Scuola di Ingegneria Industriale e dell'Informazione

Laurea Magistrale in Ingegneria Meccanica

POLITECNICO
MILANO 1863

# Food demand forecasting with Machine Learning: the prospects of cross-sectional training

Supervisor: Prof. Andrea Matta, Politecnico di Milano
Co-supervisors: Prof. Yves Dallery, CentraleSupélec
Prof. Evren Sahin, CentraleSupélec

**Lorenzo PEREGO**

900147

ACADEMIC YEAR 2018/2019

# Abstract

*Machine Learning (ML) methods have been increasingly employed for time series forecasting. Recent studies have asserted the dependency of the obtained performance on the amount of training data. Cross-sectional forecasting is a technique that employs data from different time series to train the ML model. Also defined as cross-sectional training, it was recently developed to cope with the insufficient data given by short time series. This thesis will investigate the application of cross-sectional forecasting on supply chain demand. The work will be divided in three sections which will concern the following topics: (i) application of cross-sectional forecasting to the whole dataset by means of several ML methods; (ii) experimentation of four clustering approaches to cross-sectional forecasting; and (iii) inclusion of exogenous variables besides the historical demand data for the creation of the forecasts employing cross-sectional training. The experiments, performed on two datasets related to food distribution, resulted in the ML methods outperforming the statistical benchmarks. It was also shown that ML methods' performance could be sensibly improved applying the right clustering approach, and that they were able to consider the influence of additional variables influencing demand reducing the forecasting error.*

***Key words**: Cross-sectional forecasting, demand forecasting, Machine Learning, clustering*

# Sommario

*Nonostante l'applicazione dei metodi di Machine Learning (ML) per la previsione di serie temporali sia sempre più diffusa, sono state individuate delle problematiche legate alla quantità di dati disponibili per la fase di training. Per ovviare a questo problema è stata ideata la pratica del cross-sectional forecasting, che consiste nell'utilizzare dati provenienti da molteplici serie temporali. Questa tesi tratterà l'applicazione del cross-sectional forecasting per la previsione della domanda. I dati utilizzati, forniti da una società di consulenza francese, provengono da due aziende attive nel settore alimentare. Nel lavoro, diviso in tre sezioni, vengono affrontati i seguenti argomenti: (i) confronto tra metodi statistici e di Machine Learning mediante full cross-sectional forecasting; (ii) sperimentazione di quattro approcci al clustering delle serie temporali a supporto del cross-sectional forecasting; (iii) e inclusione di variabili esogene per la generazione delle previsioni tramite cross-sectional forecasting. I risultati degli esperimenti condotti testimoniano una superiorità dei metodi di Machine Learning rispetto ai benchmark statistici utilizzati. Si è inoltre verificato che l'applicazione di una corretta tecnica di clustering permette di migliorare sostanzialmente la qualità delle previsioni e, inoltre, che l'inclusione di variabili esogene è facilmente ottenibile tramite i metodi di ML e può portare ad un miglioramento delle loro prestazioni.*

***Parole chiave**: cross-sectional forecasting, previsione della domanda, Machine Learning, clustering*

# Table of Contents

# List of Figures

# List of Tables

# Chapter 1: Introduction

*The scope of this chapter is to introduce the reader to the research environment, providing the essential knowledge that will allow the proper understanding of the thesis. Furthermore, the research goals will be defined, and the structure of the work will be delineated.*

## 1.1 What is demand forecasting

"A forecast is a statement of what is expected to happen in the future, especially in relation to a particular event or situation." (Collins dictionary)

Thus, forecasting is a discipline which finds its applications in a high number of different fields. It can be used to predict from weather to volume of phone calls, from road traffic to products' demand. The best practice in each of these fields is specific and, depending on its characteristics, can be extremely straight forward or more complex. The sunrise time tomorrow can be precisely forecasted by an analytical model, on the other hand environments with high uncertainty features, like stock exchange, will have the need of more statistically based approaches. The predictability of a certain event will be influenced by several characteristics of the forecasting environment (Hyndman and Athanasopoulos, 2018):

- How precisely we understand the factors which influence the variable we want to forecast (also referred to as "target variable").
- How much data is available.
- Whether the forecast can affect the value of the target variable itself.

Given the great extension, variety and complexity of forecasting, this study will solely focus on its application to supply chain demand.

We define supply chain demand forecasting as the action of estimating the values of future demand in the supply chain context. Supply chain demand forecasting is an important part of a field that has undergone a great expansion in recent years: supply chain analytics. Souza (2014) and Wang et al. (2016) define supply chain analytics as the ability to gain insights from data to make better decisions. Literature divides analytics into three categories:

*Descriptive analytics*: which employs data to gain insights on what happened and/or what is happening.

*Predictive analytics*: which extracts information from the data available in order to describe what will happen in future.

*Prescriptive analytics*: which aims at identifying the best course of action according to the scenario highlighted by the data.

Clearly, demand forecasting belongs to the second category. It is nonetheless of structural importance to understand the connections between these three categories in this field of application. Descriptive analytics will provide help identifying patterns and relationships between demand and different variables which may influence it. Thus, it is of valuable contribution to decide what to base the forecasting procedure on. Prescriptive analytics on the contrary, may use the prediction output with the objective to improve decision making by the means of optimization and simulation.

Given this introduction, it is clear how demand forecasting represents a key activity in supply chain operations planning and management (Hyndman, 2019), by the means of helping managers to make well informed business decisions. Bad forecasts lead to poor planning and can be harmful for the well-being of supply chains. Forecasts can be classified based on the forecasting horizon considered, which is strictly connected to their scope: (a) short-term forecasts are needed for the scheduling of personnel, production and transportation; (b)medium-term forecasts are needed to determine future resource requirements, in order to purchase raw materials, hire personnel, or buy machinery and equipment; (c) long-term forecasts are used in strategic planning. Such decisions must take into account market opportunities, environmental factors and internal resources. Investigation on long term forecasting will be beyond the scope of this study.

## 1.2 Quantitative and qualitative forecasting

Let us pose a forecasting problem: in a defined task, we are interested in forecasting demand in the future, where the demand in time unit $t$ is denoted by $D_t$. The time unit (or time bucket) selected can be the hour, the day, the month, etc. Assume that the present is defined as time unit $t$. The future time intervals will then be described as $t+1, t+2, \ldots$, and the past as $t-1, t-2, \ldots$ .

Given the statements above, the scope of the forecasting task can be delineated as the computation of $\widehat{D}_{t+1}, \widehat{D}_{t+2}, \ldots, \widehat{D}_{t+h}$ (where $h$ stands for the time horizon selected) that better approximate the future value of demand $D_{t+1}, D_{t+2}, \ldots, D_{t+h}$, unknown at time $t$. In order to reach this goal several are the methodologies available. These can be divided in two main families: quantitative and qualitative methods.

*Quantitative methods* are applicable only when data is available. This data can be of various nature. Time series data (collected at regular intervals in time) is the most common input, but cross-sectional data (collected at a single point in time) can be used as well. The second “sine qua non” for applying quantitative methods is the reasonable assumptions that patterns and relationships observed in the past will repeat similarly in the future. Thus, demand can be forecasted by observing the past regularities (patterns) and causal relationships (e.g., advertising campaigns and demand increase) and projecting them into the future.

*Judgmental (or qualitative) methods* are generally used when the data is either not available or not relevant in order to predict the target variable. The accuracy of such methods will be highly dependent on (I) the forecaster’s domain knowledge and (II) the information he bases the judgement on (Lawrence et Al., 2006). This typology of forecasting is not pure guesswork. Several well-structured and systematic approaches have been studied to aid judgment and to limit the biases coming from its nature.

A common approach to produce forecasts in practice, is the application of adjustments to the result produced by quantitative methods in order to consider some aspects which would be otherwise neglected. The use of the so called "judgmental adjustments" has been vastly discussed in literature showing controversial results. Refer to Syntetos et al. (2016) for a comprehensive review on the topic.

Due to the high uncertainty present in the demand forecasting environment and to demand's extreme dependency on several factors predictions are never completely accurate. The focus of this study will be oriented towards the use and experimentation of quantitative methods applied to short/medium-term forecasts. Improvements in performance of demand prediction methods would in fact lead to several beneficial effects in inventory management like reduction in excess inventory and stock outs, thus reducing costs and waste of resources.

## 1.3 Forecasting approaches

Among quantitative methods, three different forecasting approaches can be identified based on the data used to produce the forecast (Hyndman and Athanasopoulos, 2018):

***Time series approaches:*** the forecast is produced using as input the past recorded demand.

$$\widehat{D}_{t+1}, \widehat{D}_{t+2}, \dots, \widehat{D}_{t+h} = f(D_{t-1}, D_{t-2,} \dots)$$

Most of the widely used classical methods belong to this category.

***Causal (or explanatory) approaches:*** in this category predictor variables such as promotions, advertising campaigns, and product attributes can be used to forecast future demand. In other words, demand is correlated to the variables given as input. These correlations can be either rooted or not in causality relationships. Predictor variables can be either dynamic (they depend on time) or static.

$$\widehat{D}_{t+1}, \widehat{D}_{t+2}, \dots, \widehat{D}_{t+h} = f(R(t), L, K, \dots)$$

where $R(t), L, K$,... are the predictor variables. Examples of predictor variable that could be used to forecasts future values of demand are strength of economy, temperature, day of the week, holidays, promotions and so on.

***Mixed approaches:*** the forecasting method exploits both the information coming from recorded past demand and the predictor variables available. This methodology consists in nothing more than the mix between the previous approaches:

$$\widehat{D}_{t+1}, \widehat{D}_{t+2}, \dots, \widehat{D}_{t+h} = f(D_{t-1}, D_{t-2,} \dots, T, L, K)$$

## 1.4 The structure of time series

Before tackling forecasting tasks in the next paragraphs, it is necessary to spend some words describing time series. In demand time series several patterns can be identified. The following patterns are not necessarily present but, when they are, they can be formalized and projected by apposite methods in order to predict future sales.

*Trend*: which consists in a long-term increase or decrease in the data. It can be linear or present a more complex behavior.

*Seasonal*: which consist of a periodic variation in demand given by seasonal influencing factors. The pattern shows fixed frequency.

*Cyclic*: a cycle occurs when rises and falls in demand show a non-perfectly periodic behavior. This usually rises from economic cycles.

The effect of these patterns can be isolated, and time series decomposed. The equations reported underneath stand for the additive and multiplicative way to describe time series

$$\mathrm{D}(t) = T(t) + S(t) + R(t)$$

$$\mathrm{D}(t) = T(t) * S(t) * R(t)$$

Where *T(t), S(t)* and *R(t)* are respectively the Trend, Seasonality and Noise (or Remnant) components. Given the long-term effect of the cycle, its influence is in fact generally considered constant and represented by the trend. The multiplicative approach to the decomposition is more appropriate when all the components increase due to an increase in the trend.

## 1.5 Supply chains' operational dimensions

When facing a demand forecasting problem, it is necessary to keep an eye of regard for the supply chain physiology in order to approach the problem with the best performing tools and methods. The supply chain forecasting environment can be described by three main operational dimensions (Syntetos et al., 2016): length, depth and time.

*Length*: forecasting is needed at different locations in the supply chain. The demand at a retailing level will generate a demand at the next upstream link (distributer), which will subsequently respond placing an order to the next link (manufacturer) and so on. Length can be defined as the dimension constituted by all links in the supply chain. The characteristics of the demand will vary based on the location in the supply chain.

*Depth*: forecasting is used for various levels of decision making, from inventory control to strategic planning. Depth is defined as the level of detail at which the information is needed. This level of detail develops around several key features: products, suppliers, customers and locations.

*Time*: this dimension involves both operational choices (for example time buckets and forecast horizon) and data characteristics (history of the data, frequency of the demand, etc.).

As described in Chapter 1.1, forecasts are needed for various decisions in different contexts along the supply chain. These decisions will be based on forecasts with different characteristics. For example, inventory management for a retailer will require levels of detail different to those required by the manufacturer. Given these necessities, it is common practice to hierarchically aggregate (or disaggregate) the demand recorded in order to adapt it to the necessity of the practitioner. This procedure can concern all three dimensions, since it is mainly operated on products, locations and time buckets.

Different levels of aggregation will entail time series characterized by different traits. A higher level of aggregation would generally result in the stabilization of demand through noise reduction. When aggregating in fact, the random component of demand from the products aggregated, averages towards zero. Thus, it will be easier for forecasting methods to identify underlying patterns and module the expected demand accordingly.

Sometimes forecasts are needed at granular level but, due to the noisiness of demand, conducting direct forecasting can be challenging and result in low accuracy. In order to exploit the advantages given by this practice*, cross-sectional* aggregation could be an option. This practice is conducted aggregating demand from products of which I want to forecast sales individually. The forecasted cumulative demand can then be reassigned to the single products following alternative practices (Dalhart, 1974; Withycombe, 1989)

Aggregation concerns many alternative approaches which have been vastly discussed in literature. Even if employed in the study (see Chapter 3.1), its analysis will be beyond the scope of the study. For more information on the topic refer to the overview given by Syntetos et al. (2016).

## 1.6 Statistical methods for forecasting

Also referred to as classical methods, they represent the historical and still dominant tools for supply chain demand forecasting (Syntetos et al., 2016). Regression-based and smoothing based forecasting represent two of the most commonly employed families of statistical methods (Hyndman and Athanasopoulos, 2018).

Regression-based forecasting methods make use of relationships between dependent and independent variables. According to the independent variables used as input, these methods can be employed either through causal, mixed or time series approaches. Given the use in the study, the focus will be placed on their application through time series approaches. Several are the methods which stick to this classification. Naïve and Simple Moving Average represent the simplest methods part of this category. Expanding the horizon to less exceptional cases, linear autoregressive models produce the forecast as the linear combination of the variable's past value. These methods developed and increased in complexity during the second half of last century: in his 1951 thesis, Whittle (1951) developed the Autoregressive-moving-average (ARMA) model which integrated the concepts of autoregression and moving average in one single model. Successively, Arima model was developed with the aim to add an integration procedure in order to allow forecasting of time series presenting a trend component (Box et al., 1970). Furthermore, Seasonal Arima was developed in order to consider seasonality.

Smoothing-based forecasting: exponential smoothing forecasting methodologies are based on the assumption that the most recent points in the time series will have a greater influence than older ones. Whereas Simple moving Average methods considered past sales equally weighted, Simple (or single) exponential smoothing introduced exponentially decreasing weights going back in time. The main smoothing methods were developed by three authors during the late fifties. This approach was proposed in 1957 by Brown (1957) and was later expanded by Holt (1957) and Winters (1960). The expansions, known as Double and Triple Exponential Smoothing (or simply Holt-Winters method), allowed the application of exponential smoothing methodologies to time series which could present both trend and seasonal components. Holt-Winters is one of the most widely diffused methods in industry due to its reliability, adaptability and velocity in producing the forecasts.

Performance of statistical methods is of key interest for their application in industry and in this study. While it can be assessed some methods are more powerful than others (e.g. Holt-Winters compared to Simple Exponential Smoothing), it is difficult to file a ranking based on their performance. Depending on the Dataset characteristics, methods have different performances. A previous study (Petropoulos et al., 2014), aimed at evaluating and underlining the potential of eight statistical methods according to several time series features. The seven features considered were seasonality, trend, cycle, randomness, number of observations, average inter-demand interval (IDI) and coefficient of variation ($CV^2$). The first four features represent the components of time series and can be obtained through one of the decomposition approaches. The number of observations is descriptive of the amount of historical data employed in the forecast generation.

## 1.7 Machine Learning methods for forecasting

The term machine learning was popularized by Samuel (1959) and has been later formally defined: *"A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E."* (Mitchell, 1997). Even if conceived many years ago, ML has been gaining increasing importance in many fields of application recently and its use in forecasting has been frequent object of studies for the latest decades.

Part of Artificial Intelligence (AI), ML in turns contains Deep Learning (DL) as its subcategory (Abiodun et al., 2018). ML methods trespass in the DL category when they increase in complexity. For example, ANNs can be considered DL methods when they present a high number of layers and a complex structure given by the multiple ways the layers interact with each other. As a result of the higher DL method's complexity, they generally require a greater computational capacity compared to simple ML methods. In the following paragraphs, all methods appertaining to ML field will be addressed as ML methods with no distinction between ML and DL (unless underlined).

ML methods are divided into supervised and unsupervised learning methods. These two categories are delimited by the need of a target variable (also referred to as "dependent variable") during the training phase. An example of unsupervised learning can be found in clustering algorithms, which automatically group the records based on their features. When considering a forecasting task, the target variable is represented by the unknown future demand. In order to train the algorithm, a time series is fed as input and the following time point as output. Thus, the methods needed to produce the forecast belong to the supervised family. Unsupervised learning can be used in the forecasting process (see Chapter 5.1), but it does not actively produce forecasts.

In turn, supervised learning forecasting methods are divided into two families: regression and classification methods. The distinction between the two is based on the typology of dependent variables.

Classification methods are used to predict a categorical target variable (e.g., a variable with 2 categories: belong to a customer class, or not). These methods are employed in many fields of application, from image recognition to fraud detection.

Regression methods are used to predict a numerical value. The output can either be continuous or discrete. Since the desired output in forecasting corresponds to a discrete numerical value representing product demand, these are the methods of concern for the study.

## 1.8 Statistical vs ML methods: the difference

The difference between ML and statistical methods is not clearly defined in literature. The boundaries between the two fields are blurred to the point that the same method can be found both in statistics and in ML. Exemplary is the case of linear regression which as described before can be attributed to regression-based statistical methods or alternatively to ANN with no hidden layer and linear activation function. According to Stewart (2019), "*The major difference between* *machine learning and statistics is their purpose. Machine learning models are designed to make the most accurate predictions possible. Statistical models are designed for inference about the relationships between variables".* And again following the same line, "*Statistics draws population inferences from a sample, and machine learning finds generalizable predictive patterns*" (Bzdok et al., 2018).

These assessments, even if central to the topic, are not exhaustive. Considering a less philosophical and more methodological difference, it can be underlined that statistical methods try to adapt the model to the patterns contained in the data and subsequently use the structure obtained to predict demand. Machine learning methods, on the other hand, do not impose a strict structure, allowing it to be shaped by the data through more complex nonlinear relationships between the input and the output. The deeper the method is, the more complex relationships are allowed. Due to this complexity, it is generally impossible to understand the connections between inputs and outputs. Thus, ML methods sacrifice interpretability for predictive power (Stewart, 2019). The deeper the method, the lower the interpretability. For most of the cases, ML methods act like "black boxes".

Regarding their application, ML methods benefit from the employment of large datasets. The larger the dataset used the more generalized the relationship and the lower is the risk to incur into an overfitting situation. Overfitting occurs when the model is too closely fit to the training set, which results in highly specific predictive patterns not suitable for future demand due to the loss of generalization. Given that the structure is imposed and only a few parameters must be trained, statistical methods require less data to conduct their predictive task (Shmueli et al., 2017).

## 1.9 The potential of Machine Learning

As anticipated in Chapter 1.7, Machine Learning has been around for over half a century but has only recently seen great expansion and development. Given its versatility and its adaptable structure, Machine Learning is radically redefining many fields of employment up to the point of disruption (Yallop, 2019). ML market is increasing with outstanding speed: the compound annual growth rate (CARS) of ML is assessed to be around 23% (Forbes); 28.5 billion dollars were allocated worldwide to ML practices in the single first quarter of 2019 (Statista, 2019); McKinsey estimates that the potential economic given by AI will reach 9 to 13 trillion dollars by 2030.

In supply chain demand forecasting, Machine Learning methods still have not reached maturity and their employment is not widely used. Statistical methods are in fact still preferred by practitioners (Syntetos et al., 2016). The reason for this fact is said to be the interpretability of ML methods, seen as the main drawback of such techniques. Furthermore, scarce evidence is provided in literature about the objective superiority of ML methods compared to statistical ones.

While the analysis of the many studies conducted in literature will be object of the following chapter, it is opportune to immediately tackle the interpretability issue in order to allow the reader to approach the findings of the study with the optimal mindset.

We are living in decades of radical change for what regards technologies, customs and businesses. The increased capacity to store and process data, allows managers to collect and use more and more information. Considering retailers, a huge quantity of data about customers' behaviors during online and/or offline purchases is available. What products were visualized, how people navigated through the website, how much time they spent looking at products and so on. New technologies are able even to recognize customers entering in physical stores and to create "personas" analyzing how people behave (Hofmann and Rutschmann, 2018). This contemporary phenomenon has been given the name of "Big Data Revolution" (McAfee and Brynjolfsson, 2012). Professional figures able to process and manage this amount and variety of data have never been more requested (Davenport and Patil, 2012). In order to embrace and exploit the opportunities coming from this data, it is necessary, as suggested by Mayer-Schönberger and Cukier (2013), a radical change in mindset: "*Society will need to shed some of its obsession for causality in exchange for simple correlations: not knowing why but only what. This overturns centuries of established practices and challenges our most basic understanding of how to make decisions and comprehend reality.*" Once this change in mentality reached, data-driven companies can flourish and reach higher efficiency. It has been in fact proven that data-driven companies result 5% more productive and 6% more profitable than competitors (McAfee and Brynjolfsson, 2012).

Thus, practitioners in supply chain forecasting should drop their "obsession for causality" and base the choice of forecasting methods mainly on the predictive performances achievable.

## 1.10 Scope of the study

ML methods have been widely investigated in literature (see Chapter 2.3). As highlighted by Makridakis et al. (2018), these studies are mainly functional to a specific application and often interest a method created ad hoc for the forecasting environment considered. According to the authors, this characteristic tends to result in an upwards bias on the performance of ML, thus creating hype on their application which does not reflect their actual performance. Other criticalities they identified were: (i) employment of small datasets, (ii) consideration of short-term forecasting horizons and (iii) lack of an objective and unbiased evaluation technique. Furthermore, they underlined the need of a defined and constant benchmark. Their valuable contribution was to produce the first large-scale study which compared several ML methods to the top performing statistical ones relatively to the topic of single time series forecasting. The outcome of the study, which involved forecasts for 1045 time series from different fields, resulted in the net superiority of statistical methods.

Following in the footstep of Makridakis et al. (2018), this study aims at achieving a large-scale comparison of several ML and statistical methods. Differently from the previous investigation, it employs only data on historical demand coming from a couple of companies in the food distribution industry, thus appropriate for an investigation oriented to the field of supply chain forecasting. These companies' datasets present extremely heterogeneous characteristics, therefore showing particular suitability to test methods' predictive power according to diverse forecasting environments.

The key variation from the aforementioned analysis and originality of this study stands in the attempt to generalize the historical sales' patterns among several products. The assumption that justifies the different approach is that several products sold by a company can potentially present the same behavior and therefore be associated to the same patterns. Given the generalization power of ML methods, this behavior is "learned" if the model is properly trained with historical demand from all products belonging to the group of interest. The ML models trained will therefore be as many as the number of product groups identified in the dataset and not one for each product as in the case of the previous study from Makridakis et al. (2018).

The use of multiple product's historical time series for the training phase (cross-sectional forecasting) could be the key approach which allows ML methods to significantly improve performance. The dimension of the training set was in fact identified by Shmueli et al. (2017) and Cerqueira et al. (2019) as of central importance to the predictive power of a ML method.

The contribution of this investigation to the scientific community can be delineated among three points which reflect in three different section of the study:

1. The first section concerns the analysis of performances achieved through "full cross-sectional forecasting" (meaning cross-sectional forecasting applied to the whole dataset) approach to ML methods compared to statistical benchmarks. For this section of the study several machine learning methods are employed (see Chapter 2.4) using different methodologies to tackle multistep forecasting.

