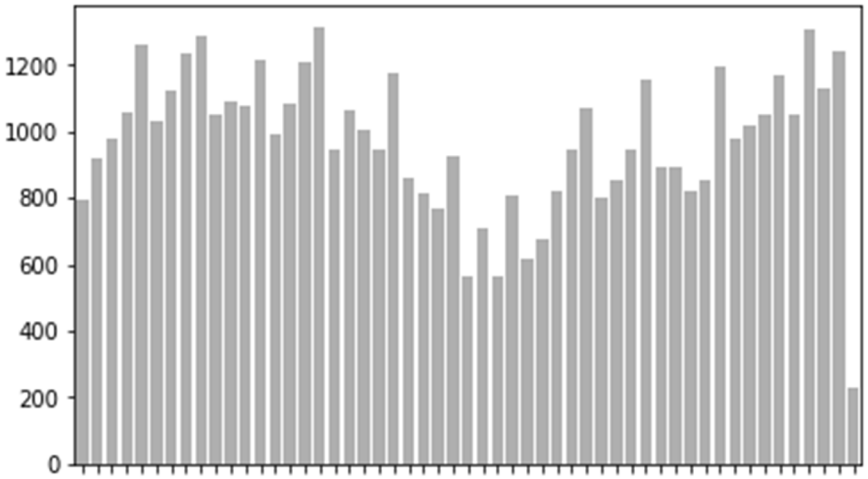

Figure 3.7: Number of observations per week

Data

The histogram of the *Data* variable reflects the movement of products, namely sales and losses, per day. Through its analysis I was able to identify at least eleven days in which the number of movements was much higher than usual, what might indicate the end of each month. I could also notice, as in the histograms related to the month and week, a decrease in the number of movements in the summer period.

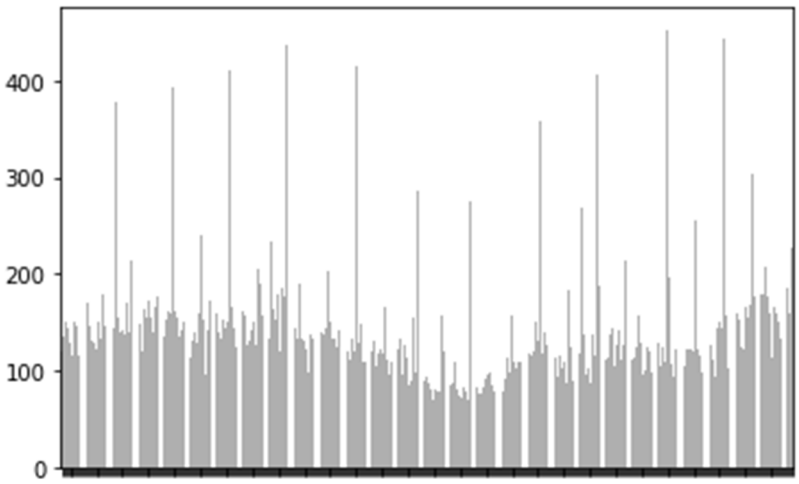

Figure 3.8: Number of observations per day

3.3. DATA PREPARATION

After exploring all variables, data needs to be processed with the goal of converting it into high-quality data that will lead to the creation of high-quality models. Due to the existence of incomplete, noisy and incoherent data this is a fundamental stage of the Data Mining process, taking approximately 80% of the total process effort. Among the several possible tasks to perform in the data preparation phase, I performed the following: Data Import, Data Cleaning, Data Transformation and Data Reduction.

3.3.1. Data import

I started by importing the Excel file provided by the company to Spyder, since I am using Python 3.6.5 to develop this project. Using the pandas *read_excel* function I obtained a DataFrame with 51 612 rows, corresponding to the number of observations, and 21 columns, which corresponds to the number of features available.

3.3.2. Data cleaning

After importing the file and consequently creating the Dataframe, it is important to proceed to the Data Cleaning phase, that consists in a set of crucial steps such as removing duplicated observations and identifying and dealing with missing values, outliers and incoherent data, that will allow the algorithm created to have a better performance.

3.3.2.1. Removing duplicated observations

To remove any possible duplicated rows in the original dataset, although in a very broad look across the Excel file it didn't seem that there would be any duplicated entries, I used the drop_duplicates function that returns only the dataframe's unique values. After applying it, the DataFrame kept having the same number of rows, wherefore I could conclude that as expected there were no duplicated rows.

3.3.2.2. Missing values

In the Data Understanding chapter, more precisely in the Variable Analysis phase, I concluded that there were no missing values in any variable. To be sure of this assumption, I applied the *isnull* function,

that returns the number of null entries for each variable, allowing me to confirm that there are no missing values in any variable of the dataset. Therefore, I don't need to make any changes to the original dataset.

3.3.2.3. Outliers

An outlier consists of an observation whose value is either much higher or much lower than most of the data, indicating that it may be some type of mistake. The big concern when dealing with outliers, is that, on one hand, altering them can lead to a better performance of the models, but on the other hand, they can be legitimate observations and sometimes the most relevant and interesting ones. That said, it is very important to understand the nature of an outlier before deciding how to act, once they shouldn't be removed unless proven to be erroneous data.

I started by visually analysing the number of outliers in the original dataset but I found it a bad approach since, for instance, the quantity of sold products varies a lot between categories, namely products that belong to the Bananas category are sold a lot more than any product belonging to the Citrinos category. That said, I divided the original dataset into five different datasets, one per each category of product, namely: Banana, Citrinos, Maca, Pera and Uvas. First, I used the Z-Score method to identify outliers as observations that stand more than 3 standard deviations away from the mean of each variable. By using this approach, I obtained the following number of outliers, per category and per variable:

	Banana	**Citrinos**	**Maçã**	**Pera**	**Uvas**
Venda Quantidade	76	197	411	99	169
Venda Valor PV	76	207	387	88	148
Venda Valor PC	72	211	392	93	164
Venda VUM	2	31	181	60	29
Stock Quantidade	57	165	89	92	134
Stock Valor	18	156	99	88	154
Stock Cobertura	1	128	197	58	4
Quebra Identificada Quantidade	109	196	178	97	187

Quebra Identificada Valor	7	7	20	4	3
Quebra Inventário Quantidade	117	200	238	108	201
Quebra Inventário Valor	194	192	231	103	163

Table 3.7: Number of outliers using the Z-score method, grouped by variable and category of product

After observing the results, I decided to take a closer look at the outliers, per variable, by analysing the respective boxplots, with the purpose of looking for the most significant outliers and also understanding which variables are most likely to contain wrong values, generated by human error, since I defined that only in that case an outlier should be removed. That said, I decided to remove outliers only on the two following variables:

1. *Venda Valor Unitário Médio*

I transformed the negative values, always equal to -1, to the median of the variable, per category, creating a new variable: *Venda VUM*.

2. *Stock Quantidade*

I consider this variable requires a lot of human input, what can cause several measure errors and, therefore, I decided to remove the most significant outliers. By observing the boxplot of this variable, I decided to replace its outliers with the median of the variable, also per category obviously, creating a new variable: *Stock Qt*.

Regarding to the outliers of the remaining variables, which didn't suffer any change I can highlight a few reasons that led me to that decision, namely:

- *Venda Quantidade*

Initially I was determined to remove the outliers identified in the Citrinos, Maca and Uvas categories, due to their high number in comparison with Banana and Pera categories outliers. My idea was to remove any observation in which this variable assumed values higher than 300, 150 and 100, regarding the categories Citrinos, Maca and Uvas respectively, but after analysing the dates in which these observations occurred I concluded that the identified outliers correspond to Saturdays, Sundays or Tuesdays, which precisely correspond to weekends and to the first day of promotions of each week in the Pingo Doce stores. This led me to assume that probably the identified observations don't correspond to errors but rather to correct values of sales on days in which customers are more likely to purchase these products.

