Østfold University College

MASTER'S THESIS

Data Science for Decision Support:

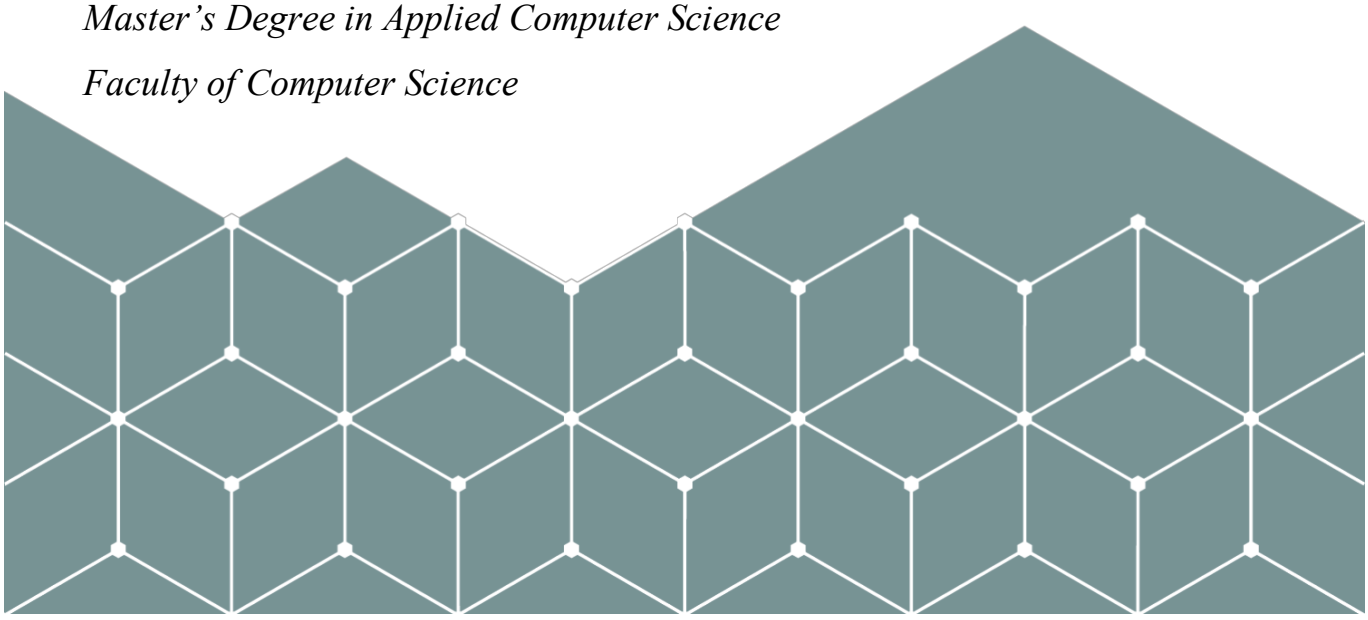
Using Machine Learning and Big data in Sales Forecasting for Production and Retail

Alireza Khakpour

Spring 2020

Master's Degree in Applied Computer Science

Faculty of Computer Science



Abstract

Sales forecasting plays a significant role in developing business analytic solutions. It is crucial for companies to have an accurate sales forecast to support their sales and operation procedures (S&OP). On the other hand, sales and demand forecasting is even more essential for the production of Fast Moving Consumer packaging Goods (FMCG) and retail industry due to the short shelf life of the products as well as their exposure to the various sale's uncertainties. This requires decision-makers to have a fast, accurate, and efficient sales forecasting solution to be integrated into their business processes. In this study, our contribution is twofold. The first one is methodological, where we examine some of the Machine Learning approaches for sales forecasting, in which conventional methods used for this task are extended to fit into this application area, and using our experiments, we demonstrate that they yield satisfactory predictive results. The second contribution is an applied one, where we use our proposal in a real-world problem for demand forecasting in the FMCG and retail industry, developing a machine learning pipeline for sales prediction that helps demand management and other operative and strategic decisions. The focal company in which the case study has been carried out is Brynild Gruppen AS, which is a manufacturer of chocolate and confectionary products located in Norway. The results of the study, are presented as a machine learning pipeline, integrating various machine learning techniques and methods, and showing promising accuracies for sales and demand forecasting.

Keywords: Machine Learning, Data Science, ML Pipeline, Sales Forecasting, FMCG, Retail

Acknowledgments

I would like to thank my dear supervisor, Dr. Roland Olsson for his precious and exceptional guidance and feedbacks as well as his positive attitudes towards my not always analytical reasonings. His level of knowledge and experience has always inspired me to seek to learn more and not being proud.

I would also like to thank Dr. Ricardo Colomo-Palacios for his valuable supports and enlightenments towards completing this master thesis. He has always been kind in helping me to make progress and get over problems.

This master thesis would have not been possible without the kind collaboration of Mathias Holm and Haris Jasarevic at Brynild Gruppen AS, who provided me the opportunity to work with a real-world business problem at first, and then assisting me to complete this project with their generous supports and contributions. The great experience of working in Brynild Gruppen as a summer intern provided me the ground for planning and constructing this Master thesis.

I would also like to express my gratitude to all professors and faculty members of Computer Science at Østfold University College that have thought me a lot of priceless and valuable knowledge in the field of Computer Science. Monica Kristiansen Holone, Cathrine Linnes, Susanne Koch Stigberg, and Harald Holone, you have always been kind to me and I learned a lot from you during your exceptional courses, and thank you for opportunities that you have given to me.

I also would like to thank my dear Iranian friends who have always been by my side during last two years and not only because of their presence during difficult moments, but also because they did everything they could in proving their friendships.

Last but not least, I would like to thank my parents, my brother, and my sister who have always been my inspiration in life and have supported me in every part of my life.

Content

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

With the advancements of data engineering and analytics, business analytics became an integral part of every business support system [1]. In this regard, sales and demand forecasting plays a significant role in developing business analytics solutions and it is crucial for companies to have an accurate sales forecast to use in their sales and operation procedures (S&OP). In fact, having an accurate estimate of the prospective sales of a particular product can help both manufacturers and retailers to make better decisions in their marketing, sales, production, and procurement planning [2].

On the other hand, sales and demand forecasting is even more essential for the Fast Moving Consumer packaging Goods (FMCG) and retail industry [3]. Indeed, many of the consumer packaging goods have a short shelf life as well as being prone to various sale's uncertainties and requires decision-makers to have a fast, accurate, and efficient sales forecasting solution to be integrated into their current processes [4].

In general, sales forecasting is a significant consideration for manufacturers, wholesalers and retailers, and it is a central endeavor for many organizations involved in supply chain activities [5]. The benefits of sales forecasting in various activities of supply chain differ [6]. Manufacturing companies can benefit from sales forecasting in all of their planning and decision supports, taking from inventory management and production planning, to sales and marketing activities. However, despite the importance of sales forecasting, lack of an accurate and efficient demand forecasting solution leads to unreliable forecasts that can have less or no effect on an organization's sales and operation processes[7]. The amount of financial benefits for an organization as the results of an accurate sales forecasting is difficult to estimate, although the sources of these benefits can be estimated to be primarily in marketing and sales such as product alterations, promotional efforts, and pricing [8]. While, the effects of a proper and accurate demand forecast over production and inventory planning is inevitable as well [9].

Currently, various methods are being used for sales and demand forecasting. However, present techniques are normally based on conventional statistical methods and are either unsatisfactory or inefficient because of having low accuracy and not making use of all available data sources, respectively [10]. These methods are commonly based on judgmental decisions of domain experts and can be inaccurate due to complexity of the effects various variables impose on the amount of sales [11]. Whereas, the demand pattern of the customers can be altered with respect to the holidays, weather, seasonal patterns, economic situations, and other variables [12].

On the other hand, the current methods are normally based on linear models, such as autoregressive moving average (ARMA) and autoregressive integrated moving average (ARIMA) that have the constrains of assuming the linearity of the problem[13]. However, FMCG and retail sales forecasting typically have nonlinearities in their time series problem and hence, require more advanced nonlinear models to tackle this issue [14].

Artificial Intelligence (AI) techniques, specifically Machine Learning (ML) methods, are providing the ability to develop more complex nonlinear models that are more generalizable to FMCG and retail Industry cases [13]. On the other hand, the ability of ML techniques to take into account various uncertainties, such as weather variables, economic situations, seasonal patterns, and demographic conditions, overcomes the limitation of traditional time series techniques. Indeed, machine learning based predictive analysis can have a great impact on sales and demand forecasting tasks to have more accurate and timely predictions.

The Structure of this thesis is as follows: We first introduce our problem statement and research questions that we seek to answer. Then, in the rest of this section, a brief background study is provided to familiarize the reader with the concepts that have been used throughout this study. Section 2 presents a systematic literature review around this topic to understand prior efforts regarding the problem stated. We then provide the methods and techniques that have been used to tackle the problem in section 3. Section 4 presents the results and evaluations of the study, which is then followed by a discussion around findings in section 5, and a conclusion and future works in section 6.

1.2 Problem Statement

In general, maintaining the best possible accuracy and efficiency while deploying machine learning methods is a challenging task that requires an extensive effort by the researchers

and practitioners. Adding to that, is the domain specific characteristics that can alter the outcomes of various machine learning techniques deployed for sales forecasting. Therefore, it is required to investigate the development of advanced machine learning based methods for sales forecasting of various products, taking into account their specific sales location and product characteristics [15].

Consequently, our contribution in this study is twofold. The first one is methodological: Where we examine novel Machine Learning approaches for time series forecasting, in which conventional methods used for this task are extended to fit into this application area and using our experiments, we demonstrate that they yield satisfactory predictive results. The second contribution is an applied one: Where we use our proposal in a real-world problem for demand forecasting in the FMCG Retail Industry, developing a machine learning pipeline, integrating various machine learning techniques and methods, and showing promising accuracies for sales and demand forecasting.

Therefore, in this study, we first investigate the deployment of various machine learning algorithms and techniques for the purpose of sales forecasting. Then we examine the possible best solution on a real-word problem for sales forecasting in the FMCG retail industry. The focal company for our study is a manufacturer of chocolate and confectionary products named as Brynild Gruppen AS. The company is located in Norway and provides around 4% of the country's confectionary consumptions. In our work, we examine the application of various machine learning techniques in order to develop a sales forecasting solution for the company. In what follows, we present our research question that we try to answer as the result of this study.

1.3 Research Questions

1. How should FMCG and retail data be translated into sales and demand forecasting indicators? How should be the processing of this data? *(What to study?)*
2. What are the suitable Machine Learning algorithms for sales and demand forecasting using FMCG and retail data? (How to study?)

1.4 Background Study

In this section, we briefly introduce and explain the concepts and backgrounds related to this study. This explanation not only helps all the readers to better interpret the contents of

this study but also assist the non-expert audiences to understand the workflow of the project in an easier manner. We first introduce the FMCG and retail industry along with their requirements in supply chain management, then, the importance of sales and demand forecasting is outlined along with definitions of some important terms. After which, a brief overview of the focal company, Brynild Gruppe AS, is given in order to introduce the aspects of the data and the problem. Finally, some of the machine learning concepts and methods are briefly explained.

1.4.1 FMCG and Retail

Fast Moving Consumer Goods are those types of products that are sold in a high volume, low price, and a rapid manner [3]. Most of the FMCG products are having short-shelf life due to the high consumer demands or the early expiry date of the product itself. Food products, such as beverages, candies, confectioneries, and pre-processed foods are some of the main categories of the FMCG [4]. The main delivery channels of FMCG products are retailers. Retail industry identify and satisfy the consumer demands for the FMCG products through a supply chain [16]. There are various aspects affecting retail industry, such as weather situations, holidays, economic factors, and trends, to name a few. It has always been a challenge towards understanding the effect of such uncertainties and preparing to have a proper strategy. A decision support system can lead planning, preparing, performing, and monitoring the FMCG supply chain progress to improve.

In general, supply chain is the process of delivering products from supplier to manufacturer to wholesaler to retailer, and eventually to consumers, including management of both the product and the information flow, which is also known as supply chain management (SCM) [16].

1.4.2 Sales and Demand Forecasting

Sales forecasting is the process of predicting future sales in order to help decision makers make better decisions in planning, production, supplying, and marketing activities [1]. Companies are using several strategies to maintain their sales level during different periods of a financial year. One of such tactics is to hold sales campaigns, during which, a range of products is offered at lower prices to the retailers in order to present more quantity of a product for a specific time period. Understanding the amount of a particular product which is likely to be sold in that specified period in a specified location at a specified price is an

important task to achieve [17]. As in this project, we will analyze several large data sets in order to carry out a predictive analysis of Brynild Gruppen's sales. The company will be introduced in a later section.

Sales forecasting is conventionally considered to be a time series problem, when statistical analysis and models are used to make predictions. Whereas, it rather can consider to be a regression problem too, where machine learning methods can be used to find underlying trends and patterns in historical time series sales data and use it to predict the future sales, either in short-term or long-term [18]. Various predictors, being demographics, trends, competitors, marketing activities, etc., can add to the quality of the prediction model.

Various machine learning algorithms can be used for sales prediction. When a reasonable amount of data is available, several supervised machine learning methods and algorithms, such as Random forest, Gradient boosting machines, and neural networks can come into play [19]. On the other hand, when the aim is to predict a new product sale, unsupervised machine learning methods, such as K-nearest neighbor can be used [20]. In a real world problem, it is very important to optimize the models (e.g. by Hyperparameter tuning) to get the best prediction accuracy, on one hand, and generalize the model to fit for the actual prediction data (e.g. by Model stacking, Ensembles of models) on the other hand [21].

However, sales forecasting is more than a prediction, and it is rather projecting uncertainties [22]. There are various factors that can cause uncertainty in sales, such as, promotions, weather variations, competitors' activities, etc. Calculating the prospective uncertainties is a significant part of the task to be addressed. On the other hand, since the pattern in time series data are normally dynamic and the distribution of predictors are varying in course of time, the one-time model creation may produce wrong result after a short period of time. Hence the process of creating new models with new data should be considered, where automatic model selection can come into picture.

To conclude, sales forecasting as an important task for every business entity can be achieved using machine learning methods on historical time series data. In this project, we aim at deploying various machine learning algorithms to find the best possible prediction accuracy, along with maintaining model optimization and generalization, while considering uncertainties and time series data specific characteristics.

1.4.3 Brynild Gruppen

As mentioned earlier, this study investigates the sales data of a manufacturing company called Brynild Gruppen AS. Brynild Gruppen AS (BG) is one of the Norway's largest manufacturers of confectionary products founded in 1895, headquartered in Fredrikstad, Norway. The company has the turnover of about 760 million NOK per year out of 170 billion NOK value of the whole retail market in Norway. The current number of employees are around 220 and growing. Brynild company has 114 standard products in FMCG ranging from chocolates and confectionaries to nuts and dried fruits.

BG produces a range of confectionary products within candies, nuts, chocolates, dried fruits. Some of these products are displayed in Figure 1 [23]. The company delivers a total of 200 Stock Keeping units (SKU) per months to more than 4000 stores around Norway, through 40-50 different distribution centers that are operated by some of the major wholesalers of Norway. The company sells its products in other Scandinavian countries as well, but the main market is in Norway itself. One of the main distribution channels for Brynild's products is a wholesaler, named NorgesGruppen [24], which has 1850 grocery stores around Norway.