2. The second section aims at identifying the appropriate choices concerning the criteria for the creation of the subgroups to conduct cross-sectional forecasting on. Here, only one of the methods previously applied is selected to conduct the analysis. The choice is based on computational requirements and performances. This topic has not been object of interest in previous studies; thus, the analysis aims at filling the gap in literature described in Chapter 2.1.
3. Furthermore, the implementation of mixed approaches through ML models is briefly considered. Demand forecasting is traditionally based on time series approaches. Meaning, the demand is predicted using exclusively historical sales. Other factors influencing demand (for example temperature or marketing campaigns) are generally taken into account through judgmental adjustments or separate predictions of their effects on demand. ML, thanks to the complex relationship between input and output variables, can potentially include these factors in the inputs and learn their effect on demand. This approach is not original (see Chapter 2.2), but its potential to exploit Big Data information make it either way worth of further attention.

The study will develop around several Chapters. Chapter 2 will introduce the reader to the state of the art of cross-sectional forecasting and will provide the essential knowledge to fully understand the rest of the thesis. Chapter 3 will address the data preparation and preprocessing phase which is of central importance to all of the three contribution of this study previously overviewed. These three contributions will be treated in Chapter 4, 5 and 6 respectively. Finally, in Chapter 7, conclusions are drawn and possibilities for further studies are suggested.

It is necessary to underline that the study does not claim to evaluate the performance of optimized methods. The use and optimization of these methodologies in fact require ad hoc data treatment and time/computing power consuming procedures. The focus is on an easy to implement approach to forecasting which may be used by practitioners. The methods applied are retrieved from open source packages available for python and preprocessing is limited to the traditional essential steps usually applied.

# Chapter 2: Overview of ML techniques

*The scope of this chapter is to introduce the reader to the ML techniques which were used along the thesis. The first section will focus on the state of the art of cross-sectional forecasting, highlighting the literature gap this study aims to address. Successively the framework for time series clustering which will be used in Chapter 5 is presented. Furthermore, sections 2.3 and 2.4 are intended to provide knowledge regarding the use of ML methods in demand forecasting and the principles governing the methods selected for the study.*

## 2.1 Cross-sectional forecasting

The study of Makridakis et al. (2018) was focused on single time series forecasting, meaning, time series were individually considered. It assessed lower performance of ML methods compared to statistical methods. The complexity of ML accounts in fact for the possibility to approximate up to an infinity of diverse functions (Hornik, 1991), but can incur in the fallback of overfitting. This could happen when few data are fed as training set to a complex method, which could fit the error instead of the generalizable patterns useful to produce the forecasts (Bandara et al., 2020). Thus, the results achieved by ML methods in the aforementioned study, perfectly reflected the thesis of Hyndman (2016) and Yan (2012) which highlighted the limitation of complex ML models when producing forecasts based only on the limited amount of data coming from single time series.

Nonetheless several fields of application present a multitude of time series to be forecasted. This is for example the case of our field of interest, where the estimation of future demand values is required for all time series considered, each one standing for a product. The time series from this kind of dataset often present common traits and patterns which could be generalized among them. For example, groceries retailers could sell different brands of the same good (e.g. water or milk), which will approximately follow the same demand patterns.

These forecasting environments provide a competitive advantage for ML methods compared to statistical methods. ML methods are indeed able to exploit the shared patterns and train the models across multiple products.

Hartmann et al. (2015) recently proposed this approach, which they defined “cross-sectional forecasting”, in order to deal with incomplete sales histories and missing values. The authors proposed a single model which, trained with data from the whole dataset, would have been able to incorporate knowledge from all time series considered. Smyl and Kuber (2016) proposed cross-sectional forecasting as one of the two alternatives to address the lack of data problem for complex methods. Bandara et al. (2020) welcomed the proposal and further investigated cross-sectional forecasting. They underlined its potential to fundamentally improve ML methods performances by exploiting similarities in behavior across time series. Furthermore, they identified a criticality of the approach standing in the training across potentially disparate time series, which may reduce forecasting accuracy. To face this risk, the authors proposed a clustering phase based on time series attributes to group those time series which supposedly presented similar traits. Cluster specific training was then employed for Long Short Term Memory NN (LSTM) and these utilized to make predictions. Their proposed model, applied to the datasets from the CIF2016 and the NN5 forecasting competitions, managed to outperform most of the other participating methods. The outcome of their experimental study highlighted that, depending on the dataset’s time series homogeneity, the approach could improve performance with statistical significance (high homogeneity) or not (low homogeneity) compared to the “single time series” approach to LSTM.

The previous study from Bandara et al. (2020) focused on the general concept of applying the clustering procedure to cross-sectional forecasting, but did not place particular attention on how to create the clusters and what is the appropriate clustering approach allowing to optimize predictive performance. The authors performed this procedure selecting arbitrary numbers of clusters for several clustering methods. Concluding the paper, they emphasized the importance of the cluster selection methodology and highlighted the need of a study concerning the topic.

This thesis embraces the suggestion for future work of Bandara et al. (2020) and underlines the importance of such investigation. Of central significance is the necessity to find a tradeoff between the advantages given by an increase in training data availability and the disadvantages coming from considering disparate time series, both effect of clusters' increasing dimension. Thus, in the following investigation, the approach of Bandara et al. (2020) is proposed again focusing on the empirical evaluation of performance as a function of the number of clusters the dataset is divided in. The dependency between these two variables is shown and results are critically analyzed. This study wants to improve the comprehension of the clustering phase implications on cross-sectional forecasting, therefore representing the first step towards the achievement of a more sophisticated methodology.

## 2.2 Clustering time series

As introduced in the previous paragraph, the second section of this study evolves around the need to select the appropriate clustering dimension in order to optimize performance in a cross-sectional forecasting task. This operation results extremely complex for several reasons:

- since the best solution to the clustering problem is strictly related to the ML algorithm performance and ML methods work as black boxes, it results extremely difficult to set a precise objecting function to base the clustering on.
- Dependence on the Characteristics of the dataset, thus the optimal dimension of cluster is specific to the considered application.
- Variation of ML method complexity, which affects the clustering optimal solution and does not only depend on the ML method employed but also on the parameters selected as input.

Furthermore, the sole procedure of clustering time series is not to be considered an easy task. It has been, in fact, a challenge which many researchers have addressed in recent years due to the vast application in several fields like Biology, Medicine, Finance, Supply chain management and many more (Aghabozorgi et al., 2015). The criticality in the task stands in grouping time series which usually are characterized by high dimensionality, could show different lengths, present missing values, and be subjected to noise and irregularities.

### 2.2.1 Time series clustering framework

The scope of this paragraph is to provide the reader with the framework necessary to understand the models developed and described in Chapter 5.1, and not to give a complete overview of the literature on time series clustering. For an in-depth literature review on the topic, refer to the work of Aghabozorgi et al. (2015).

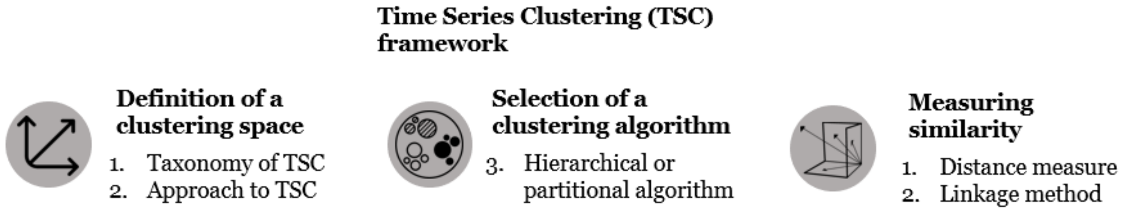
Time Series Clustering (TSC) framework
Definition of a clustering space
1. Taxonomy of TSC
2. Approach to TSC
Selection of a clustering algorithm
3. Hierarchical or partitional algorithm
Measuring similarity
1. Distance measure
2. Linkage method

*Figure 1: Time series clustering framework*

When clustering time series, numerous alternatives are available adopting a combination of operational decisions. This decisions give structure to the framework presented in Fig. 1 and concern the following five aspects of the task (Aghabozorgi et al., 2015):

1. **Taxonomy of time series clustering** (Keogh and Lin, 2005):
    - *Whole time series clustering*: the whole length of the time series is used as input for clustering.

- *Subsequence time series clustering*: a set of subsequences is employed to define the similarity between time series.
- *Time point clustering*: the value of one (or a few) points defined in time is used as similarity measure.

Since the concern of the study is grouping products which present similar patterns, all historical sales from a certain product should be considered when assessing similarity. Thus, the sole approach considered in the study is *whole time series clustering*.

2. **Approaches to *whole time series clustering*** (Warren Liao, 2005)*:*
   - *Distance-based*: algorithms work directly with time series' raw data and group them based on a distance measure.
   - *Feature-based:* algorithms work indirectly with features extracted from raw data.
   - *Model-based:* time series are indirectly grouped based on the model extracted from raw data.

Distance-based approaches seem as the most straight forward: most of them consist in applying an unsupervised ML algorithm to group *N* points defined in *M* dimensions, where *N* is the number of time series and *M* the number of time points in each time series. When applying this procedure to real world time series the approach present some criticalities (Wang et al., 2006):

i. When time series are very long, some clustering algorithm become intractable due to the high dimensionality.
ii. When time series present different lengths or missing values, distance measures fail to evaluate similarity.
iii. When the environment is characterized by high uncertainty, distance measures performance could be heavily penalized by the presence of noise.

Many methods have been devised to cope with these problematics. One among all, the diffused methodology of applying Dynamic Time Warping (DTW), an elastic distance measure which can deal with different time series lengths.

Model-based approaches would require to assign a model for each time series of interest, which is not suggested for irregular demand patterns and could involve high computational requirements (Wang et al., 2006).

Feature-based approaches have the characteristic not to be subjected to the same problematics (Räsänen and Kolehmainen, 2009). Thus, given the characteristics of the data available for this study, feature-based approaches represented the most investigated choice for this study (see Chapter 4).

### 3. Clustering algorithms

Clustering algorithms can be divided in two main families: hierarchical and partitional (or commonly referred as K-means) clustering techniques (Karypis et al., 2000).

- *Hierarchical Clustering:* production of a nested sequence of partitions which at the highest level presents a single cluster comprehensive of all *N* time series and, at the lowest level, singleton clusters (formed by a single time series). Each of the intermediate level is formed either splitting clusters from higher levels (divisive approach) or agglomerating clusters from lower levels (agglomerative approach). The nested partitions can be graphically represented by a dendrogram, which can be then employed by the user to decide how to conduct the necessary action of selecting the preferred clustering level. The dendrogram (of which an example is reported in Fig. 2), in fact, qualitatively shows the number and the enclosed dissimilarity related to each clustering level, thus being a key tool to select the appropriate level for the application considered.

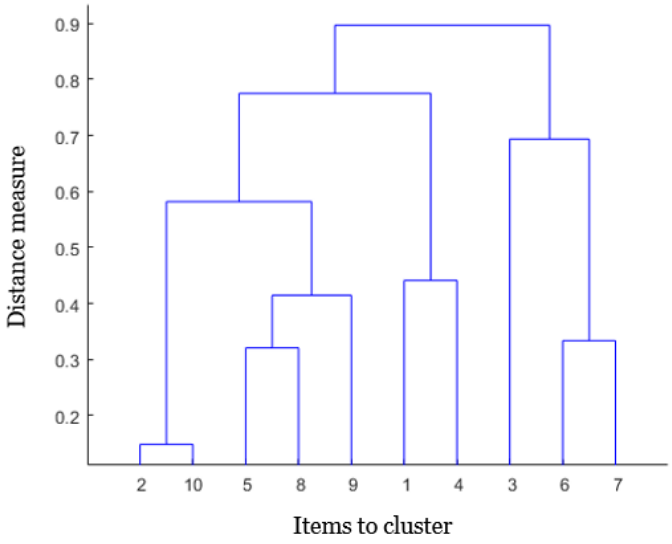

*Figure 2: Example of a dendrogram (retrieved from www.mathworks.com)*

- *Partitional (or K-mean) Clustering: K* clusters are iteratively formed based on the similarity between the time series and the *K* centroids. The *K* centroids (where *K* must be fixed a priori by the user) are initially randomly picked in the *M*-dimensional space (defined by the time series' features of concern). In each iteration, all points (representative of the time series) in the space are assigned to a centroid's cluster. For the following iteration, the centroid is moved to the barycenter defined by the points included in its cluster. The iterative process stops when the centroids stop moving.

4. **Distance measure:**

   This choice concerns the quantitative assessment regarding the distance between two points in the clustering space. Depending on the necessity of the study, diverse functions can be selected to compute the distance. Here are described three of the most used distance measures:

   - Euclidean distance: representing the "straight line" distance between two points ($p$ and $q$) in a $M$-dimensional Euclidean space.

$$d(p,q) = \sqrt{\sum_{i=1}^{M}(q_i - p_i)}$$

- Chebyshev distance: defined as the maximum difference between couples of coordinates of the two points considered.

$$d(p,q) = max_i(q_i - p_i)$$

- Manhattan distance: or also "block distance" is defined as the sum of the absolute difference of coordinates.

$$d(p,q) = \sum_{i=1}^{M} |q_i - p_i|$$

5. **Linkage method:**

When applying hierarchical clustering algorithms, it is necessary to define the linkage method, that is the principle governing the agglomeration of two clusters (or division in two clusters for the divisive variant). Here are reported some of the most commonly used:

- *Complete linkage*: the distance between clusters is given by the distance of the two furthest points (one in each cluster) which they contain (Fig. 3).
- *Single linkage*: the distance between two clusters is computed as the distance of the closest points (one in each cluster) which they contain (Fig. 4).
- *Average linkage*: the distance between two clusters is equal to the average distance from any member of one cluster to any member of the other cluster (Fig. 5).
- *Ward method*: while in the previous cases listed, the parameter governing the agglomeration/division of clusters was a distance,

Ward method aims at the minimization of variance inside the clusters. Thus, the agglomeration is practiced between those two cluster that, when merging, produce the lowest increase in the sum of squares. When selecting Ward's method, no choice regarding the distance measure is needed, since it is naturally based on Euclidean distance.

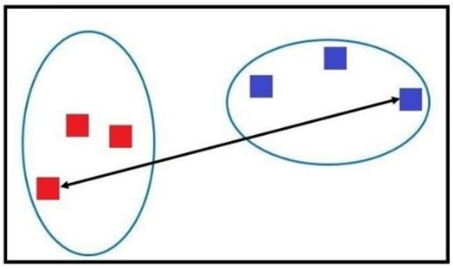

*Figure 3: Complete Linkage*

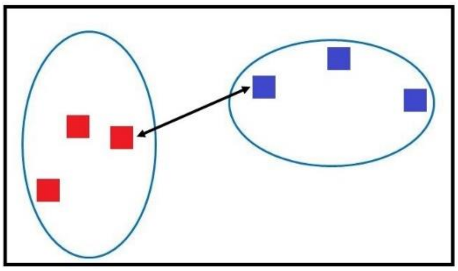

*Figure 4: Single Linkage*

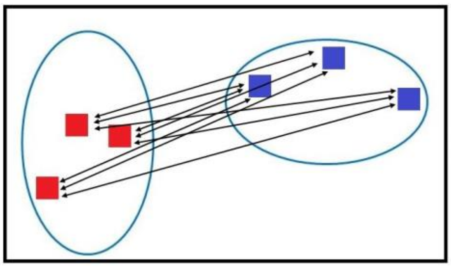

*Figure 5: Average Linkage*

(figures retrieved from: www.datavedas.com)

### 2.2.2 Feature-based clustering

Two of the clustering techniques employed in this study (which will be further discussed in Chapter 4.1) were inspired from the work of Wang et al. (2006) and concern whole sequence time series clustering by means of a feature-based approach. The authors proposed to base the selection of the feature on their significance and interpretability. The nine features selected have indeed been identified in literature (Armstrong, 2001) as able to capture the "global picture" of time series. The outcome of their approach resulted in a clustering performance comparable to the distance-based approaches used as benchmark, but, differently to these last ones, showed low computational requirements (due to the low dimensionality) and lower sensitivity to missing values. Previa additional experiments, they also concluded that high accuracy could be achieved limiting the number of features to just a subset of the nine 'global' features, and that this subset should be selected according to the dataset characteristics and the clustering needs. Furthermore, they highlighted that the selection of interpretable features could favor the collection of relevant insights regarding the clusters' characteristics. The same approach has been employed also in the previously described model proposed by Bandara et al. (2020), who used a slightly enlarged set of feature automatically extracted from the data.

## 2.3 Machine Learning performance

The performance of machine learning methods in demand forecasting has been object of study since the end of the last century. From the nineties, several studies were conducted on the ANNs applications in demand forecasting. Their usage was investigated on time series approaches by Tang et al. (1991). The authors compared the performances of various feedforward, back-propagation ANNs with the Box-Jenkins method. Findings revealed that, depending on the time series' characteristics, ANNs could achieve similar results or even outperform the statistical method.

Later in the same decade, Thiesing and Vornberger (1997) proposed the application of ANNs through a mixed approach. Time series (from past recorded demand) and other predictor variables were exploited in order to produce real data forecasts for a German supermarket. The mixed approach proved to be more effective than the statistical techniques which were used at the time by the company.

Aburto and Weber (2007) more recently proposed the use of neural networks as a constituent part of a hybrid methodology, which employed both Arima and ANNs in order to forecast demand for a Chilean supermarket.

An extensive analysis of various ML techniques was conducted by Carbonneau et al. (2008): forecasts based both on simulated and real-world data, were produced employing ANN, Recurrent ANN (RNN), Support Vector Regression (SVR) and various statistical methods. The outcome asserted a slight superiority of the ML methods but without statistical significance.

Ali et al. (2009) applied Regression Trees and SVR to grocery retailer's data. These data manifested periods with and without promotions. The outcome of the experimental studies showed that, in periods without promotions, the techniques' performances were comparable. However, the ML methods managed to greatly outperform the benchmark (exponential smoothing with promotional adjustments) where the influence of promotion was relevant.

Given the continuous development of ML methods in recent years, the need of an extensive analysis study was met by Makridakis et al. (2018). They compared ten popular ML methods to eight traditional statistical ones focusing on accuracy and computational requirements. The dataset comprehended 1045 time series retrieved from the M3 forecasting competition. The investigated ML methods comprehended SVR, K-nearest neighbor regression (KNN), gaussian processes, Regression Trees and six variants of ANN algorithms. All these methods were outperformed by the statistical ones in terms of both accuracy and computational complexity.

In the aforementioned paper by Makridakis et al. (2018), the authors explained the contradictory results in literature with the employment of small datasets (i), with the consideration of short-term forecasting horizons (ii) and with the lack of an objective and unbiased evaluation technique (iii). On the matter they underlined the need of a defined and constant benchmark.

In addition to the issues highlighted, this study wants to underline the existence of further problematics in finding benchmarks in order to have a better assessment of a certain methodology's performance.

The first challenge is conveyed with the support of the paper "'Horses for Courses' in demand forecasting" (Petropoulos et al., 2014). The authors express the need of approaching the forecast of the time series with the most appropriate tool for the job. The authors affirmed that the right question to address a forecasting problem with, is not "What is the best method?" but more correctly "What is the best method for my data?". Since the best method will depend on the data characteristics, it will result difficult to identify a well-defined and constant benchmark to compare ML methods' and statistical methods' performances. A second challenge this study will have to dwell on, is that employing ML methods require setting many parameters. For example, Multi-Layer Perceptron (MLP) (one of the simplest ANNs) requires to initially set the number of hidden layers, the number of nodes (in the input, output and hidden layers), the activation functions, and the learning rate. On the other side, simple statistical methods can be used without any parameter such as the Naïve method or with small number of parameters such as exponential smoothing. Another challenge is characterized by the different needs of data pre-processing phases whether it concerns ML or statistical methods.

Given the last two challenges (methods parameters and pre-processing), the performance of ML methods will greatly depend on the user's capacity to optimize the parameters and to pre-process the data. Thus, the evaluation would lose part of its objectivity.

Compared to the previous study, this work does not aim at providing a comparison which pretends to be objective and all-encompassing. The limitations are therefore taken into account and when possible tackled. When not, the issue is considered in order to define the boundaries and scope of the study itself.

## 2.4 Machine Learning Methods

As introduced in Chapter 1.7, ML methods are divided into supervised and unsupervised ML methods. Furthermore, a supervised learning method is labeled as classification or regression method respectively based on the categorical or numerical form of the output variable. The estimation of the future amount of sales places demand forecasting in the regression category.

The scope of this paragraph is to give a brief overview of the methods that are used for demand forecasting and, subsequently, explain the reasons for the selection of the methodologies considered in this study. Furthermore, it aims at providing the reader with the knowledge necessary to understand the differences between the employed methods in order to provide the tools for the comprehension of the following chapters.

Several machine learning methods exist for regression tasks. The most diffused typologies are Artificial Neural Networks (ANNs) and Decision Trees (DTs) (Seif, 2019). Additionally, other methods have been developed and applied to forecasting. Support Vector Regression (SVR) (Smola and Schölkopf, 2004), a variant of Support Vector Machines, has for example proved to be a quite performing method for financial time series forecasting (Tay and Cao, 2001). Also simpler methods like K-nearest neighborhood (KNN) have been employed (Makridakis et al., 2018). New variants of methods are often investigated in order to improve predictive performance. In the last decade, thanks to an increase in computational capabilities, the interest in the more complex Deep Learning methods has grown (Abiodun et al., 2018). Artificial Neural Network have in fact evolved to highly nonlinear configurations which, as a result of the many degrees of freedom in the model, are able to "learn" underlying relationships between input and output. Due to this renewed potential in ANN methods, many studies have been recently conducted on the topic (Kong et al., 2019; Smyl and Kuber, 2016; Bandara et al., 2020).

Although some DL algorithms, like RNN, Convolutional Neural Networks (CNN) and LSTM have proved to be valuable forecasting methods, they were not taken into consideration for this study due to the limited computational power available.

Most of the ML methods considered in this thesis are part of the Decision Trees family. Decision Trees (DTs), are non-parametric supervised learning methods used for classification and regression. The goal, when a regression task is concerned, is to create a model that predicts the numerical value of a target variable by learning decision rules inferred from the data features. In the application considered, features correspond to the past value of demand of the involved product (which is associated to a time series). Models are achieved by recursively dividing the data space and fitting a simple model on the partition. The operation is concluded when all features have been considered. The outcome of the recursive operation can be graphically represented by a Decision Tree. Fig. 6 shows an example of the Regression Tree and space partition.

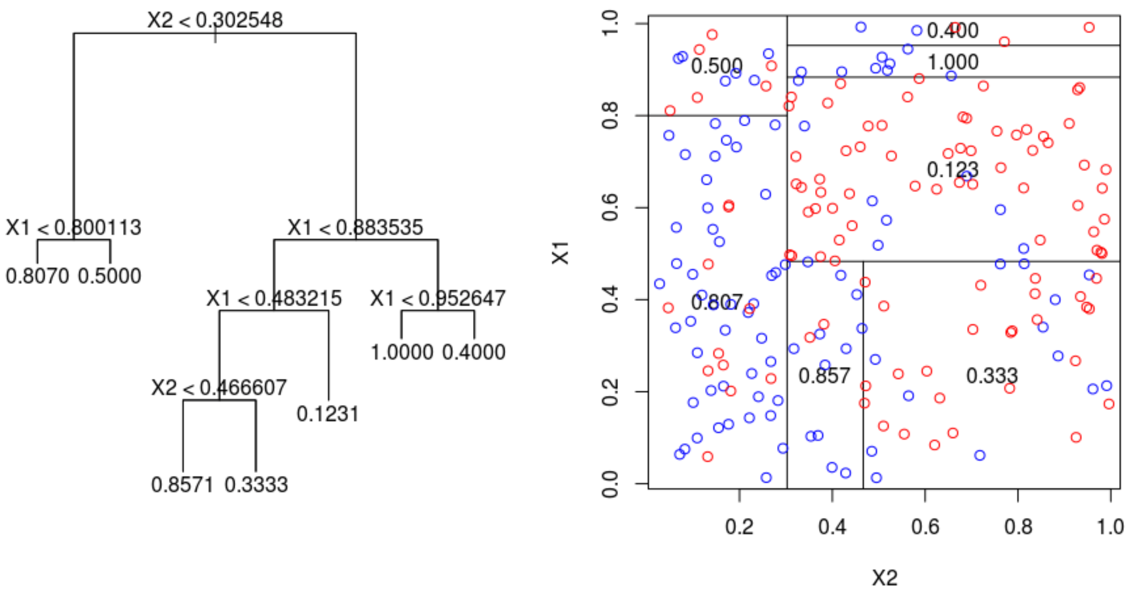

*Figure 6: example of RT's generation process (retrieved from www.datacamp.com)*

The procedure described is the basis for multiple ML methods which have been developed and refined along the years. These developments are all based on the combination of various simple models in order to create a more performing one (Dietterich, 2000) increasing stability, reducing variance and avoiding overfitting. The combinations, which are also called "ensembles" can be constructed employing different methodologies: Bagging, Boosting and Stacking.