- *Venda Valor PV* and *Venda Valor PC*

Since these variables are a product of the Venda Quantidade variable together with prices (PV and PC) that are automatically loaded into the company's database, I found it hard to contain any wrong value.

- *Quebra Identificada Quantidade*, *Quebra Identificada Valor, Quebra Inventário Quantidade* and *Quebra Inventário Valor*

Although these are the variables with the higher number of outliers, I decided to maintain all observations regarding these four variables, since they are the most relevant ones to this project and removing them might lead to loss of important information.

3.3.2.4. Coherence checking

While analysing the dataset looking for something that seemed illogical, I found several possible incoherencies among the data provided, namely:

1. *Venda Valor PC* > *Venda Valor PV*

I created a flag variable that assumes a value of 1 if the value of the variable *Venda Valor PC* is higher than the one of the *Venda Valor PV* variable, what means that the cost price of certain product was higher than the price it was being sold to the customers. I concluded that there are 1 759 observations in which this condition is observed. Although this might seem illogical, I didn't consider it an incoherence since when a product is on sale it might be sold below its cost price. That said, I decided to maintain those 1 759 observations.

2. *Stock Cobertura* and *Stock Qt* have different signals

I created another flag variable that has a value of 1 if the variables *Stock Qt* and *Stock Cobertura* have different signals, being one positive and the other negative. This repreents an incoherence since logically, if the quantity of stock is positive then its coverage would still be positive and analogously, if the first is negative then the second should be negative. After analysing the flag variable created, I identified 158 observations in this situation. To understand this, I analysed the *Stock Cobertura* formula, that relates the *Stock Qt* variable with the *Venda Valor PC* variable and concluded that this happens when the second variable (*Venda Valor PC*) assumes negative values or is equal to zero. Since it is not possible to divide by 0, the company decided that when *Venda Valor PC* is equal to zero, the *Stock Cobertura* variable takes automatically the value of -1. Therefore, since this was the best way found by the company to deal with this situation, I decided to maintain these observations, not considering it an incoherence.

3. *Venda Quantidade* <= 0

The third flag variable I created has the objective of identifying the observations in which the variable *Venda Quantidade* assumes non positive values. I considered this an incoherence since from my perspective the company can only sell positive quantities of products. After analysing the flag variable, I concluded that there are 275 observations in which this happens. There are 9 observations in which the quantity of product sold is negative and 266 in which it is equal to zero. By performing a deep analysis of the dataset, I understood that in all observations with zero quantity of product sold, at least one of the variables regarding the identified or unidentified loss quantity is filled with a value different than zero. That said, I concluded that these observations, although not representing any sales information, are representing a moment in which the losses (identified or not) were measured. Therefore, I decided to maintain the 266 observations with value equal to zero. Regarding the 9 observations with negative values, I considered it an incoherence and since all those values range between 0 and -2, being little significant, I decided to change the negative values to zero, creating a new variable *Venda Qt*, that will be used instead of the *Venda Quantidade* variable from now on.

4. *Venda Valor Unitário Médio* <= 0

Another flag variable I created signals whether the variable *Venda Valor Unitário Médio* assumes values that are negative or equal to zero. Trough the analysis of the count values of that variable, and as I already knew from the Outliers analysis, I could identify 266 cases in which that happens. Analogously to the previous incoherence I understood that this happens in all the observations in which the quantity of product sold is equal to zero. Since it is not possible to divide by zero, and since the variable *Venda Valor Unitário Médio* is equal to the division of the *Venda Valor PV* with the *Venda Quantidade*, I concluded that once the calculation couldn't be performed, the company decided to automatically fill the *Venda Valor Unitário Médio* variable with the value -1. This incoherence was already solved in the Outliers phase, while replacing the variable *Venda Valor Unitário Médio* for the *Venda VUM* variable, in which the values equal to -1 were replaced by its median value, depending on the category of the product that is being analysed.

Lastly, I find it important to highlight a few interesting facts I discovered while looking for possible incoherencies:

- *Quebra Inventário Quantidade* > 0

It might happen when the stock is measured by an employee, as it is supposed to happen at least weekly, and is obtained a higher stock quantity than the one that was expected, considering the information on previous measures of stock and sales.

- *Quebra Identificada Quantidade* > 0

This might happen each time a store receives transferred products from other stores, per example. There are only 30 observations in this condition, but I didn't consider it an incoherence, thus not making any changes to the dataset.

3.3.3. Data transformation

In this phase I used the Z-score standardization method to rescale data and created new variables, through both the interaction between previous existent ones and by using new features that were not being measured in the original dataset. Finally, I transformed all categorical variables, since the algorithm available in scikit-learn can't deal with it, obtaining a new set of dummy variables.

3.3.3.1. Statistical transformations

Since the dataset has several variables with values in different ranges as well as different measurement units, I found it important to rescale all numerical variables, by bringing all columns in the dataset to a more similar range. Due to the remaining existence of outliers in all variables, I decided to use the Z-score standardization method, although this technique doesn't produce data with the exact same scale, as happens in the Min-Max normalization. I also considered using the Log transformation but due to the several negative and equal to zero observations it was not possible. That said, I standardized all numerical variables, obtaining new ones with equal means and standard deviations, equal to 0 and 1 respectively, but with different ranges.

3.3.3.2. New variables

I have used two approaches to create new variables: interaction between variables already existent and creation of new features based on interesting characteristics of the data that were not being analysed in the original dataset.

Interaction between variables

Through the interaction between the variables that were already part of the original dataset, I was able to create some new interesting features that might be useful for building the classifier (table 3.4).

Variable	Formula
Quebra Total	Sum of the two variables that measure the quantity of identified and unidentified losses
Venda/Quebra	Division of the *Venda Qt* variable by the *Quebra Total* variable
VendaPV/VendaQt	Division of the *Venda Valor PV* variable by the *Venda Qt* variable
VendaPV/QuebraTotal	Division of the *Venda Valor PV* variable by the *Quebra Total* variable
Margem	Difference between the *Venda Valor PV* and *Venda Valor PC* variables
Venda/Stock	Division of the *Venda Qt* variable by the *Stock Qt* variable

Table 3.8: Summary of the new variable's formula

Creation of new variables

I also created new variables, mostly dummy ones, considering interesting characteristics of the municipality to which each store belongs as well as the season of the year and the day of the week in which an observation occurred. All the created variables are explained below:

Criminality Index

I have created a variable that represents the criminality rate (number of crimes per 1000 inhabitants, regarding the year of 2018, according to ("Statistics Portugal - Web Portal", 2019)) of the municipality to which each one of the 20 stores belongs. Therefore, the model will be able to comprehend the influence of the criminality rate on the existence of a bigger number of losses. I found it interesting to create this variable since theft is one of the major causes pointed by the company to the high number of Unidentified/Inventory losses.