Figure 1 Product examples of Brynild Gruppen AS

1.4.4 Machine Learning

Since the methods and techniques that have been used throughout this master thesis is based on machine learning techniques and methods, we briefly explain various concepts and

phases of a typical machine learning workflow in this section. Followings are the steps to carry out for almost every machine learning based technique [25]:

1. Pre-processing of the available data for the desired Machine learning problem.
2. Division of the pre-processed data set into training, validation and test sets. (Varies based on the model and whether performing Cross-Validation or not)
3. Training of a model over the training data set.
4. Prediction of the target variable values on the test data set using the model.
5. Calculation of the accuracy and precision of the predicted target values.
6. Improvement of the model using optimization techniques.
7. Compare several models to select the best one.
8. Interpretation of the models, reporting and visualizing the results.

In what follows we briefly explain various steps along with concepts required to understand them.

1.4.4.1 Pre-processing of Data

The first step towards every machine learning procedure is the pre-processing of data, where the data undergoes various checks and processes to be ready for the desired machine learning model. Handling missing values, variable transformation, variable encoding, and normalization are some of the possible processes need to be considered in this step [25].
After preliminary data processing, such as handling missing values, we need to divide the data set into the features or the predictors set, and the target or the dependent variable. The target variable is in fact the variable that we are trying to predict using our machine learning model, which is the amount of sales in our case. Features or predictors are all other explanatory variables that we have utilize, comprising of product information, retail store information, and other socioeconomics variables. Next data preparation step is the task of dividing the data into training and testing sets. Machine learning problems are generally classified into being either a supervised or an unsupervised problem. It is called a supervised machine learning, since we have the historical labeled or target variable, for example Point-Of-Sales (POS) data, to train our model based upon. The training data set has chosen conventionally to comprise of 80% of the whole data, based on which the model trains itself by looking at the features and the target variables. In fact, the relationship between the features and target variables is identified during the training activity which is carried out over the training data set. We then evaluate the generated model by predicting

the target variables for the test data set and comparing the predicted values with the actual values of the target variable from the test set to measure the accuracy of the model [26].

1.4.4.2 Encoding

One of the procedures in preprocessing is to encode the categorical variables to have a numerical representation. This process is to convert the categorical features into numerical features ready for regression analysis. There are various encoding techniques available [27]. One of these techniques is called One-Hot encoding, which is the most common encoding way. One-Hot encoding is a binary style of categorization that allows the computer to understand different categories without interpreting them as labels. Another approach of encoding is called Hashing. Hashing uses a technique called hashing trick to carry out the encoding task. Comparing to One-hot encoding, the hashing technique uses less number of newly generated features. However, hashing technique introduces the problem of some information loss due to the incidence of collision that should be handled. The problem with one hot encoding is that, it creates a new variable column for every category present in a variable, which is making the dataset extremely big if number of categories are high, for example, store names in the case of retail data includes more than thousands of store names. Instead, one of the most efficient encoding techniques is Target Encoding. Target encoding uses the mean of the target variable corresponding to each category to calculate the new replacement for that particular category in the independent variable. The detail explanation of the steps taken to implement this approach is presented in the methodology section of this thesis.

1.4.4.3 Normalization

Since the target variable in the case of sales forecasting is the number of sales for each product, which is a numerical value, the problem that we address here is in the form of a regression problem. Basically, regression problems assume that the distribution of the data is normal. When a feature data is said to have a normal distribution it can be seen as a bell shape or a gaussian form. Different methods and techniques can be used to normalize a feature data, namely, RankGauss, BoxCox, Yeo-Johnson, and cubic root. Various Normalization techniques operate differently with respect to characteristics of the data. RankGauss is a technique introduced by one of the Kaggle competition winners, Michael Jahrer [28]. In this method, the rank of each value in a given feature column is first calculated using a rank based sorting function called argsort. Then the ranks are

transformed to lie in the range of -1 and 1. Finally, in order to make the distribution of the values in the form of gaussian, an inverse error function is applied [29].

1.4.4.4 Model training

After preprocessing the data has been completed the next step is to train a model using the training dataset. Different machine learning algorithms can be used to train multiple number of models and comparing the results of them to select the best algorithm, whereas different algorithms perform differently with respect to different types of data. Many algorithms are available to use, such as Random Forest (RF), Extreme Gradient Boosting (XGBoost), and Adaptive Boosting (AdaBoost) algorithms, to name a few. Neural Networks are another type of algorithm known as a deep learning technique, that can achieve great results in different problems. Furthermore, there are several other traditional algorithms as well, which are more based on statistical techniques, such as Support Vector Regression (SVR), K-Nearest Neighbor (KNN), Bayesian Ridge regression (BR), and Gaussian Process regression (GPR). Later in this section, a brief explanation of each of these algorithms are given.

After the training step, the model will learn the correlation between the features or independent variables and the target or dependent variable. To simplify, the machine learns from previous examples provided in training data to predict the future cases. This is when the trained model can be used to predict the values or labels of the test data set in order to evaluate the accuracy of the prediction. Here, we predict the values of the target variable and compare them with their actual values. One of the metrics to evaluate the performance of a desired model is to use the mean absolute error (MAE). Using the MAE, we can measure how far the average prediction is away from the exact values recorded in the data set [30].

1.4.4.5 Improving the Model

Normally, the generated models have an average predictive accuracy, and there is a need to improve the model to make it more reliable, performing better in terms of accuracy and efficiency. Currently, various methods are available for optimizing a so called predictive model. Following are the three common approaches that makes it possible to achieve highest possible accuracy. These methods are as follows:

- Feature Engineering
- Hyper-parameter optimization

- Using different machine learning algorithms to create new models.

Feature engineering is the process of extracting desired features by using either the experts' domain knowledge about the data or some statistical techniques [31]. Feature engineering involves the reduction of number of features if required. In some of the problems, extra number of features leads to a lower predictive accuracy as well as lowered speed of training. It is possible to carry out the task of feature reduction by following two techniques: (1) Feature Importance, and (2) Principal component analysis (PCA)

1.4.4.6 Feature importance

Feature importance is to identify those features that are acting the main roll in predicting the target variable. With this technique, it is possible to find out which features are more significant for the prediction of the target variable, hence removing the unnecessary variables to reduce the dimension of the model. Feature importance considers the effect of a variable over the prediction accuracy to select best participating features from the feature space.

1.4.4.7 Principal component analysis (PCA)

PCA is another dimensionality reduction technique that helps in lowering the complexity of the final model as well as speeding up the model training process. This dimensionality reduction is achieved by reducing the number of features into a lower dimension presentation. The number of component parameters are normally chosen to be 0.95, which means that the minimum number of principal components is chosen such that 95% of the variance of the variable is maintained.

1.4.4.8 Hyper-parameter Optimization

In order to optimize the performance of different algorithms, it is possible to tune their hyperparameters in such a way that they provide higher accuracy. Hyperparameters differ from model parameters that are learned during the model training. Different algorithms have different set of hyperparameters to tune. It is important to understand the effect of each parameter in order to tune it accordingly. Algorithms such as XGBoost have many number of parameters to tune, and the results of the prediction can dramatically change as a response to a different parameter set configuration. For example, some of the hyperparameters of XGBoost are as follow: min_child_weight, which is used to control overfitting, learning_rate, where lower values results in learning the specific characteristics of the trees, n_estimator, reg_alpha, and subsample, to name a few.

In order to find the best set of parameters different models with different combination of the parameters should be generated and compared. There are two methods for this task: (1) Grid search and (2) Random search. While tuning the hyperparameters, there is the possibility of overfitting. The term overfitting means that the model is so well fitted to predict the training dataset that is not generalizable to future test cases. In order to solve the issue of overfitting we can use Cross-Validation. In the following, we first explain Cross-Validation and then random and grid search Cross-Validation is explained.

1.4.4.9 Cross Validation

Cross Validation is a method that can be used to overcome the problem of overfitting. Overfitting is the situation where a model is too good to be true, that is, it is not generalizable to the real word data, and the results are perfect only for training data. This is normally occurs when the data set is small, or the training data is biased with a correlation with the target variable. The most common method of cross validation is K-Fold Cross-Validation. In this method the data is divided into k partitions and then the model is trained K times for K-1 of the partitions, which are also called as folds [32]. However, when it comes to Big data, in our case, having a high volume of records, there is no need to implement Cross Validation, since the chance of overfitting is assumed to be negligible.

1.4.4.10 Random Search Hyper-parameter selection with Cross-Validation

One of the technique to search the parameter space for the desired configuration is random search Hyper parameter selection that can also be carried out along with the Cross validation. In random search, an interval of the hyperparameters are chosen, within which the parameters are chosen randomly every time a model is created. This techniques results in finding a good set of parameters with minimum number of model creation [33].

1.4.4.11 Grid Search Hyper-parameter Selection with Cross-Validation

Instead, the Grid search technique, receives as input a fixed number of possible values for each parameter. Then, all of the combinations are considered as one potential set of parameters, upon which the model is trained and compared with others. This technique can be used after a smaller interval of parameters are identified in random search [34]. This is also possible to implement it using cross-validation technique.

Random search is performing faster, since it can be set to have specific number of sample parameters combination. However, random search tries to find parameters only between the given intervals and it is possible to miss some of the best fit parameters. Therefore, once

the limitation of the parameters are identified, Grid Search can be used for more precise examination, given that the computation power during this stage should not be an issue.

1.4.4.12 Generation of alternate models with different algorithms

However, an optimized model is not always the best option, and the choice of the algorithm itself can change the results of the prediction considerably. Therefore, once a model is optimized it should be interchanged with other algorithms to compare the best performing model in a particular application. In the rest of this section, we present some of the machine learning algorithms, among which three of them are chosen to be used in this study.

1.4.4.13 Regression Trees

One of the main types of machine learning algorithms are regression trees. These methods use decision trees to solve regression problems where the target variable is of type of numerical variables. Some of the regression trees are Random forest, Extreme Gradient Boosting (XGBoost), and Adaptive Boosting (AdaBoost). Although the performance of this applications differs in different scenarios, XGBoost normally presents better results [35]. In terms of speed, AdaBoost is much faster than XGBoost, especially when the amount of data is high.

1.4.4.14 Extreme Gradient Boosting (XGBoost)

Boosting is one of the machine learning techniques where an ensemble of weak learners is used to make a powerful classifier. The term weak learner relates to those models that are holding an accuracy of only about the average baseline. One of the latest and improved methods of Extreme Gradient Boosting is XGBoost. XGBoost is one of those models that is very flexible to tuning. That is, there are many number of hyperparameters that can be configured to increase the performance of the model. In recent years XGBoost has shown very good results in many different applications [36]. Hence it is important to study the performance of XGBoost in sales forecasting too. It is important to consider the size of the data when optimizing the XGBoost. This algorithm executes faster compared to older conventional methods, however, algorithms such as AdaBoost can operate faster, especially if the dataset is Big. XGBoost also has the possibility to be parallelized easily in order to make the generation of the trees in a parallel way. Setting of the hyperparameter ‘n-job=-1” results in full utilization of all CPU threads to run the XGBoost parallelly.

1.4.4.15 Support Vector Machine

One of the techniques that has been used in many applications of machine learning is known as Support Vector Machine (SVM). In supervised learning, this technique can be used for both classification and regression problems. The algorithm uses an n-dimensional space to define the feature variables. Then, a hyperplane is defined to be the best representative of the data points. In this project we have used Support Vector Regression. In Support Vector Regression (SVR) a linear function is learned based on a kernel function considered to be a non-linear function[37]. Parameter tuning can affect the results of the SVR into a great extent and should be carried out to get the best result.

Chapter 2

Related Works

2.1 Overview

In order to best understand various aspects of the study framework, we conducted a Systematic Literature Review (SLR). The scope of our review is based on the guidelines for systematic literature review in software engineering provided by Kitchenham and Charters [38]. The steps proposed by Kitchenham and charters guideline are as follows: (1) Detecting the essentials of the study, (2) Outlining the review protocol, (3) Identifying and choosing the primary researches, and (4) Performing the data extraction. Therefore, conducting a systematic literature review is to achieve an overview of the state of the science in order to identify, evaluate, and interpret relevant researches in the field of interest. In order to develop a protocol for the review we have used our previously formulated research questions, being as follows:

1. How should FMCG and retail data translated into sales and demand forecasting indicators? How should be the processing of this data? *(What to study?)*
2. What are the suitable Machine Learning algorithms for sales and demand forecasting using FMCG and retail data? (How to study?)

We then generated a search string to retrieve the related primary studies about our topic from scientific databases. In this line, we chose some broad terms and formulated our search string as follows:

(“Sales Forecasting” OR “Sales Prediction” OR “Demand Forecasting” OR “Demand Prediction”) AND “Machine Learning” AND “Sales”

The search term is formulated based on the method proposed in [39]. In this method, a Boolean AND is used to link the major terms and a Boolean OR is providing different possibilities of a term. In order to conduct a comprehensive literature review, we have used 6 of the most popular scientific paper databases as the source of primary studies, namely, Science Direct, IEEE Explore, Springer Link, ACM Digital Library, Wiley Online Library,

Taylor & Francis, and the Google general search engine to find any available white papers from the related industry.

The retrieval of the papers are based on a set of inclusion and exclusion criteria. An iterative approach is used to analyze the search results and the respected criteria for each of the iterations are as follows:

Criteria used in first iteration:

- Studies that are related to a Machine Learning approach.
- Studies that are efforts towards accuracy improvement.
- Studies that are efforts towards Model selection.
- Studies that are efforts towards Feature Extraction.
- Studies that examined the stability of an algorithm in the long term.
- Studies that are efforts towards Feature Selection
- Studies that are efforts towards optimizing the Machine Learning method in performance metrics, such as speed.
- Studies that are related to the topic but are not related to e-commerce and social media data such as sentiment analysis from reviews.
- The forecasting technique used in a study should have a machine learning based method to qualify that study for selection.
- If the Study focus is just the application of machine learning in a forecasting task, the focus should be on the food retail and FMCG and other industries such as fashion, energy, tourism, service, or airline, are excluded. However, if other specific cases such as optimization, feature selection, or model selection is considered, the study shall be included regardless of the context of the data.
- Studies related to promotions sales forecasting.

Criteria used in second iteration:

- Found irrelevant by not answering research questions
- Couldn't pass the quality checks

We have reviewed the resulted studies in two rounds. In the first round, all of the results from the search process are investigated by their topic, abstracts, keywords, and

conclusions, and first iteration criteria is used to either include or exclude the paper into our net stage review process. In the second round, the studies have gone through scrutinization by studying the full-text of the papers. Second iteration criteria is used to either include or exclude the papers in this step. After which, the final set of studies for our review are selected. The results of the process is demonstrated in the Table 1.

Database	**Number of Initial Search Result**	**Number of Retrieved Papers In 1st Iteration**	**Number of Selected Papers in 2nd Iteration**
Science Direct	490	62	28
IEEE Xplore	241	37	24
Springer Link	528	20	3
ACM Digital Library	67	8	2
Wiley Online Library	78	4	3
Taylor & Francis	46	4	1
Google Search	100	11	1
Total	1550	146	62

Table 1 Number of search result in each phase

Furthermore, the quality assessment of the studies are carried out by answering to a set of quality check questions based on the method provided in [40]. Each question is answered as being Yes or No and is recorded as 1 and 0, respectively. These questions are to assess the studies against one of the quality criteria, namely, bias, validity, and generalizability. The average value for a study to pass the quality assessment should be equal or higher than 0.5. The set of questions are shown in Table 2.

Quality Concept	Question Number	Question	Yes	No
Selection Bias & Measurement Bias	1	Does the study choose the subjects under study randomly?		
	2	Are the outcomes of the study interpreted based on the subjects under study?		
Validity	3	Is the study carried out with a scientific methodology?		

Quality Concept	Question Number	Question	Yes	No
	4	Are the methods used well-defined and verifiable?		
Generalizability	5	Is there a proper use-case to test the results?		
	6	Are the results general enough to be expandable to other situations?		
Novelty	7	Whether the study uses novel approaches		

Table 2 Quality assessment questions

In the next section the result of the literature review is provided which is in fact the answers to the literature review research questions.