*Bagging*: which stands for "Bootstrap aggregating" (Breiman, 1996), consists in running the learning algorithm several times in order to train various DT models ("weak learners") and subsequently averaging the forecasting result. Each time a weak learner is trained, a different subset of the training set is employed in order to produce $n$ different results, where $n$ is the number of weak learners trained. The $n$ subsets considered are of the same dimension of the original training set, but contain $1 - {}^{1}/_{e}$ of its samples (≈ 63,2%), while the remaining spots are filled with duplicates (Dietterich, 2000). Once the $n$ weak learners have been trained, forecasts are produced and averaged in order to get a final forecast which shows less variance. In Fig. 7 the bagging process is schematically reported.

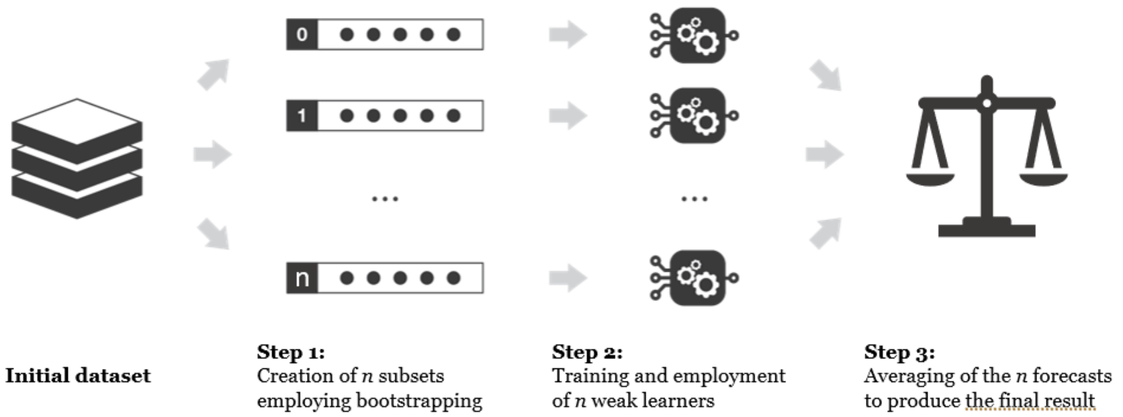
0
1
...
n
...
Initial dataset
Step 1:
Creation of *n* subsets employing bootstrapping
Step 2:
Training and employment of *n* weak learners
Step 3:
Averaging of the *n* forecasts to produce the final result

*Figure 7: Ensemble methods; boosting procedure*

*Boosting*: similarly to bagging, boosting makes use of several heterogeneous subset of the original training set in order to train weak learners which provide results. These results are then averaged to obtain the final forecast. The difference stays in the mutual independence between these subsets. While in bagging the subsets are created simultaneously through the bootstrapping process and result independent, in boosting, trees are grown sequentially based on the information from previously grown trees (Elith et al., 2008). This sequential optimization process (graphically represented in Fig. 8) can concern different techniques and is based on the performance of each tree on the "holdout set", which is a fraction of the dataset not employed in the training phase.

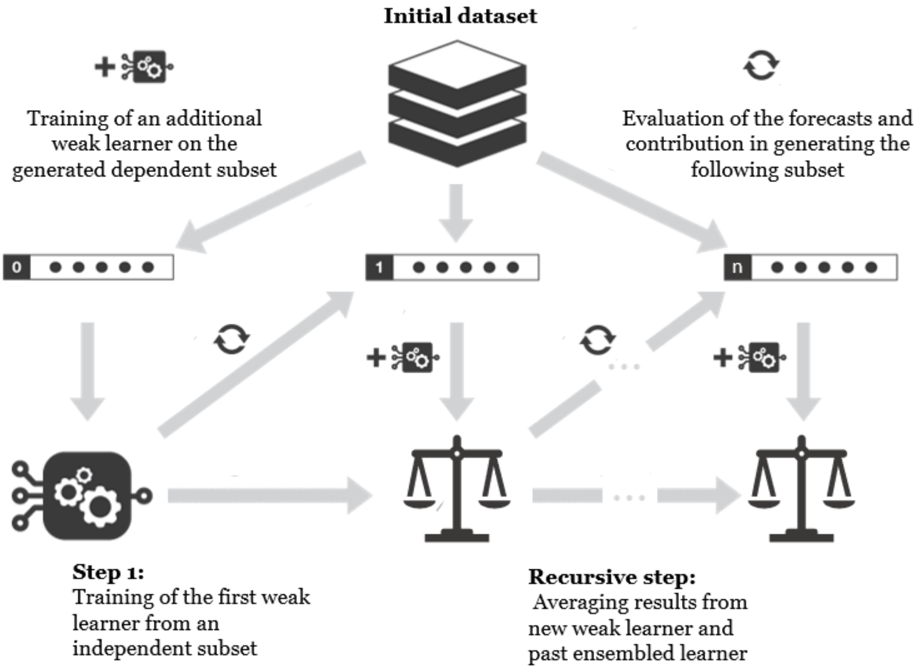
Initial dataset
Training of an additional weak learner on the generated dependent subset
Evaluation of the forecasts and contribution in generating the following subset
0
1
n
...
Step 1:
Training of the first weak learner from an independent subset
Recursive step:
Averaging results from new weak learner and past ensembled learner

*Figure 8: Ensemble methods; boosting procedure*

*Stacking*: differently to the previous embedding methods, stacking employs the forecasts of the *n* weak learners as input for a meta-model which is trained and used to produce the final forecast (Fig. 9). Given the need to "save" part of the data for the meta-model training, stacking is advisable only when the original dataset is of considerable dimensions (Rocca, 2019).

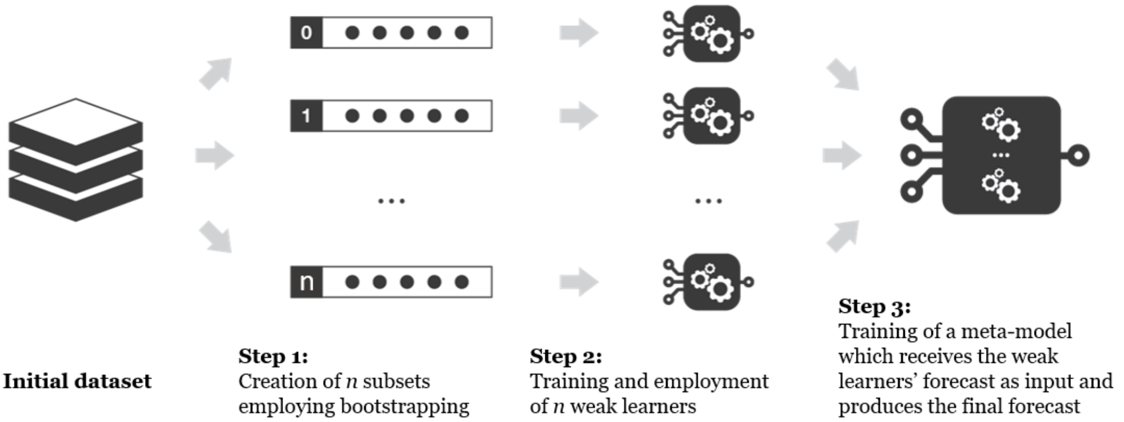
0
1
...
n
...
Initial dataset
Step 1:
Creation of *n* subsets employing bootstrapping
Step 2:
Training and employment of *n* weak learners
Step 3:
Training of a meta-model which receives the weak learners' forecast as input and produces the final forecast

*Figure 9: Ensemble methods; stacking procedure*

The three methodologies described can be used independently or combined, giving birth to several well-known methods like Random Forests, AdaBoost, Extremely Randomized Trees and many more. The methods employed for this study will be introduced in Chapter 3.2.

## 2.5 Multistep forecasting

As already introduced in the previous chapter, forecasting tasks can involve different horizons of interest: sometimes it is necessary to know the demand one day ahead and sometimes much before. When considering a task which requires a forecasting horizon different from 1 time step ahead, we can relate to the term "multistep forecasting". Meaning, forecasts are required at time $t + 1, t + 2, \ldots, t + h$ where $h$ is the number of time steps in the horizon considered and $t$ is representative of the time when the forecast is produced.

Statistical and ML methods work differently in such circumstances. Statistical methods, as explained in Chapter 1.8, are concerned with the inference of the global structure of the population starting from a sample of it. Applying an imposed structure and bending it to fit the training data, they are able to extract the fitted patterns and provide a forecast which is valid, theoretically speaking, for infinite steps ahead. On the other hand, ML methods can be viewed as black boxes which receive a number of inputs and return a predefined number of outputs. Thus, since they do not infer anything about the data population, the usefulness of the forecast is limited to the outputs provided. Different approaches have been develop to tackle a multistep forecasting task with a ML method (Hamzaçebi et al., 2009):

*Direct approach*: the forecasts for all the time steps considered in the forecasting horizon are produced at the same time by the same model. Thus, the model will provide *h* outputs where *h* is the number of steps in the horizon considered (Fig. 10).

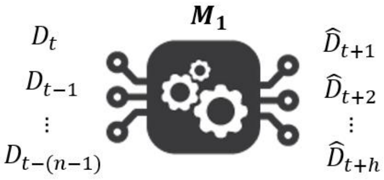

*Figure 10: Direct multistep forecasting approach*

*Indirect approach*: the model produces the output only for 1 time step ahead. The forecast produced is then inserted as an input with a sliding window procedure in the same model in order to predict the expected value for the following time step. This process continues iteratively until all *h* forecasts have been generated (Fig. 11).

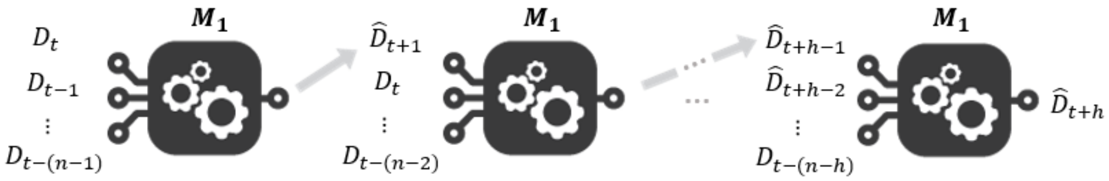

*Figure 11: Indirect multistep forecasting approach*

*Multi-model approach*: as many models as the number of time steps in the forecasts are trained ($h$). Each model produces exclusively the forecast related to one of the $h$ time steps (Fig. 12).

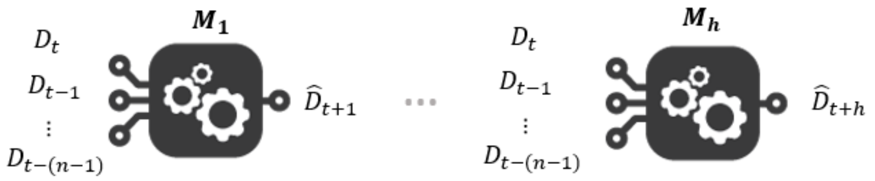

*Figure 12: Multi-model multistep forecasting approach*

The characteristic traits of the presented approaches are summed up in Table 1.

| Approach | N. of models | N. of inputs | N. of outputs | N. of iterations |
|---|---|---|---|---|
| **Direct** | 1 | $n$ | $h$ | 1 |
| **Indirect** | 1 | $n$ | 1 | $h$ |
| **Multi-model** | $h$ | $n$ | 1 | 1 |

*Table 1: Approaches to Multistep demand forecasting with ML methods*

None of the approaches has proved in literature to consistently outperform the alternatives (Hamzaçebi et al., 2009). This thesis wants to investigate the behavior of multistep approaches when applied to the specific case of cross-sectional forecasting. Due to the high computational requirements required to train several models, only the direct and the indirect approaches were considered.

# Chapter 3: Preprocessing

*The scope of this chapter is to introduce to the reader to the work which was necessary before the beginning of the experiments treated in the next chapters.*

*The preprocessing was conducted following the guidelines provided by Shmueli et al. (2017). In their book the authors identified the poor understanding of task and experimental environment as the greatest threat to data analytics projects. They described several steps (Fig. 13) which were not only used as guidelines during the unfolding of the study but will also serve in the following text as the structure to describe the preprocessing phase.*

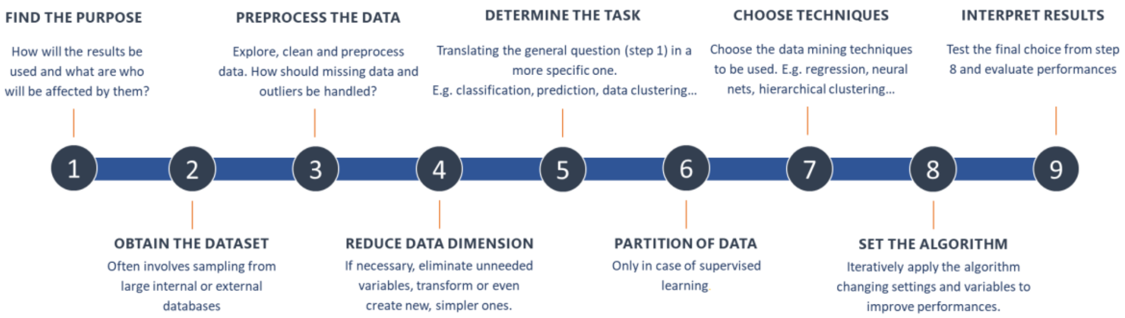
FIND THE PURPOSE
How will the results be used and what are who will be affected by them?
PREPROCESS THE DATA
Explore, clean and preprocess data. How should missing data and outliers be handled?
DETERMINE THE TASK
Translating the general question (step 1) in a more specific one.
E.g. classification, prediction, data clustering...
CHOOSE TECHNIQUES
Choose the data mining techniques to be used. E.g. regression, neural nets, hierarchical clustering...
INTERPRET RESULTS
Test the final choice from step 8 and evaluate performances
1
2
3
4
5
6
7
8
9
OBTAIN THE DATASET
Often involves sampling from large internal or external databases
REDUCE DATA DIMENSION
If necessary, eliminate unneeded variables, transform or even create new, simpler ones.
PARTITION OF DATA
Only in case of supervised learning
SET THE ALGORITHM
Iteratively apply the algorithm changing settings and variables to improve performances.

*Figure 13: steps of a data mining task*

*The first six of the nine steps will account for the rest of Chapter 3.1 which will describe the preprocessing phase. Steps 7 and 8 will be treated independently in Chapter 3.2. Finally, the clean data is employed through the algorithm selected and step 9 will take the form of Chapter 4, 5 or 6 depending on the analysis involved.*

## 3.1 The preprocessing phase

1. *Develop an understanding of the purpose of the data mining project.*
   The first scope of the study was identified as the need to evaluate the

potential of cross-sectional forecasting applied to supply chain demand. To do so, it was decided to conduct two experiments: the first concerned comparing full cross-sectional forecasting to other forecasting tools and methodologies employing a time series approach; the second wanted to verify cross-sectional forecasting adaptability to mixed approaches. A further contribution of this study was set to be an investigation on the appropriate way to apply a clustering approach to demand forecasting.

2. *Obtain the dataset to be used in the analysis.*
   The datasets employed in the study where provided by a French consulting company specialized in supply chain demand forecasting. From the several datasets available, two were selected due to their heterogenous characteristics.
   - *"Dataset A"* contained data about sales, product attributes and promotions of a B2B company providing food to canteens and cafeterias distributed among France. Specifically, the data available interested 401 products belonging to the "yogurt and fresh cheeses" product family. Due to the business' characteristics, the historical sales records presented extremely noisy and intermittent demand which would have consisted in a challenge for the forecasting methods.
   - *"Dataset B"* came from a French company specialized in retail having multiple stores all around France. The data available concerned past sales from 834 products, product and site attributes but, differently from *Dataset A,* no information about promotions or marketing campaigns effectuated during the concerned time span was available. Compared to the previous dataset, the demand time series presented on average a more regular demand.

- Information about holidays were retrieved from an open platform for French public data. The dataset reported the state of holiday (0 or 1) as a function of the "zone scolaire" and of the date.

The contents and characteristics of both datasets are schematically reported in Table 2.

| | Dataset A | Dataset B |
|---|---|---|
| **Historical Sales** | yes | yes |
| **Product Attributes** | yes | yes |
| **Store Attributes** | yes | yes |
| **Promotions** | yes | no |
| **Number of products** | 401 | 834 |
| **Company** | B2B, food provider for canteens and company restaurants | B2C, BIO products retailer |
| **Details** | Extremely noisy demand | More regular demand |

*Table 2: Datasets characteristics*

Both *Dataset A* and *B* contained information about past sales reported accordingly to different criteria: not only plain and simple records of past sales but also a "corrected" time series in order to account for the influence of exceptional sales and promotions.

The data provider suggested to adopt a forecasting horizon corresponding to two months ahead. This horizon was based on the interests of the involved companies.

3. *Explore, clean, and preprocess the data.*
   Data preprocessing represented a consistent part of the project. According to literature (Shmueli et al., 2017), this practice can represent around 80% of the total work in a data mining task. In this case study, given the high number of methods compared and the relevance of the post processing phase, the preprocessing accounted for broadly one fourth of the total work.

   An initial qualitative analysis was necessary to understand the datasets, the meaning of each information contained and to have an initial idea of the future work needed. After this initial data evaluation practice, the central section of the preprocessing phase took different forms depending on the data involved:

   - **Historical sales preprocessing**: requesting a major contribution, this procedure aimed at transforming raw sales recorded from Point of Sales (POS) into ready-to-forecast time series. It involved several steps and data manipulations which were conducted following the process flowchart (Fig. 14) reported underneath.

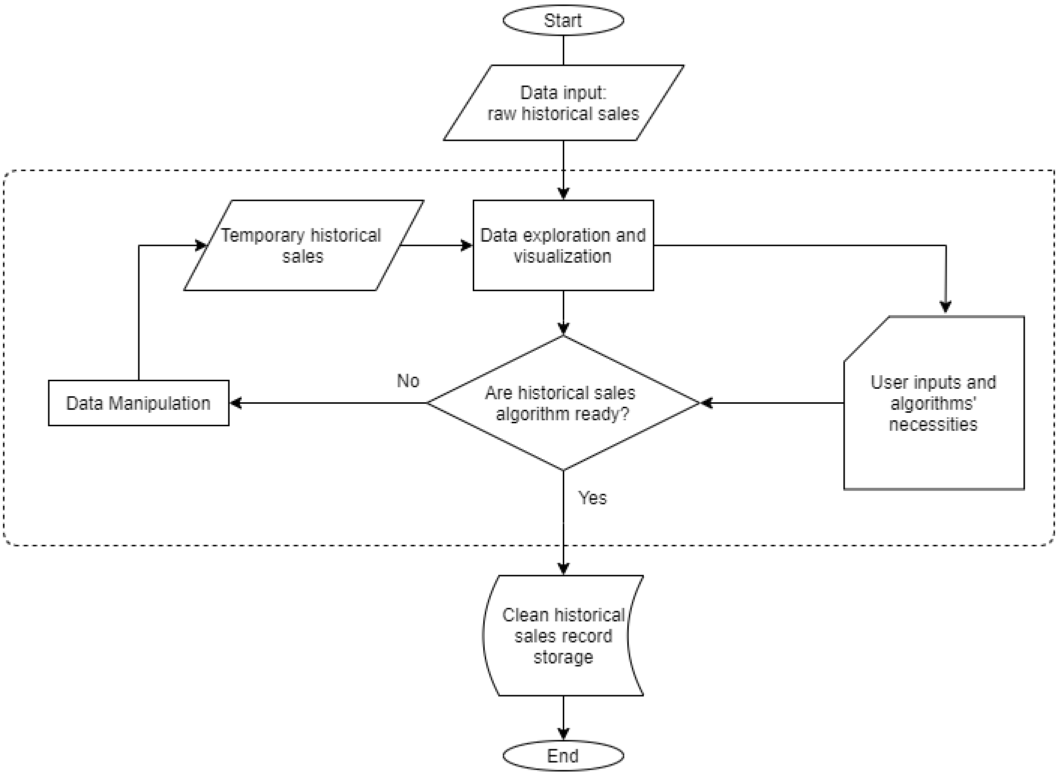
Start
Data input:
raw historical sales
Temporary historical sales
Data exploration and visualization
User inputs and algorithms' necessities
No
Data Manipulation
Are historical sales algorithm ready?
Yes
Clean historical sales record storage
End

*Figure 14: Data preparation process flowchart*

User inputs and algorithmic necessity were selected in order to comply to the research scope conditions and to respect algorithmic constrains. The dotted line contains the part of the process which is operated recursively. Hypothetically, if all data manipulation practices needed were known from start, historical sales preprocessing would have been a straightforward process. However, since some of the input choices entail decision which were aided by data visualization, the process resulted recursive and not automatable. In the following list reports the main data manipulation practices conducted in Python.

a. *Dates management and daily aggregation:* dates were processed in order to be interpretable by Python. Then, single sales records were aggregated by daily time buckets.

b. *Null sales adjustments:* POS records present only information about sales happening, but nothing is recorded when a certain product does not sell at all. It is therefore necessary to integrate daily historical sales in a structure which presents all time intervals from the beginning until the end date of the records.

c. *Aggregation:* sales were aggregated on both location and time dimension. Weekly time buckets were selected based on the study necessities. Indeed, daily time buckets proved to be too small to be applied to the extremely noisy and intermittent demand of *Dataset A*. On the other end, monthly time buckets would have decreased the difference between the various products and would have decreased the significance of holidays and promotions, thus departing from the study's third mentioned goal. The granularity level chosen for the location dimension involved the aggregation of all sales recorded for a certain product in all stores.

d. *Treatment of NPI* (new product introductions): several new products were introduced during the time span of interest for both datasets. Since their inclusion's effect in the study would have produced unknown outcomes, they were eliminated from the study in order not to bias the result.

e. *Treatment of interrupted historical sales*: a few products presented a long-lasting halt in sales. Since information about the reason of their behavior was not available, they were deleted in order to avoid biases.

Points d and e accounted for the elimination of several products from both datasets. *Dataset A*'s dimensions were reduced from 401 to 179 products, while *Dataset B*'s from 834 to 411.

- **Holidays preprocessing** (both datasets): holidays are dependent on the "zone scolaire", meaning a French classification of holiday systems for schools. For Dataset A every sale reported the "zone scolaire" of destination, therefore it was possible to estimate the influence of holidays on demand in each zone. The weights were computed based on the percentage of sales of the specific product on the zone of interest weighted over the total sales in the three zones. For Dataset B information was not available about the zone scolaire, therefore an arbitrary weight of one third was given to each zone. The format of the holidays was therefore left as a number included between 0 and 7, where 0 meant that none of the days in the considered week was holiday and 7 that all of them were.
- **Promotions**: as previously anticipated, information about promotions was available exclusively for Dataset A and was limited to beginning and ending date, product of concern and name of the promotion. One between the many limitations penalizing the mixed approaches investigation is the lack of data on the typology of promotion, which is known to be an important variable in characterizing the sales. Thus, the promotions were processed in order to have a table which, for each combination of product and time point, presented a Boolean variable.

4. *Reduce the data dimension, if necessary.*
   No data reduction was considered necessary thanks to the manageable dimension of the cleaned dataset and the need for a vast amount of time series.