Loja Nome	Criminality Index
Arneiro	31.4
Canidelo – Lavadores	30.3
Carnaxide	24.6
Damaia	34
Espinho	33.6
Estados Unidos da América	75.6
Faro – Tridente	49.9
Gaia – Av. República	30.3
Loulé – Andrade de Sousa	53
Oliveira do Hospital	15.2
Ourique	28.1
Pombal – Barco	24.4
Portimão – Craveiros	44.8
Queijas	24.6
Ramalde – S. João de Brito	74.3
Samora Correia	38.7
Santo António dos Cavaleiros	28.3
Serpa Pinto	74.3
Tavira EN	41.6
Vale do Lobo	53

Table 3.9: Criminality Rate of each store

Creation of dummy variables

1. Climate Class

According to the Köppen-Geiger climate classification, the most frequently used climate classification system (Kottek et al., 2006), continental Portugal can be divided into two types of climate: Csb and Csa (Instituto de Meteorologia de Portugal & Agência Estatal de Meteorologia de Espanha, 2011). Thus, I created the variable Climate Class that, according to the store location, returns the value 1 if it belongs to a *Csa* climate zone or 0 if the store is in a *Csb* climate zone. This information might be useful since, if the fruits are not correctly stored in a refrigerated location in each store, the stores that located in a specific climate type can be more prone to a faster rotting of the products.

2. Semana Category

I created a flag variable that categorizes the weeks depending on their position on each month. I did this by assigning all the weeks to two different categories that assume the values 1 and 0, depending on whether it is an extreme week or a middle week, respectively. While the first case contains the weeks that are in the beginning or in the end of each month (containing the first or the last day of a month), the second contains the remaining weeks.

3. Above AVG Quebra Mensal/ Above AVG Quebra Semanal/ Above AVG Quebra Categoria

In order to create the target variable, I had to create 3 flag variables that signal if an observation has a total loss value higher than the average for the month, week and product category that is being analysed.

I started by calculating the average values per month of the *Quebra Total* variable, obtaining the average total loss per month. After, I used the obtained values to create the *AboveAVGQuebraMensal* variable, that assumes the value of 1 if the total loss of an observation is higher than the average of that month and 0 if not. It is important to notice that in this case, higher will always mean “smaller than”, since the quantity of losses is measured in negative values, meaning that the more quantity of losses, the more negative the *Quebra Total* variable will be.

Analogously, to create the *AboveAVGQuebraSemanal* variable I calculated the average values of total losses per week and used those values to create this variable that returns a value of 1 if the total loss of the observation is higher than the average of the corresponding week and 0 otherwise.

Lastly, and as in the previous two variables, I calculated the total loss average per category of products and used those values to create the *AboveAVGQuebraCategoria* variable that assumes the value of 1 if the total loss of the observation is higher than the average of that category and 0 otherwise.

Category	Quebra Total (Average)
Banana	-7.242859
Citrinos	-2.128615
Maça	-1.462564
Pera	-1.944239
Uvas	-3.296596

Table 3.10: Average of the *Quebra Total* variable, per category of product

4. Seasons

Considering the meteorological seasons of the year in the north hemisphere ("Épocas / Estações do ano", 2019), I created a feature that assigns each month to its respective season. Therefore, March, April and May belong to the spring; June, July and August to the summer season; September, October and November to the autumn and December, January and February to the winter.

5. Day of Week

By using Pandas' *dt.weekday_name*, I created a variable that assigns every date to its respective day of the week, namely: Monday, Tuesday, Wednesday, Thursday, Friday, Saturday or Sunday. Additionally, I found it relevant to highlight when a date corresponds to a national holiday in Portugal, since it is expected to have a higher number of sales on those days, which I did by analysing the list of national holidays in Portugal for the year of 2018 ("Calendário 2018", 2019) and changing the day of the week obtained by the Pandas function for "Holiday".

6. Significant Loss (Target Variable)

Since I will use supervised ML algorithms, the dataset needs to contain a target variable that, being this a classification problem, must take one of two possible values in order to belong to one of two possible categories. The target variable on this project is called *Significant Loss*, meaning that if the variable is 1 the observation is a significant loss and if it is 0 then it is not. I defined that an observation must be considered a significant loss if and only if the value of the *Quebra Total* variable is greater than the average total loss of that month, week and category. That said, I obtained 11 264 observations that are considered significant losses among the 51 612 observations available in the dataset.

7. Above Weekly Avg Quebra Inventário

I could identify 10 926 observations with a value different than zero on the Quebra Inventário Quantidade variable, what means that a loss happened but was only discovered while filling the inventory. While analysing the graph that represents the spread of these 10 926 observations among all the weeks in the dataset (figure 3.4), I could identify some patterns probably related to the end and beginning of the month. That said, I decided to create a flag variable that assumes the value 1 if a week has more than the average value of different than zero observations on the Quebra Inventário Quantidade variable and 0 otherwise. Since the average number per week of observations with a value different than zero in the *Quebra Inventário Quantidade* variable is approximately 206, I could flag 18 weeks that have more observations than the average.

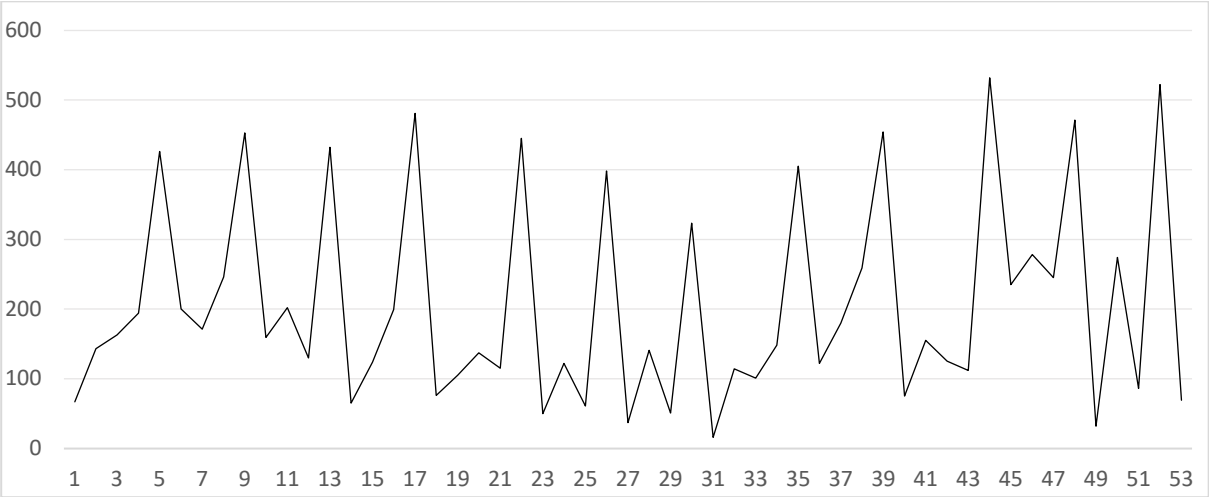

Figure 3.9: Number of instances with an unidentified loss quantity different than zero, per week

8. Above VUM

I have created a flag variable that compares the *Venda VUM* value for each observation with its average per product, with the objective of understanding in which observations a product is being sold in a lower or higher than average price per unit. This flag variable returns a value of 1 if the *Venda VUM* value is above the average and 0 otherwise. After creating this variable, I could conclude that among the 51 612 observations, 29 633 represent observations in which the *Venda VUM* value is higher than the average value for that product, while the remaining represent instances in which the product is likely on sales.

Changing categorical to dummy variables

Another important step was transforming all categorical variables, such as *Categoria Nome, Seasons, Day of Week* and *Loja Nome*, into a set of dummy variables, since the *DecisionTreeClassifier* cannot handle categorical features. Therefore, creating dummy variables, using the pd.*get_dummies* function was the best way I found to solve this issue, since it creates a new column for each value of the categorical feature that is being transformed, filling each row with 1 if the row contained that column's value and 0 if not. I didn't perform this transformation on the *Artigo Nome* variable, not only due to the fact that it would create 81 new features, given the high number of different products that are being studied, but also due to the fact that in a later phase I will decide not to use this variable, since it is highly correlated with *Categoria Nome*.