2.2 Findings of the Literature Review

The results of this literature review is presented in this section by answering the respected research questions. Hence we try to answer our two main questions based on the literature, trying to identify 1) What to study, and 2) How to study.

2.2.1 What to study?

RQ1: How should FMCG and retail data be translated into sales and demand forecasting indicators? How should be the processing of this data?

2.2.1.1 Feature selection

According to the literature, various efforts have been done for using the data that the FMCG and retail industry collect for a possible sales forecasting solution. One of this efforts is regarding feature selection. In this context, the study presented in [41] uses a feature selection strategy called Multi-objective evolutionary feature selection in order to select optimal variables for online sales forecasting. They have implemented a wrapper feature selection mechanism which is basically to select the best combination of variables from a feature set search space. Considering multiple criteria while choosing the combination of variables is making it to be a multi-objective method. The authors of the aforementioned study, tested their proposed technique against some of the well-known approaches of

feature selection with the help of a test called hypervolume values, and resulted in a more efficient dataset.

In the same context, [42] proposed the usage of a method called Multivariable adaptive regression splines (MARS) for the variable selection process. MARS is an approach for finding the optimal variable combinations in a high-dimensional data. The authors of this study, used the hybrid approach of combining MARS algorithm with a support vector regression (SVR) for sales forecasting of different computer products. They have examined their proposed technique over a dataset and found that it is not only better than some other techniques, such as genetic algorithm combined with SVM [43], and ARIMA, but also has the ability to identify important predictor variables. [43] used the genetic algorithm based wrapper feature selection technique to analyze the data and select the set of appropriate variables, after which an SVM is used for the demand forecasting task. However, they have claimed that this approach presents a better result compared to SVM without feature selection or other approaches such as Winter Model.

Another hybrid approach is proposed by [4], where a combination of genetic algorithm and neural network is used for variable selection and sales forecasting, respectively. This study that is related to the FMCG industry and food products, investigates the performance of the proposed technique over a fresh milk sales data, and the result shown to be more efficient in terms of performance, compared to other conventional timeseries methods.

In another study, a stepwise linear regression is used for variable selection [44]. In this techniques, the most relevant variable is used to start the prediction process using a linear regression algorithm. Then, among the candidate variables the one that is most contributing to enhance the prediction accuracy is kept and the one that is less contributing is removed in each step. In a more recent study, authors chose to use the Weka tool for the sales forecasting task [45]. Hence, they have used the numerical feature selection method included in Weka, which has two parts: 1. Attribute evaluator with the help of correlation and relief method, and 2. A search method such as BestFirst.

2.2.1.2 Cluster-based approach

From a different prospective, one of the approaches that have been used by a number of sales forecasting studies is use cluster-based forecasting models [46]–[48]. This method makes use of a clustering algorithm to divide the training data into separate partitions and creating a specific forecasting model for every partition or cluster. However, [46] specifies

that the clustering method, the measurement of similarities, and the choice of variables, will influence the efficiency of the clustering-based methods. This study uses a K-means algorithm to cluster the sales training data, and an Extreme Learning Machine is used to create the forecasting models. They have compared their proposed method with other combinations of clustering and forecast modeling methods, and they found that the result in terms of accuracy is proved to be better.

The study presented in [48] instead, utilized a Self-Organizing Map (SOM) neural network to partition the sales data based on the characteristics of their sales behavior. In this study, the aim was to cluster the items based on the life curve, after which, a classification algorithm is used to assign the new items to the defined clusters. The results of the examination over a textile industry data shown an accuracy improvement of about 25% compared to other base models. In a similar manner, [47] used SOM to achieve the clustering task. The difference of their work is the use of principal component analysis to reduce the dimensionality of the data as well as removing the noisy data, prior to the clustering and modeling activity. This technique presents some improvements in the performance of the forecasting model.

2.2.1.3 Feature Engineering

There have been other attempts in the literature based on feature engineering to enhance the demand forecasting results. One of the more recent one is the work presented in [49], where a new set of customer related features is created based on their previous purchase time and value. Some of these newly generated features that are contributing in the task of future demand prediction are: Number of purchases, mean time between purchases, standard deviation of times between purchases, maximal time without purchase, time since last purchase, mean value of the purchase, and median value of the purchase. However, these features requires the data related to the customer behaviors, such as the customer loyalties and memberships.

Another approach of feature engineering is carried out by [50], where the sales forecasting problem is converted into a classification task, by transforming the sales data into three classes of substantial, middle, and inconsiderable sales. However, this method is used to identify whether a particular product sells well or not. As mentioned by the authors, this task is more useful on fashion retail, and in their study it is tested over a fashion retail dataset as well.

Last but not least in terms of feature selection and engineering, is the task of adding more number of explanatory variables to the sales data. One of such variables which have been used in different prediction tasks, is weather data. The impact of weather data over people's behavior have been explored in many number of previous studies [51]–[55]. A more recent study presented in [51], investigated the impact of weather variables over sales data of brick and mortar retailing sales. They have found that the weather condition has a huge impacts of 23% over the sales, based on the location of the stores, and about 40% based on the sales theme. However, these values can vary based on the industry under consideration. Hence it is interesting to investigate the effect of weather fluctuations related to the food industry. In a similar effort but different approach, the study presented in [56] studied the influence of weather variables over the data as well. The authors of this study, divided the prediction task into two separate cases of short-term to predict the sales in near future, and long term for control of long lead times.

2.2.2 How to study?

RQ2: What are the suitable Machine Learning algorithms for sales and demand forecasting using FMCG and retail data?

Recently, [10] presented a work regarding the criteria for classifying forecasting methods. Based on their findings, classifying forecasting methods as being either machine learning or statistical results is misinterpreting the results of comparison between different methods and their performances. Hence, they have suggested to categorize the forecasting methods into two main classes of (1) Objective: considering the mathematical properties of the models, and (2) Subjective: considering the methodological dimensions of the models. They have further identified a set of dimensions in each category based on which models are further classified. Since the methods we have found in our study are also overlapping between machine learning and statistics in many cases and there is not a clear line between them, we have used some of the dimension terms presented in this paper in order to structure our findings, all of which are listed in the Table 3.

Category	Dimensions
Objective	Global vs. Local Methods
	Probabilistic vs. Point Forecasts
	Computational Complexity

Category	Dimensions
	Linearity & Convexity
Subjective	Data-driven vs. Model-driven
	Ensemble vs. Single Models
	Discriminative vs. Generative
	Statistical Guarantees
	Explanatory/Interpretable vs. Predictive

Table 3 Forecasting method categories

However, some of the literature believe that from a forecasting method selection perspective, methods can be categorized into being as either statistical or machine learning methods. In this regard, [57] conducted a study to compare the performance of statistic methods with various machine learning based methods. They have classified the items in the historical data of a large grocery store as perishable and non-perishable products. The result of the study investigated the performance of ARIMA, SVM, RNN and LSTM with respect to predictive performance, generalization ability, runtime, cost and convenience. It is eventually demonstrated that SVM, RNN, and LSTM have a high predictive accuracy regarding perishable items, whereas ARIMA has a better runtime aspect. LSTM is shown to be better regarding cost and accuracy in non-perishable items.

2.2.2.1 Single Models

Literature also consists of various individual machine learning algorithms that have been used and tested for the task of sales and demand forecasting, which are categorized as single models with respect to the dimensions given in the Table 3. However, each study has its own goals and a corresponding approach to achieve these goals. One of the older studies in this context is the work presented in [58], where a model updating strategy is proposed to update support vector regression. This strategy is based on adding new data into the training data during the course of time. In the proposed approach, the training data contains two parts of historical and most recent data. This way it is guaranteed that the most recent data also affects the model building process. This technique have been tested using a real world data to predict the sales of a company one week ahead of time. As a result, improvement in the accuracy was obtained by the proposed so called dynamic SVR method.

Another study that explored the performance of SVR algorithm with retail dataset is the work presented in [59]. Mentioning that the huge size of retail data is a barrier for SVR.

Hence, they have proposed an algorithm called Row and Column Selection Algorithm (ROCSA) which selects a small but informative sample of the dataset for training the SVR model. While the row selection process picks a fraction of the whole dataset which represents the underlying patterns and their characteristics, the column selection process reduces the dimensionality of the data to increase the interpretability of the models. The result of the test cases over a real world retail dataset presented an increase of 39% of the accuracy compared to the original SVR model, which can considered to be an outstanding result.

In a more recent and advanced study, [60] proposed a big data framework by developing a back-propagation neural network-based classifier model that is trained by fuzzy inputs. In this study, other than historical sales data, a number of explanatory variables from advertisements, expenses, promotions, and marketing data are also considered. These variables are used to formulate a demand shaping effect form marketing activities. The framework is then tested on a supply chain data set and compared with a set of results from other statistical and machine learning algorithms such as, ARIMA, SVM, and random forest. The results have shown to be promising, specifically after consideration of demand shaping effect where the MSE decreased from 33.2 to 6.7, which is a huge difference.

The study presented in [61] conducted a comparison study between various machine learning based techniques for sales prediction of a retail store. Generally, a number of regression techniques are compared against gradient boosting algorithms and as the result it has shown that the boosting techniques outperform other regression methods in retail store sales forecasting. This study demonstrated that gradient boosting is showing a better result than Linear regression, Polynomial Regression, Lasso Regression, Ridge Regression, and AdaBoost. However, optimization techniques that can contribute in model improvements are not considered and hence the results cannot be reliable.

With another perspective, [12] has considered the demand prediction of semi-luxury items of the retail market. These products are those that are not purchased regularly and normally their prices are noticeably higher than regular products. In Norway, confectionary products containing sugar ingredients are lying under this category due to the high amount of tax imposed by the government on sugar containing products. The study presented in [12] investigated the performance of Random Forest algorithm over the weekly sales prediction of this particular types of products, incorporating a number of other variables such as, holidays, discounts, and regional factors. Although non-food products are considered in this study, its approach towards considering the high seasonality and variations of semi-

luxury products are noteworthy. Clustering of data based on regional factors such as holidays, unemployment rates, fuel prices, and store locations has shown a great impact on demand prediction adjustments.

2.2.2.2 Ensemble Models

Another approach towards sales forecasting in retail industry has been the combination of various methods to create an ensemble of models. This approach has been scrutinized to a great extent in literature and presented promising results in various settings [4], [21], [44], [46], [47], [62]–[65]. These studies believe that, the prediction accuracy of the combined models are higher than an individual model, hence they have suggested various combination of methods in order to take the advantage of multiple methods. In what follows, we scrutinize these studies in order to understand the cons and pros of each approach.

One of the older attempts around ensemble methods is the work presented in [62], where authors proposed a hybrid method by combining the SOM of neural network with case-based reasoning (CBR) for sales forecasting of new released books. This study combined two ML methods, namely ANN and KNN with case-based reasoning to cluster the past cases that is required to compare with the present cases. As an attempt to optimize the clustering of past cases a SOM is used to improve conventional CBR which requires a lot of time to distinguish between a new case and each of the past cases. The result of the study shown that the SOM neural network has better accuracy for sales forecasting compared with the K-mean method.

[21] analyzed the stability of a prediction model for a particular SKU over longer period of time, considering that it is not only the accuracy of the forecast that should be good but also, the algorithm is required to be stable over a long period of time. The authors proposed a new ensemble method using the averaging technique which considers both the accuracy and the stability to select the best model. In this approach two models of time series and regression based are used to create a primary forecast. Then a weight is generated for each of the models by the deviation of the forecasts, which is then multiplied to the forecasted values as the final forecast. The results of an experiment over a 3 months historical data shown that the ensemble method performs better compared with individual methods.

[63] conducted a study in sales forecasting of a drug store company. The authors implemented various linear, non-linear, and hybrid approaches to compare the performances. Adding to that, a composite model using Seasonal Trend decomposition

using Locally estimated scatterplot smoothing (STL) is designed. Three decomposed components of seasonal, trend, and reminder were forecasted by Snaive, ARIMA, and XGBoost. The results shown a better performance in STL than in individual or hybrid methods.

[64] suggested that Back Propagation Neural Network (BPNN) can be used for the prediction of market demand which has shown promising results compared to conventional statistical approaches. However, authors pointed that BPNN has some limitations such as the local optimization due to the random initialization, slow convergence, and low precision. Adding to that, the BPNN is not performing well with small sample size and more random uncertainties in data. Therefore, the authors proposed a method to enhance the performance of BPNN by using an AdaBoost algorithm, taking the neural network as a weak learner. The combine predictor model generated in this study was then tested by simulation of market demand statistical data and has shown improvements to the individual neural network.

Recently, [65] proposed a technique to combine deep learning models for sales prediction of a food shop that sells fresh prepared dishes, sandwiches and desserts. Two deep learning models of LSTM and CNN was combined to capture the long temporal dependencies of the data characteristics and to learn the local trend features, respectively. Since the parameter optimization of these models is a challenging task, two approaches of Particle Swarm Optimization (PSW) and Differential Evolution (DE) were used to automate the optimal architecture search process. The performance of the proposed technique is compared to a SARIMA model as a baseline solution and a better result was achieved in terms of prediction accuracy.

2.3 Literature Review Summary

In this study, a systematic literature review is conducted to investigate the related works to the sales and demand forecasting for FMCG and retail industry using machine learning techniques. The goal of this literature review is to identify the previous efforts regarding the sales forecasting and finding a ground for the rest of this master thesis. The studies presented in the literature provide the grounding and knowledge for finding the underlying scientific techniques and theories related the subject area. Therefore, it was required to review the literature rigorously to identify all of the techniques and methods used related to our formulated research questions. At the same time high number of not high quality

studies required a quality assurance check for study selection. Hence, a systematic literature review method have been chosen in order to scrutinize and review all the high quality related previous works.

In this regard, two research questions that have been formulated for this study are addressed with the aim of collecting and scrutinizing the related research studies. First one is to understand that how the FMCG and retail data should be translated into sales and demand forecasting indicators, as well as identifying possible variables and features that can be used in this task. Second one is to identify the suitable machine leaning algorithm and method for sales and demand forecasting task. To amalgamate the findings, the results of the literature search have been divided according to the research questions, aiming at answering the questions precisely based on the literature. After multiple rounds of literature review based on some inclusion and exclusion criteria, we have identified 62 research papers form 7 most common research databases.

Furthermore, studies answering the first research question are divided into three categories of either studying feature selection, cluster-based approach, and feature engineering. As a result it has been found that some of the possible feature selection techniques that have been utilized so far are Multi-objective evolutionary feature selection, Multivariable adaptive regression splines, genetic algorithm wrapper, and stepwise linear regression. Another approach towards sales forecasting is to divide the sales data into separate partitions with the help of clustering algorithms. This approach is carried out in a number of studies, however one important point here to consider is that the choice of clustering method is affecting the prediction result into a great extent. In general, k-means algorithm and self-organizing map are two main approaches for this task.

Feature engineering is another activity that contributes to the enhancement of the demand forecasting task. Creating a new set of related features based on past data have been the focus of some of the studies. Another approach in this regard is to convert the forecasting problem into a classification task by transforming the sales data. Adding to that, one of the most important approaches is to add more explanatory variables to the data set that not only improves the forecasting accuracy but also can overcome the issue of uncertainty in the sales patterns. Some of these variables have been weather, advertisement, marketing, expenses, holidays, discounts, and regional factors.