5. *Determine the data mining task.*
   The precise data mining task is defined along the Chapters 4, 5, and 6

depending on the analysis considered.

6. *Partition of the data.*
   Long sales history was available for both datasets. They contained approximately 4 years of records, from 2015 to 2019. Thus, the employment of cross-validation was considered unnecessary and a simple train and test set split was effectuated. The proportions for the subsets creation was chosen to be around 80-70% for the train set and 20-30% for the test set according to guidelines in literature (Liu and Cocea, 2017; Shmueli et al., 2017). For both datasets, the test set was set to comprehend the last 40 weekly sales recorded. Considering the forecasting horizon (set to 8 weeks) the partition allowed to produce 33 forecasts for each product. Given the total amount of products contained in both Datasets after preprocessing (equal to 580), 19140 individual predictions were produced with each of the methods included in the analysis.
   Different algorithms require different inputs. Thus, once train and test sets were defined, they were modeled in order to adapt to each of the methods applied in the study. Three different families of methods created the need for three diverse formats in order to meet the input requirements (Table 3):
   *Direct multistep forecasting ML methods*: the number of predictor variables was defined as equal to 52, with the intention of including an whole year of historical sales as input in order to allow the recognition of annual seasonality. The train set was then reorganized, following the instructions found in literature (Vandeput, 2018), to assume the form of more than 80000 records made by 52 points time-sequences. To each record of predictor variables, 8 labels (or dependent variables) were associated consisting in the 8 weekly sales following the time sequence.
   *Indirect multistep forecasting ML methods:* similarly to the previous case, the records were set to contain 52 points time-sequences. Due to the

recursive approach to multistep forecasting, only one dependent variable was associated with the record.

*Statistical methods:* differently from ML methods, the statistical methods don't require an independent training phase. The parameters which need tuning are fitted on the time series when the forecast is needed. The formatting of the algorithmic input aimed at keeping the comparison between the different families of methods consistent. The dataset was rearranged in order to have time sequences comprehending all historical sales of the concerned product until the point of forecast. The points of forecast were selected in order to mirror the 19140 predictions in the test set of ML methods.

| Family of methods | Independent Training | N. of inputs | N. of outputs |
|---|---|---|---|
| **Indirect ML** | yes | 52 | 1 |
| **Direct ML** | yes | 52 | 8 |
| **Statistical** | no | All demand history before the forecast | 8 |

*Table 3: Data partition driving factors*

## 3.2 Algorithms employed and settings

7. *Choose the data mining techniques to be used.*

As anticipated in Chapter 2, the selection of the methods to test was quite restricted by the limitation of the computational power available. In fact, due to the high power required for their optimization, DL methods were excluded from the analysis. This study should in future be extended to such techniques, which have proved to be valuable tools for forecasting (Kong et al., 2019; Zhao et al., 2017). The methods selected for the study were a set of Decision Trees (DTs) based ensembles chosen to test different ensembling techniques and growing complexities.

*Regression Trees (RT):* also called Classification and Regression Trees (CART) are the simplest method of the Decision Tree's family and represent a single DT with as many nodes as the number of inputs.

*Random Forest (RF):* is an ensemble ML method which is based on the bagging technique (see Chapter 2.4) but operates a further randomization in order to reduce similarities between the trees. It involves in fact, not only the use of bootstrapping to train multiple trees, but also the use at each node split of one feature selected based on the "most discriminative" principle from a limited set of features (Liaw and Wiener, 2002).

*Extremely Randomized Trees (ETR):* makes use of bagging applying a principle extremely similar to the one used for RF. In fact the methods work essentially in the same way, but while in RF the feature is selected from the subset using the "most discriminative" principle, in ERT the features are selected completely randomly (Geurts et al., 2006).

*Extreme Gradient Boosting (XGBoost):* is a typology of Boosting ensemble method which employs  a gradient descent optimization technique on a differentiable, arbitrary loss function (Friedman, 2002). It has proved to be a valuable method in the competitions where it was proposed (Sandulescu and Chiru, 2016; Volkovs et al., 2017) .

As reported in Table 4, all methods apart from XGBoost were employed both considering a direct and indirect approach to multistep forecasting. The open source python package employed for XGBoost did not allow multiple outputs. The direct approach was therefore not feasible.

| Method | Direct | Indirect |
|---|---|---|
| **RT** | yes | yes |
| **RF** | yes | yes |
| **ETR** | yes | yes |
| **XGBoost** | no | yes |

*Table 4: Multistep forecasting approaches*

8. *Use algorithms to perform the task.*

   One of the ML challenges which were introduced in Chapter 2.3, is the complexity of parameters selection in order to have the best performing method possible. For what concerns most of the ML methods, it is in fact required for the user to set several constraints which define the boundaries of the structure then customized by the data. For example, the parameters in input could be the maximum depth of the DTs, the number of leaves on each node, the number of features considered for each node (e.g. for RF).

   Each of the methods employed was optimized through a function which allows a Random Search (RS) of the optimal hyper-parameters to be given as input to a ML method. This function, compared to the alternatives, allows a good quality optimization with low computational expenses (Bergstra and Bengio, 2012). Thus, it was not necessary to manually select the best parameters for the regression task, but only to define a range of possible values they could take. The range of values employed for the methods in this study are reported in Table 5 and 6, and they were set based on the guidelines given by Vandeput (2018).

| Parameters | RT | RF | ETR |
|---|---|---|---|
| **max_depth** | 5-15 | 6-11 | 8-15 |
| **min_samples_leafs** | 5-20 | 5-15 | 2-10 |
| **max_features** | / | 3-8 | 6-11 |
| **max_samples_split** | / | 5-15 | 2-10 |

*Table 5:Parameters' range for RT, RF and ETR*

| Parameters | XGBoost |
|---|---|
| **max_depth** | 4-12 |
| **colsample_bylevel** | [0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9] |
| **learning_rate** | [0.001, 0.005, 0.01, 0.025, 0.05, 0.1] |
| **n_estimators** | 1000 |
| **subsample** | [0.2, 0.3, 0.4, 0.5, 0.6, 0.7] |

*Table 6: Parameters' range for XGBoost*

# Chapter 4: Full cross-sectional forecasting

*The preprocessing procedure described in Chapter 3 allowed to have clean data ready to apply the methods on. The scope of this chapter is, on the first place, to introduce the reader to the requirements and challenges of evaluating the performance of several different methods while forecasting on a large and diverse dataset. On this matter, the benchmarks, accuracy measures and evaluation system employed are explained. In the second section of this chapter, cross-sectional approach to forecasting with ML methods is applied to both datasets and results are shown and critically analyzed.*

## 4.1 Statistical benchmarks

In order to evaluate the performance of the ML methods, several statistical methods were selected in order to have reliable benchmarks. The benchmark methods selected were of various entity. On one hand simple methods, classically used as benchmarks were chosen. These methods rely on really simple logics in order to produce the forecasts. Here listed the methods employed:

*Naïve:* is the simplest way to produce a forecast. It is based on the belief that what happened today (or this week) will also happen tomorrow (or next week) in the same way. Thus, $y_{t+1=}y_t$. In the experimental setting considered where a multistep forecast is needed, the method can be described by the following formula:

$$y_{t+i} = y_t \quad \text{for} \quad i = 1, \dots, h$$

*Moving Average (MA):* it is a simple methodology which was extremely diffused in the past and even know represents the starting basis for many of the most advanced methods (Hyndman and Athanasopoulos, 2018). Moving averages methods are used for many application (Hyndman, 2011) but, when a forecasting task is involved, they consist in estimating the future value of time series averaging the *k* most recent values. These averages can be arithmetic or weighted depending on the practitioner's preferences. The choice for this study was to generate the forecast through the arithmetic average of the demand during the latest 3, 5, 12 and 52 weeks and along all the product lifetime. In the future chapters, these methods will be addressed as MA3, MA5, MA12, MA52 and MA_all respectively. For a multistep setting the forecast will be created as follows:

$$y_{t+i} = \frac{1}{k}\sum_{j=0}^{k-1} y_{t-j} \quad \text{for } i = 1, \dots, h$$

*Seasonal Naïve (SNaïve):* the forecasts, instead of assuming the values of the latest time step, refer to the time step one seasonal cycle past. Since in the concerned case, the seasonal cycle is a yearly cycle made of 52 weeks, the associated formula will be the following one:

$$y_{t+i} = y_{t+i-52} \quad \text{for } i = 1, \dots, h$$

*Seasonal Moving Average (SMA)*: similarly to Seasonal Naïve, it refers to the values one seasonal cycle past, but operates a centered moving average to obtain the forecast desired. This method was not previously used in literature and can be considered an originality trait of this study. The only way of application considered was the setting where $k = 3$, therefore:

$$y_{t+i} = \frac{1}{3}\sum_{j=-1}^{1} y_{t+i-52+j} \quad \text{for } i = 1, \dots, h$$

*Linear Regression (LR):* the last simple benchmark method employed in the analysis is Linear Regression. This tool has the characteristic trait to be claimed both from the statistical and the ML field (Hastie et al., 2009). It is in fact the method which establishes the basis for the autoregressive statistical methods category but, can also be reconducted to the specific case of a feedforward NN with a linear activation function.

On the other hand, also other more sophisticated methods were employed. This was done in order to provide a term of comparison to assess the performance of ML methods compared to those statistical methods which are most commonly employed in industry. Only two methods have been employed as they can be considered representative of their categories (which have been introduced in Chapter 1.6).

*Seasonal Autoregressive Integrated Moving Average (SArima):* described in Chapter 1, it is able to consider level, trend and seasonality in the generation of the forecasts. It was implemented through the function available in R (Hyndman and Khandakar, 2007) which automatically fits the 6 parameters needed without the need for user intervention.

*Holt-Winters (HW):* also known as triple exponential smoothing, it is equally capable to handle both trend and seasonality. For this method was chosen a multiplicative setting for modelling seasonality. This choice was determined by the high dependency between the level and the seasonality qualitatively examined in the preprocessing phase.

## 4.2 Comparing results

As introduced in Chapter 3 while describing the train and test set creation, many forecasts were produced for each product. Considering 33 forecast (each of 8 time points) for each product, 5907 and 13563 predictions were made with each method for Dataset A and Dataset B respectively. It was therefore necessary to find a way to evaluate the accuracy not only of one prediction, but across all predictions of all products in the dataset.

This awareness dictated some of the boundaries for the choice of the accuracy measure to employ in the study. It had in fact not only to be resilient to null historical sales in the forecasting horizon (frequent mainly in Dataset A), but also to be comparable across products which would have presented different average values of demand. Thus, the accuracy measure was set to be dimensionless.

Of the accuracy measures commonly used in practice (Vandeput, 2018), Mean Absolute Percentage Error (MAPE) was not chosen because of the problems handling zero-demand, while MAE and RMSE where discarded due to their dimensionality characteristics.

$$MAPE = \frac{1}{h}\sum_{i=1}^{h}\frac{|e_{t+i}|}{y_{t+i}}$$

$$MAE = \frac{1}{h}\sum_{i=1}^{h}|e_{t+i}|$$

$$RMSE = \sqrt{\frac{1}{h}\sum_{i=1}^{h}e_{t+i}^{2}}$$

To cope with the dimensionality problem of MAE and RMSE it is possible to apply a variation to the formula, which, dividing by the average real value in the forecasting horizon, normalizes the result, making it comparable across different products:

$$MAE\% = \frac{\frac{1}{h}\sum_{i=1}^{h}|e_{t+i}|}{\frac{\sum_{i=1}^{h}y_{t+i}}{h}} * 100 = \frac{\sum_{i=1}^{h}|e_{t+i}|}{\sum_{i=1}^{h}y_{t+i}} * 100$$

$$RMSE\% = \frac{\sqrt{\frac{1}{h}\sum_{i=1}^{h}e_{t+i}^{2}}}{\frac{\sum_{i=1}^{h}y_{t+i}}{h}} * 100$$

MAE% and RMSE% where both chosen to evaluate the results.

Each of the methods performs the optimization on a specific objective function. This function is generally pre-selected and optimized for the algorithm of concern, but in some cases, it can also be modified by the user changing the preferences of the python package. Even considering this possibility, in this study it was decided to employ the pre-selected objective function. Since in most of the cases it was either the MAE or the RMSE, a third criticality was identified: the fictious improvement in performance according to an accuracy measure for those method which use it also as objective function. This issue was tackled with the decision to consider both indicators at the same time, averaging the MAE% and the RMSE%:

$$KPI\% = \frac{\dfrac{\sum_{i=1}^{h}|e_{t+i}|}{\sum_{i=1}^{h}y_{t+i}} + \dfrac{\sqrt{\frac{1}{h}\sum_{i=1}^{h}e_{t+i}^{2}}}{\frac{\sum_{i=1}^{h}y_{t+i}}{h}}}{2} * 100$$

## 4.3 Results

The forecasts were finally generated, evaluated and ranked as previously discussed. As expected, due to the great diversity between Dataset A and Dataset B, the results are considerably different depending on the dataset involved.

### 4.3.1 Dataset A

| Rank | Method | KPI% | MAE% | RMSE% | Rank | Method | KPI% | MAE% | RMSE% |
|---|---|---|---|---|---|---|---|---|---|
| 1 | **SNaïve** | **30.6** | 25.1 | 36.0 | 11 | **LR** | **76.3** | 71.4 | 81.2 |
| 2 | **SMA** | **33.1** | 27.9 | 38.3 | 12 | **MA3** | **77.4** | 73.6 | 81.2 |
| 3 | **Naïve** | **33.7** | 27.9 | 38.3 | 13 | **LR_1SA** | **79.9** | 73.3 | 86.6 |
| 4 | **MA_all** | **35.9** | 29.6 | 42.3 | 14 | **XGBoost_1SA** | **97.8** | 94.2 | 101.4 |
| 5 | **ETR** | **48.0** | 45.1 | 50.8 | 15 | **SArima** | **106.2** | 100.5 | 111.9 |
| 6 | **MA52** | **51.7** | 48.6 | 54.8 | 16 | **MA5** | **116.2** | 114.4 | 119.4 |
| 7 | **SES** | **68.8** | 64.8 | 72.7 | 17 | **RT_1SA** | **121.8** | 112.7 | 131.0 |
| 8 | **MA12** | **72.4** | 68.4 | 76.5 | 18 | **RT** | **135.1** | 128.4 | 141.8 |
| 9 | **RF** | **73.5** | 71.3 | 75.8 | 19 | **RF_1SA** | **178.6** | 173.2 | 184.1 |
| 10 | **HW** | **74.5** | 72.4 | 76.6 | 20 | **ETR_1SA** | **181.0** | 177.9 | 184.1 |

*Table 7: Dataset A; Methods' ranking by performance*

Table 7 shows the results obtained for Dataset A. The acronym "1SA" stands for "1 step ahead" and refers to the indirect multistep forecasting approach. In order to interpret these results, it is essential to keep in mind the characteristics of the datasets which strongly influence the performance of every method considered. Dataset A contains product which, due to the nature of the company, present a time history of sales with an extremely high random component. Furthermore, it is characterized by high lumpiness and a significantly intermittent demand. These characteristics make up for a really complex forecasting environment, where patterns in historical sales are hidden and difficult for methods to understand. It is for this reason that methods which rely on seasonality and trend patterns do not perform well (see Holt-Winters and SArima) compared to methods which rely on simple logics. The predominance of simple methods when forecasting on complex datasets is not a new concept in literature (Hyndman and Athanasopoulos, 2018). ML methods show unsatisfyingly low accuracy, with the only exception of ETR when applied with a direct forecasting approach. More in general, almost every method considered in the analysis scores a lower KPI% when applying a direct multistep methodology.

Any attempted assessment on the comparison among the ML methods employed results out of place. It is in fact necessary to state that Dataset A with these characteristics strongly limits the outcome of the analysis due to the high random component of the time series the forecasts are based on.

### 4.3.2 Dataset B

Dataset B, on the other hand, thanks to the highly aggregated demand, presents more stable and regular time series. It can be noticed from the results reported in Table 8, that with these settings Holt-Winters and SArima which are supposed to perform well, actually show unsatisfying performances. The comparison in fact, places Holt-Winters and Arima, past the second half of the ranking, losing position even to methods as simple as SNaïve, MA_all or SMA. These unconventional results, can be justified by both statistical methods’ issues in handling seasonality for weekly data on an yearly seasonality (Hyndman and Athanasopoulos, 2018). Such conditions, which were set by the consulting company, highlight a strong limitation of the statistical methods compared to the ML ones. Some methodologies have been designed to cope with this problem: for example, instead of applying Holt-Winters directly, it is suggested a deseasonalization with the subsequent application of Holt's Methods. Nonetheless, the application of a further preprocessing step would have gone beyond the scope of the study to evaluate methods performances applied to rough time series data.

A second observation to be done on the results concerns the good performance of ML methods applied with the indirect multistep forecasting approach (1SA). Apparently, the more complex the method is, the better it performs. The only exception to the trend is RT_1SA which managed to outperform both ETR_1SA and RF_1SA. It must be underlined as a limitation of the comparison that all of these methods show variance in the results. In order to reduce this variance, multiple experiments would be recommended. Due to the limited computational capacity available for the study, running the test many times was not achievable. Thus, the results must be carefully evaluated: for example, RT_1SA can apparently perform better than the two ensemble methods, but its results should be considered less reliable because of the higher variance. As explained in Chapter 3 in fact, ensemble methods, being the average of several RTs (weak learners), should provide a more reliable result.

Furthermore, it is reasonable to affirm that ML methods, applied employing cross-sectional training, perform better than statistical ones. The validity of this statement must be anyway weighted over a couple of factors:

1. As mentioned above, the results are the outcome of one single experiment. More should be performed in order to prove ML methods' superiority with statistical significance.
2. The experiments are performed in a forecasting environment where the statistical methods cannot show their potential. Thus, the stated superiority should be only considered circumstantial.

| Rank | Method | KPI% | MAE% | RMSE% | Rank | Method | KPI% | MAE% | RMSE% |
|---|---|---|---|---|---|---|---|---|---|
| 1 | **XGBoost_1SA** | **18.9** | 16.1 | 21.7 | 11 | **SArima** | **48.6** | 45.0 | 52.2 |
| 2 | **RT_1SA** | **19.2** | 16.4 | 22.0 | 12 | **LR** | **49.3** | 42.6 | 56.0 |
| 3 | **ETR_1SA** | **19.7** | 17.3 | 22.1 | 13 | **RT** | **50.6** | 44.7 | 56.5 |
| 4 | **RF_1SA** | **21.9** | 19.6 | 24.2 | 14 | **MA52** | **51.4** | 46.2 | 56.6 |
| 5 | **LR_1SA** | **26.7** | 24.9 | 28.5 | 15 | **HW** | **52.5** | 47.0 | 58.0 |
| 6 | **RF** | **38.8** | 34.2 | 43.4 | 16 | **MA3** | **53.0** | 47.9 | 58,.1 |
| 7 | **ETR** | **40.8** | 34.3 | 47.3 | 17 | **MA5** | **53.8** | 48.7 | 58.9 |
| 8 | **SNaïve** | **43.8** | 40.3 | 47.3 | 18 | **Naïve** | **57.3** | 52.1 | 62.5 |
| 9 | **MA_all** | **47.7** | 44.1 | 51.3 | 19 | **SES** | **64.7** | 60.1 | 69.3 |
| 10 | **SMA** | **48.5** | 44.1 | 52.9 | 20 | **MA12** | **82.5** | 77.9 | 87.1 |

*Table 8: Dataset B; Methods' ranking by performance*

Noteworthy is the difference in value between the MAE% and RMSE% considered separately. The MAE% results consistently lower thanks to its higher resistance to the outliers (Vandeput, 2018). In order to verify the suitability of the KPI%, defined as the average of the two aforementioned indicators, it was decided to understand how these two were linked. Fig. 15 and Fig. 16 represent the performances computed with the two accuracy measures for Dataset A and Dataset B respectively. Since in both of the cases is observable a linear relationship (standing for a similar behavior), it was concluded that the KPI% was indeed appropriate to rank the methodologies.

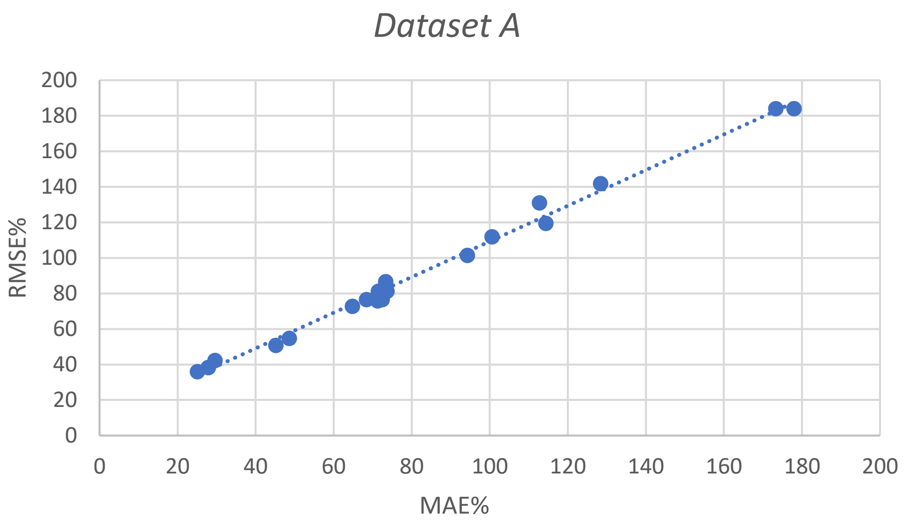

*Figure 15: Dataset A; Relationship MAE%-RMSE%*

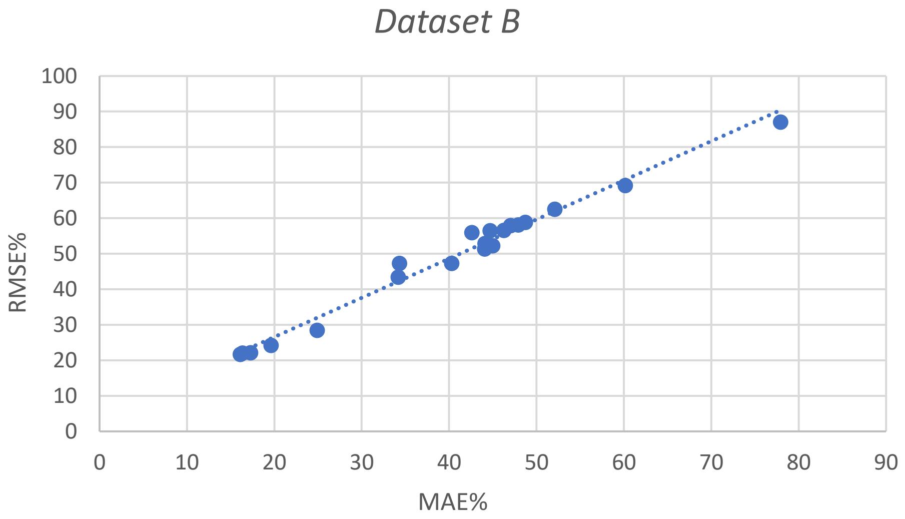

*Figure 16: Dataset A; Relationship MAE%-RMSE%*

## 4.4 Critical analysis

The results coming from *Dataset A* are not perfect to understand the real potential of forecasting methods due to the high random component. They can nonetheless be of great use to understand, through the comparison with Dataset B, some underlying phenomena which affect ML performance.

The first remark is about the difference in the findings regarding the approach to conduct multistep forecasting. It is clear, looking at the result from Dataset B, that the indirect approach greatly outperforms the direct one. This is due to the higher specificity of the ML methods in predicting 1 output instead of 8 at the same time (Hamzaçebi et al., 2009). This principle is not as effective for dataset A, where supposedly the high randomness causes a high error which is reinserted in the model, making it diverge from the correct forecasts.