3.3.4. Data reduction

3.3.4.1. Choosing relevant variables

Before checking for correlation and applying PCA, I chose the most relevant variables from both the already existing ones and those created, removing a set of variables I considered irrelevant (Table 3.11).

Removed variable	Reason for removal
Loja Nome, Loja, Categoria Nome, Categoria, Artigo Nome, Artigo	I will use the dummy version of these variables
Ano	It has the same values for all observations
Mês	I will use the *Seasons* variable instead
Semana	I decided to use the *Semana Category* variable instead
Data	I decided to use the *Day of Week* variable instead

Table 3.11: Removed variables and respective reason for removal

I also removed all non-standardized variables and used the ones obtained after applying the Z-Score method. Lastly, I removed the remaining categorical variables (Seasons and Day of Week), maintaining their dummy version created on the previous phase of this project. Consequently, I obtained a dataset with only numerical variables.

3.3.4.2. Dealing with correlated variables

A data mining model shouldn't be fed with correlated variables, since it can overemphasize one data component or even create and unstable model that originates uncertain results. Therefore, in order to prevent this from happening and with the objective of creating a faster and more interpretable model, I analysed the correlation between all the relevant variables that remained in the dataset. To do this I started by observing the correlation matrix (Figure 3.9), which allowed me to detect the variables that seem to be correlated.

After a brief analysis of the correlation matrix I applied the *corr* function to the transformed dataset to assess the coefficient of correlation between each variable, with the objective of identifying the ones with a coefficient above than 0,7, since it was the threshold that I chose to use for removal. By analysing the results, I identified the following variables as involved in correlations above the defined threshold: *Venda Qt, Venda Valor PV, Venda Valor PC, Stock Qt, Stock Valor, Banana, VendaPV/QuebraTotal, Venda/Quebra, Quebra Total, Quebra Inventário Quantidade, Quebra Identificada Quantidade, AboveAVGQuebraSemanal, AboveAverageQuebraMensal* and *AboveAVGQuebraCategoria.* Therefore, I decided to remove the following variables: *Venda QT, Venda Valor PV, Venda Valor PC, Stock Qt, VendaPV/Quebra Total, Quebra Total, AboveAVGQuebraSemanal, AboveAVGQuebraMensal* and *AboveAVGQuebraCategoria.*

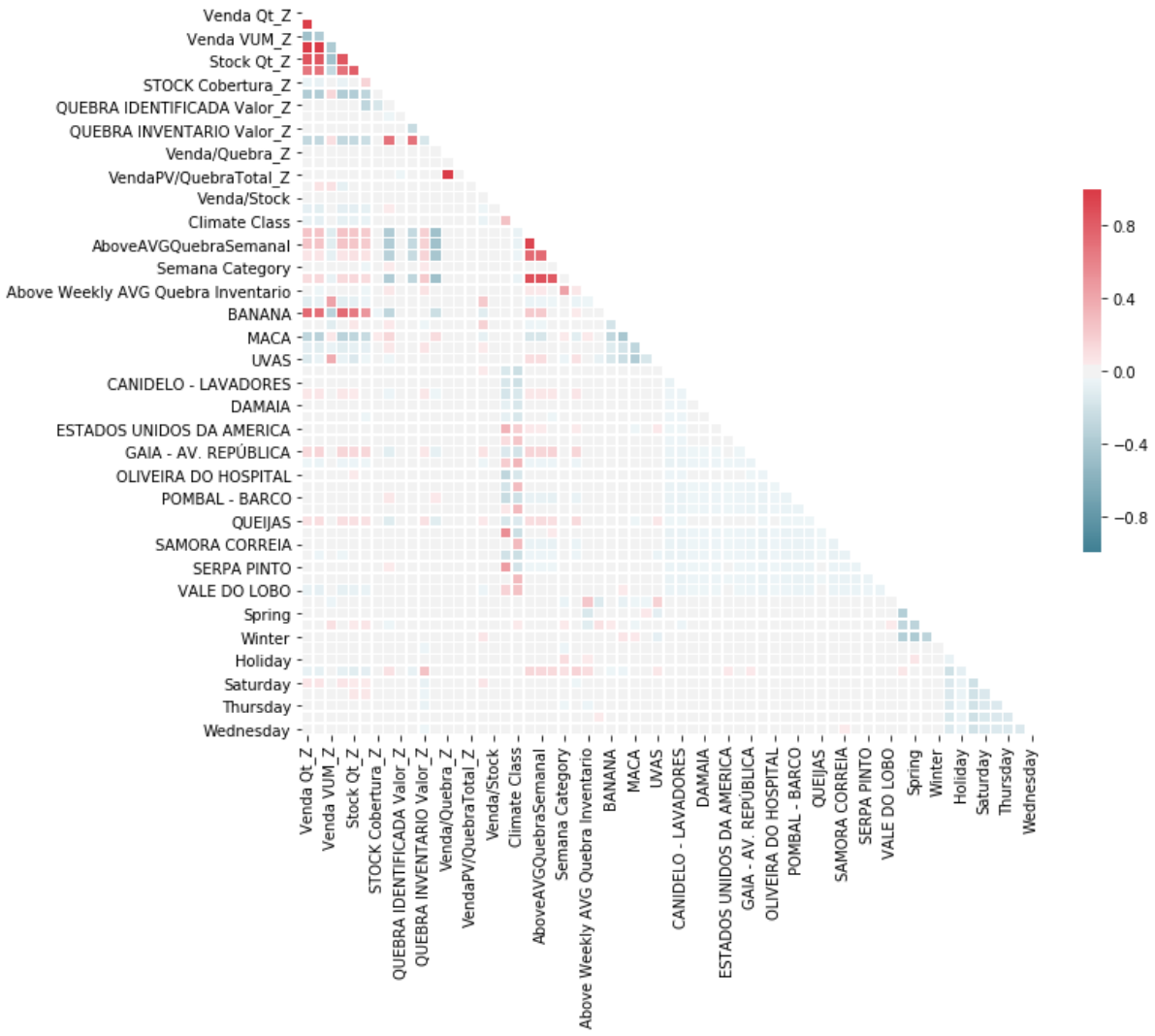

Figure 3.10: Correlation Matrix of the transformed dataset

3.3.4.3. PCA

After dealing with correlated variables I applied Principal Component Analysis, a dimensionality reduction algorithm, that calculates a projection of the original data into fewer dimensions. I did this with the main objective of creating a smaller set of components that are completely independent from each other and that summarize the original dataset. The main question is how many components to choose while applying the PCA. In order to answer to this question, I analysed the plot of the Cumulative Sum of the Explained Variance, concluding that four components are enough to explain almost 100% of the dataset that previously contained 53 variables, excluding the target variable. Therefore, I applied PCA and obtained a dataset to which I called *tese_PCA* with only four input variables.