To answer the second research question, we have divided studies into two categories of using either a single model for prediction or an ensemble of models. In the case of single model approach, various algorithms have been utilized. Support vector regression, Back-

propagation neural network, extreme gradient boosting, and random forest have been among the most used techniques that also shown promising results. On the other hand, the combination of different machine learning techniques in order to enhance the prediction accuracy have been the focus of many studies. In this regard, combination of neural network with case-based reasoning, time series with regression techniques, ARIMA with XGBoost, BPNN with AdaBoost, and LSTM with CNN have been investigated.

To sum up, there have been various attempts in sales and demand forecasting, however, there are still a lot of research gaps in this era. Firstly, the sales forecasting specific to the FMCG industry have not been investigated comprehensively. Secondly, a complete comparison study between various machine learning techniques with proper optimization techniques that can maximize the precision have not been yet carried out. Thirdly, Inclusion of various explanatory variables and their contribution to the enhancement of forecasting task by overcoming the uncertainties are not investigated broadly. lastly, there have not been a similar study within Norway retail market. Therefore, A comprehensive study covering all of these gaps can be useful for both the industry and the research community.

Chapter 3

Methodology

3.1 Introduction

In this section, the methods and techniques used in this project are explained in detail. We first briefly provide an overview about the research methods that have been used in this study. Later, a roadmap towards answering the research questions will be presented along with a detailed step by step explanation of techniques and methods that have been used to answer those research questions. The research method that has been used in this study is the Design Science Research method presented in [66]. Design science research is defined as the efforts towards creating an innovative solution to a real-world problem and challenge. As discussed in [67] and shown in Figure 2, fundamentally, design science research method is based on three main activities. First, a relevance cycle, whereas the underlying application domain and problem statement is analyzed, second, a rigor cycle, which is to connect the problem with the knowledge based on scientific foundations,

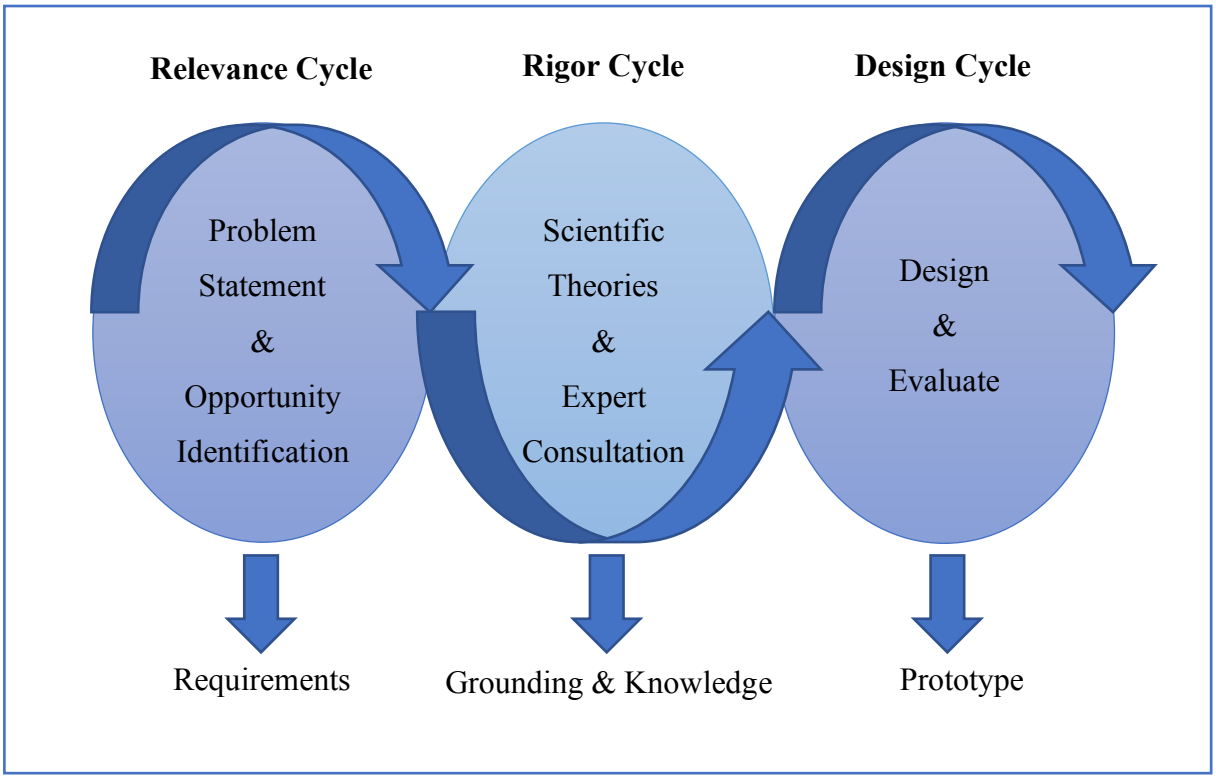


Figure 2 Study roadmap based on design science research

experiences, and expertise, and third, a design cycle which is to design and evaluate a prototype as a solution to the identified problem with respect to the scientific foundations. Hence we formulate the roadmap of this project based on these three stages as demonstrated in Figure 2.

3.1.1 Relevance Cycle

This stage of the process comprises of analyzing the application domain of the study and identifying the requirements and the problem to be addressed. In this project, the problem environment is a manufacturing company that is pursuing the improvement of their sales and operations. In order to understand the problem area, there is a need to gather some information about the company's day to day activities to rigorously formulate the problem and recognize the opportunities available within the organization and outside. Hence, a set of interviews and meetings with some of the managers and employees in the company were organized, along with a visit to the production site. As the result, the main problem of lack of a reliable method for predicting the customers demand has been identified, which leads to many uncertainties in planning, production, marketing, and generally decision makings. At the same time, the potentials of available data collected during sales and operation have been acknowledged to shed a light towards deploying data science and machine learning techniques as a solution to this problem. The relevance cycle activity will later expands towards verification of the answers to the research questions and the presence of knowledge added to the research area.

3.1.2 Rigor Cycle

After identifying the requirements and formulating the problem statement, we entered to the rigor cycle as the second stage of study. In this stage, a comprehensive literature review has been conducted to identify the underlying scientific theories and methods as a way of connecting the problem into a scientific research field. Given the availability of massive amount of data form heterogenous sources, the application of Machine Learning and Big Data research fields in Fast Moving Consumer Goods industry have been recognized as the main research focus of the study in order to synergically find a solution to the problem domain as well as contributing in complementing a new knowledge to the research field.
Furthermore, several rounds of consultation with experts in Machine Learning and Big Data fields have been conducted to formulate the walkthrough towards this project. It is

then decided to found the base of this project in scrutinizing the application of machine learning techniques along with managing the company's massive amount of data with the help of Big Data handling methods. Some of the main challenges regarding the enormous amount of unstructured and heterogenous data have been decided to be addressed with the help of Big Data handling methods. This leads to an understanding of how the future data should be collected, stored, and processed in order to facilitate future efforts in this particular Big Data application domain. Adding to that, the application of different data processing and machine learning techniques became the main focus of the study to analyze the issues and potential solutions to address the above said challenges.

3.1.3 Design Cycle

Design Cycle comprises of building a prototype grounded by the knowledge base and the investigation of different ways of optimizing the performance of this prototype. This process will later expands to the evaluation of the prototype as the main outcome of the study. The proposed prototype for this study is outlined as being a machine learning pipeline for the sales forecasting of Brynild company products with the goal of providing the ability to integrate it in their sales and operation processes. The pipeline is to be used for consolidating the data, preprocessing the data, creating different machine learning models based on different algorithms and various combination of their hyperparameter configurations, evaluating the models and picking the best reliable performing model, and finally, using it to predict future sales of a particular product in a particular store. Evaluation of the prototype will be carried out by conventional statistical evaluation methods. In what follows, the steps carried out in this stage is explained in detail.

3.2 Planning and Design

This study is an effort towards conceptualizing the main aspects of deploying data science approaches for the application of retail, and FMCG manufacturing decision makings. In fact, the main idea behind is to create a framework that acts as a roadmap for implementing data science and machine learning techniques for the problem of sales and demand forecasting. At the same time, since the amount of data available have the characteristics of Big data, a Big data handling approach should be taken into consideration while we design our framework. This framework is developed based on a machine learning pipeline

and is presented as a prototype implemented based on software engineering concepts. The approach of developing a framework based on software engineering concepts such as software architecture and design helps the corresponding practitioners, in Brynild company in our case, although generalizable, to use this study as a repeatable solution to their future

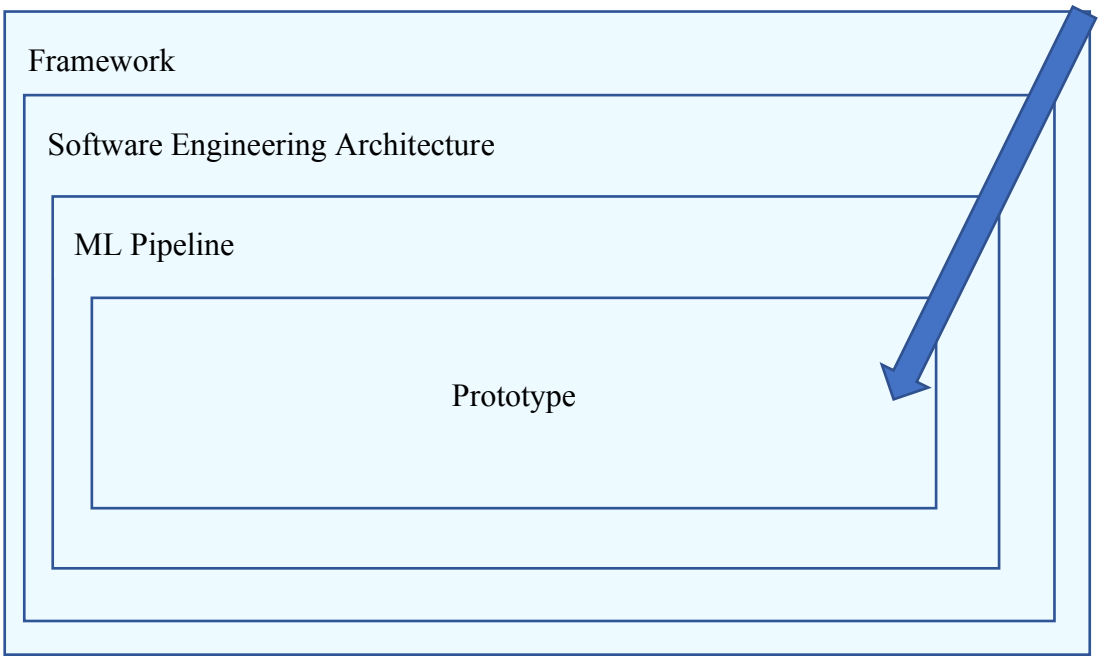


Figure 3 Overview of the structure design of the study

problems in the field of data science and machine learning. Figure 3 demonstrates the overview of the structure design of this study.

In what follows, we first explain the software architecture that we use in order to ground the development of our framework, and then we present the machine learning pipeline along with detail implementation of its components, and finally a prototype of the pipeline is implemented and exposed in order to deliver a touch of the principal workflows and processes.

3.3 A Software Engineering Architecture

The first step towards a software engineering based design solution is to found an underlying architecture that acts as a ground for further developments. In this study, we have utilized the Lambda architecture. Lambda architecture as demonstrated in Figure 4 has three main layers: 1) Batch processing layer, 2) Real-time processing layer, and 3) Serving layer. This architecture is used to handle both real-time streaming data as well as the stored batch historical data in the application of predictive analysis. The Batch

processing layer is where the processing of the historical data takes place. It is where all the massive amount of historical data is stored, pre-processed, and analyzed in order to create the most accurate and reliable predictive model. The real-time processing layer comprises of the streaming of the data using Big data processing technologies and the new incoming data is used to create predictive model out of the most recent generated data. Finally the serving layer is where the user interacts with the system by sending the desired queries for prediction either based on the historical data or based on the real time streaming data.

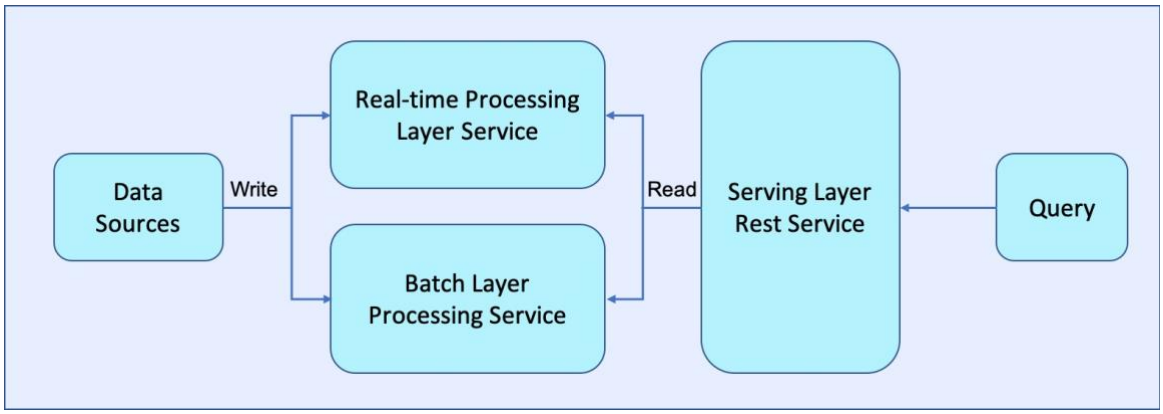


Figure 4 Lambda Architecture

In this project, the fact that the uncertainties in sales and demands are notably high due to the long list of factors that affect the market, such as the spread of the Covid-19 epidemic in 2020 that affected the supply and demand of the FMCG products around the world enormously, reported by Nielsen [68], leads to the applicability of the lambda architecture significantly. The Batch processing layer can be used for prediction of the sales and demand during the most of the periods of the year and the real-time processing layer can be used when the uncertainties are expected to be high, as another example during the festive seasons.

In our case study, we have the lack of streaming data at the moment, but this project serves as a proof of concept in order to convince decision makers for investments in developing streaming data collection processes in the future. In this regard, in order to present the workflow of the system in our prototype, we have exposed part of the data as a new streaming data into the system. The Apache Spark is used for the development of both the Real-Time and Batch layer, where Spark specific configurations have been part of the challenges of implementing this process, since these configuration are application dependent and a set of standard configuration is not available.

3.4 Machine Learning Pipeline

The Machine Learning pipeline in this project comprises of two main workflows: 1) Rel-time Prediction workflow, and 2) Historical Prediction workflow. Conventionally, real-time machine learning prediction applications are working overnight by processing the most recent collected data and make the predictions based on that data for next day's decision makings. In our pipeline, we have designed to have two processes in the real-time prediction layer of the architecture. First to analyze the most recent stored collected target data for the most up to date model creation and processing the features of the next week feature data to make the prediction. This part of the architecture can be used for short-term prediction during uncertain situations where historical data may result in miss-calculations. Historical data on the other hand is used to make the most accurate predictive model based on the full utilization of the historical data and make the predictions for the newly generated feature data. This part of the framework, can greatly be used for long-term prediction and planning for achieving better decision makings based on deeper analysis of the historical data. Figure 5 demonstrates the working flow of the pipeline:

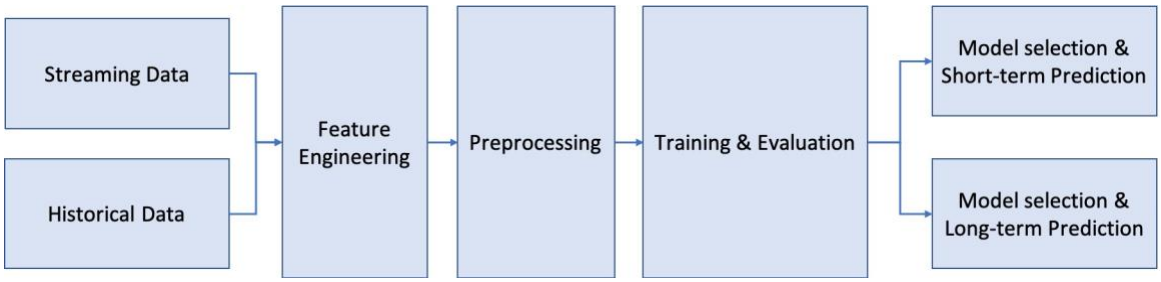


Figure 5 Pipeline Work Flow

3.4.1 Data Consolidation and Preprocessing

The first step towards every machine learning application is to make the available data ready for the utilization of possible machine learning algorithms in the first step, called data consolidation and preprocessing. This process normally involves, collecting relevant data from different data sources, consolidation of data into a unified tabular format, removing extra non-informative or less important features, adding new features either inferring from within the present data in a feature engineering step or from other sources of data, handling of missing datapoints, encoding, scaling, normalizing and finally, dividing the data into train and test sets. We explain these steps in the rest of this section.