While the randomness of the time series is for sure a reason for the bad performances related to Dataset A, it could be not the only one. Following the observation of Bandara et al. (2020) (see Chapter 2.1), a second reason for the disappointing performance of ML methods could be the diversity of time series used to train the cross-sectional forecasting model. In order to test this hypothesis, the correlation between randomness, quantified by the Coefficient of Variation (CV) and the Average Inter-Demand Interval (IDI) (see Chapter 1.6), and the product-specific performance was computed across both the datasets. Analyzing the following results, it is obvious that, while for IDI the correlation results dubious, there is a relationship between the CV and the performance of the ML method considered for the analysis. Figure 17 and 18 report the scatterplots CV-KPI% and IDI-KPI% respectively for Dataset A and Dataset B, together with the numerical value of the Pearson Correlation Coefficient (PCC) associated to the plot.

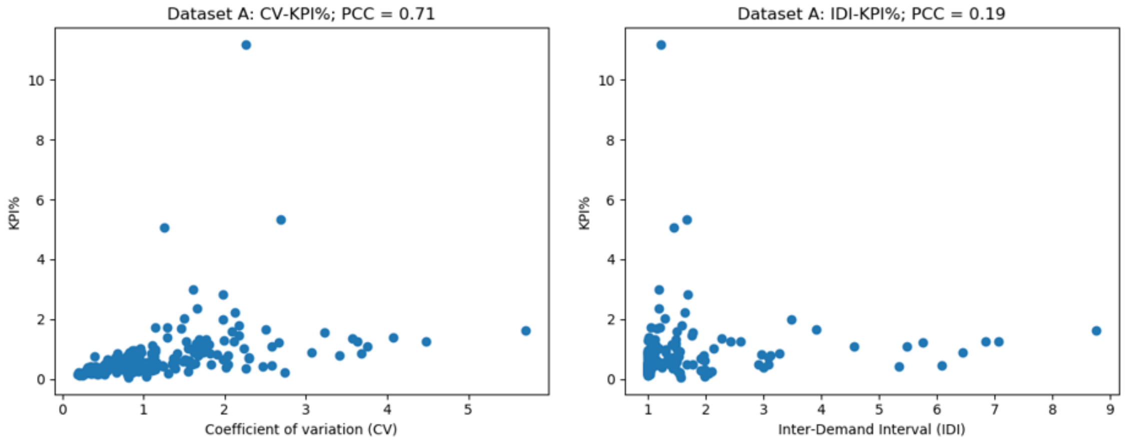

*Figure 17: Dataset A; Scatterplots KPI%-CV (on the left) and KPI%-IDI (on the right)*

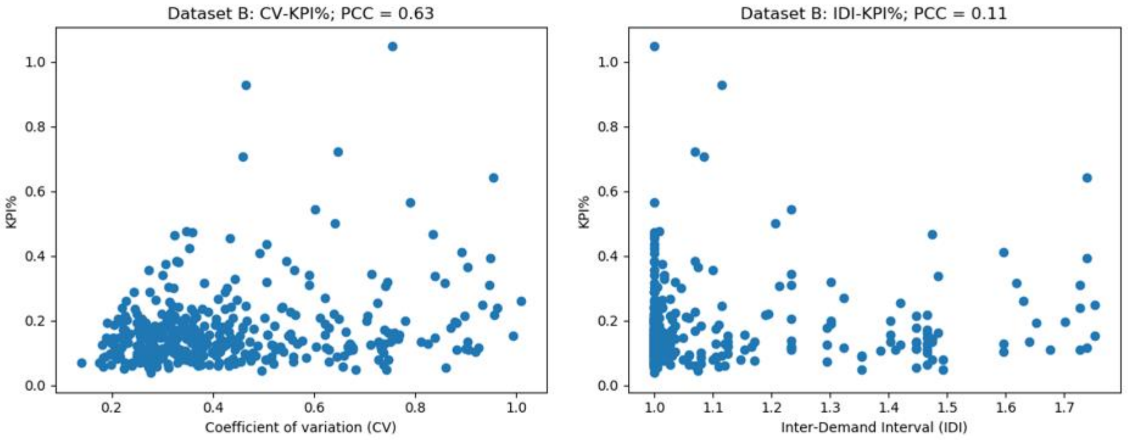

*Figure 18: Dataset B; Scatterplots KPI%-CV (on the left) and KPI%-IDI (on the right)*

Now that the correlation between performance and CV has been proved, it is necessary to show that it is not the only factor determining the difference in accuracy between Dataset A and Dataset B. To do so, it was decided to consider only the products of Dataset A which showed less noise, meaning the products having a CV lower than 1 (around 50% of Dataset A). For these items, the average KPI% was computed from the same forecasts of the previous experiment. In Table 8 the results are reported, listing in order of performance the forecasting accuracies achieved.

| Rank | Method | KPI% |
|---|---|---|
| 1 | **XGBoost_1SA** | 45.9 |
| 2 | **RF_1SA** | 57.4 |
| 3 | **RT_1SA** | 60.0 |
| 4 | **ETR_1SA** | 64.0 |
| 5 | **RF** | 66.8 |
| 6 | **RT** | 67.3 |
| 7 | **ETR** | 40.8 |

*Table 9: Dataset A (subset with CV<1), Methods' ranking by accuracy*

Two different observation can be made regarding the results showed in Table 9:

1. Considering the most regular group of products (CV<1) from Dataset A, the indirect multistep approach seems again to consistently outperform the direct one, thus strengthening the thesis previously stated.
2. The evaluation of the previous experiment's forecasts over the subset restricted to the most regular time series, resulted in lower KPI% compared to the KPI% obtained over the whole dataset (see Table 7).
3. Even for this selection of more regular items from Dataset A, the forecasts result extremely less accurate than for Dataset B.

The second statement can be explained by the higher heterogeneity of the products contained in Dataset A compared to Dataset B. This characteristic is graphically represented by the boxplot showed in Fig. 19. Considering these hypotheses, the importance of a clustering procedure in order to group similar products before applying cross sectional forecasting becomes clear. This procedure will indeed be the topic of the following Chapter.

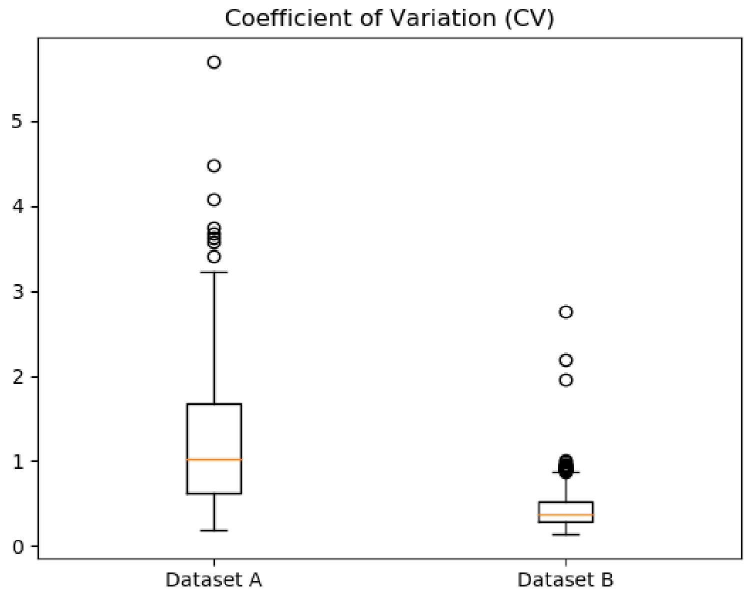

*Figure 19: CV values distribution for the two datasets*

# Chapter 5: Cluster influence on cross-sectional forecasting

*As mentioned previously, the scope of this Chapter is to investigate the best approach to cluster time series for cross-sectional forecasting. Given the lack of literature on the topic, several options were tested, and the results compared among them.*

*This chapter will be divided in two sections. The first one will treat the description of the clustering approaches tested. In the second one, results will be shown and the differences between the approaches analyzed.*

## 5.1 Clustering approaches

The problem of clustering time series was already discussed in Chapter 2.2. As anticipated, in this section the reader will be introduced to the four clustering approaches which were chosen to partition the dataset.

### 5.1.1 PCA-based clustering

The first approach examined, adopted the same time series clustering methodology used by Bandara et al. (2020), Wang et al. (2006) and Räsänen and Kolehmainen (2009), namely whole sequence time series clustering by means of a feature-based approach. Similarly to the model proposed by Bandara et al. (2020), the features considered for the clustering (see Table 10) where extracted through the *tsmeasures* function from the *anomalous_ACM* package tool available on R.

| Feature | Description |
|---|---|
| **lumpiness** | Variance of remainder |
| **entropy** | Spectral entropy from ForeCA package |
| **ACF1** | First order of autocorrelation |
| **lshift** | Level shift on a rolling window |
| **vchange** | Variance change |
| **cpoints** | Number of crossing points |
| **fspots** | Flat points using discretization |
| **trend** | Strength of trend |
| **linearity** | Strength of linearity |
| **curvature** | Strength of curvature |
| **spikiness** | Strength of spikiness |
| **KLscore** | Kullback-Leibler score |
| **change.idx** | Index of the maximum KL score |
| **CV** | Coefficient of variation |

*Table 10:List of features extracted from the time series*

Wang et al. (2006) suggested a better clustering performance was achievable employing only a limited set of features. Embracing their observation, for this clustering approach it was decided to select only those features which showed the most variation in the dataset. To perform this dimensionality reduction task, Principal Component Analysis (PCA) (Wold et al., 1987) was selected with an eye of regard for its ulterior application, namely the possibility to visualize the data along the principal components and graphically identify the clusters (if any was present). The inevitable interdependency between the feature employed was not considered as an issue for the applicability of the PCA because of the descriptive and not inferential scope of the analysis (Jolliffe, 2002).

Due to the different units of measure and scales of the various features, it was necessary to conduct a process of normalization before feeding them as input to the PCA. The normalization was operated with the function *StandardScaler* from the *Scikit-learn* package available on Python. According to this function, the input value $x$ is subjected to the following transformation:

$$z = \frac{x - \mu}{\sigma}$$

Where $\mu$ is equal to the mean of the input vector, $\sigma$ to its standard deviation and $z$ is the output value.

The PCA was conducted on the features extracted from both the datasets. For Dataset B, the procedure unveiled the existence of three PCs which alone could explain almost 70% of the variance contained among the products' features (Fig. 20). These PCs are given by a weighted combination of all the features previously listed (Table 10), where the weight of each feature (factor loading) is reported in Fig. 21. Fig. 22 shows a 3D scatterplot where each point represents a time series and the axis correspond to the three main PCs identified by the PCA. A graphical analysis of the plot was sufficient to notice the presence of one single cluster, where most of the products reside. This could be one of the reasons why ML methods, applied through cross sectional forecasting on the entire dataset, show really good performances even without a clustering approach beforehand (see Chapter 4.3).

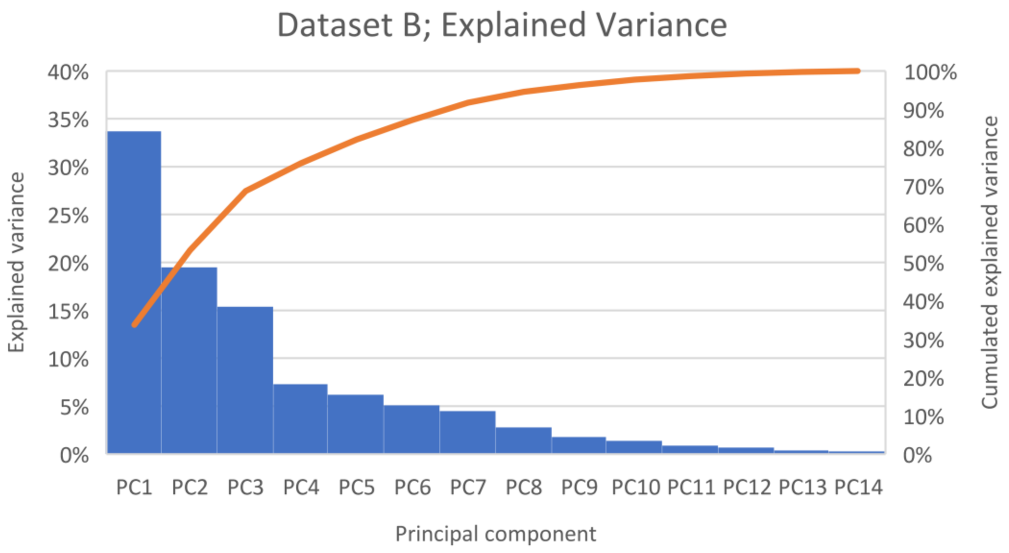

*Figure 20: Dataset B; Percentage of explained variance related to each PC*

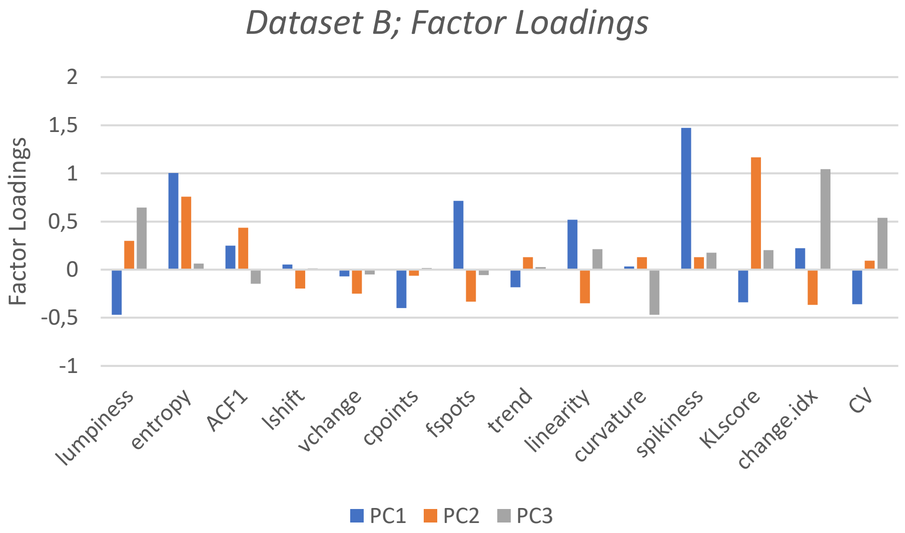

*Figure 21: Dataset B; Factor loadings*

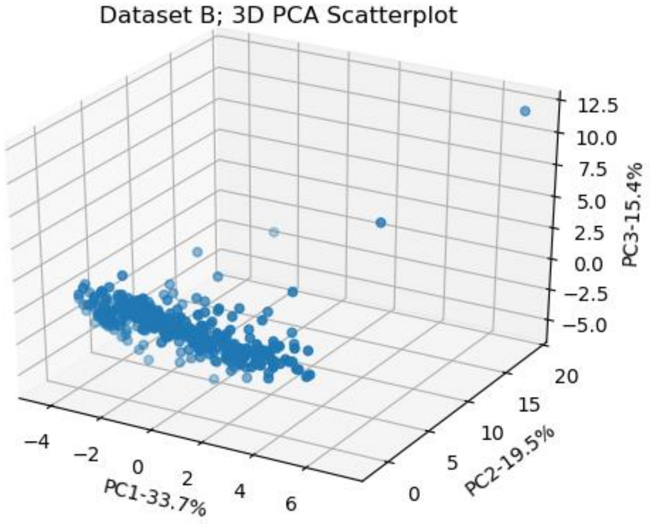

*Figure 22: Dataset B; Distribution of the time series in a 3D space defined by PC1, PC2 and PC3*

The process for Dataset A followed the same procedure as for Dataset B. In this case only two are the PCs which stand out from the group and they account for approximately 60% of the total variance. Fig. 23 shows the percentage of explained variance for each PC. For the two explaining the most variance, their composition is portrayed by the factor loadings (Fig. 24) and the 2D related scatterplot plotted (Fig. 25). While for Dataset B it was possible to identify a cluster, this second dataset's time series seem to take on a sparse distribution from which no insights can be drawn.

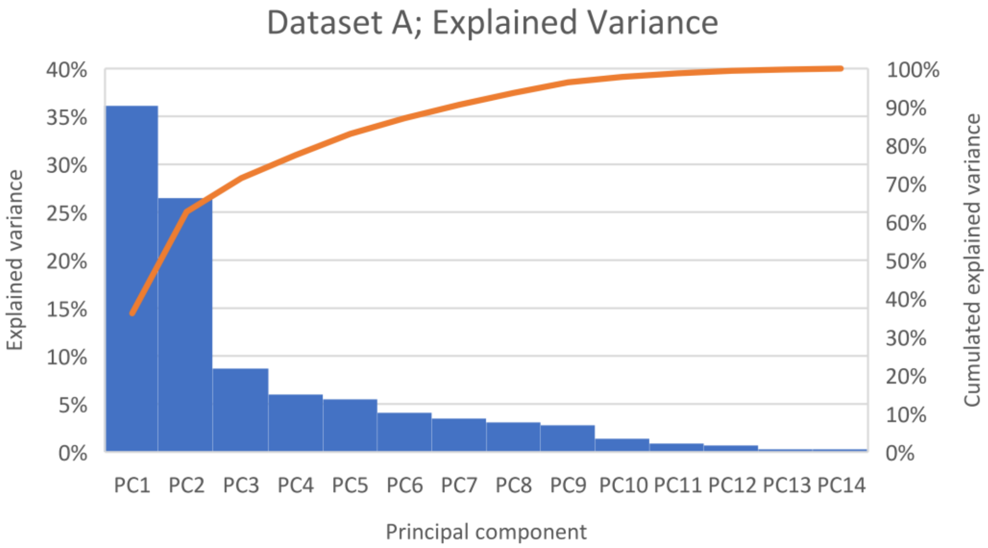

*Figure 23: Dataset A; Percentage of explained variance related to each PC*

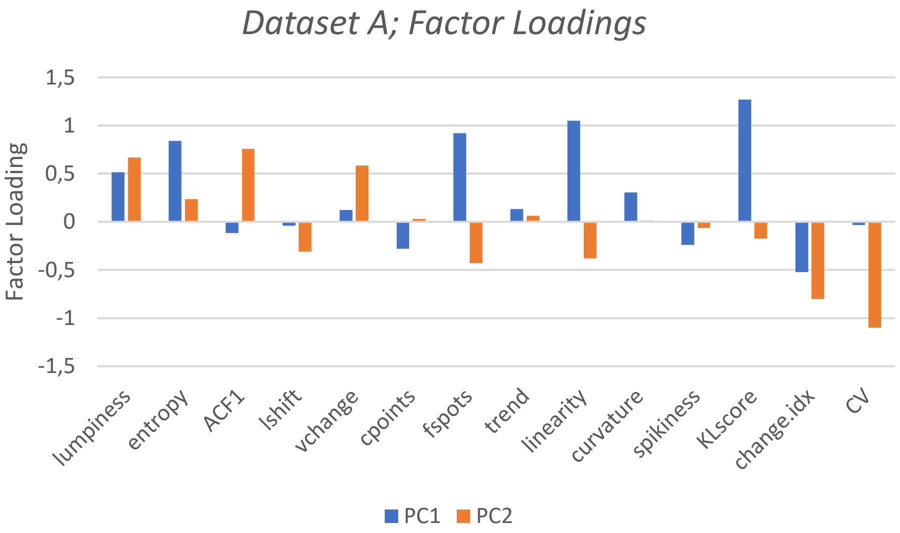

*Figure 24: Dataset A; Factor loadings*

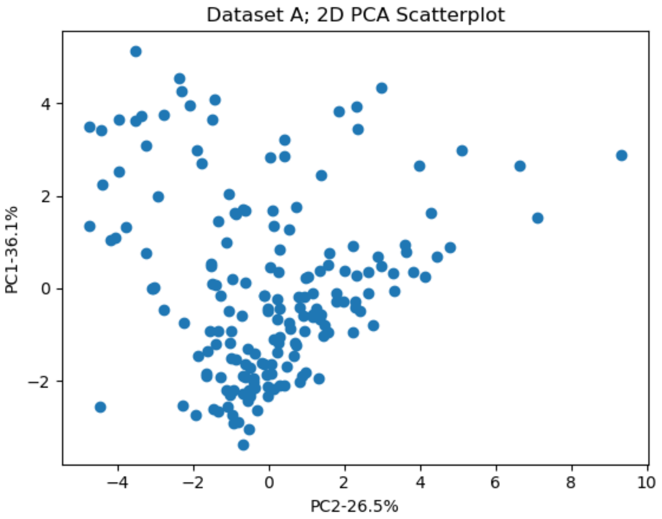

*Figure 25: Dataset A; Distribution of the time series in a 2D space defined by PC1 and PC2*

The creation of a clustering model is a complex process comprehensive of the multiple steps depicted in Chapter 2.2. While the choices regarding the taxonomy and the approach to time series clustering have been already defined, it is still necessary to select the appropriate algorithm, distance measure and linkage method to utilize.

Hierarchical clustering, compared to K-means clustering, does not require the a priori selection of the number of clusters and provides an interpretable graphical representation of the clustering process. Thus, it seemed more appropriate for an investigation on the relation between ML methods performance and dimension/homogeneity of the clusters. The "agglomerative" variant of the algorithm was selected.

Given the necessity to keep the homogeneity inside each cluster as high as possible, Ward's method, which minimizes the increase in variance for each agglomerative step, was chosen as linkage method. Consequence of the principle governing Ward's method, Euclidean distance was the obligated choice for the distance measure.

The combination of Hierarchical clustering with Ward linkage method and Euclidean distance was adopted for this clustering approach as for the following ones.

In Fig. 26, the dendrograms related to the clusters' formation are reported for both datasets. In order to define the clusters, the structures can be cut to a certain height. According to the height selected, a precise number of clusters with the associated number of products is defined. For example, cutting the dendrogram produced for Dataset B at a height corresponding to a Euclidean distance equal to 30, four clusters would be obtained and one of them would be composed only by a few products.

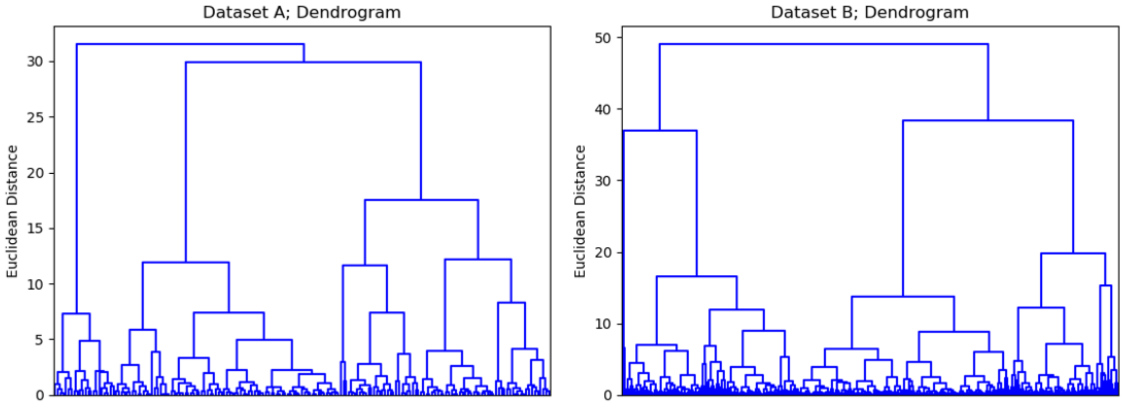

*Figure 26: PCA-based clustering dendrograms*

### 5.1.2 Correlation-based clustering

This second clustering approach is based on the same features listed on the previous paragraph (Table 10) but, differently from the last approach described, PCA is not employed.

The selection of the subset of features was conducted under the assumption that the most relevant features would have been those which have the greatest impact on ML methods' performance. Thus, the results from the large-scale analysis conducted in the first section of this study where employed in order to evaluate the relationship between performance of ML methods and each of the features in

Table 10. Pearson's correlation coefficients (PCC) were computed in order to roughly quantify the effect of each feature on ML methods performance. The results for the two datasets are shown in Table 11 and 12.