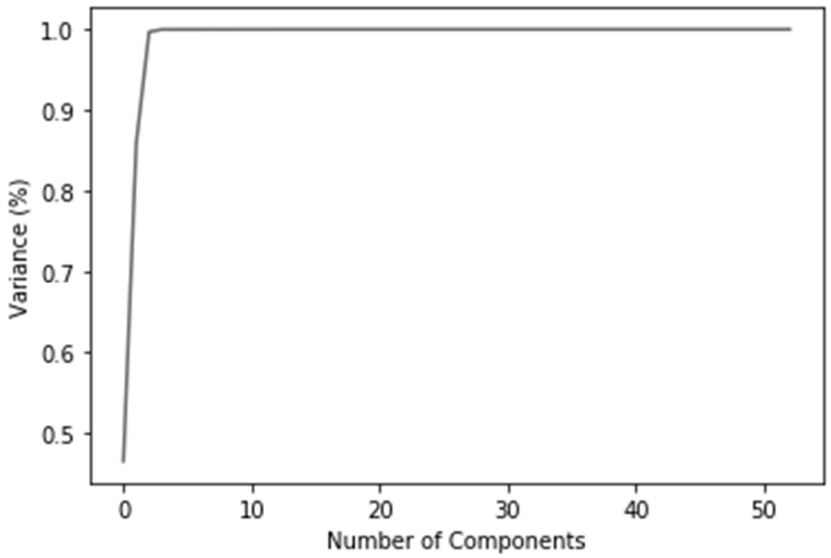

Figure 3.11: Plot of the Cumulative Sum of the Explained Variance per Number of Components

In addition to creating this dataset, I decided to maintain the previous one, without applying PCA, that represents a dataset with 53 no correlated variables to which I will refer as *tese_NoCorr* from now on. I understand that it contains a huge number of input variables, but I wanted to build a classifier from a dataset without PCA, in order to maintain the interpretability of each feature.

Lastly, I decided to create two additional datasets: *tese_ChosenVar1* and *tese_ChosenVar2*, that were formed by the removal of a few variables from the *tese_NoCorr* dataset. I decided to create these datasets in order to be able to infer losses without any information regarding previous losses, with the obvious exception of the target variable. That said, to create the *tese_ChosenVar1* dataset I removed all variables that contained information directly linked to the quantity and value of previous losses, maintaining some variables regarding stocks and sales. To create the *tese_ChosenVar2* dataset I also removed the features that measured sales and stock quantities and values, only maintaining the ones with information regarding the store's characteristics, category of products and time of the week, for instance. In my opinion, although this last dataset is likely to create less accurate results, it could represent the most actionable classifier, since it could easily be applied by the company.

Dataset	Number of Input Variables	Correlated removed	PCA applied
Tese_NoCorr	53	Yes	No
Tese_PCA	4	Yes	Yes
Tese_ChosenVar1	46	Yes	No
Tese_ChosenVar2	41	Yes	No

Table 3.12: Summary of the four created datasets

Tese_ChosenVar1	Tese_ChosenVar2
Venda VUM_Z	Criminality Index
STOCK Cobertura_Z	Climate Class
VendaPV/VendaQt_Z	Semana Category
Margem_Z	Above VUM
Venda/Stock	Categoria Dummies
Criminality Index	Loja Dummies
Climate Class	Seasons Dummies
Semana Category	Day of week Dummies
Above VUM	
Categoria Dummies	
Loja Dummies	
Seasons Dummies	
Day of week Dummies	

Table 3.13: Detail on the input variables of the tese_ChosenVa1 and tese_ChosenVar2 datasets

3.4. MODELING

After performing data preparation, transformation and reduction I can now create and apply the models that I hope will be capable of predicting significant losses in the analysed products. As already explained, I decided to apply Decision Trees, since I found this the most advantageous algorithm for the company to understand and further apply. I started by dividing the datasets into training, consisting in 70% of data, and testing, which consisted in the remaining 30%. Due to the several drawbacks of this spilt method, I also decided to perform 10-fold cross validation while looking for the best parameters of each DT, which means that the data was divided into 10 different subsets, using 9 to train the model and the last as test data.

Due to the high interpretability of the DT algorithm, I believed it would be of great interest of the company for me to apply it to this problem. Therefore, I used the DecisionTreeClassifier class from scikit-learn, that uses an optimised version of the CART algorithm to generate a model that predicts the class of the *Significant Loss* variable, by learning simple decision rules inferred from the data features.

Initially I used the default values for all parameters, in order to understand what results would be obtained, but I understood that I needed to limit some parameters in order to avoid overfitting to the

training data. That said, I implemented *GridSearchCV*, which returned the best values for the parameters I found relevant, namely: *criterion, max_depth, min_impurity_decrease, min_samples_leaf* and *min_samples_split*, and then applied the selected parameters to each corresponding dataset. The values tested using the *GridSearchCV* were defined by me depending on what made sense to use on this problem. Therefore, I used *gini* and *entropy* as options for the splitting criterion; 3, 4 and 5 as possible values for the maximum depth parameter since I didn't want to obtain a tree with too many levels; 50, 100, 200, 400 and 500 as options for the minimum number of instances required to split a node; 10, 50, 100 and 150 as the minimum observations required to be at a leaf and 0, 0.01, 0.02, 0.05, 0.1 and 0.15 as the minimum decrease in impurity required for splitting a node. It is also important to mention that I defined the *class_weight* parameter as *balanced* due to the ratio between observations with *Significant Loss* equal to 0 and 1, since approximately only 22% of the dataset had a value of 1 for the mentioned target variable.

After creating the DT classifier, I thought it might be useful for the company to be able to visualize it. With that in mind, I used the scikit-learn *export_graphviz* function, as well as the Python module *pydotplus*, to generate the visual representation of the classifiers obtained from the analysis of the three datasets without PCA. I didn't apply this to the remaining dataset since it wouldn't be interpretable for the company due to the new variables originated after performing PCA.

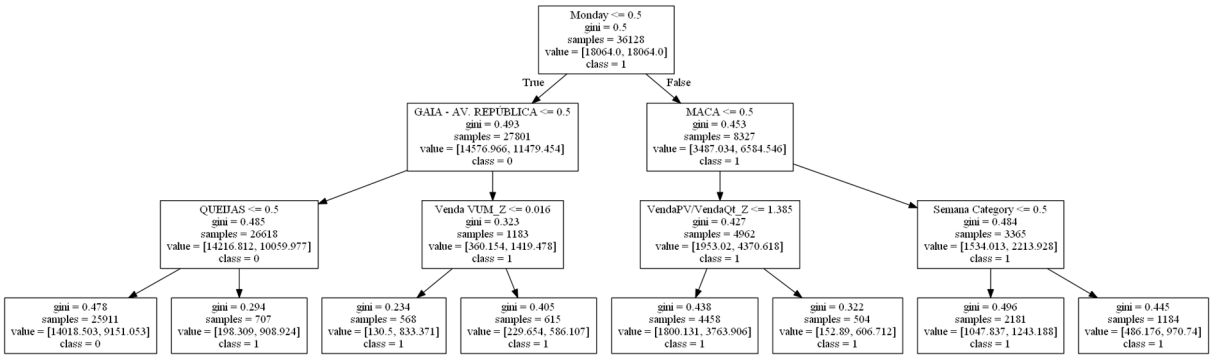

Figure 3.12: DT built from the ChosenVar1 dataset, with maximum depth of 3

Above is an example of the DT obtained from the *tese_ChosenVar1* dataset (Figure 3.11), but only with 3 levels of depth instead of the four levels returned by the *GridSearchCV*, since both classifiers present the same confusion matrix and therefore the same evaluation measures, and I understood that a 3 level DT would be easier to comprehend and interpret than a more long and complex one. This visual representation allows the company to identify the most relevant variables that influence the existence of significant losses, since those are the variables that are used as nodes. Moreover, the DT visualization indicates the gini value for each node, obviously starting with a higher gini on the parent node and decreasing until it reaches the leaf nodes. The company can also identify the number of samples in each node before splitting it, as well as the number of observations at that node that belong to each category. Finally, each node is assigned to a class meaning that most samples inside that node belong to the specified class.

Regarding the DT obtained with the remaining datasets, as well as the one from *tese_ChosenVar1* with a maximum depth equal to four, they all can be consulted in the Appendix.

3.5. EVALUATION

After creating all the classifiers, I moved on to the next step: measuring the performance of the models built from the four datasets, in order to evaluate and compare each one of them. To do this, I analysed the following evaluation measures: Accuracy, Sensitivity, Specificity, Precision and F-measure for each classifier (Table 3.14).