3.4.2 Data collection and consolidation

Collection of the data into a unified data store is the first step for every machine learning pipeline. One of the best solutions for data collection is a to have a data lake, which provides the ability of storing data in different formats and structures with conserving the original form of the dataset. It is important for the pipeline that its processing layers can have access to an immutable form of the original dataset. Deploying a data lake is out of the scope of this project, however, we use a storage server to gather and store all of the data which reflects the characteristics of a data lake to some extent, considered to be acceptable for our prototype development. Basically, in the historical batch processing layer, data are collected from all available sources with the help of a separate ingestion service for each of the sources. These services collect the data in their original format and store the data in the storage server. Every dataset is then receives an identification number in order to later get accessed easier. The real-time streaming of data is also carried out through an online ingestion service powered by a real-time streaming engine such as Apache Spark. Figure 6 shows the data consolidations components of the pipeline at this stage:

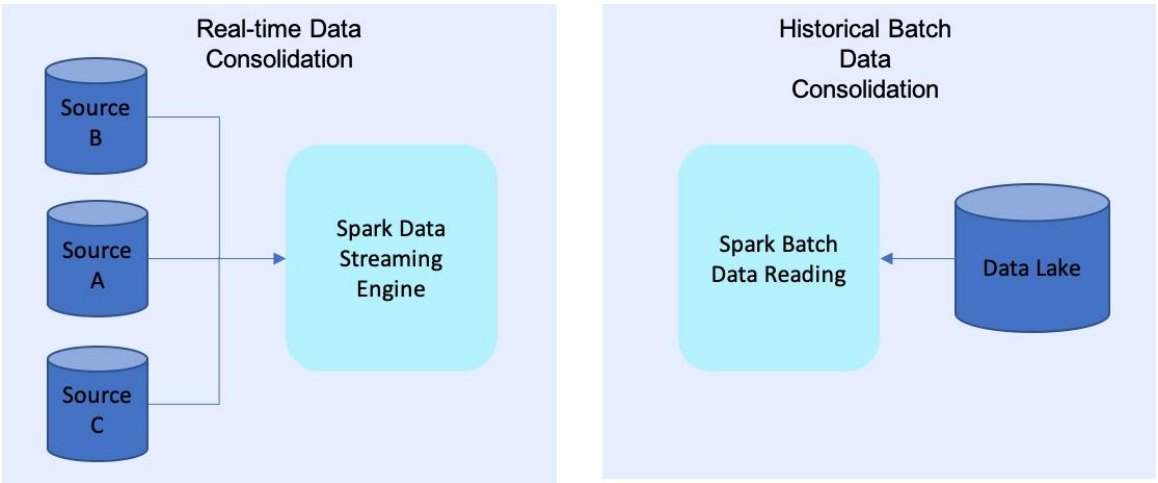


Figure 6 Data Consolidation Component of the pipeline

Brynild confectionary manufacturing company collects data from various sources in order to understand the situation of the market and evaluate their sales and operations processes to enhance their decision making activities. The data that have been made available to use in this study have been collected and stored in heterogenous data formats and needed to be consolidated into a well-structured tabular data in order to be used for machine learning application. Table 4 presents various primary raw data sources available:

#	Data Source	Description

1	DimensjonKunde	Data about each store
2	DimensjonVare	Data about each Product
3	DimVareAlle	Data about each Product
4	Grossistsalg	Total Aggregated Sales data
5	Krediteringer	Credits given to each store
6	PriserButikkNG	Price of the products
7	Nielsen	Total sale of the store company
8	TradesolutionData	Data about each store income
9	Ng_uttrekk_brynild	Total Item sales data (Point-Of-Sales Data)
10	AntallVarerButtikMoned	Total product in each month in each store

Table 4 Data Sources

The main source of data is the Point-Of-Sales (POS) data that have been collected from one of the main wholesalers of Norway, called NorgesGruppen that have been the main channel for Brynild's products distribution as well in recent years. The POS data is corresponding to about three years of Brynild's product sales starting from 2016 to the beginning of 2019 within various retail stores of NorgesGruppen. The data present the total number of consumer packages (known as F-pak (Forbrukerpakning) in Norway) sold for each particular product in each particular store for each day. There are several other attributes that come along the data regarding products' characteristics and stores' specifics. Table 5 presents initial attributes of available POS dataset:

Attribute	Description
#BUTIKKNAVN	Name of the store
KOMMUNE_NR	Code of the municipality of the store
FYLKE_NR	Code of the county of the store
DATODAG	Date and time of data recorded
UKEDAG_NV	Day of the week of data recorded
VARENAVN	Product name
VAREEAN_NR	Product Identification number (F-pak)
OMSETNING_INK_MVA	Total amount of sold at store from each product including tax
ANTALL_SOLGTE	Total number of products sold in each day at each particular store (F-pak)

Attribute	Description
ANTALL_HANDLER	Total number of customers buying a particular product
ANTALL_SOLGT_KAMPANJE	Total number of products sold in sale campaigns
ANTALL_HANDLER_TOTALT	Total number of customers buying from a store in a particular day

Table 5 Point-Of-Sales Data Attributes

Another source of data is the total number of D-paks sold and delivered to wholesaler's stores. Although this data is not representative of the sales of the products at each particular day, but BG can make a use of this data to predict the amount of orders in a particular duration in the future, compare it with predictions from points of sales data and plan accordingly. This data are collected from different tables provided by BG. Each table comprises of many number of variables but, for our particular application some of the attributes are discarded and the final set of features are presented in the Table 6 which presents the name of the each tables along with their attributes and description. Attributes marked with '*' are chosen as the key for each table. These datasets are joint to form our final dataset for model training.

Table	Attribute	Description
Grossistsalg	FylkeNR	Code of the county of the store
	Key_Kunde	Identification number of each store
	Oppstartsdato	Date of opening of store
	Profil	Store profile and size
	Snittomsetning	Store Average Turnover
	K Varenr*	Product Identification number
	Kjedeprofil_Id	Store company Identification number
	Tid_Id	Date and time of record
	Antall D-pak	Total aggregated number of Distribution packages sold at each date
DimensjonVare	Varenr	Internal Product Identification number
	Netovekt dpk	Net weight of the D-pack
	Seasongtype	Seasonality of each Product
	Lanseringsår	Lunch year of each Product

Table	Attribute	Description
DimVareAlle	Varenr	Product Identification number
	Materialart	Design group of the package
	Varegruppe	Product group
DimensjonKunde	Key_Kunde	Product Identification number
	PostNr	Post number of each store

Table 6 Aggregated Data

3.4.3 Data preprocessing

Preprocessing of the data includes, exploration of the raw data available in the data store, extracting features and converting them into a tabular dataset, feature engineering and transformation. In the collected data, there are many attributes recorded in various datasets that have been fetched from different sources, however, not all of these attributes are useful for model creation, since many of them are not informative, such as store address and telephone number, and many are only a different representation of others, such as the store name which is only a different representation of the store identification number. This process can be automated with some of the statistical feature selection techniques such as stepwise regression, canonical correlation, or Personal Component Analysis (PCA)..
Finally, all the desired variables are fetched from various tables and merged into a unified tabular format for further processing. In our prototype, this process is carried out with the help of Scikit-learn and Pandas data management library package of the Python programming language.

3.4.4 Feature Engineering

One of the main aspects of every machine learning pipeline is feature engineering. Regarding the sales and demand forecasting we have scrutinized the literature to find reliable techniques and methods based on which we have decided to deploy two methods in this particular application as follows: 1) Hidden Feature extraction, which is to create new set of features from already available features, and 2) External Features, which is to add more explanatory variables and checking their importance and correlation with our target variable, being number of sales for each product. In the first method, we have created

four types of new features, 1) Date-Related Features, 2) Domain-Specific Features 3) Lag Features, and 4) Expanding Mean Window.

Date-Related features are fetched from the date variable, to create following new attributes: Weekdays, Weekends, Week Start, Day of the year, Day of the week, Week of the year, Quarter of the year day, Month, and Year. These variables are chosen with prior consideration that sales and demand are pretty much related to the time of the purchase, and for example, the sales is very much affected during weekends, summer, or the month of December which is near to the new year. Hence, these variables could be an important factor for the prediction of the sale. Second set of new features are those specific to the sales data. These data are considered based on investigation of the literature and discussions with the domain experts. In this regard, the age of the shop is fetched from its date of opening, since it can have an effect over the store reputation and customer's loyalty and trustworthiness to the store, as well as the fact that people normally buy their needs from the well-known stores that they have used to do shopping from. Another feature is created form the date of first introduction of the product. Given that people use to buy products that they are familiar with, therefore, the age of product feature is also generated. Next type of attribute to be added to our feature set is the Lag feature. Lag uses target variable to create a new feature. This is in fact, to consider the previous recorded sales of a particular product for the prediction of the future sales. It is making the assumption that the amount of sales at time 't' is greatly related to the amount of sale at time 't-1'. These past values are known as Lags. Hence we create Lag-1 feature, which is the amount of sales of a particular product in its last recording. Finally, the expanding mean window feature is created. This feature is also based on the target variable and is generated by calculating the mean of the previous sales for each particular product in each particular store. It is known as expanding window since the calculated mean is expanding to the extent of all the previous data points in every new data point of the time series data.

Second method in feature engineering is to add other attributes called explanatory variables to the data set. These variables are normally gathered with a consultation with the domain experts. In this project, after a series of investigation, the set of features that are added to the data set are, average turnover of the shop, type of the shop, that is being either a supermarket or a hypermarket, Net weight of each product package, seasonality of the product, that is being either a normal product representation or a festive season product, the type of the packaging design, the category of the product, location of the shop, and the profile of the shop being one of the subdivisions of the NorgesGruppen chains. Following

this step, all of the data rows having null values are discarded, which is a required step prior to the machine learning model creation.

3.4.5 Target Encoding for Big Data

One of the important aspects for every machine learning pipeline is a mechanism for converting categorical variables into numerical variables. This is basically required for many machine learning algorithms, such as neural network, SVM, and linear regressions. Categorical variables are those variables that are either not having a numerical representation or those that are demonstration of a categorical distinction between data points. The interpretation of variables to either being a categorical feature or not is very much based on the domain area. For example, the variable 'Year' can be a categorical variable in a situation where different years have distinctive values of target variable and year itself being older or newer does not have any effect on the target variable. In our dataset there are many number of categorical variables that need to be transformed to some continuous numerical representation.

There are various approaches to address this challenge, the choice of which depends on the number categories in each variable. One of the common techniques is One-hot encoding, however, given that it creates one variable for every category of each variable and as we are dealing with Big Data having thousands of categories in some of the variables, it is not applicable and can cause memory and efficiency issues. Target Encoding is one of the promising techniques that not only handles the issue of high number of variables but also takes the relation of each variable with the target variable into consideration during generation of the new data points.

Target encoding is an easy method to implement by taking the mean of the target variable for each category and replacing the values of those variables with that mean. However, in order to stay away from the problem of overfitting, that is, the reflection of target variables in prediction, there is a need to do it in a special way. In fact, relying only on the average value of the target variable is not always the best option, since there maybe categories with very few number of items leading to an overfitting situation. There are two methods to overcome this issue. One is to use cross validation and computing the means in each out-of-fold dataset. In this project we use another approach called additive smoothing which is known to be IMDB method to rate it's movies [69]. The underlying idea behind additive smoothing is that, if a particular category has few number of elements, we should not rely

on its target mean anymore and we add the overall mean to our calculations. Mathematically it is equivalent to the Equation 1:

Equation 1 Target Encoding Formula

$$\mu = \frac{n \times \bar{x} + m \times w}{n+m}$$

Where:

- μ is the new data point we are calculating
- n is the number of items present in that category
- $\bar{x}$ is the mean of the target variable corresponding to each category
- m is the "weight" to allocate to the overall mean of the target variables
- w is the overall mean of all the target variables

In this technique, the only parameter to set is m, which should be decided based on the number of elements of the category that has the least number of items. As a result, we have converted all of the categorical variables into their target encoded numerical representation. The important point to consider here is that, the encoding of the variables in test set should be carried out based on the target variables in training set, since the target variables in test set are supposed to be unknown. Variables that are target encoded are as follows: ShopId, PostNo, Profile, ShopProfileId, StateId, ProductId, Seasonality, MaterialArt, ProductGroup, Day, Month, DayOfYear, WeekOfYear, DayOfWeak, QuarterOfYear, WeekEnd, and WeekStart. And those that are remained unchanged are: AvgTurnover, AvgTurnover, NetWeight, AgeOfProduct, Year, expanding_mean, and Lag_1.

3.4.6 Normalization

Input normalization has a great impact on neural network models, both in terms of accuracy and training speed. Instead, it is said to have no effect on other gradient-based methods such as XGBoost or tree based methods such as Random forest. In this project we have applied normalization to our dataset and compared the results for distinguishing between these effects on time series data sales. Our machine learning pipeline also includes an integrated Normalization module. RankGauss is used to implement normalization. This technique is previously explained in the background section of this thesis. The results of the Normalization will be presented later in the result section. From a theoretical point of

view, Since this technique uses a ranking of datapoints by sorting them, this examination becomes important to see if the RankGauss technique works on the target encoded data.

3.4.7 Train-Test Data split

The next piece of the pipeline is the split of data into train and test set. Given that the problem of sales forecasting is a time series problem by its nature, the time series characteristics should be taken into consideration. It is to divide the data in such a way that the timestamps on the recorded data are preserved. It is important not to shuffle the data to prevent using future data for prediction of the past. There is not a separate strategy for every layer of our pipeline in this part and we just need to establish a method such as an API call in order to give the ability to rest of the pipeline to call it as a service, whenever required. In this project, the train-test split function is implemented using the Scikit-learn library package and integrated into our prototype.