Since the reported coefficients were computed based on the accuracy achieved on one single experiment, they are subjected to the same problematics which characterize the results previously shown (Table 7 and 8), namely the randomness in the results. In this case, the goal of the analysis is to roughly quantify the influence of each of the features on the performance in order to select the most influencing ones. Thus, it is reasonable to assume that the randomness of the results would have not greatly affected the feature selection process.

| Method | PCC | Method | PCC |
|---|---|---|---|
| **CV** | 0,364 | **trend** | -0,157 |
| **KLscore** | 0,311 | **entropy** | 0,132 |
| **cpoints** | -0,283 | **fspots** | 0,128 |
| **vchange** | -0,261 | **linearity** | -0,082 |
| **lshift** | -0,259 | **ACF1** | -0,069 |
| **lumpiness** | 0,226 | **change.idx** | 0,019 |
| **spikiness** | 0,222 | **curvature** | -0,010 |

*Table 11: Dataset A; PCC KPI%-Feature*

| Method | PCC | Method | PCC |
|---|---|---|---|
| **lumpiness** | 0,565 | **ACF1** | -0,226 |
| **spikiness** | 0,496 | **fspots** | 0,133 |
| **CV** | 0,430 | **curvature** | -0,073 |
| **linearity** | -0,351 | **lshift** | 0,070 |
| **KLscore** | 0,342 | **vchange** | 0,066 |
| **trend** | -0,289 | **change.idx** | 0,034 |
| **entropy** | 0,263 | **cpoints** | 0,028 |

*Table 12: Dataset B; PCC KPI%-Feature*

The next step consisted in the arbitrary choice of the correlation coefficients' thresholds to establish the employment of the related features in the Clustering procedure. All features were discarded if they presented a Pearson's coefficient lower than 0.2 for Dataset A or 0.25 for Dataset B, leaving with the following list (Table 13):

| Dataset A | Dataset B |
|---|---|
| **CV** | **lumpiness** |
| **KLscore** | **spikiness** |
| **cpoints** | **CV** |
| **vchange** | **linearity** |
| **lshift** | **KLscore** |
| **lumpiness** | **trend** |
| **spikiness** | **entropy** |

*Table 13: List of the selected features*

Clustering is a procedure which is generally performed on $M$-dimensions, each one presenting the same scale. This is for example verified when applying distance-based approaches to time series clustering and all $M$-dimensions stand for the $M$ values assumed by the time series in different time points. In the case of concern, where the $M$-dimensions are related to the $M$ features selected, the axis scales would not be the same, thus the clustering procedure would result biased.

To cope with this problematic all features must undergo a process of normalization, which saw again the employment of the *StandardScaler* function. The feature selected, once normalized, were used to define the M-dimensional clustering space where the same clustering algorithm described in the previous paragraph was applied. The resulting dendrograms are shown in Fig. 27.

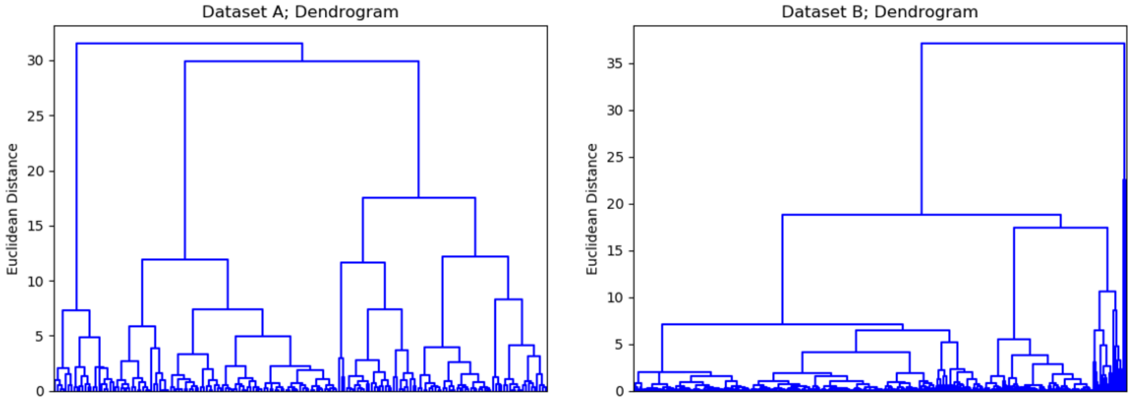

*Figure 27: Correlation-based clustering dendrograms*

### 5.1.3 SI-based clustering

The third and last feature-based approach examined in this study is built on the extraction of the Seasonality Indexes (SI).

As introduced in Chapter 1.4, time series can be decomposed in the Trend, Seasonality and Remainder components. The seasonality component, being the expression of how demand cyclically varies along the year, has a great influence in forecasting methods performances. Thus, it was devised a way to group the products based on their historical sales seasonality patterns.

The *seasonal_decompose* function from the *statsmodels* package on Python was employed to extract the 52 SI which quantitatively describe the influence of the seasonality component in each of the 52 weeks of a seasonal cycle (1 year).

The clustering procedure was then performed in a 52-Dimensional space with a hierarchical methodology, ward linkage function and Euclidean distance measure. Represented in Fig. 28 the dendrograms for the two datasets related to this clustering approach.

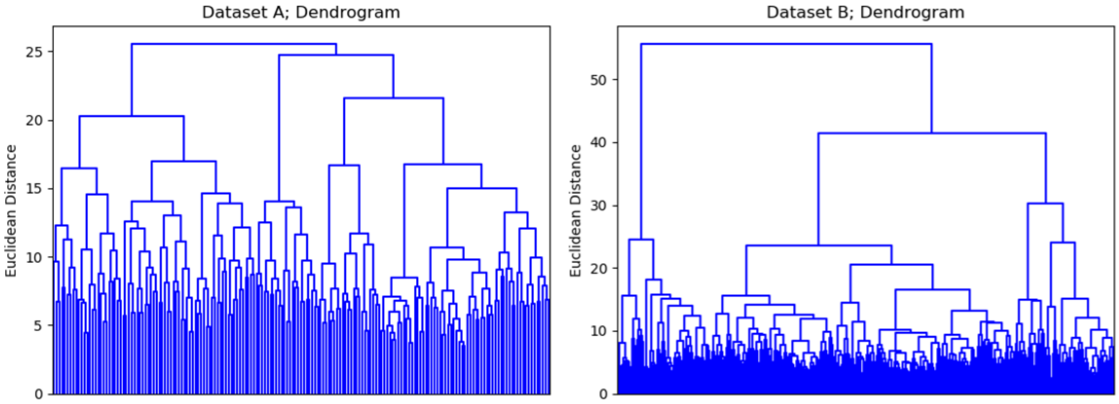

*Figure 28: SI-based clustering dendrograms*

### 5.1.4 Distance-based clustering

The last approach tested relied on a whole time series, distance-based methodology. Meaning, all historical demand values were employed as coordinates in a $M$-Dimensional space, where $M$ is equal to the length of the time history considered in the training set.

This procedure is probably the most diffused way to apply time series clustering, but it presents several disadvantages which were examined in Chapter 2.2. The process is quite straight forward, since no normalization was needed, and the clustering algorithm employed was the same as the previous cases. Fig. 29 shows the dendrograms related to this approach.

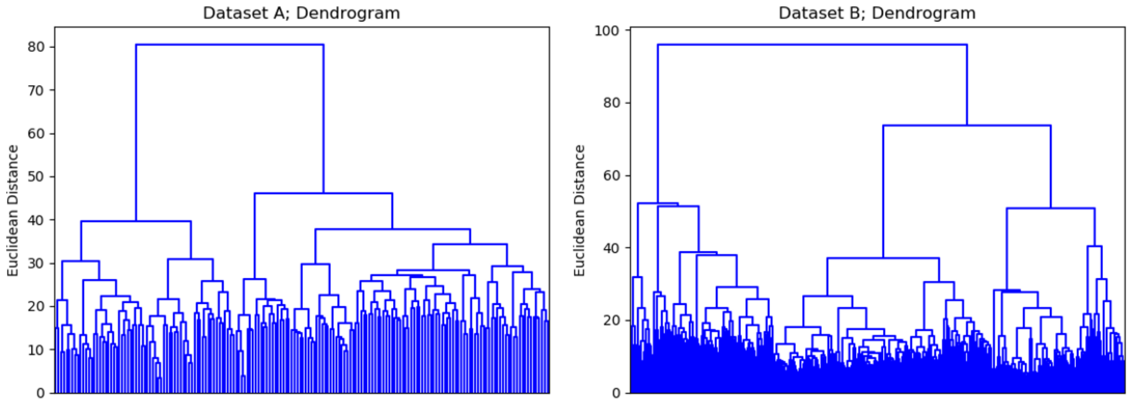

*Figure 29: Distance-based clustering dendrograms*

## 5.2 Results analysis

Since the behavior of ML methods is sealed in a black box and can not be easily interpreted, it is also difficult to establish the optimal clusters' dimension and the most effective way to form them. Lacking therefore a way to establish the optimal procedure, it was decided to extensively run the experiments over all the clustering approaches listed in the previous paragraph.

The dendrograms achieved with the hierarchical clustering approaches previously described, were iteratively cut in order to form $i$ clusters, where $i$ varied from 2 to 70. For each value assumed by $i$, $i$ ML models were trained with the time series related to the products attributed to the associated cluster. The overall accuracy of the dataset was given by the weighted average of the cluster's related accuracy over the number of products in the cluster considered. The iterative procedure is described in Fig. 30.

The iterative computation described, allowed to plot the overall accuracy on the dataset as a function of the number of clusters. From a theoretical point of view, when $i$ assumes a small value, a great number of products compose each cluster and therefore it will be more likely to incur in what was defined as *specification error* (see Chapter 2.1). On the other hand, when the number of clusters becomes too high, only a little data is available for training, thus the models run the risk of incurring in the *overfitting error*. The theoretical model described is graphically supported by Fig. 31.

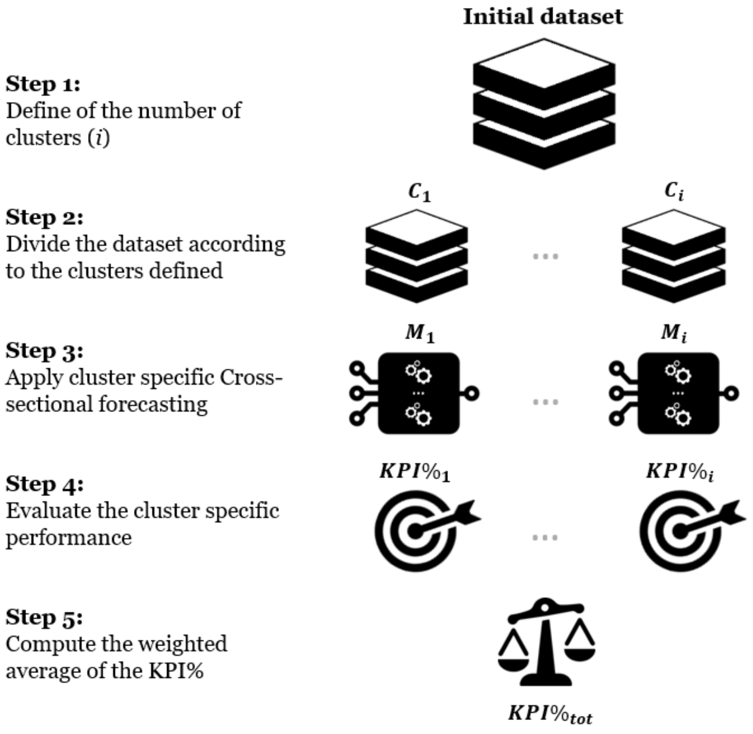
Initial dataset
**Step 1:**
Define of the number of clusters ($i$)
**Step 2:**
Divide the dataset according to the clusters defined
$C_1$
$C_i$
**Step 3:**
Apply cluster specific Cross-sectional forecasting
$M_1$
$M_i$
**Step 4:**
Evaluate the cluster specific performance
$KPI\%_1$
$KPI\%_i$
**Step 5:**
Compute the weighted average of the KPI%
$KPI\%_{tot}$

*Figure 30: Iterative procedure for the evaluation of the clustering approaches*

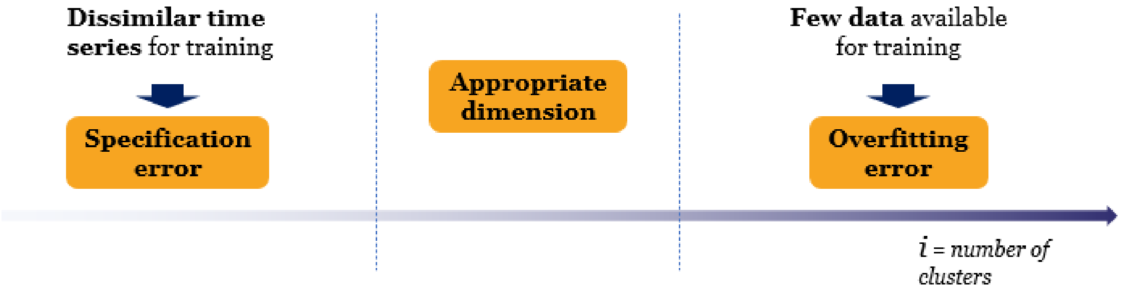

*Figure 31: ML accuracy as a function of the number of clusters*

The problem given by the lack of an objective function to assess the quality of the clusters' formation process was therefore addressed with the evaluation of the coherence between the expected behavior and what the clustering procedure was actually able to generate.

A further issue related to this section of the study is the high processing power required. In fact, in order to extract the results from the iterative procedure described, it was necessary to train a number of models equal to $\sum_{i=2}^{70} i = 2484$ . Thus, these experimental settings were applicable only to the methods which required less training time, namely Regression Trees and Random Forests. Between the two alternatives the choice fell on the employment of RF over RT. This decision was based on the lower variance which theoretically should characterize ensemble methods (see Chapter 2.4). Since, due to the computational requirements of the experiments, multiple runs where not feasible, RTs' results randomness would have risked undermining the outcome of the study. RF were applied through the indirect approach to forecasting, which, when conditions get more stable, has shown more promising results than the direct variant.

### 5.2.1 Dataset B

Starting from Dataset B, the results obtained applying PCA-based, Correlation-based, SI-based and Distance-based clustering approaches are reported in Fig. 32, Fig. 33, Fig. 34 and Fig. 35 respectively. The KPI% obtained through full cross-sectional forecasting with the considered ML method is also graphically represented.

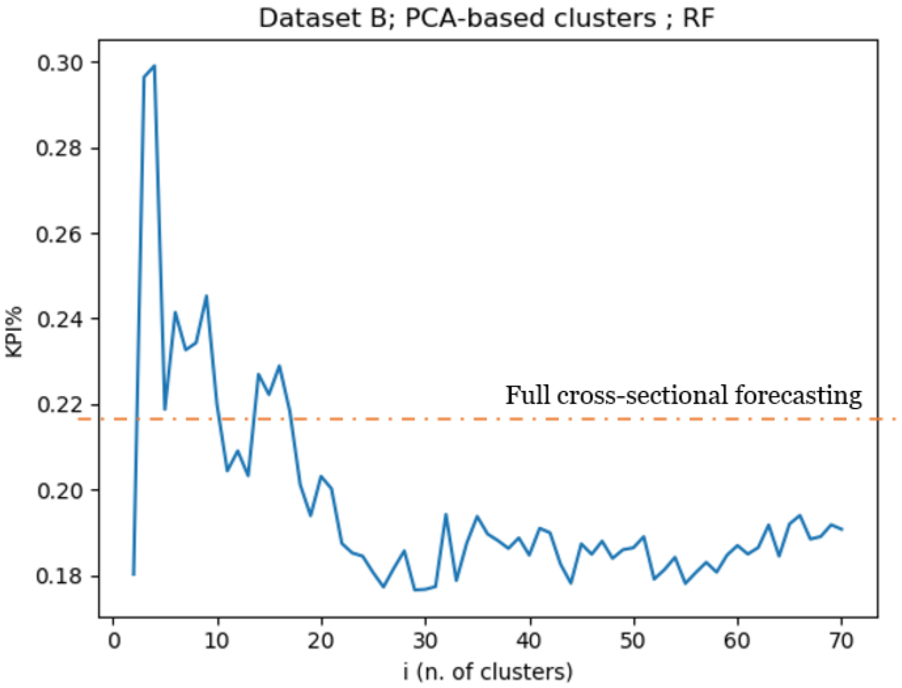

*Figure 32: Average performance of RF on Dataset B as a function of the number of clusters formed used the PCA-based clustering methodology*

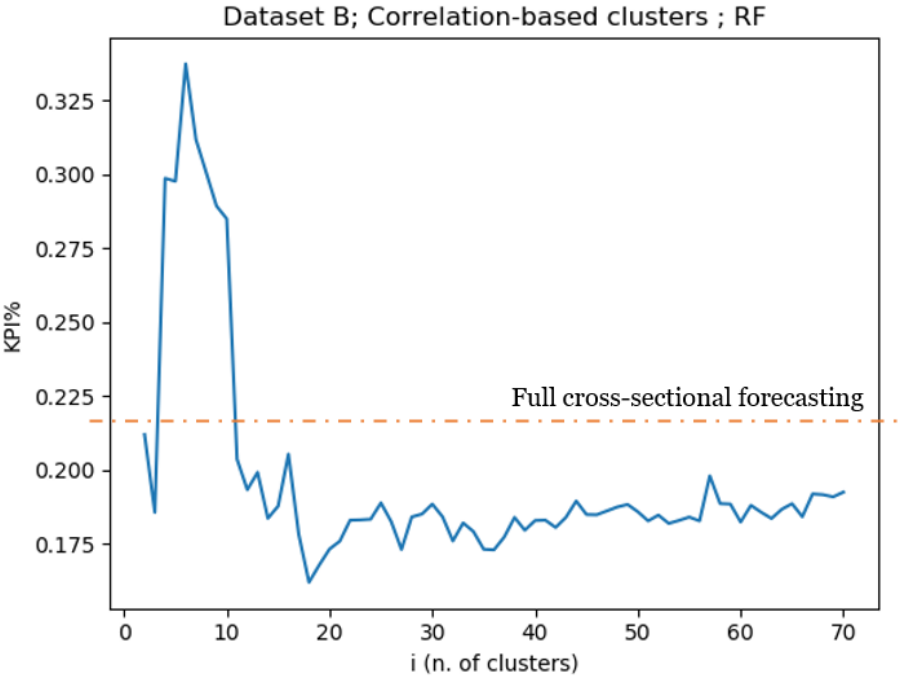

*Figure 33:Average performance of RF on Dataset B as a function of the number of clusters formed used the Correlation-based clustering methodology*

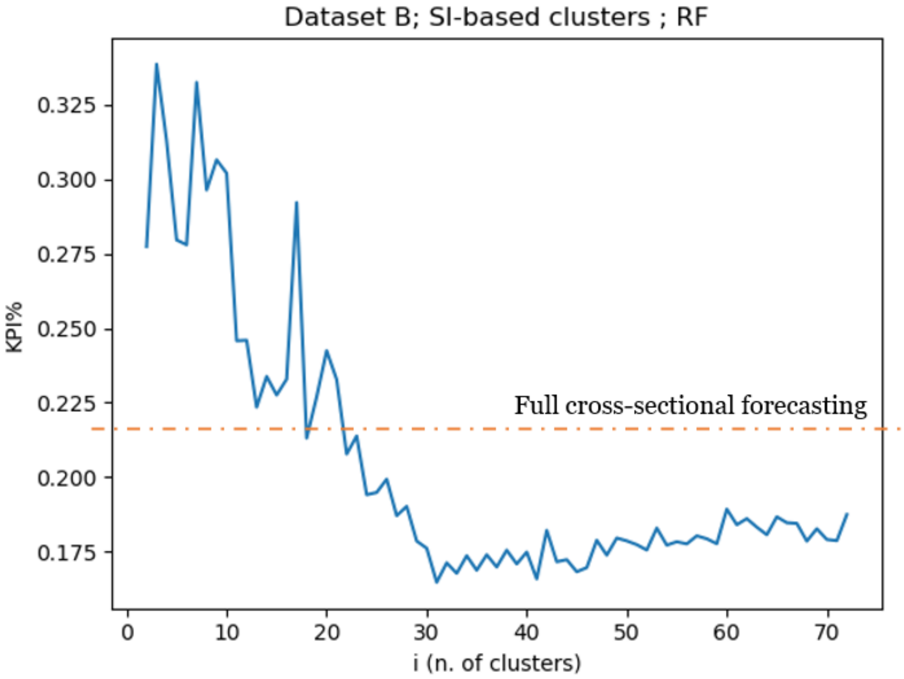

*Figure 34: Average performance of RF on Dataset B as a function of the number of clusters formed used the SI-based clustering methodology*

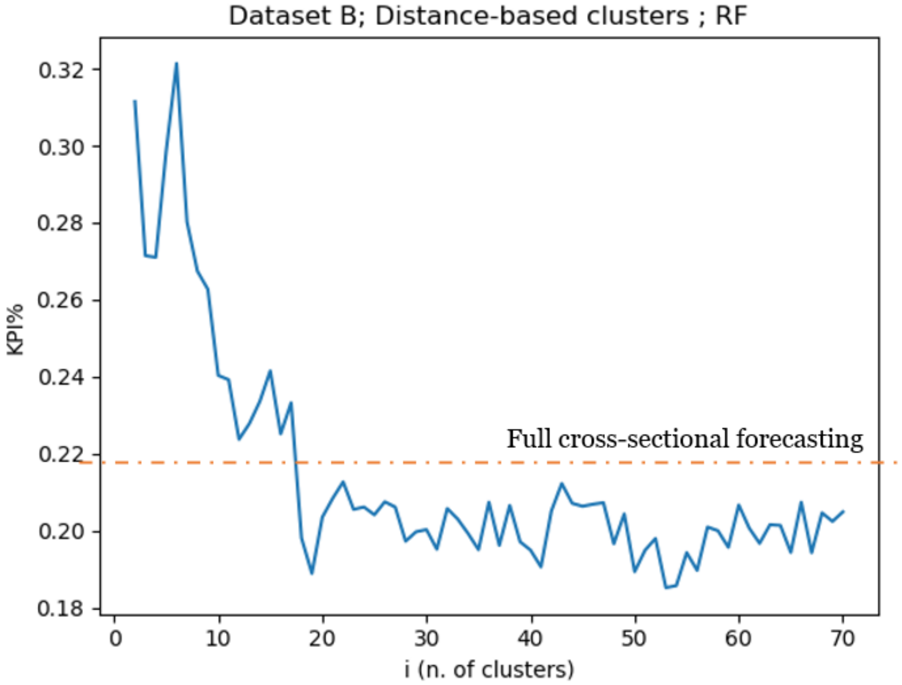

*Figure 35: Average performance of RF on Dataset B as a function of the number of clusters formed used the Distance-based clustering methodology*

In order to give a realistic and unbiased interpretation of the results shown it is necessary to further stress a couple of this Dataset's characteristics already mentioned. Firstly, Dataset B includes 411 products meaning that the average cluster dimension will be equal to $411/i$. It is important to underline that it is only the average cluster dimension and that the clusters' number of attributed products will vary based on the Ward's linkage measure computation in the defined clustering space. The second noteworthy observation previously made comes from the results of the PCA (see Chapter 5.1.1) and consists in the detection of one single cluster which comprehends all of the products.

Other than these reminders, it is important to state that the randomness of the results varies along the plot. The random component can in fact be imagined as a positive or negative factor which provokes a displacement of the computed accuracy from the "real" one. When the cluster is only one (like in the previous analysis), this randomness heavily affects the accuracy. When the clusters grow in number on the other hand, the cumulated random component given by each model employed tends to average towards zero. Here explained why, in all of the graphs shown, the closer *i* is to 2, the spikier the graph gets.

The increase in error that for each clustering approach verifies when the number of clusters is in the proximity of 5 can be justified with the observation on the PCA analysis mentioned above. Possibly, the division of the dataset in a limited number of clusters would still comport heterogeneous products to be attributed to the same group. On the other hand, this group can rely on less products' historical demand available in the training phase. Since for a higher number of clusters the average accuracy seems to improve, it is possible that smaller clusters in the dataset involved can determine more homogeneous products. These products, even if limited in number, can provide an amount of training data which is enough to train a model in a more homogeneous forecasting environment.