		NoCorr	PCA	ChosenVar1	ChosenVar2
Parameters	**Criterion**	Entropy	Gini	Gini	Gini
	Max_depth	5	4	4	4
	Min_samples_split	50	50	50	50
	Min_samples_leaf	10	10	150	150
	Min_impurity_decrease	0	0	0	0
Results	**Accuracy**	0,97	0,74	0,72	0,74
	Sensitivity	0,97	0,76	0,78	0,82
	Specificity	0,99	0,63	0,48	0,45
	Precision	0,997	0,89	0,84	0,84
	F-measure	0,98	0,82	0,81	0,83

Table 3.14: Parameters and Evaluation Measures of all datasets

The parameters used above, as previously mentioned, were the ones obtained after the use of *GridSearchCV*. After exploring some parameters that I considered interesting to change, such as the maximum number of depth for the DT, I concluded that for the *tese_ChosenVar1* dataset the evaluation measures remain exactly the same whether the max_depth parameter is equal to 4, as returned by *GridSearchCV*, or 3, returning a simpler tree. Therefore, I decided to create both models from this dataset, allowing the company to decide which one they consider the best solution. Regarding the *tese_ChosenVar2* dataset, I also tried to use a lower value for the max_depth parameter but the evaluation measures slightly changed, decreasing from the ones obtained with a max_depth equal to 4 to the ones obtained with the *tese_ChosenVar1* dataset, regardless of its max_depth parameter.

4. RESULTS AND DISCUSSION

4.1. MODEL 1 – TESE_NOCORR

As expected, this dataset originated the classifier with the highest evaluation measures, being all equal or higher than 0,97, due to the simple fact that it contains variables with information regarding the quantity and value of both types of losses. For this reason, although this is apparently the best model, I don't find it particularly interesting once it requires information that might not be available in advance to prevent future losses. While visualizing the DT (Figure 8.1), it can be noticed that the first split occurs on the feature that measures the quantity of identified losses, just as most of the following splits which are related to both identified and unidentified losses' quantity and value. The next nodes are related to the categories of each product, as well as the seasons of the year, but also including the already mentioned features with information about losses. Therefore, despite the very good evaluation measures, I defend that this is not a very useful model.

4.2. MODEL 2 – TESE_PCA

Since this classifier was obtained with a dataset that is the same as the previous one but after applying PCA, I maintain my opinion concerning the influence of the already mentioned set of features related to losses. Additionally, I consider this the worst of the four models, since it provides no interpretability for the company due to the appliance of PCA, being this the reason why I haven't created a visualization of this classifier. Furthermore, it presents some evaluation metrics lower than the ones of the *tese_ChosenVar2* dataset, such as sensitivity that has a value of 0,76 and f-measure, with a value of 0,82.

4.3. MODEL 3 – TESE_CHOSENVAR1

The third classifier was built using the *tese_ChosenVar1* dataset, which doesn't include variables regarding the quantity and value of both identified and unidentified losses but includes some features that provide information related to stocks and sales. The DT obtained, as can be consulted in the appendix (Figure 8.2), presents Monday as the parent node, using that feature to create the first split in the data. After that it uses features such as the stores' name and the category of the products, ending up relying on the stocks and sales variables to split the data. This is the model with the worst accuracy and f-measure values, although it requires more input variables than the next and last one of the four models, providing enough reasons not to be considered the best and most useful classifier.

4.4. MODEL 4 – TESE_CHOSENVAR2

The last DT (Figure 8.3) was created using the *tese_ChosenVar2* dataset, that only uses variables linked to the characteristics of the stores and products, as well as features that represent periods of time, such as seasons and days of the week, not including any information related to losses, sales or even stocks, with the exception of the *Above VUM* variable, and being for this reason the most useful algorithm in my opinion. This classifier, that resulted in a DT very similar to the previous one, also starts by dividing the data using the Monday variable. After that, it relies on features regarding the category of the products, the season and the name of the store, for example. Although it doesn't represent the model with the best evaluation metrics, as already stated, I consider it the most useful and applicable

for the company's problem, since it doesn't require any information that is in continuous change such as stock measurements, sales and especially losses.

Now that I have briefly assessed all models' visual representation and evaluation measures, I find it relevant to assess the results as a whole. Therefore, in my opinion, the DT created using the *tese_ChosenVar2* dataset is the best classifier, followed by the one originated by the *tese_ChosenVar1*. While comparing these last two classifiers, it is clear that both have problems with the specificity, meaning that they are good at predicting cases of actual significant loss but also have a high rate of false positives. This means that both algorithms wrongly classify observations as having a significant loss when it is not significant. This can be preferable or not by the company but, since the purpose here is to prevent significant losses in order to act in advance, this score for specificity isn't necessarily bad. It would be worse if the model wrongly classified instances as a not significant loss when in fact it was significant, since in that case we would be missing the objective of preventing future significant losses.

Overall, in terms of having a good performance with a set of variables that are not directly linked to losses and sales, relying mostly on static features, I am satisfied with the result of the *tese_ChosenVar2* model that was able to reach an accuracy of 0,74 and a sensitivity, precision and f-measure above 0,8.

5. CONCLUSIONS

As referred in the Introduction chapter, losses represent a major problem to this business, since its occurrence reduces profit whilst creating a lot of food waste that could be avoided. Solving this problem, by allowing the company to identify the factors that lead to the significant loss of a specific product, thus being able to prevent future major losses, was a great way to show the applicability of the data mining area, namely the use of machine learning, in the study of the high number of losses in the retail sector, being this the main objective of this project. Concerning the remaining objectives, I proposed to pre-process the data provided by the company, which I did in the Data Preparation chapter, as well as to create several models to the pre-processed data, which I completed in the Modelling chapter. Additionally, I defined the objective of evaluating each one of the obtained models, as I did in the Evaluation chapter, and lastly, in the Results and Discussion chapter, I identified the best model as the one created using the *tese_ChosenVar2* dataset, being this the one that should be used by the company to take any meaningful insights.

From a more technical perspective, I can conclude that the model obtained from the *tese_ChosenVar2* dataset is a good classifier, correctly predicting the class of 74% of the observations. Furthermore, the model is very good at identifying instances that belong to the positive class (*Significant Loss* = 1), correctly predicting 82% of the observations that are a significant loss. As already stated, specificity is the worst evaluation measure of this model, indicating that the obtained classifier wrongly classifies instances as being a significant loss when its loss is not significant, only correctly classifying 45% of the instances that belong to the negative class (*Significant Loss* = 0).

In terms of more business-related conclusions, by analysing the visual representation of the classifier, I concluded that we are facing a significant loss if the respective observation happens on a day of the week different that Monday on the Queijas store, especially if it happens in the autumn, or Gaia-Avenida da República stores, being these the two most problematic stores in terms of losses. Additionally, significant losses are also likely to happen on Mondays, on products like apples (belonging to the *Maca* category) and in weeks that are in the beginning or the end of each month, especially if it is not Summer season. Lastly, an observation that occurs on Monday and on products like apples, but in weeks that belong to the middle of a month, is likely to have suffer from a significant loss if the price per unit at which the product is being sold is below its average (being probably on sales). In a summarized manner, the day of the week has a huge impact on whether there will be a significant loss or not, as well as the category of the product we are analysing. Additionally, seasons are relevant to the prediction of significant losses, just as the position of a week in the respective month (beginning, end or middle). Lastly, the price at which the product is being sold as a big impact on this topic also, since prices below average are more prone to originate significant losses.