3.4.8 Model Training and Evaluation

After rounds of data processing and preparations, we finally have the desired dataset based on which we can create a predictive model. The model training part of the pipeline should be in such a way that it can interchange different machine learning algorithms. These algorithms should be optimized in an efficient manner in order to identify the best performing model. In this project we have developed various machine learning algorithms and every algorithm have been undergone the process of hyperparameter optimization in order to find the best performing model. Given the enormous amount of data, the process of hyperparameter optimization which involves the process of model training to be carried out for various combinations of hyperparameters takes a lot of time and should be optimized itself. The implementation of the hyperparameter tuning using Apache Spark provides the ability to parallelize this process in a distributed environment. We have used a 64 core standalone server to examine the working progress of the Spark application.
Machine learning algorithms that have been used in this application are among some of the well-known machine learning techniques that follow different approaches for model training. These methods have been selected carefully in order to compare the performance of different approaches for the application of sales and demand forecasting. Three machine learning algorithms of XGBoost, Random Forest, SVM have been used. These algorithms

are based on various approaches, namely, Gradient Boosting Tree, Ensemble trees, and Statistical techniques, respectively. Every algorithm undergone the hyperparameter optimization process and the best performing model out of each method is captured. After which the models are compared together in order to pick the best model for future predictions. The model evaluation process fetches the test data from the data split service of the pipeline and evaluated the performance of the model based on the most reliable metric for regression analysis, being Adjusted R-Squared, which can be used as the accuracy of the regression model. This metric shows a better performance as it is becoming closer to the value of 1. R-squared or coefficient of determination is the proportion of the variance in the target variable that can be predicted from the independent variable [70].
Spark has some configuration settings that need to be managed in order to control the memory utilization of the processes. The advantage of applying Spark over the standard Python library packages is the difference in the way they use the memory resource for processing the data. Python fetches all the data into memory at the beginning of the process, but Spark only loads the required part of the data into memory only when it needs it for its computations. However, since in this project we are parallelizing a multi-core standalone server, we need to manage the memory management of the processes with the help of Spark memory management configuration settings. This is in fact, part of the challenges required to handle Big Data in this project. Table 7 presents the final set of options that have been used. These options have been set by monitoring the application memory usage in Linux Operating System and the application behavior in Spark User Interface.

Setting	Value
spark.driver.memory	90g
spark.executer.memory	90g
spark.memory.offHeap.enabled	True
spark.memory.offHeap.size	15g

Table 7 Spark Memory Configurations

3.4.9 Optimization technique

The optimization technique used in this project is a Randomized search technique to find the optimal parameters through a predefined interval of a set of parameters. This Randomized search is implemented from scratch in order to design a parallel processing

technique for Hyperparameter search. Every random combination of the hyperparameters are given to one core of the system to train a model using Spark. A 64 cores Linux server have been utilized to parallelly train 64 models with 64 different configurations. However, the limited amount of memory available on one server constraints the full implementation of this technique, and a real distributed high performance computing cluster is required to achieve the highest performance and speed. In fact, Hyperparameter optimization is required to generate single model for every hyperparameter combination corresponding to an algorithm. This process can be parallelized to speed up the hyperparameter selection which is an important factor in dealing with Big data and achieving best model accuracies. In order to implement the parallelized Hyperparameter space random search, we have used the PySpark, which is a Python API for Spark. The latest release of Apache Spark (2.3) provides the ability to run native Python code with PySpark in an efficient manner. To be specific, the pandas library package of pandas can be used by having it in a user-defined functions (UDF) in apache spark. The user defined function here is the implementation of the random selection of hyperparameters. Then the data is replicated 64 times, and a replication identification number is attached into each replicated data. Then each group of data is feeded into one of the cores using Spark to run parallelly. The implementation of this technique is provided in Appendix A.

Chapter 4

Results and Evaluation

In this section we present the detail description of the results and evaluation of the study. Results will include the outcome of various stages of the progress in order to present a verification of the methods and techniques deployed. This section acts as both the report of the findings and a demonstration of the progress of the study. We first demonstrate a full presentation of the machine learning pipeline developed in this study. Then the results of applying this ML pipeline for our case study data is provided. To show the progress of the study, we demonstrate the effect of the transformations over the dataset, and then the results of applying each of the machine learning algorithms over the dataset along with their gradual optimization progress are presented. Finally these results are compared and evaluated.

4.1 FMCG Sales Forecasting ML Pipeline

Figure 7 shows the overall presentation of the machine learning pipeline developed in this study. This pipeline acts as the underlying architecture of the prototype, being the outcome of the design science research method deployed in this study. The proposed machine learning pipeline presents the working and the data flow for a sales forecasting application adapted for the FMCG and retail industry. The adaptation considered the sales uncertainties and short shelf life characteristics of the application domain. The design of this pipeline is based on the Lambda architecture of software engineering which starts with the consolidation of data from two channels of real-time data, and batch historical data. It then enters to the Feature Engineering module, where FMCG sales forecasting related features are generated. The output of the this stage is the preprocessed data that goes to preprocessing step under Missing Value handling, Feature Encoding, and Normalization. Then the train-test data split divides the data to train and evaluation sets which is to be accessed through an API service.

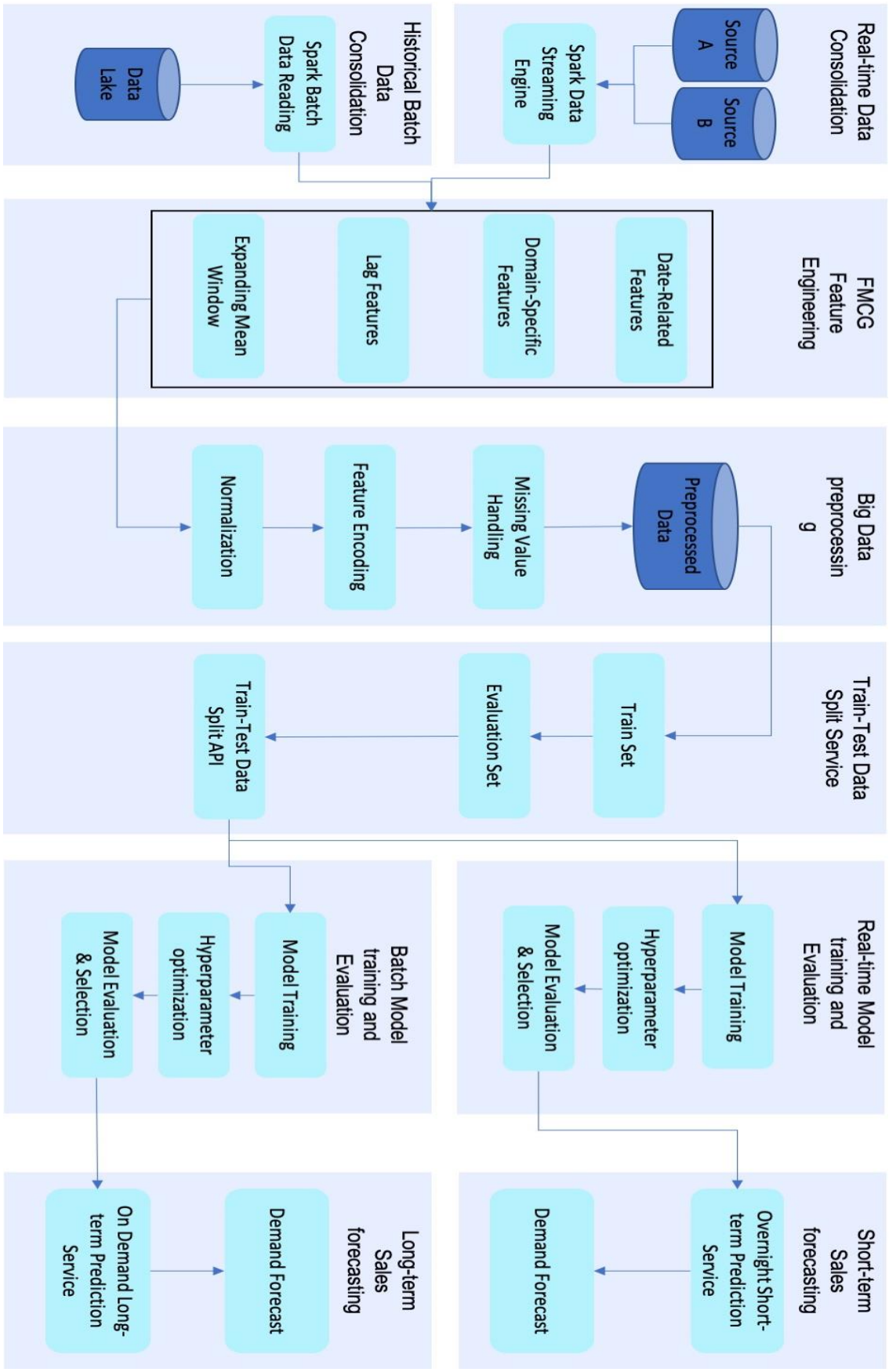


Figure 7 Machine Learning Pipeline for FMCG and Retail IndustryFeature Engineering and Transformation

At this stage, the prototype presents two services to either train the machine learning models based on historical batch data or the real-time streaming data. Various models are deployed, optimized, and evaluated at this stage to choose the best predictive model.

In this section the effect of the normalization over the dataset is presented using graphs generated with Python plotting library. The results demonstrate that the data is transformed into a proper gaussian bell shape as desired in order to be zero centered. Figure 8 shows the Kernel Density Estimate (KDE) plot for some of the features and the target variable. KDE plot is used to visualize the probability density of the variables. The plots demonstrated the Normalization behavior over four different variables. One of which is the target variable.

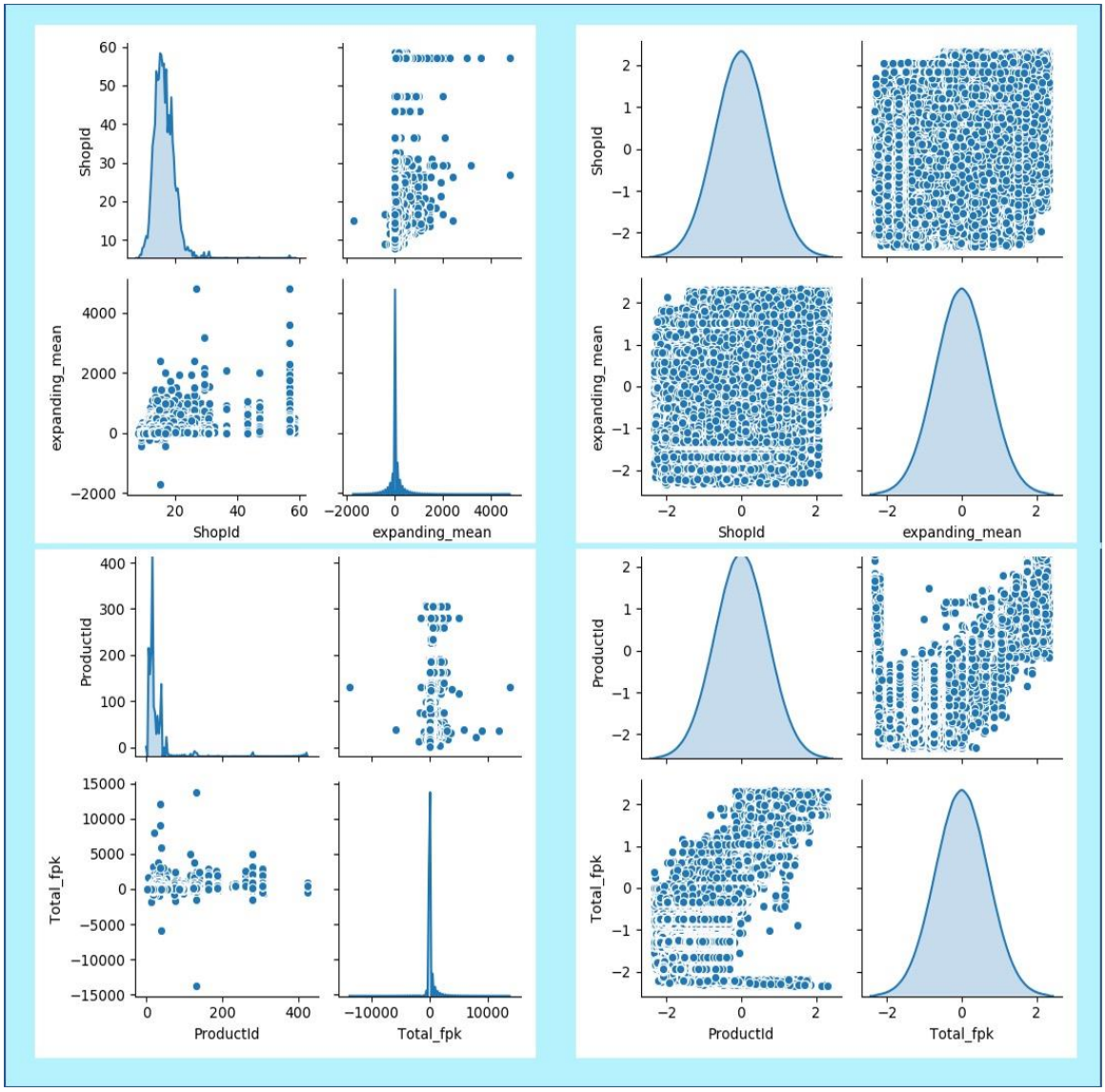


Figure 8 Before and After Normalization effect on four of the variables

After normalizing the feature vectors, all the columns are transformed into a common scale which prevents machine learning algorithms from the misleading difference in attribute's scales. Although, conventionally, normalization is said to have less significant effect on gradient boosting and Tree-based algorithms, we found that the results of the model prediction significantly increased after the normalization of the data. Anyhow, normalization is proved to have an important effect over Support Vector Regression.
Another consideration is that whether to Normalize the target variable or not? This question is also answered by testing the algorithm over the dataset with both normalized target data and Non-Normalized target data, and results shown a significant improvement when the target variable is also normalized. Table 8 shows the model accuracies before and after Normalization for both the datasets.

Model	Adjusted R-Squared for Dataset with Normalized Features only	Adjusted R-Squared for Dataset with Normalized Feature and Target
XGBoost	0.82	0.87
Random Forest	0.80	0.86
Support Vector Regression	0.63	0.83

Table 8 Results of the Models

4.2 Model Training and Optimization

XGBoost is one of the best performing algorithms in machine learning, although its performance is very much related to the setting of its hyperparameters. Hence a great effort have been put into hyperparameter optimization in this project. As the result of the hyperparameter optimization we have achieved a high value of adjusted R-squared. We have conducted 100 rounds of model training with randomized parameter space search to examine various combinations of hyperparameters. The results of some of the training rounds that had significant improvement of 0.01 in adjusted R-squared for the XGBoost algorithm along with their parameter settings are presented in Table 9. The results are sorted in a descending order based on the adjusted R-squared. Although the primary comparison metric for models is root mean square error, this metric does not show the acceptability level of the model. This is because the variance of the target variable prevents

the performance representativeness of this metric. Hence, the adjusted R-squared has been chosen.

Iteration	adjusted R2	Col_ sample_ bytree	gamma	min_ child_ weight	learning_ rate	max_ depth	n_ estimators	reg_ alpha	reg_ lambda	Sub sample
30	**0,87**	**0,5**	**0,05**	**10**	**0,05**	**9**	**50**	**0,75**	**1,00E-05**	**0,8**
32	0,86	0,5	0,07	8	0,07	9	50	0,75	1,00E-05	0,8
4	0,85	0,5	0,07	12	0,07	7	100	0,5	1,00E-05	0,8
20	0,84	0,5	0,05	4	0,05	7	100	0,5	1,00E-05	0,6
56	0,83	1	0,05	12	0,07	9	100	0,75	1,00E-05	0,8
60	0,82	1	0,05	4	0,09	7	200	0,5	1,00E-05	0,6
34	0,81	1	0,07	10	0,07	7	200	0,5	1,00E-05	0,6
40	0,80	0,5	0,03	4	0,5	7	200	0,5	1,00E-05	0,8
42	0,79	1	0,07	12	0,5	5	50	0,75	1,00E-05	0,6
57	0,78	1	0,07	12	0,5	7	100	0,75	1,00E-05	0,6
82	0,77	0,5	0,05	8	0,5	7	200	0,75	1,00E-05	0,6
46	0,75	1	0,03	4	0,5	7	100	0,5	1,00E-05	0,6
51	0,74	0,5	0,05	12	0,5	7	200	0,5	1,00E-05	0,6
93	0,71	1	0,03	8	0,5	9	200	0,75	1,00E-05	0,8

Table 9 Hyperparameter optimization of XGBoost

Random Forest algorithm have shown a greater change of adjusted R-Squared with different parameter settings. As it can be seen from the results of the hyperparameter optimization in Table, the difference between the lowest and highest the adjusted R-Squared is as high as 0.15, which is a significant value. Hence, iteration number 75 has shown the best result of the Adjusted R2 of 0.81, which is very much near to the XGBoost result.