Thus, this behavior would justify the existence of the two minimums identifiable along all of the plots for Dataset B, the first close to the condition of full cross-sectional forecasting, the second for higher values of $i$. This second one, seems to be the global minimum of the function, reflecting the expected theoretical model. The trend for an increasing value of $i$ seems in fact to be characterized by the foreseen worsening in performance. The trend is expected to further penalize forecasting accuracy for $i > 70$ but it was impossible to continue all experiments due to the computational time.

When it comes to comparing the different clustering approaches, SI-based appear to be the best performing one. This assessment is rooted on the following key points:

- The plot related to the SI-based clustering approach reports a regular behavior. This stands for a better clustering logic when it comes to decide how to form the groups. If the aggregation of two clusters (decreasing the number of clusters by 1) heavily affects the average error, it means that that Ward's linkage method (see Chapter 2.2) on the clustering space defined, does not reflect the optimal aggregation strategy. Since Ward's method has been already evaluated to be a good approach for the formation of the clusters, the problem stands in the definition of the clustering space connected to the clustering approach employed. When comparing the outcomes of the different clustering approaches it is also necessary to take into account the natural randomness given by the RF and its influence's indirect proportionality to $i$ (as underlined above). Thus, the evaluation of the irregularities given by the alternative clustering approaches is better when done on a high number of clusters in order to diminish the influence of randomness given by other factors. Both Correlation-based and SI-based clustering methodologies seem to behave better than the alternatives under this point of view.
- The two aforementioned methodologies manage to outperform the alternative options scoring comparable average errors, with SI-based clustering allowing RF to score a minimum KPI% equal to 16.5% against the 16.2% manageable employing the Correlation-based approach. On one

hand SI-based, compared to the other best performing clustering approach (Correlation-based), does slightly worse. On the other, the superiority of the Correlation-based variant is verified only on the minimum point (which can be biased due to the random component). Differently, SI-based approach manages to keep a stable good performance in the surrounding of the global minima.

- The KPI% behavior along the graph produced with the SI-based clustering approach better resembles the expectations coming from the theoretical model describes before. It shows in fact a noisy but constant decrease in error until the global minimum, followed by a net inversion of the behavior. In order to test this assertion and verify the continuity of the increasing trend, it was decided to run, at least for this configuration, a second experiment. The iterative procedure was set to stop when *i* assumed a value equal to 140. From the outcomes of this second run (Fig. 36), it is possible to observe that the prolongation of such slow increase in error happens linearly until the end of the plot. Thus, strengthening the claim of SI-based approach on the coherence with the foreseen theoretical model.

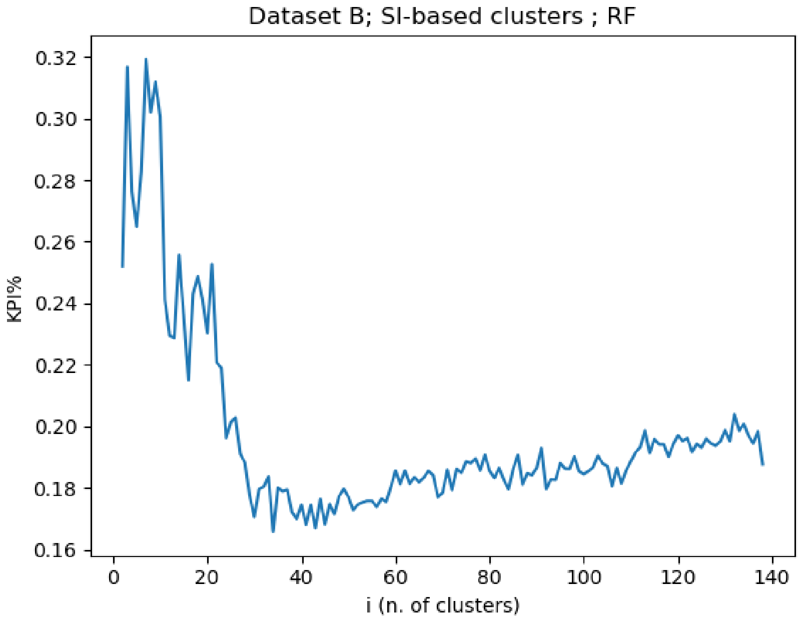

*Figure 36: Average performance of RF on Dataset B as a function of the number of clusters formed used the SI-based clustering methodology (second run)*

Nonetheless, the characteristics of Dataset B, which account for quite regular time series and similar products, make these observations a weak proof of this methodology superiority compared to the others. The plots show in fact several similar traits, which are a sign that, for such homogeneous datasets, how clusters are formed could have less influence than how big the clusters are. Thus, Dataset B could not be the best testing ground if the scope stands in understanding the validity of a clustering approach.

### 5.2.2 Dataset A

Extremely different is the situation for the more heterogeneous Dataset A. The results, reported in Fig. 37, Fig. 38, Fig. 39 and Fig. 40 show different behaviors. The approach based on the PCA analysis is the best when it comes to an evaluation based in the minimum error obtained, scoring a KPI% equal to 71.2%. It is nonetheless due to mere "luck", since along the rest of the plot consistently shows a lower performance compared to the alternatives. Distance-based clustering approach shows an extremely erratic behavior (Fig. 40), thus proving its inadequacy to the task discussed in Chapter 2.2. The correlation-based clustering approach, even if reaching good performances, shows irregular behavior with a sudden increase in the error for $i \approx 20$. It would be nonetheless interesting to proceed with the experiment for an higher number of clusters in order to establish how the improving trend continues and if a lower minimum can be found. Similarly to the case of Dataset B, the approach based on the extraction of the Seasonality Indexes (SI) performs apparently better. Fig. 34 shows in fact a more regular pattern and a resemblance to what was expected in the theoretical model.

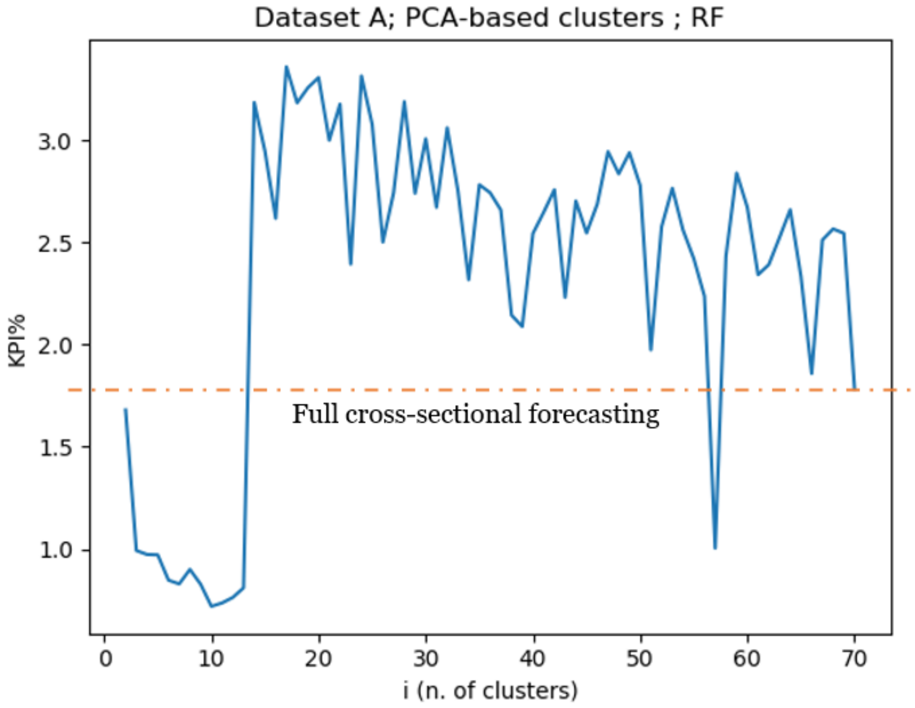

*Figure 37:Average performance of RF on Dataset A as a function of the number of clusters formed used the PCA-based clustering methodology*

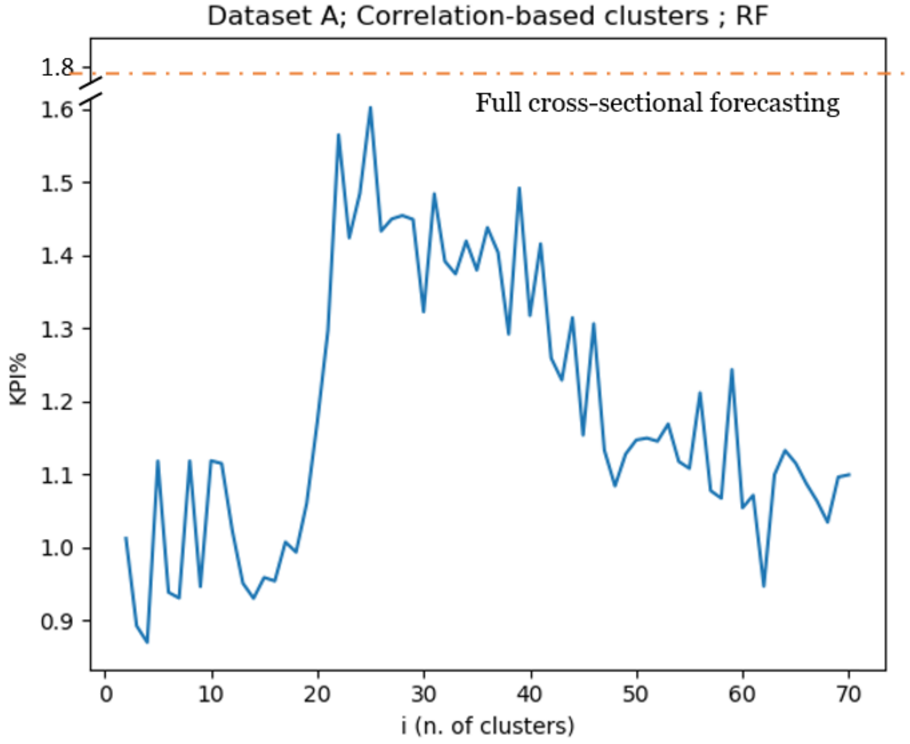

*Figure 38:Average performance of RF on Dataset A as a function of the number of clusters formed used the Correlation-based clustering methodology*

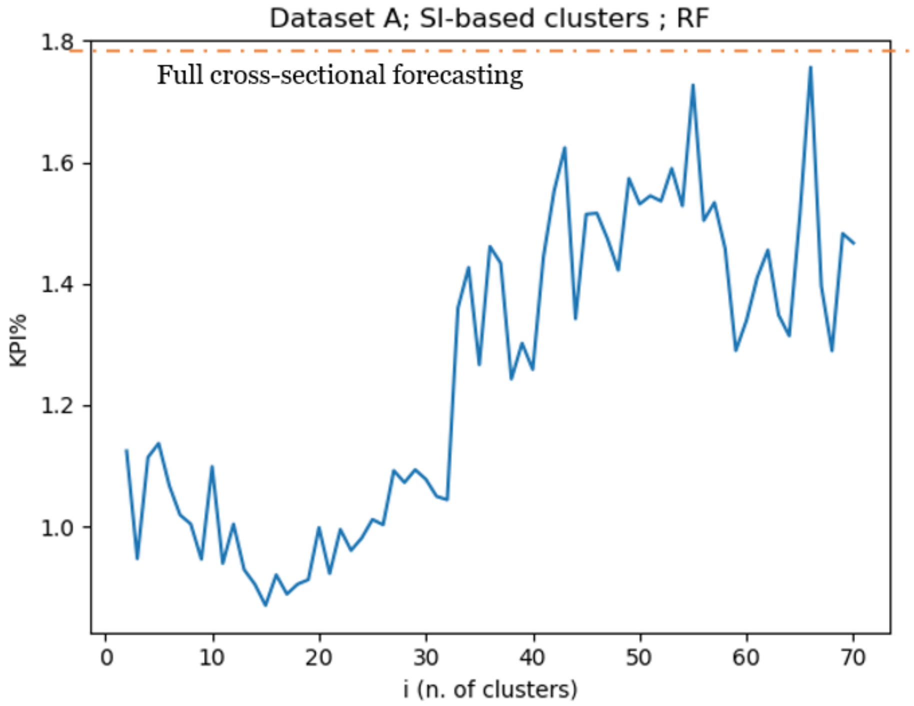

*Figure 39: Average performance of RF on Dataset A as a function of the number of clusters formed used the SI-based clustering methodology*

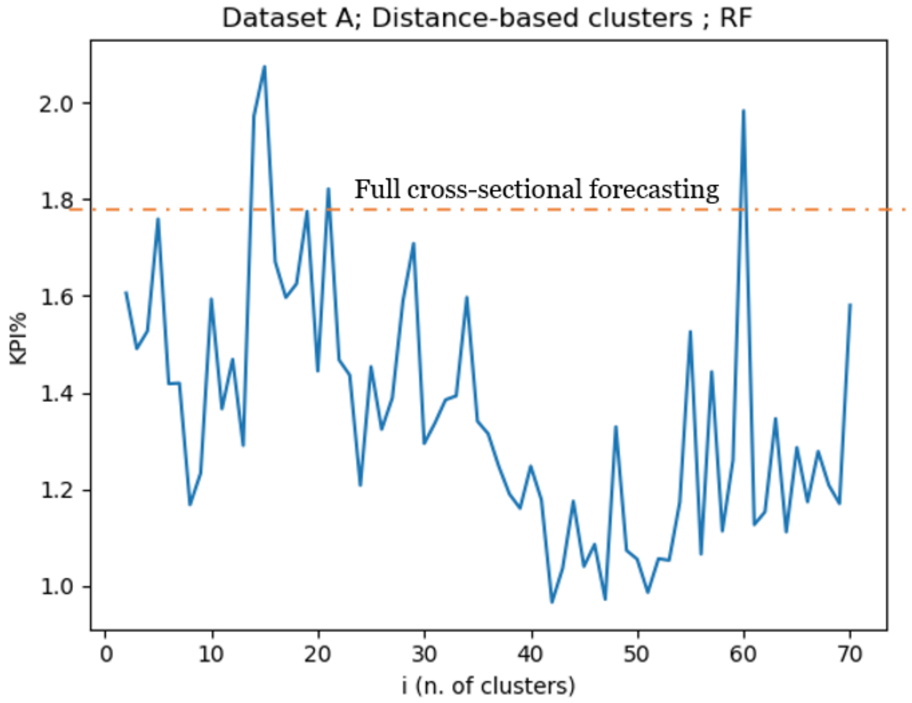

*Figure 40: Average performance of RF on Dataset A as a function of the number of clusters formed used the Distance-based clustering methodology*

### 5.2.3 Findings

Considering the observations which have been discussed in the previous two chapters, it is possible to conclude that the clustering approach can add considerable value to the cross-sectional forecasting procedure. The global minimum in the KPI% achieved with the clustering approach was indeed found to be consistently lower than the KPI% obtained through full cross-sectional forecasting. The decrease in KPI% obtained reached 26% for Dataset B and up to 52 % for Dataset A, where full cross-sectional forecasting was penalized by the high heterogeneity of the dataset.

The second (unexpected) finding consists in the possible existence of one or several local minima other than the global one. This presumably finds root in the complexity of the tradeoff between *specification error* and *overfitting error*.

Furthermore, a considerable outcome of this section is the better performance of SI-based and Correlation-based (on a smaller scale) clustering approaches compared to the alternatives examined. The interpretation of this outcome finds basis in the different focus of the approaches considered. The two best performing methodologies trespass from the clustering problem to the forecasting one. In the case of SI-based approach for example, the clusters are not formed uniquely to maximize the similarity of time series, but to maximize the similarity in a trait which heavily affects the forecasting accuracy (seasonal component).

# Chapter 6: Mixed approaches

*The difference between time series, causal (or explanatory) and mixed approaches to time series forecasting has been previously introduced (see Chapter 1.3). While until this point the study has been focused exclusively on time series approaches, this chapter will be focused on the experimentation of mixed approaches in order to improve ML methods accuracy. This analysis aims at evaluating the potential increase in performance which could be reachable through the consideration of the multitude of information coming available with the Big Data Revolution (McAfee and Brynjolfsson, 2012).*

*This chapter will be divided in three sections. The first one will present the approach employed, the difference between statistical and ML methods and the selection of the exogenous variables which were used to produce the forecasts. The second section will address the implementation and the modeling decision taken for the integration of the exogenous variables as inputs. Finally, the result will be presented and examined in the last section.*

## 6.1 Considering exogenous variables

The difference between time series approaches and mixed approaches stands in the nature of the data the forecast is based on. While time series approaches employ exclusively past historical demand data as input to the algorithms, mixed approaches integrate historical demand information with data from exogenous variables that have an effect on demand.

Both statistical and ML methods can potentially consider the influence of exogenous variables, but they can do it in fundamentally different ways.

Statistical methods are built in order to receive as input only data from time series. The influence of further variables, for example of marketing campaigns, is generally separately considered and successively integrated in the forecast (Syntetos et al., 2016). This approach works as far as only a few variables are considered but fails in an environment where Big Data provides information in a variety of different forms which require to be integrated in the forecasts. For statistical methods the integration of the variables in the forecast generally requires ad hoc modeling which is hardly scalable. The external modeling of exogenous variables is beyond the scope of the study. Thus, statistical methods will not be taken into account for the investigation on mixed approaches.

ML methods, on the other hand, are characterized by a malleable structure which gets automatically adapted to the inputs. This difference with statistical methods allows ML methods to receive data as input from various origins and the influence of that input can be automatically learned during the training phase. Since no additional model is needed, the integration of extra variables results straight forward, therefore making ML easy to adapt to the Big Data environment.

Considering this difference, even if this analysis selected only a limited number of variables which influence could have been modeled by statistical methods, it was decided to focus uniquely on mixed approaches to ML methods. The results will then be evaluated using, as term of comparison, those from the previous section of the study (see Chapter 4.3).

Regarding the selection of the exogenous variables, as described in Chapter 3.1, Datasets A and B had available, other than the historical sales, also information about the products and the stores. Additionally, For Dataset A, also knowledge about promotions was available. In order to apply the mixed approaches, it was previously necessary to analyze what factors influenced demand, how they influenced it and why. Of the information available it was decided that data on the product attributes were of little use. Different products attributed to the same category were proved to present extremely different sales behavior. Thus, the inclusion of the category could have been essentially harmful for the forecasting performance. The only piece of information provided of use for this approach was about the promotions related to the products in Dataset A.

Previa an analysis on the sales' behavior in the datasets, it was established that the presence of holidays was of fundamental importance for Dataset A. This phenomenon becomes clear when considering the nature of the company. The demand of a B2B company which provides food to canteens can only be extremely dependent on the holidays calendar. It was in fact observed that, approaching the holidays (near to Christmast for example), the demand experiences a decrease in value followed by a peak after them. This behavior can be explained considering the canteens' needs to finish the perishable goods and to refill the stocks when the schools or companies are back to work.

This was not equally true for the company which provided Dataset B. This second one, being a supermarket chain and therefore staying open for business also during the holidays, presented sales which showed non observable direct relationships to holidays. This, nonetheless, does not preclude the possibility of the existence of an underlying relationship.

Of the other variables that could have had an influence on the recorded demand, for example the weather, none were taken into account. This choice was made in order to help ML methods to rightly understand the relationships between the sales and the exogenous variables considered. Adding other variables, where the relationships to demand are not as clear as for the considered ones, could influence the learning phase and negatively affect the performance due to spurious correlations.

Given these conditions, the following analysis will be focused mainly on Dataset A. The analysis is nonetheless conducted also for dataset B, in order to verify if the performance would benefit from the use of exogenous variables also in the case of a possible (meaning not as evident as for Dataset A) relationship to demand.

The data regarding the holidays, as anticipated in Chapter 3.1, are publicly available and were downloaded from an official French data repository.

## 6.2 Experimental setting

Once the problem regarding what variables to consider was tackled, it was time to define how to manage them in order to have input-ready information.

As briefly described in the previous paragraph, ML methods are able to receive any kind of numerical input and potentially learn the relationships with the outputs. Practically speaking the input vector will not be formed by only the 52 numerical values related to the previous year's historical sales but also by other $s + p$ numerical values related to holidays ($s$) and promotions ($p$). $s$ and $p$ are the outcome of some arbitrary decision made according to the following considerations:

- **Holidays**: the influence of holidays on the historical demand recorded in Dataset A has been explained and justified in the previous paragraph. In order to allow ML methods to understand the underlying relationships, the inputs have to be selected and appropriately treated. The treatment of the holiday data has been already discussed in the chapter concerning the preprocessing phase (see Chapter 3.1). The outcome of such treatment was a numerical value, addressed as "holiday factor", between 0 and 7, where 0 meant that none of the days in the considered week was holiday and 7 that all of them were. The inclusion of these values in the input vector must be limited to those features that can have an influence on the demand to be forecasted. The holiday factors which have been considered relevant are attributable to three categories:
    1. Present holidays: comprehend those holiday factors which are related to the weeks included in the forecasting horizon.
    2. Future holidays: comprehend the holiday factors related to the weeks coming after the forecasting horizon. These factors were considered useful for the influence of the coming weeks on the decision of canteens to purchase a defined quantity of products. For example, if Christmas is coming in 1 week, the purchase will be influenced by the need to empty the stocks. It was arbitrarily

decided to take into account the influence of 4 weeks after the time the forecast is needed.

3. Past holidays: comprehend the holiday factors related to the weeks coming before the forecasting horizon. Similarly to future holidays, they were considered in order to take into account canteen's purchase behavior. After a holiday period they need in fact to totally replenish the stocks of the involved perishable products. Again, 1 month (4 weeks) was the past horizon selected for this category.

Recapping, to consider the influence of the holidays, a set of holiday factors, meaning numerical value between 0 and 7, was included in the input vector of the ML methods. This set was composed by 4 factors related to future holidays, 4 factors related to past holidays and *h* factors related to the holidays in the forecasting horizon. *h* assumes a different value according to the approach to multistep forecasting adopted: when the direct approach is applied, 8 future values of demand are forecasted simultaneously, therefore *h=8*. When instead the indirect approach is used, only 1 holiday factor is sufficient to cover the forecasting horizon.

- **Promotions**: only a few information were available regarding promotions, namely the day and the product of concern. Since the promotions' periods were found to be of a modular length of 1 week, a dummy variable (which assumes values equal to 0 or 1 alternatively) was sufficient to embody all data available on the matter. Similarly to the approach adopted for the holidays, it was not only considered the influence of promotions at the time of the forecasting horizon but also of those which happened near it. The dummy variables which were defined as "promotion factors" can be therefore grouped in two categories:
  - Present Promotions: comprehend those promotion factors which are related to the weeks included in the forecasting horizon.
  - Past Promotions: comprehend the promotion factors related to the weeks coming before the forecasting horizon. These factors were

included in the input vectors due to the belief that promotions affect demand not only during the time they are active but also in the future. To make a simple example applicable to the case study, it is possible that canteens would buy in excess to take advantage of the current promotions and successively reduce the orders to reduce the excess stock. In this case as well, a horizon of 4 weeks in the past was considered.

Future promotions were established not to influence the demand and therefore were not included in the input vectors.