My advice is for the company to reinforce the surveillance in the periods and in the stores that present a higher number of significant losses, to understand the nature of those loss (whether it is mostly due to a theft or rotten fruits, for example). Also, the stores with a higher number of significant losses, such as *Queijas* and *Gaia-Avenida da República* for instance, should be frequently analysed to understand how the products are being stored and if they have all the right conditions to that purpose, such as the right temperature and humidity per example. Additionally, losses monitoring should happen more frequently when a product is being sold at a lower than average price, maybe

measuring and updating the inventory of those products more than just once per week, as currently happens.

Finally, I hope that the company can recognize the relevance of data mining as a very useful tool to their business sector nowadays, as I already did, considering this a vastly interesting field to be applied to a bigger set of retail problems.

6. LIMITATIONS AND RECOMENDATIONS FOR FUTURE WORK

As with most studies, the design of the current project is subject to several limitations that affect the quality of the findings as well as the ability to solve this company's problem. The first is the small period of time represented in the original dataset provided by the company, the second is the randomness of the topic that is being analysed and predicted and the last one is the lack of information regarding some variables that might be useful to solve this problem.

The dataset can be considered too small for supervised classification, since smaller training sets lead to the fact that a slightly change in some variable might result in a very different final classification. Furthermore, I consider one year a small time period to develop this analysis, since data regarding a period longer than one year could help the classifier on finding a pattern in the occurrence of significant losses between the same months or seasons of different years. Regarding the second limitation, the huge range of possible causes for significant losses makes this a very unpredictable subject. Since losses can occur due to several reasons from random theft to perishables' rotting, and even to mistakes made by the stores or the distribution centres, I consider this a very hard to predict topic. In my opinion it might be useful to perform this analysis but only regarding the identified losses, not considering the non-identified losses since those are the ones whose causes are unknown. The last limitation I felt while developing this project was the lack of information regarding some relevant aspects that could help to understand the major causes of significant losses. For example, assessing the temperature of each stores' section of fruits might be helpful, as well as knowing if the perishables are being stored in a refrigerated area as it is supposed or not. Therefore, there are a lot of variables that are not being measured by the company or, even if they are, they rarely correspond to the reality of the stores. A more accurate monitoring of each store could be made in order to have more variables available to analyse as well as more reliable ones. Although the mentioned limitations don't invalidate the developed study as well as its conclusions, I am sure that overcoming these constraints would help to create an even better and useful model.

That said, my recommendations for future work on this topic are that this project covers a period of time of more than one year, using data from two years or more, even if that means reducing the number of stores that are being analysed, for example, or even analysing the data per store, instead of having all stores together in the same dataset. Therefore, I also recommend extending this analysis to a broad range of products, both perishable and non-perishable ones as well as developing this study using the stores that the Group holds in other countries, namely Poland and Colombia. Given the second mentioned limitation regarding the randomness of this topic, my advice as already mentioned is to perform this study but only regarding the quantity and values of identified losses, since most of those are due to perishable rotting allowing the company to identify the main factors that cause it and obviously allowing them to prevent it. The non-identified losses, whose causes are always unknown can also be analysed but in a different model and always considering that the results might be very unpredictable. In a summarized manner, my advice is to focus on the identified losses analysis. Finally, my last recommendation is to measure and register the temperature of each store in order to include that as a reliable variable to build the classifier, as well as monitoring the local in which the products are being stored since whether it is refrigerated or not should have an impact on the quantity of identified losses among perishables.

Overall, I consider this a very fascinating problem that is of great interest for the company to solve. Nevertheless, in order to achieve even better results, I think the company needs to adopt a more data-driven attitude by measuring a set of different factors that might be relevant to the problem.

7. BIBLIOGRAPHY

Agard, B., Morency, C., & Trépanier, M. (2006). Mining public transport user behaviour from smart card data. *IFAC Proceedings Volumes (IFAC-PapersOnline)*.

Alpaydin, E. (2010). Introduction to Machine Learning Third Edition. *Introduction to Machine Learning*. https://doi.org/10.1007/978-1-62703-748-8_7

Azevedo, A., & Santos, M. F. (2008). KDD, semma and CRISP-DM: A parallel overview. *MCCSIS'08 - IADIS Multi Conference on Computer Science and Information Systems; Proceedings of Informatics 2008 and Data Mining 2008*.

Calendário 2018. (2019). Retrieved 28 August 2019, from https://www.calendarr.com/portugal/calendario-2018/

Chen, D., Sain, S. L., & Guo, K. (2012). Data mining for the online retail industry: A case study of RFM model-based customer segmentation using data mining. *Journal of Database Marketing and Customer Strategy Management*. https://doi.org/10.1057/dbm.2012.17

Daryanto, A., & Sahara, D. (2016). Food loss in supermarkets: what can supermarkets do to reduce food loss? *Proceedings of the Crawford Fund 2016 Annual Conference*.

Data created worldwide 2010-2025 | Statista. (2019). Retrieved 27 August 2019, from https://www.statista.com/statistics/871513/worldwide-data-created/

Davison, B. D., & Weiss, G. M. (2010). Data Mining. The Handbook of Technology Management, Supply Chain Management, Marketing and Advertising, and Global Management, volume 2, 542–55

Épocas / Estações do ano. (2019). Retrieved 28 August 2019, from https://www.calendario-365.pt/epocas-estacoes-do-ano.html

Han, J., Kamber, M., & Pei, J. (2012). *Data mining*. Amsterdam: Elsevier/Morgan Kaufmann.

Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). From data mining to knowledge discovery in databases. *AI Magazine*.

Instituto de Meteorologia de Portugal, & Agência Estatal de Meteorologia de Espanha. (2011). Atlas climático ibérico: Temperatura do ar e precipitação (1971-2000). In *Agencia Estatal de Meteorología, Ministerio de Medio …*.

Jadhav, S. D., & Channe, H. P. (2016). Comparative Study of K-NN, Naive Bayes and Decision Tree Classification Techniques. *International Journal of Science and Research (IJSR)*, *5*(1), 1842–1845. https://doi.org/10.21275/v5i1.nov153131

Jagdev, G. (2018). *Augmenting Revenue Growth in Retail Segment via Data Mining*. *5*(3), 1–8.

K, K., M, M. N., & R, S. (2018). Applications of Data Mining Techniques in Healthcare and Prediction of Heart Attacks. *International Journal of Data Mining Techniques and Applications*. https://doi.org/10.20894/ijdmta.102.007.001.027

Kader, A. A. (2005). Increasing food availability by reducing postharvest losses of fresh produce. *Acta*

Horticulturae, *682*, 2169–2176. https://doi.org/10.17660/ActaHortic.2005.682.296

Karamshuk, D., Noulas, A., Scellato, S., Nicosia, V., & Mascolo, C. (2013). Geo-spotting: Mining online location-based services for optimal retail store placement. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, *Part F128815*, 793–801. https://doi.org/10.1145/2487575.2487616

Kleissner, C. (1998). Data mining for the enterprise. *Proceedings of the Hawaii International Conference on System Sciences*.

Kotsiantis, S. B. (2007). Supervised machine learning: A review of classification techniques. *Informatica (Ljubljana)*.

Kottek, M., Grieser, J., rgen, Beck, C., Rudolf, B., & Rubel, F. (2006). World Map of the Koppen-Geiger climate classification updated. *Meteorologische Zeitschrift*.

Larose, D. T., & Larose, C. D. (2014). Discovering Knowledge in Data. In *Discovering Knowledge in Data*. https://doi.org/10.1002/9781118874059

Manyika, J., Chui Brown, M., B. J., B., Dobbs, R., Roxburgh, C., & Hung Byers, A. (2011). Big data: The next frontier for innovation, competition and productivity. *McKinsey Global Institute*.