Support Vector Regression is implemented similar to a support vector machine, but the corresponding hyperplane is the one which is best representative of the target variable data points instead of classifying them. SVR has few parameters to set. It starts with the kernel type. Kernel helps in finding the corresponding hyperplane in higher dimensional spaces without increasing the computational cost. We have tried different kernel functions to find the best one for our problem. Three of the available kernel functions have been examined in the optimization process: 1) Polynomial Kernel, 2) Linear Kernel, and 3) Radial Basis Function (RBF). The result of the model fitting for each of the kernel functions with different combination of hyperparameters are presented in Table 10.

Iteration	Adjusted R2	Kernel	C	gamma	Epsilon	Degree	Coef0
1	0.833	Poly	100	Auto	0.1	3	1
2	0.795	RBF	100	0.1	0.1	NA	NA
3	0.660	Linear	100	Auto	NA	NA	NA

Table 10 Support Vector Regression Model Optimization Results

The result of the SVR model clearly demonstrates that using the polynomial kernel function produces the best result.

Comparing the results of the three chosen algorithms, namely XGBoost, Random Forest, and Support Vector Regression shows that XGBoost presents a better result with an Adjusted R2 of 0.87. However, this result and the configuration of hyperparameters are pretty much related to the amount of data and the process of algorithm selection should be automated in the ML pipeline as a continues service.

Chapter 5

Discussion

In this study, we have utilized a design science research method in order to investigate the best way of applying machine learning techniques into the problem of sales and demand forecasting within Fast Moving Consumer Goods and retail industry. The result of the study was twofold. First, we have investigated the steps required for the development of a machine learning pipeline that fits the application of sales and demand forecasting. This machine learning pipeline servs as a baseline for a prototype, which is the required outcome of a design science research. Then, we have applied various machine learning algorithms and techniques to find the best solution for the specific case of FMCG and retail industry. Second, a case study about the sales of the confectionary manufacturing company, Brynild Gruppen, has been conducted to examine the outcome of the proposed prototype. As the result of these activities, the result of this thesis is answers two research questions. Hereinafter, we present the answers to our research questions that we present here for recall:

1. How should FMCG and retail data be translated into sales and demand forecasting indicators? How should be the processing of this data? *(What to study?)*
2. What are the suitable Machine Learning algorithms for sales and demand forecasting using FMCG and retail data? (How to study?)

The answer to the first question is achieved with the development of a machine learning pipeline as a prototype fitted for sales and demand forecasting in FMCG and retail industry. The development of the prototype has been carried out based on a Software Engineering architecture design, and the Lambda architecture has identified to be the best solution for this particular application. This is because the nature of the FMCG and retail industry is found to be in such a way that, many factors are affecting the sales and demand. On the other hand, the occurrence of special situations such as the an epidemic outbreak or a festive season leads to drastic rise of uncertainties in sales and demand. In this case, there is a need to have two types of prediction, a short-term prediction, and a long term prediction. The proposed machine learning pipeline based on a Lambda architecture provides the possibility of having both the predictions based on both real-time data processing and the

historical batch data processing. In the case of the short-term prediction during normal situations, both the services can be used, and during special external factor influenced situations, only the short term prediction will be considered. Moreover, the historical batch data can be used for long-term prediction which is useful for long-term planning of the production.

Another focus of this study was to efficiently process the company's Big Data. To do this, we have utilized Big data processing techniques and tools to examine the applicability of them in our application area. In this regard, we organized our workflow into two directions: First, to preprocess the data in such a way that the massive amount of data taken into consideration, where we have applied an specific way of target encoding and normalization to not only preserve the originality of the data, but also increase the quality of the data by encoding the feature variables based on the target variable. These procedures act as the preprocessing stage of the proposed machine learning pipeline. Second, the model training and hyperparameter optimizations should have been handled with the help of Big data handling applications in order to not only accelerate the process but also to improve the efficiency of model training stage of the pipeline.

To answer the second research question we applied machine learning techniques over Brynild Gruppen's sales data. During the preprocessing stage, two steps of normalization and feature engineering have been carried out with an in depth knowledge of the data characteristics that was captured from the domain experts presents in the Brynild company. Normalization have been carried because the data have been analyzed to be sparse in distribution and given the intrinsic requirements of regression models and specially algorithms such as Support Vector Machine and Neural Network. Hence we have first utilized the standard scalar technique of adding the overall mean and dividing by the standard deviation. However, given the massive amount of data and the strong sparsity of the data, being Big data's characteristics, this techniques found to be not the best option. Hence the especial method of RankGauss have been used that presented a promising and reliable outcome. Therefore, it is of a significant value to consider this technique in future developments.

On the other hand, the nature of the data being a timeseries data provides the opportunity to scrutinize the underlying hidden features in data and extract them accordingly. This process that leads to generation of new features has a significant effect on the predictive power of the models. Extracted features have been either date related features, or inferred from the target variable itself.

Another significant outcome of this study to consider is the utilization of different machine learning algorithms and carrying out a comparison study to identify the best performing model in this particular application. We have implemented three of the most efficient machine learning algorithms, namely XGBoost, Random Forest, and Support Vector Regression. The selection of algorithms have done carefully in order to include different approaches of machine learning, being gradient boosting, tree-based, and statistical methods. Although the results have shown satisfying in almost all of the methods, basically due to the reliable outcome of the preprocessing and feature engineering step, the performance of the XGBoost shown to be the best among others with an adjusted R-squared of 0.86.

Chapter 6

Conclusion & Future Work

Sales and demand forecasting has always been one of the main issues of the FMCG and retail Industry. Having an accurate prediction of the amount of sale helps all the supply chain actors to plan and operate accordingly. This leads to a more efficient, robust, effective, and sustainable supply chain operation. At the same time, the amount of data generated and stored by supply chain actors are becoming enormous. These data are collected from heterogenous sources and satisfies the characteristics of Big data by having six main Vs of Big data, being, Volume, Velocity, Variety, Variability, Veracity, and Value. Therefore, it is crucial to formulate a roadmap towards utilization of this data. This formulation should be in such a way that, it considers the specifics of FMCG and retail industry, being demand uncertainties. On the other hand, machine leaning techniques have shown a great potential in providing a solution for this type of problem, although many challenges remained unanswered. In this regards, this study is formulated to answer two main research question: First, how to use the FMCG and retail data for sales and demand forecasting, and 2) Which machine learning techniques and methods should be used for sales and demand forecasting.

We then formulated a design science research method towards conducting this study, where three stages of relevance cycle, rigor cycle, and design cycle have been carried out. In relevance cycle we have recognized the specific problem statement along with the opportunities present in this domain with the interviews and meetings hold with companies experts. As the result of this stage requirements for the potential solution have been identified. In the rigor cycle, the goal was to identify underlying related scientific theories in the study field in order to study and add to the field knowledge. This part have been done with the help of conducting a systematic literature review and consultation with machine learning experts. Finally, in the design cycle we have designed a prototype, being a machine learning pipeline, presenting various stages that the data should pass through for developing an efficient and reliable predictive model for sales and demand forecasting.

Research questions have been addressed by conducting a case study over a confectionary manufacturer company, named Brynild Gruppen AS. The results answered the research

question by providing a detail and step by step procedure for developing a machine learning pipeline that performs different operations of data ingestion, data preparation, feature engineering, data split, model training, and prediction. We have deployed a software engineering approach for implementation of the pipeline. Hence, a Lambda architecture have been developed, given the especial characteristic of the application domain being demand uncertainty. Both real-time data processing, and historical batch data processing have been considered for making a distinction between short-term and long-term prediction.

Furthermore, the results of applying three different machine learning algorithms, namely, XGBoost, Random Forest, and Support Vector Machine shown that after preprocessing of data, XGBoost has the best outcome. At the same time the process of hyperparameter optimization can be achieved in an accelerated manner with the help of the Apache Spark as a Big data Handling tool.

At the time of this study, there was a limitation of confidentiality about some the available data, and more data regarding the customers behavior and competitors sales could not be added to our dataset. However, given the opportunities and potentials of this type of solution for all the supply chain stakeholders, raises the significance of information sharing throughout the supply chain. Hence, in the future, we are considering adding more explanatory variables to the data in order to develop more reliable predictive models. Adding to that, the unavailability of real-time data at the moment limits the examination of the pipeline in the case of real-time data ingestion. Therefore, the examination of the FMCG real-time data collection, processing and model creation should be considered as a future study, both using real word data and simulation techniques. Last but not least, is the deployment and integration of such a machine learning pipeline into the company's sales and operation that will introduce new set of challenges, that should be studied in the future as well.

Bibliography

[1] P. Trkman, K. McCormack, M. P. V. de Oliveira, and M. B. Ladeira, "The impact of business analytics on supply chain performance," *Decision Support Systems*, vol. 49, no. 3, pp. 318–327, Jun. 2010, doi: 10.1016/j.dss.2010.03.007.

[2] T. Huang, R. Fildes, and D. Soopramanien, "The value of competitive information in forecasting FMCG retail product sales and the variable selection problem," *European Journal of Operational Research*, vol. 237, no. 2, pp. 738–748, Sep. 2014, doi: 10.1016/j.ejor.2014.02.022.

[3] D. Gupta, "FMCG Case Study," in *Applied Analytics through Case Studies Using SAS and R: Implementing Predictive Models and Machine Learning Techniques*, D. Gupta, Ed. Berkeley, CA: Apress, 2018, pp. 345–396.

[4] P. Doganis, A. Alexandridis, P. Patrinos, and H. Sarimveis, "Time series sales forecasting for short shelf-life food products based on artificial neural networks and evolutionary computing," *Journal of Food Engineering*, vol. 75, no. 2, pp. 196–204, Jul. 2006, doi: 10.1016/j.jfoodeng.2005.03.056.

[5] E. J. Marien, "Demand Planning and Sales Forecasting: A Supply Chain Essential," 1999. https://www.semanticscholar.org/paper/Demand-Planning-and-Sales-Forecasting%3A-A-Supply Marien/f277c96fb50be9346f69bc7ca354c14619b00042 (accessed May 13, 2020).

[6] J. T. Mentzer and M. A. Moon, *Sales Forecasting Management: A Demand Management Approach*. SAGE Publications, 2004.

[7] M. Lawrence, M. O'Connor, and B. Edmundson, "A field study of sales forecasting accuracy and processes," *European Journal of Operational Research*, vol. 122, no. 1, pp. 151–160, Apr. 2000, doi: 10.1016/S0377-2217(99)00085-5.

[8] J. T. Mentzer, C. C. Bienstock, and K. B. Kahn, "Benchmarking sales forecasting management," *Business Horizons*, vol. 42, no. 3, pp. 48–56, May 1999, doi: 10.1016/S0007-6813(99)80021-4.

[9] R. Fildes and C. Beard, “Forecasting Systems for Production and Inventory Control,” *International Journal of Operations & Production Management*, vol. 12, no. 5, pp. 4–27, Jan. 1992, doi: 10.1108/01443579210011381.

[10] T. Januschowski *et al.*, “Criteria for classifying forecasting methods,” *International Journal of Forecasting*, Aug. 2019, doi: 10.1016/j.ijforecast.2019.05.008.

[11] M. Bohanec, M. Kljajić Borštnar, and M. Robnik-Šikonja, “Explaining machine learning models in sales predictions,” *Expert Systems with Applications*, vol. 71, pp. 416–428, Apr. 2017, doi: 10.1016/j.eswa.2016.11.010.

[12] T. Qu, J. H. Zhang, F. T. S. Chan, R. S. Srivastava, M. K. Tiwari, and W.-Y. Park, “Demand prediction and price optimization for semi-luxury supermarket segment,” *Computers & Industrial Engineering*, vol. 113, pp. 91–102, Nov. 2017, doi: 10.1016/j.cie.2017.09.004.

[13] N. Syam and A. Sharma, “Waiting for a sales renaissance in the fourth industrial revolution: Machine learning and artificial intelligence in sales research and practice,” *Industrial Marketing Management*, vol. 69, pp. 135–146, Feb. 2018, doi: 10.1016/j.indmarman.2017.12.019.

[14] J. P. Karmy and S. Maldonado, “Hierarchical time series forecasting via Support Vector Regression in the European Travel Retail Industry,” *Expert Systems with Applications*, vol. 137, pp. 59–73, Dec. 2019, doi: 10.1016/j.eswa.2019.06.060.

[15] H. N. Perera, J. Hurley, B. Fahimnia, and M. Reisi, “The human factor in supply chain forecasting: A systematic review,” *European Journal of Operational Research*, vol. 274, no. 2, pp. 574–600, Apr. 2019, doi: 10.1016/j.ejor.2018.10.028.

[16] J. T. Mentzer *et al.*, “Defining supply chain management,” *Journal of Business Logistics*, vol. 22, no. 2, pp. 1–25, Sep. 2001, doi: 10.1002/j.2158-1592.2001.tb00001.x.

[17] “Forecasting and Demand Modeling,” in *Fundamentals of Supply Chain Theory*, John Wiley & Sons, Ltd, 2019, pp. 5–44.

[18] O. Valenzuela *et al.*, "Hybridization of intelligent techniques and ARIMA models for time series prediction," *Fuzzy Sets and Systems*, vol. 159, no. 7, pp. 821–845, Apr. 2008, doi: 10.1016/j.fss.2007.11.003.

[19] N. Vairagade, D. Logofatu, F. Leon, and F. Muharemi, "Demand Forecasting Using Random Forest and Artificial Neural Network for Supply Chain Management," in *Computational Collective Intelligence*, 2019, pp. 328–339.

[20] K. Afrin, B. Nepal, and L. Monplaisir, "A data-driven framework to new product demand prediction: Integrating product differentiation and transfer learning approach," *Expert Systems with Applications*, vol. 108, pp. 246–257, Oct. 2018, doi: 10.1016/j.eswa.2018.04.032.

[21] N. C. D. Adhikari, R. Garg, S. Datt, L. Das, S. Deshpande, and A. Misra, "Ensemble methodology for demand forecasting," in *2017 International Conference on Intelligent Sustainable Systems (ICISS)*, Dec. 2017, pp. 846–851, doi: 10.1109/ISS1.2017.8389297.

[22] C. I. Papanagnou and O. Matthews-Amune, "Coping with demand volatility in retail pharmacies with the aid of big data exploration," *Computers & Operations Research*, vol. 98, pp. 343–354, Oct. 2018, doi: 10.1016/j.cor.2017.08.009.

[23] "Brynildgruppen - Hjem." https://www.brynildgruppen.no (accessed Nov. 22, 2019).

[24] "NorgesGruppen." https://www.norgesgruppen.no/ (accessed May 14, 2020).

[25] P. Harrington, *Machine Learning in Action*. Greenwich, CT, USA: Manning Publications Co., 2012.

[26] I. Bruha, "From machine learning to knowledge discovery: Survey of preprocessing and postprocessing," *Intelligent Data Analysis*, vol. 4, no. 3–4, pp. 363–374, Jan. 2000, doi: 10.3233/IDA-2000-43-413.

[27] K. Potdar, T. S. Pardawala, and C. D. Pai, "A Comparative Study of Categorical Variable Encoding Techniques for Neural Network Classifiers," *International Journal of Computer Applications*, vol. 175, no. 4, pp. 7–9, Oct. 2017.