Table 14 sums up the content of the input vector, while Fig. 41 provides a graphical representation of how the vector was assembled. For the correct interpretation of both, it is important to remind that *h* assumes different values whether a direct (*h*=*8*) or indirect (*h*=*1*) approach to demand forecasting is considered.

| Content | Form | Past | Present | Future |
|---|---|---|---|---|
| **Demand** | Amount of sales per week | 52 weeks | / | / |
| **Holidays** | Holiday Factor (from 0 to 7) | 4 weeks | *h* weeks | 4 weeks |
| **Promotions** | Promotion Factor (0 or 1) | 4 weeks | *h* weeks | / |

*Table 14: Content of the ML methods' input vector*

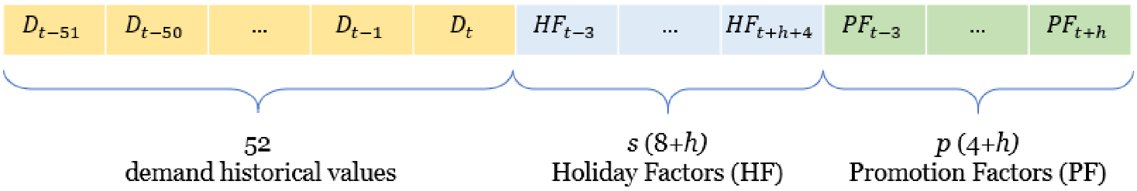

*Figure 41: Structure of the ML methods' input vector*

## 6.3 Results analysis

The vector introduced in the previous paragraph was used to produce the forecast in the same way adopted for the time series approaches (see Chapters 3 and 4). Since the mixed approaches to statistical methods were beyond the scope of the thesis, the forecasts were produced employing exclusively the ML methods.

### 6.3.1 Dataset A

Both indirect and direct multistep forecasting were investigated for every method. For each configuration, 5907 forecasts related to Dataset A were produced. The accuracy indexes (KPI%) obtained are presented and compared to those achieved for the time series approaches in Table 15.

| Rank | Method | KPI%<br>(mixed approaches) | KPI%<br>(time series approaches) | Change in KPI% |
|---|---|---|---|---|
| 1 | **XGBoost_1SA** | 34.0 | 97.8 | -55.8 |
| 2 | **RT_1SA** | 42.0 | 121.8 | -79.8 |
| 3 | **ETR_1SA** | 43.2 | 181.0 | -137.8 |
| 4 | **ETR** | 49.4 | 48.0 | +1.4 |
| 5 | **RF_1SA** | 49.8 | 178.6 | -128.8 |
| 6 | **RF** | 61.3 | 73.5 | -12.2 |
| 7 | **RT** | 86.8 | 135.1 | -48.3 |

*Table 15: Dataset A; Mixed approach to ML methods performances and comparison with time series approach*

The results presented for Dataset A show promising results. The additional variables included as input seem in fact to bring to a significant increase of performance for almost all the methods tested. The only exception is ETR which apparently didn't manage to further improve the forecasts' quality. Nonetheless, given the randomness of the forecasts, such slight variation of performance can be considered not significant.

A noteworthy observation to discuss is the behavior of the methods when employed with the two different multistep forecasting approaches. In the previous investigation, meaning the application of time series approaches to Dataset A (see Chapter 4.3), the performance of the ML methods when forecasts were produced with the direct multistep approaches resulted in most of the cases higher than their indirect counterparts. This behavior is no longer verified when considering the mixed approaches. The situation is in fact reversed, since in all of the cases depicted in Table 13, indirect approaches manage to outperform the direct approaches. As commented in Chapter 4, the indirect approach has proved to perform scarcely in extremely complex environment probably due to the higher error which is reinserted in the method at each iteration. In this case apparently, thanks to the addition of holiday and promotion factors as input, ML manage to better learn the pattern and minimize the error which is reinserted at each iteration.

### 6.3.2 Dataset B

Despite the lack of relevant information discussed in Chapter 6.1, it was decided to expand the investigation of mixed approached also to Dataset B. As anticipated, this was done in order to examine the effects of the inclusion of variables with weaker influence on demand in the input vector employed to predict its future value. The structure and content of the input vector is the same as for Dataset A when promotions are excluded. Differently from what observed for Dataset A, from the values reported in Table 16, no increase or decrease in accuracy is clearly visible along the whole set of methods. Worth mentioning again, the differences in behavior shown by the alternative multistep forecasting approaches. While in fact for the indirect approaches a weak increase in error is on average verified, the direct approaches apparently perform significantly better than when they are given as input exclusively historical demand data.

| Rank | Method | KPI% (mixed approaches) | KPI% (time series approaches) | Change in KPI% |
|---|---|---|---|---|
| 1 | **RF_1SA** | 16.9 | 21.9 | -5.0 |
| 2 | **XGBoost_1SA** | 19.0 | 18.9 | +0.1 |
| 3 | **RT** | 19.0 | 50.6 | -31.6 |
| 4 | **RT_1SA** | 22.4 | 19.2 | +3.2 |
| 5 | **ETR_1SA** | 23.4 | 19.7 | +3.7 |
| 6 | **ETR** | 25.0 | 40.8 | -15.8 |
| 7 | **RF** | 25.1 | 38.8 | -13.7 |

*Table 16: Dataset B; Mixed approach to ML methods performances and comparison with time series approach*

# Chapter 7: Conclusions

*This chapter will present the conclusions drawn from each of the analysis discussed in the previous chapters. It will be divided in four sections: the first three paragraphs will discuss the outcomes of Chapter 4, 5 and 6 respectively. Here the findings will be schematically summed up and the limitations underlined. The fourth and last section of the Chapter will concern some of the possible research topics which can be identified from the outcomes of this thesis.*

## 7.1 Full cross-sectional forecasting

The investigation treated in Chapter 4 brings to several conclusions connected to the previously shown results:

1. The indirect multistep forecasting approach has proved to allow ML methods to reach, in case of a regular forecasting environment, a better performance compared to the direct variant. This was verified for each of the methods tested on Dataset B. When more chaotic forecasting environments are involved (Dataset A), indirect approaches suffer from the error injection mechanism happening due to the iterative procedure, thus being outperformed by the direct ones.
2. ML methods prove to perform better than statistical methods when applied to the conditions which defined the boundaries for the study, namely, time series characterized by (i) weekly time buckets, (ii) yearly seasonality and (iii) minimum preprocessing performed. This conclusion is based uniquely on the results achieved on Dataset B. Dataset A is in fact not appropriate to reach such conclusion. Its high level of irregularity affects in fact all kind of methods but the simplest ones (which in fact outperform the competitors). Supporting this statement, stands the decision by the consulting company providing the data (specialized in demand forecasting) to rely on a higher level of aggregation to produce the forecasts for Dataset A.

3. The performance of ML methods when applied in a cross-sectional forecasting form, do not depend only on the products' historical demand predictability but also on its homogeneity among the products in the dataset. This outcome poses the basis for the employment of the clustering procedures treated in Chapter 5.

Regarding this section of the study, it is important to underline a couple of limitations which affected the extent of the outcomes. The restricted processing power available undermined the possibility to run the algorithms multiple times in order to strengthen the results. Due to the randomness of the ML algorithms outputs, it was impossible to make any solid assessment on the comparison between the ML methods employed. A second limitation is the lack of a competitive benchmark method. This does not allow to prove how the ML methods really perform. The knowledge about the good performance of the methods for Dataset B comes from the comparison with the forecasts produced by the consulting company with its own forecasting package. The accuracy of the ML methods was said to be aligned with the results the package produced.

## 7.2 Clustering approach to cross-sectional forecasting

Chapter 5 treated the employment of a clustering approach to aid cross-sectional forecasting. Several methods to form the clustering space were investigated and a critical analysis of the results was given in Chapter 5.2. Before coming to the conclusions that can be drawn from this analysis it is necessary to point out the challenges met. The plots extracted from the results resulted in fact in many cases erratic. The unavailability of a higher computing power precluded the possibility to run the experiments multiple times in order to reduce the random component characterizing the results of ML methods. The resulting plots were therefore hard to read, and the observations made on them leave space for uncertainty. Furthermore, always due to the lack of computing power, the experiments were cut once the number of clusters reached the value of 70, thus limiting the extent of the curve analyzable. That being said, the following conclusions can be drawn from Chapter 5:

1. A clustering approach can sensibly improve the accuracy of the forecast. Indeed, every clustering approach proposed managed to score a minimum KPI% lower than the one achieved with full cross-sectional forecasting.
2. The KPI% plotted as a function of the number of clusters can present several local minima other than the foreseen global minimum.
3. The clustering approaches that perform best are based, the first, on the Seasonality Indexes extracted from the time series and, the second, on the correlation coefficients between the features and the performance achieved in Chapter 4. Meaning, by those clustering approaches which focus not on grouping the most similar time series, but on grouping time series based on a logic which could help ML methods to better learn time series' patterns. Thus, the contribution of this study in filling the literature gap described in Chapter 2.1, stands in the definition of a more performing clustering logic to base the procedure on. Between those examined in this thesis, the best performing one results to be based on the Seasonality Indexes, that are features extracted from the time series, directly attributable to the patterns ML should learn.

## 7.3 Mixed approaches to cross-sectional forecasting

The outcomes of this section (Chapter 6) were limited by the scarce availability of suitable data for the task and by the same issues concerning the time series approaches (see Chapter 7.1). It was nonetheless possible to extract some valid outcomes from the analysis:

1. Mixed approaches can provide key information that allows ML method to understand the underlying patterns. This statement is rooted on the significant decrease in error that characterized all of the methods employed through the mixed approaches application to Dataset A. The

ML methods aided with the additional variables as input, manage in fact to understand the relationship between demand and holidays/promotions reducing the error of up to three times the value obtained with time series approaches.

2. The analysis of externals variables' influence assumes a central importance in the case of mixed approaches. The results related to Dataset B showed in fact that the inclusion of variables that have a dubious relationship with demand could also have detrimental effects on forecasting accuracy.

## 7.4 Future topics of research

Mixed approaches applied through the simplicity allowed by ML methods represent an interesting course of evolution forecasting should undertake. The reason which has prevented for it to be widely diffused at present time is that forecasting software providers are struggling to access and combine relevant data from external sources (Syntetos et al., 2016). An interesting topic of research would be the development of an automatic way to evaluate the relevance of the external variables to consider in order to improve demand forecasting.

The second topic suggested for further investigation concerns the conclusions reached by this study regarding clustering approaches. This thesis managed to assess the best logic to address the clustering approaches with, namely the creation of the clusters focusing on the forecasting procedure and not just on the similarity between the time series. It didn't however pretend to devise the best way to do it. Better approaches following the same logic could and should be developed. Given the promising results of the SI-based clustering approach, further attention should be especially placed on it.

# Bibliography

1. Abiodun, O.I., Jantan, A., Omolara, A.E., Dada, K.V., Mohamed, N.A., Arshad, H., 2018. State-of-the-art in artificial neural network applications: A survey. Heliyon 4, e00938. https://doi.org/10.1016/j.heliyon.2018.e00938
2. Aburto, L., Weber, R., 2007. Improved supply chain management based on hybrid demand forecasts. Appl. Soft Comput. 7, 136–144. https://doi.org/10.1016/j.asoc.2005.06.001
3. Aghabozorgi, S., Seyed Shirkhorshidi, A., Ying Wah, T., 2015. Time-series clustering – A decade review. Inf. Syst. 53, 16–38. https://doi.org/10.1016/j.is.2015.04.007
4. Ali, Ö.G., Sayın, S., van Woensel, T., Fransoo, J., 2009. SKU demand forecasting in the presence of promotions. Expert Syst. Appl. 36, 12340–12348. https://doi.org/10.1016/j.eswa.2009.04.052
5. Armstrong, J.S., 2001. Principles of forecasting: a handbook for researchers and practitioners. Springer Science & Business Media.
6. Bandara, K., Bergmeir, C., Smyl, S., 2020. Forecasting across time series databases using recurrent neural networks on groups of similar series: A clustering approach. Expert Syst. Appl. 140, 112896. https://doi.org/10.1016/j.eswa.2019.112896
7. Bergstra, J., Bengio, Y., 2012. Random Search for Hyper-Parameter Optimization 25.
8. Box, G.E., Jenkins, G.M., Reinsel, G.C., Ljung, G.M., 1970. Time series analysis: forecasting and control. John Wiley & Sons.
9. Breiman, L., 1996. Bagging predictors. Mach. Learn. 24, 123–140. https://doi.org/10.1007/BF00058655
10. Brown, R.G., 1957. Exponential smoothing for predicting demand, in: Operations Research. INST OPERATIONS RESEARCH MANAGEMENT SCIENCES 901 ELKRIDGE LANDING RD, STE ..., pp. 145–145.
11. Bzdok, D., Altman, N., Krzywinski, M., 2018. Statistics versus machine learning. Nat. Methods 15, 233–234. https://doi.org/10.1038/nmeth.4642
12. Carbonneau, R., Laframboise, K., Vahidov, R., 2008. Application of machine learning techniques for supply chain demand forecasting. Eur. J. Oper. Res. 184, 1140–1154.
13. Cerqueira, V., Torgo, L., Soares, C., 2019. Machine Learning vs Statistical Methods for Time Series Forecasting: Size Matters. ArXiv190913316 Cs Stat.
14. Dalhart, G., 1974. Class seasonality-a new approach. Publ. Forecast. 2nd Ed. Am. Prod. Inventory Control Soc. Wash. DC 11–16.
15. Davenport, T.H., Patil, D.J., 2012. Data scientist. Harv. Bus. Rev. 90, 70–76.
16. Dietterich, T.G., 2000. Ensemble Methods in Machine Learning, in: Multiple Classifier Systems, Lecture Notes in Computer Science.

Springer, Berlin, Heidelberg, pp. 1–15. https://doi.org/10.1007/3-540-45014-9_1

17. Elith, J., Leathwick, J.R., Hastie, T., 2008. A working guide to boosted regression trees. J. Anim. Ecol. 77, 802–813. https://doi.org/10.1111/j.1365-2656.2008.01390.x
18. Friedman, J.H., 2002. Stochastic gradient boosting. Comput. Stat. Data Anal., Nonlinear Methods and Data Mining 38, 367–378. https://doi.org/10.1016/S0167-9473(01)00065-2
19. Geurts, P., Ernst, D., Wehenkel, L., 2006. Extremely randomized trees. Mach. Learn. 63, 3–42. https://doi.org/10.1007/s10994-006-6226-1
20. Hamzaçebi, C., Akay, D., Kutay, F., 2009. Comparison of direct and iterative artificial neural network forecast approaches in multi-periodic time series forecasting. Expert Syst. Appl. 36, 3839–3844. https://doi.org/10.1016/j.eswa.2008.02.042
21. Hartmann, C., Hahmann, M., Lehner, W., Rosenthal, F., 2015. Exploiting big data in time series forecasting: A cross-sectional approach, in: 2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA). Presented at the 2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA), pp. 1–10. https://doi.org/10.1109/DSAA.2015.7344786
22. Hastie, T., Tibshirani, R., Friedman, J., 2009. The elements of statistical learning: data mining, inference, and prediction. Springer Science & Business Media.
23. Hofmann, E., Rutschmann, E., 2018. Big data analytics and demand forecasting in supply chains: a conceptual analysis. Int. J. Logist. Manag. 29, 739–766. https://doi.org/10.1108/IJLM-04-2017-0088
24. Holt, C.C., 1957. Forecasting seasonals and trends by exponentially weighted moving averages. Int. J. Forecast. 20, 5–10. https://doi.org/10.1016/j.ijforecast.2003.09.015
25. Hornik, K., 1991. Approximation capabilities of multilayer feedforward networks. Neural Netw. 4, 251–257. https://doi.org/10.1016/0893-6080(91)90009-T
26. Hyndman, R.J., 2016. Q&A time | Rob J Hyndman [WWW Document]. URL https://robjhyndman.com/hyndsight/qa-time/ (accessed 2.4.20).
27. Hyndman, R.J., 2011. Moving Averages. Citeseer.
28. Hyndman, R.J., Athanasopoulos, G., 2018. Forecasting: principles and practice. OTexts.
29. Hyndman, R.J., Khandakar, Y., 2007. Automatic time series for forecasting: the forecast package for R. Monash University, Department of Econometrics and Business Statistics ….
30. Jolliffe, I.T. (Ed.), 2002. Principal Component Analysis for Time Series and Other Non-Independent Data, in: Principal Component Analysis, Springer Series in Statistics. Springer, New York, NY, pp. 299–337. https://doi.org/10.1007/0-387-22440-8_12
31. Karypis, M.S.G., Kumar, V., Steinbach, M., 2000. A comparison of document clustering techniques, in: TextMining Workshop at KDD2000 (May 2000).

32. Keogh, E., Lin, J., 2005. Clustering of time-series subsequences is meaningless: implications for previous and future research. Knowl. Inf. Syst. 8, 154–177. https://doi.org/10.1007/s10115-004-0172-7
33. Kong, W., Dong, Z.Y., Jia, Y., Hill, D.J., Xu, Y., Zhang, Y., 2019. Short-Term Residential Load Forecasting Based on LSTM Recurrent Neural Network. IEEE Trans. Smart Grid 10, 841–851. https://doi.org/10.1109/TSG.2017.2753802
34. Liaw, A., Wiener, M., 2002. Classification and Regression by randomForest 2, 6.
35. Liu, H., Cocea, M., 2017. Semi-random partitioning of data into training and test sets in granular computing context. Granul. Comput. 2, 357–386.
36. Makridakis, S., Spiliotis, E., Assimakopoulos, V., 2018. Statistical and Machine Learning forecasting methods: Concerns and ways forward. PLOS ONE 13, e0194889. https://doi.org/10.1371/journal.pone.0194889
37. Mayer-Schönberger, V., Cukier, K., 2013. Big Data: A Revolution that Will Transform how We Live, Work, and Think. Houghton Mifflin Harcourt.
38. McAfee, A., Brynjolfsson, E., 2012. Big Data: The Management Revolution 9.
39. Mitchell, T.M., 1997. Does Machine Learning Really Work? AI Mag. 18, 11–11. https://doi.org/10.1609/aimag.v18i3.1303
40. Petropoulos, F., Makridakis, S., Assimakopoulos, V., Nikolopoulos, K., 2014. 'Horses for Courses' in demand forecasting. Eur. J. Oper. Res. 237, 152–163. https://doi.org/10.1016/j.ejor.2014.02.036
41. Räsänen, T., Kolehmainen, M., 2009. Feature-Based Clustering for Electricity Use Time Series Data, in: Kolehmainen, M., Toivanen, P., Beliczynski, B. (Eds.), Adaptive and Natural Computing Algorithms, Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, pp. 401–412. https://doi.org/10.1007/978-3-642-04921-7_41
42. Rocca, J., 2019. Ensemble methods: bagging, boosting and stacking [WWW Document]. Medium. URL https://towardsdatascience.com/ensemble-methods-bagging-boosting-and-stacking-c9214a10a205 (accessed 3.12.20).
43. Samuel, A.L., 1959. Some studies in machine learning using the game of checkers 13.
44. Sandulescu, V., Chiru, M., 2016. Predicting the future relevance of research institutions - The winning solution of the KDD Cup 2016. ArXiv160902728 Phys.
45. Seif, G., 2019. Selecting the best Machine Learning algorithm for your regression problem [WWW Document]. Medium. URL https://towardsdatascience.com/selecting-the-best-machine-learning-algorithm-for-your-regression-problem-20c330bad4ef (accessed 3.15.20).
46. Shmueli, G., Bruce, P.C., Yahav, I., Patel, N.R., Jr, K.C.L., 2017. Data Mining for Business Analytics: Concepts, Techniques, and Applications in R. John Wiley & Sons.

47. Smola, A.J., Schölkopf, B., 2004. A tutorial on support vector regression. Stat. Comput. 14, 199–222. https://doi.org/10.1023/B:STCO.0000035301.49549.88
48. Smyl, S., Kuber, K., 2016. Data Preprocessing and Augmentation for Multiple Short Time Series Forecasting with Recurrent Neural Networks 14.
49. Souza, G.C., 2014. Supply chain analytics. Bus. Horiz. 57, 595–605. https://doi.org/10.1016/j.bushor.2014.06.004
50. Stewart, M.S., PhD, 2019. The Actual Difference Between Statistics and Machine Learning [WWW Document]. Medium. URL https://towardsdatascience.com/the-actual-difference-between-statistics-and-machine-learning-64b49f07ea3 (accessed 1.21.20).
51. Syntetos, A.A., Babai, Z., Boylan, J.E., Kolassa, S., Nikolopoulos, K., 2016. Supply chain forecasting: Theory, practice, their gap and the future. Eur. J. Oper. Res. 252, 1–26. https://doi.org/10.1016/j.ejor.2015.11.010
52. Tang, Z., de Almeida, C., Fishwick, P.A., 1991. Time series forecasting using neural networks vs. Box- Jenkins methodology. SIMULATION 57, 303–310. https://doi.org/10.1177/003754979105700508
53. Tay, F.E.H., Cao, L., 2001. Application of support vector machines in financial time series forecasting. Omega 29, 309–317. https://doi.org/10.1016/S0305-0483(01)00026-3
54. Thiesing, F.M., Vornberger, O., 1997. Sales forecasting using neural networks, in: Proceedings of International Conference on Neural Networks (ICNN'97). Presented at the Proceedings of International Conference on Neural Networks (ICNN'97), pp. 2125–2128 vol.4. https://doi.org/10.1109/ICNN.1997.614234
55. Vandeput, N., 2018. Data Science for Supply Chain Forecast. Independently published.
56. Volkovs, M., Yu, G.W., Poutanen, T., 2017. Content-based Neighbor Models for Cold Start in Recommender Systems, in: Proceedings of the Recommender Systems Challenge 2017 on ZZZ - RecSys Challenge '17. Presented at the the Recommender Systems Challenge 2017, ACM Press, Como, Italy, pp. 1–6. https://doi.org/10.1145/3124791.3124792
57. Wang, G., Gunasekaran, A., Ngai, E.W.T., Papadopoulos, T., 2016. Big data analytics in logistics and supply chain management: Certain investigations for research and applications. Int. J. Prod. Econ. 176, 98–110. https://doi.org/10.1016/j.ijpe.2016.03.014
58. Wang, X., Smith, K., Hyndman, R., 2006. Characteristic-Based Clustering for Time Series Data. Data Min. Knowl. Discov. 13, 335–364. https://doi.org/10.1007/s10618-005-0039-x
59. Warren Liao, T., 2005. Clustering of time series data—a survey. Pattern Recognit. 38, 1857–1874. https://doi.org/10.1016/j.patcog.2005.01.025
60. Whittle, P., 1951. Hypothesis testing in time series analysis. Almqvist & Wiksells boktr.
61. Winters, P.R., 1960. Forecasting sales by exponentially weighted moving averages. Manag. Sci. 6, 324–342.

62. Withycombe, R., 1989. Forecasting with combined seasonal indices. Int. J. Forecast. 5, 547–552. https://doi.org/10.1016/0169-2070(89)90010-1
63. Wold, S., Esbensen, K., Geladi, P., 1987. Principal component analysis. Chemom. Intell. Lab. Syst. 2, 37–52.
64. Yallop, M., 2019. Machine learning: the big risks and how to manage them [WWW Document]. Financ. Times. URL https://www.ft.com/content/90ac19fe-2008-11ea-92da-f0c92e957a96 (accessed 1.30.20).
65. Yan, W., 2012. Toward Automatic Time-Series Forecasting Using Neural Networks. IEEE Trans. Neural Netw. Learn. Syst. 23, 1028–1039. https://doi.org/10.1109/TNNLS.2012.2198074
66. Zhao, Z., Chen, W., Wu, X., Chen, P.C., Liu, J., 2017. LSTM network: a deep learning approach for short-term traffic forecast. IET Intell. Transp. Syst. 11, 68–75.

# Acknowledgments

First of all, I would like to thank Prof. Yves Dallery, Prof. Evren Sahin and Prof. Andrea Matta. To them I owe the realization of the joint research project between Politecnico di Milano and CentraleSupélec which allowed the development of this thesis. I would like to thank them, along with PhD. Mohammed Hichame Benbitour, for the guidance continuously provided during the project.

A special thanks goes also to all the other members of LGI in CentraleSupélec, who warmly welcomed me and allowed me to grow personally and professionally.

This experience would have not been possible outside the Alliance4Tech program. Thus, I would like to thank everybody responsible for its ideation and coordination. I am particularly grateful to Prof. Mauro Filippini who actively helped me and my colleagues for the last two years.

Last but not least I would like to thank my family and friends who provided me with the support needed all the way to the realization of this thesis.