Moisen, G. (2008). Classification and Regression Trees. *Ecological Informatics*, 582-588.

Oladipupo, T. (2010). Types of Machine Learning Algorithms. In *New Advances in Machine Learning*. https://doi.org/10.5772/9385

Parfitt, J., Barthel, M., & MacNaughton, S. (2010). Food waste within food supply chains: Quantification and potential for change to 2050. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *365*(1554), 3065–3081. https://doi.org/10.1098/rstb.2010.0126

Podgorelec, V., Kokol, P., Stiglic, B., & Rozman, I. (2002). Decision trees: An overview and their use in medicine. *Journal of Medical Systems*. https://doi.org/10.1023/A:1016409317640

Raj S. (2015). A Review of data mining applications in banking.

RETAIL | Significado, definição em Dicionário Inglês. (2019). Retrieved 28 August 2019, from https://dictionary.cambridge.org/pt/dicionario/ingles/retail

Rokach, L., & Maimon, O. (2008). Data mining with decision trees : theroy and applications. In *Series in machine perception and artificial intelligence*. https://doi.org/10.1142/9097

Sahu, H., Shrma, S., & Gondhalakar, S. (2008). A Brief Overview on Data Mining Survey. *Ijctee*.

Shalev-Shwartz, S., & Ben-David, S. (2013). Understanding machine learning: From theory to algorithms. In *Understanding Machine Learning: From Theory to Algorithms*. https://doi.org/10.1017/CBO9781107298019

Statistics Portugal - Web Portal. (2019). Retrieved 28 August 2019, from https://www.ine.pt/xportal/xmain?xpid=INE&xpgid=ine_indicadores&indOcorrCod=0008074&contexto=bd&selTab=tab2

Sudhakar, M., & Reddy, C. V. K. (2016). Two Step Credit Risk Assessment Model for Retail Bank Loan Applications using Decision Tree Data Mining Technique. *International Journal of Advanced Research in Computer Engineering & Technology*, *5*(4), 705–718.

Tan, P., Steinbach, M., & Kumar, V. (2005). Intro to Data Mining. New Jersey: P. Ed Australia, 1–6

Vafeiadis, T., Diamantaras, K. I., Sarigiannidis, G., & Chatzisavvas, K. C. (2015). A comparison of machine learning techniques for customer churn prediction. *Simulation Modelling Practice and Theory*, *55*(June 2016), 1–9. https://doi.org/10.1016/j.simpat.2015.03.003

Wirth, R., & Hipp, J. (2017). CRISP-DM: Towards a Standard Process Model for Data Mining. *Proceedings of the Fourth International Conference on the Practical Application of Knowledge Discovery and Data Mining*.

Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2016). Data Mining: Practical Machine Learning Tools and Techniques. In *Data Mining: Practical Machine Learning Tools and Techniques*.

Zhang, K., Cheng, Y., Liao, W., & Choudhary, A. (2012). *Mining millions of reviews*. 1–8. https://doi.org/10.1145/2378104.2378116

8. ANNEXES

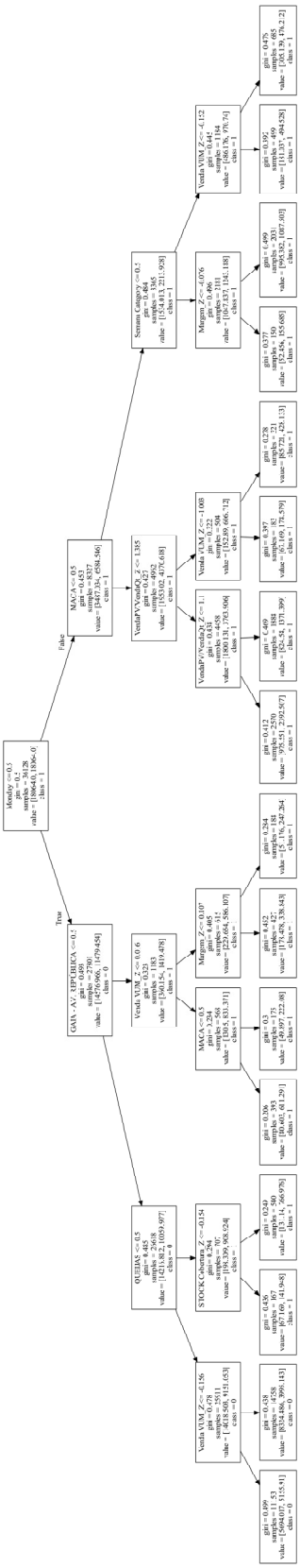

Figure 8.1: DT built from the tese_ChosenVar1 dataset, with maximum depth of 4

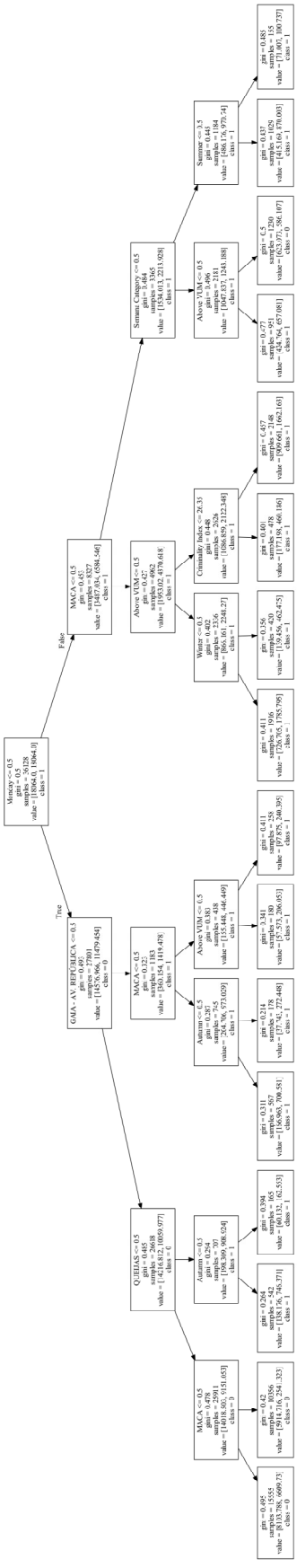

Figure 8.2: DT built from the *tese_ChosenVar2* dataset

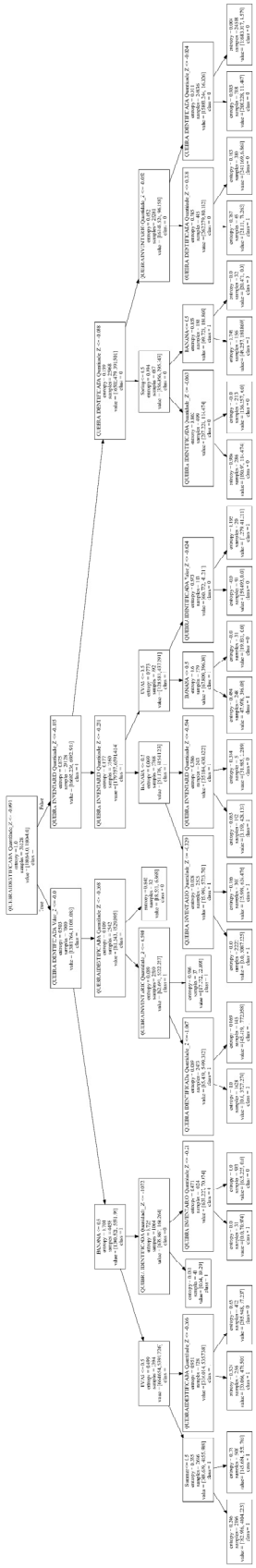

Figure 8.3: DT built from the tese_NoCorr dataset