[28] "1st place with representation learning | Kaggle," 2017. https://www.kaggle.com/c/porto-seguro-safe-driver-prediction/discussion/44629 (accessed May 07, 2019).

[29] “Preparing continuous features for neural networks with GaussRank - FastML,” 2018. http://fastml.com/preparing-continuous-features-for-neural-networks-with-rankgauss/ (accessed Nov. 22, 2019).

[30] B. Ratner, *Statistical and Machine-Learning Data Mining:: Techniques for Better Predictive Modeling and Analysis of Big Data*. Chapman and Hall/CRC, 2017.

[31] K. Ramasubramanian and A. Singh, “Feature Engineering,” in *Machine Learning Using R*, K. Ramasubramanian and A. Singh, Eds. Berkeley, CA: Apress, 2017, pp. 181–217.

[32] “3.1. Cross-validation: evaluating estimator performance — scikit-learn 0.20.3 documentation.” https://scikit-learn.org/stable/modules/cross_validation.html (accessed May 03, 2019).

[33] “sklearn.model_selection.RandomizedSearchCV — scikit-learn 0.20.3 documentation.”https://scikitlearn.org/stable/modules/generated/sklearn.model_selection.RandomizedSearchCV.html (accessed May 03, 2019).

[34] “sklearn.model_selection.GridSearchCV — scikit-learn 0.20.3 documentation.” https://scikitlearn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html (accessed May 03, 2019).

[35] M. Kuhn and K. Johnson, “Regression Trees and Rule-Based Models,” in *Applied Predictive Modeling*, M. Kuhn and K. Johnson, Eds. New York, NY: Springer New York, 2013, pp. 173–220.

[36] T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” in *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA, 2016, pp. 785–794, doi: 10.1145/2939672.2939785.

[37] “sklearn.svm.SVR — scikit-learn 0.20.3 documentation.” https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVR.html (accessed May 05, 2019).

[38] B. Kitchenham, “Procedures for Performing Systematic Reviews,” p. 33, Jul. 2004.

[39] P. Brereton, B. A. Kitchenham, D. Budgen, M. Turner, and M. Khalil, “Lessons from applying the systematic literature review process within the software engineering domain,” *Journal of Systems and Software*, vol. 80, no. 4, pp. 571–583, Apr. 2007, doi: 10.1016/j.jss.2006.07.009.

[40] V. Garousi, M. Felderer, and M. V. Mäntylä, “Guidelines for including grey literature and conducting multivocal literature reviews in software engineering,” *arXiv:1707.02553 [cs]*, Jul. 2017, Accessed: Jul. 28, 2019. [Online]. Available: http://arxiv.org/abs/1707.02553.

[41] F. Jiménez, G. Sánchez, J. M. García, G. Sciavicco, and L. Miralles, “Multi-objective evolutionary feature selection for online sales forecasting,” *Neurocomputing*, vol. 234, pp. 75–92, Apr. 2017, doi: 10.1016/j.neucom.2016.12.045.

[42] C.-J. Lu, “Sales forecasting of computer products based on variable selection scheme and support vector regression,” *Neurocomputing*, vol. 128, pp. 491–499, Mar. 2014, doi: 10.1016/j.neucom.2013.08.012.

[43] Y. Liu, Y. Yin, J. Gao, and C. Tan, “Wrapper Feature Selection Optimized SVM Model for Demand Forecasting,” in *2008 The 9th International Conference for Young Computer Scientists*, Nov. 2008, pp. 953–958, doi: 10.1109/ICYCS.2008.151.

[44] H. Lee, S. G. Kim, H. Park, and P. Kang, “Pre-launch new product demand forecasting using the Bass model: A statistical and machine learning-based approach,” *Technological Forecasting and Social Change*, vol. 86, pp. 49–64, Jul. 2014, doi: 10.1016/j.techfore.2013.08.020.

[45] P. A. Castillo *et al.*, “Applying computational intelligence methods for predicting the sales of newly published books in a real editorial business management environment,” *Knowledge-Based Systems*, vol. 115, pp. 133–151, Jan. 2017, doi: 10.1016/j.knosys.2016.10.019.

[46] C.-J. Lu and L.-J. Kao, “A clustering-based sales forecasting scheme by using extreme learning machine and ensembling linkage methods with applications to computer server,” *Engineering Applications of Artificial Intelligence*, vol. 55, pp. 231–238, Oct. 2016, doi: 10.1016/j.engappai.2016.06.015.

[47] C.-J. Lu and Y.-W. Wang, “Combining independent component analysis and growing hierarchical self-organizing maps with support vector regression in product demand forecasting,” *International Journal of Production Economics*, vol. 128, no. 2, pp. 603–613, Dec. 2010, doi: 10.1016/j.ijpe.2010.07.004.

[48] S. Thomassey and M. Happiette, “A neural clustering and classification system for sales forecasting of new apparel items,” *Applied Soft Computing*, vol. 7, no. 4, pp. 1177–1187, Aug. 2007, doi: 10.1016/j.asoc.2006.01.005.

[49] A. Martínez, C. Schmuck, S. Pereverzyev, C. Pirker, and M. Haltmeier, “A machine learning framework for customer purchase prediction in the non-contractual setting,” *European Journal of Operational Research*, May 2018, doi: 10.1016/j.ejor.2018.04.034.

[50] A. Fallah Tehrani and D. Ahrens, “Enhanced predictive models for purchasing in the fashion field by using kernel machine regression equipped with ordinal logistic regression,” *Journal of Retailing and Consumer Services*, vol. 32, pp. 131–138, Sep. 2016, doi: 10.1016/j.jretconser.2016.05.008.

[51] F. Badorf and K. Hoberg, “The impact of daily weather on retail sales: An empirical study in brick-and-mortar stores,” *Journal of Retailing and Consumer Services*, vol. 52, p. 101921, Jan. 2020, doi: 10.1016/j.jretconser.2019.101921.

[52] L. M. Bouwer, “Projections of Future Extreme Weather Losses Under Changes in Climate and Exposure,” *Risk Analysis*, vol. 33, no. 5, pp. 915–930, 2013, doi: 10.1111/j.1539-6924.2012.01880.x.

[53] S. Dunne and B. Ghosh, “Weather Adaptive Traffic Prediction Using Neurowavelet Models,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 1, pp. 370–379, Mar. 2013, doi: 10.1109/TITS.2012.2225049.

[54] A. Koesdwiady, R. Soua, and F. Karray, “Improving Traffic Flow Prediction With Weather Information in Connected Cars: A Deep Learning Approach,” *IEEE Transactions on Vehicular Technology*, vol. 65, no. 12, pp. 9508–9517, Dec. 2016, doi: 10.1109/TVT.2016.2585575.

[55] Yue Liu, Jianguo Zhao, and Junjun Gao, “Weather sensitive demand forecasting method based on SVR for shoes products,” in *2014 6th International Conference*

on Knowledge and Smart Technology (KST), Jan. 2014, pp. 29–34, doi: 10.1109/KST.2014.6775389.

[56] G. Verstraete, E.-H. Aghezzaf, and B. Desmet, "A data-driven framework for predicting weather impact on high-volume low-margin retail products," *Journal of Retailing and Consumer Services*, vol. 48, pp. 169–177, May 2019, doi: 10.1016/j.jretconser.2019.02.019.

[57] J. Wang, G. Q. Liu, and L. Liu, "A Selection of Advanced Technologies for Demand Forecasting in the Retail Industry," in *2019 IEEE 4th International Conference on Big Data Analytics (ICBDA)*, Mar. 2019, pp. 317–320, doi: 10.1109/ICBDA.2019.8713196.

[58] J. A. Guajardo, R. Weber, and J. Miranda, "A model updating strategy for predicting time series with seasonal patterns," *Applied Soft Computing*, vol. 10, no. 1, pp. 276–283, Jan. 2010, doi: 10.1016/j.asoc.2009.07.005.

[59] Ö. Gür Ali and K. Yaman, "Selecting rows and columns for training support vector regression models with large retail datasets," *European Journal of Operational Research*, vol. 226, no. 3, pp. 471–480, May 2013, doi: 10.1016/j.ejor.2012.11.013.

[60] A. Kumar, R. Shankar, and N. R. Aljohani, "A big data driven framework for demand-driven forecasting with effects of marketing-mix variables," *Industrial Marketing Management*, Jun. 2019, doi: 10.1016/j.indmarman.2019.05.003.

[61] A. Krishna, A. V, A. Aich, and C. Hegde, "Sales-forecasting of Retail Stores using Machine Learning Techniques," in *2018 3rd International Conference on Computational Systems and Information Technology for Sustainable Solutions (CSITSS)*, Dec. 2018, pp. 160–166, doi: 10.1109/CSITSS.2018.8768765.

[62] P.-C. Chang and C.-Y. Lai, "A hybrid system combining self-organizing maps with case-based reasoning in wholesaler's new-release book forecasting," *Expert Systems with Applications*, vol. 29, no. 1, pp. 183–192, Jul. 2005, doi: 10.1016/j.eswa.2005.01.018.

[63] M. Gurnani, Y. Korke, P. Shah, S. Udmale, V. Sambhe, and S. Bhirud, "Forecasting of sales by using fusion of machine learning techniques," in *2017*

International Conference on Data Management, Analytics and Innovation (ICDMAI), Feb. 2017, pp. 93–101, doi: 10.1109/ICDMAI.2017.8073492.

[64] S. Li, J. Wang, and B. Liu, "Prediction of Market Demand Based on AdaBoost_BP Neural Network," in *2013 International Conference on Computer Sciences and Applications*, Dec. 2013, pp. 305–308, doi: 10.1109/CSA.2013.77.

[65] N. Xue, I. Triguero, G. P. Figueredo, and D. Landa-Silva, "Evolving Deep CNN-LSTMs for Inventory Time Series Prediction," in *2019 IEEE Congress on Evolutionary Computation (CEC)*, Jun. 2019, pp. 1517–1524, doi: 10.1109/CEC.2019.8789957.

[66] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design science in information systems research," *MIS Q.*, vol. 28, no. 1, pp. 75–105, Mar. 2004.

[67] A. R. Hevner, "A Three Cycle View of Design Science Research," *Scandinavian Journal of Information Systems*, pp. 87–92, 2007.

[68] "Key Consumer Behavior Thresholds Identified as the Coronavirus Outbreak Evolves," 2020. https://www.nielsen.com/us/en/insights/article/2020/key-consumer-behavior-thresholds-identified-as-the-coronavirus-outbreak-evolves (accessed May 08, 2020).

[69] M. Halford, "Target encoding done the right way - Max Halford," 2018. https://maxhalford.github.io/blog/target-encoding/ (accessed May 08, 2020).

[70] N. R. Draper and H. Smith, "On Worthwhile Regressions, Big F's, and R2," in *Applied Regression Analysis*, John Wiley & Sons, Ltd, 2014, pp. 243–250.

Appendix A
Source Codes

Following is the source code for the parallelized Random search XGBoost using Pandas User defined function in Apache Spark.

```
import findspark
from pyspark import SparkContext, SparkConf
from pyspark.sql import SparkSession
from pyspark.sql import functions as F
import random
from pyspark.sql.types import *
from sklearn.model_selection import train_test_split
findspark.init()

conf = SparkConf().setAppName('mainSpark').setMaster('local[36]')
sc = SparkContext(conf=conf)
spark = SparkSession.builder \
    .config(conf=SparkConf()) \
    .getOrCreate()


df = spark.read.format("CSV").option("inferSchema", True).option("header", True).load('LatestNormalizedEncoded.csv')
replicate_df = spark.createDataFrame(pd.DataFrame(list(range(1, 100)), columns=['replicate_id']))
replicate_train_df = df.crossJoin(replicate_df)
# Declare the output schema
outSchema = StructType([StructField('replicate_id', IntegerType(), True), StructField('rmse', DoubleType(), True),
                        StructField('r2', DoubleType(), True), StructField('adjusted_r_squared', DoubleType(), True),
                        StructField('colsample_bytree', FloatType(), True), StructField('gamma', FloatType(), True),
                        StructField('min_child_weight', IntegerType(), True), StructField('learning_rate', FloatType(), True),
                        StructField('max_depth', IntegerType(), True), StructField('n_estimators', IntegerType(), True),
                        StructField('reg_alpha', FloatType(), True), StructField('reg_lambda', DoubleType(), True),
                        StructField('subsample', FloatType(), True)])


# decorating the function
@F.pandas_udf(outSchema, F.PandasUDFType.GROUPED_MAP)
def run_model(pdf):

    replication_id = pdf.replicate_id.values[0]
    colsample_bytree = random.choice([0.5, 1])   # Typical values: 0.5-1
    gamma = random.choice([0.03, 0.05, 0.07])
```

```
    min_child_weight = random.choice([4, 8, 10, 12])  # Used to control over-fitting. Higher values prevent a model from
    # learning relations which might be highly specific to the particular sample selected for a tree.
    learning_rate = random.choice([0.5, 0.05, 0.07, 0.09])  # Lower values are generally preferred as they make the
    # model robust to the specific characteristics of tree and thus allowing it to generalize well.
    max_depth = random.choice([3, 5, 7, 9])  # Used to control over-fitting as higher depth
    # will allow model to learn
    n_estimators = random.choice([50, 100, 200, 1000])  # Checked up to 10000 no change on Accuracy
    reg_alpha = random.choice([0.5, 0.75, 1])
    reg_lambda = random.choice([1e-5])
    subsample = random.choice([0.6, 0.8])  # Values slightly less than 1 make the model robust by reducing the variance.
    # Typical values ~0.8 generally work fine but can be fine-tuned further.
    X = pdf[['StoreName', 'kommuneNo', 'CountyNo', 'WeekDay', 'ProductID', 'ProductUnitPrice', 'Day',
             'Month', 'Year', 'DayOfYear', 'WeekOfYear', 'QuarterOfYear', 'WeekEnd', 'WeekStart',
             'PostNo', 'AvgTurnover', 'ShopProfileName', 'Profile',
             'expanding_mean', 'lag_1']]
    y = pdf['TotalSold']
    Xtrain,Xcv,ytrain,ycv = train_test_split(X, y, test_size=0.33, random_state=42, shuffle=False)
    xgb_model = xgboost.XGBRegressor(learning_rate=learning_rate, n_estimators=n_estimators, max_depth=max_depth,
                                     min_child_weight=min_child_weight, gamma=gamma, subsample=subsample,
                                     colsample_bytree=colsample_bytree, reg_alpha=reg_alpha, reg_lambda=reg_lambda, n_jobs=48)
    xgb_model.fit(Xtrain, ytrain)
    prediction = xgb_model.predict(Xcv)
    rmse = mean_squared_error(ycv, prediction, squared=False)
    r2 = r2_score(ycv, prediction)
    adjusted_r_squared = 1 - (1 - r2) * (len(ycv) - 1) / (len(ycv) - Xcv.shape[1] - 1)
    result = pd.DataFrame({'replicate_id': replication_id, 'rmse': rmse, 'r2': r2, 'adjusted_r_squared': adjusted_r_squared,
                           'colsample_bytree': colsample_bytree, 'gamma': gamma, 'min_child_weight': min_child_weight,
                           'learning_rate': learning_rate, 'max_depth': max_depth, 'n_estimators': n_estimators,
                           'reg_alpha': reg_alpha, 'reg_lambda': reg_lambda, 'subsample': subsample}, index=[0])
    return result


results = replicate_train_df.groupby("replicate_id").apply(run_model)
results.sort(F.desc("adjusted_r_squared")).coalesce(1).write.format('com.databricks.spark.csv')\
        .save("Result.csv", header='true')
```