Lappeenranta-Lahti University of Technology
School of Engineering Science

Joel Lahtinen

Clustering and classification of material suppliers using machine learning algorithms

Master's thesis

Examiners: Professor Pasi Luukka and post-doctoral researcher Christoph Lohrmann

ABSTRACT

Author: Joel Lahtinen

Title: Clustering and classification of material suppliers using machine learning algorithms

Year: 2021 **Place:** Helsinki, Finland

Master's thesis. Lappeenranta-Lahti University of Technology, Industrial Engineering and Management.

92 pages, 28 figures and 9 tables.

Examiners: Professor Pasi Luukka and post-doctoral researcher Christoph Lohrmann

Keywords: Data analysis process, feature selection, supplier clustering, supplier classification, classification evaluation, supplier relationship management

The purpose of this thesis is to serve as a feasibility study to investigate the usage of machine learning algorithms to enhance daily operations of procurement function. It seeks to find suitable variables from the given supplier dataset, clustering the suppliers into coherent categories, understand characteristics of each category and choosing a better classifier from two options for supplier classification. The thesis presents a supplier relationship management process and framework where the machine learning algorithms are used to create efficiencies in sourcing operations towards different suppliers. The studied dataset composed of technical material supplier data of a Finnish listed company.

The algorithms used in feature selection phase are Principal component analysis (PCA) and Pearson's correlation coefficient (PCC). Algorithms for cluster quantity determination are the elbow method, Calinski-Harabaz index and Silhouette width. Used machine learning algorithm for clustering is K-means algorithm and classifier algorithms compared are Artificial neural network (ANN) and Random forest (RF). The evaluation of classifier algorithms was done by comparing firstly the accuracy and secondly the true positive rate (TPR) of classification performance.

The research results indicate that given supplier data has some redundancy which were identified and removed in feature selection phase. The optimal and practical quantity of clusters for given data was five clusters with distinctive characteristics to be used. As comparing ANN and RF with different hyperparameter settings as classifiers for supplier data, the RF was found to be the more accurate and more suitable for the purpose. The thesis gave indication that supplier data could be clustered in meaningful categories and a classifier can classify clustering data with over 95 % accuracy into those pre-defined categories.

TIIVISTELMÄ

Tekijä: Joel Lahtinen

Työn nimi: Materiaalitoimittajien klusterointi ja klassifiointi koneoppimismenetelmiä käyttäen

Vuosi: 2021 **Paikka:** Helsinki, Finland

Diplomityö. Lappeenrannan-Lahden teknillinen yliopisto, tuotantotalouden koulutusohjelma.

92 sivua, 28 kuvaa and 9 taulukkoa.

Tarkastajat: Professori Pasi Luukka ja tutkijatohtori Christoph Lohrmann

Hakusanat: Data-analyysiprosessi, piirteenvalinta, toimittajien klusterointi, toimittajien klassifiointi, ohjatun koneoppimisen arviointi, toimittajasuhdehallinta

Tämän työ toimi tutkimuksena, jossa tutkittiin koneoppimisalgoritmien soveltuvuutta hankintafunktion päivittäisen toiminnan tehostamiseen. Työssä etsittiin soveltuvia muuttujia toimeksiantajalta kerätystä toimittajadatasta, klusteroitiin toimittajat yhtenäisiin kategorioihin, selvitettiin kategorioiden ominaispiirteet ja valittiin parempi luokittelualgoritmi kahdesta vaihtoehdosta toimittajien luokitteluun. Lisäksi työ esitteli toimittajasuhdehallinnan prosessin sekä viitekehyksen, jossa koneoppimisalgoritmeja hyödynnetään tehokkuuden lisäämisessä hankintaoperaatioissa eri toimittajien kanssa. Tutkittu data koostui teknisten materiaalien toimittajadatasta, joka oli saatu suomalaiselta pörssiyhtiöltä.

Työssä piirteenvalinnassa käytettiin pääkomponenttianalyysia sekä Pearsonin korrelaatiokerrointa. Klusterien määrän määrittämiseen käytettiin nk. kyynärpäämetodia, Calinski-Harabaz indeksiä sekä Siluettipisteytystä. Käytetyt koneoppimisalgoritmit olivat klusteroinnissa K:n keskiarvon klusterointimenetelmä ja luokittelussa Neuroverkko sekä Satunnainen metsä. Luokittelualgoritmeja arvioitiin niiden luokittelutarkkuuden sekä sensitiivisyyden perusteella.

Tutkimus osoitti, että tutkitussa datassa on tarpeettomia muuttujia, jotka tunnistettiin ja poistettiin piirteenvalinnan aikana. Optimaaliseksi kategorioiden määräksi valikoitui viisi kategoriaa, joista jokaisella oli selvästi yksilölliset ominaispiirteet. Neuroverkkoa sekä satunnaismetsää tarkasteltaessa eri asetuksilla, työssä osoitettiin, että satunnaismetsä oli tarkempi ja sensitiivisempi toimittajien luokittelussa. Tutkimus osoitti, että koneoppimismenetelmillä toimittajadata oli klusteroitavissa käytettäviin kategorioihin sekä luokiteltavissa yli 95 % tarkkuudella.

ACKNOWLEDGEMENTS

With this master's thesis my time as a student finally comes to its end. The journey as a student for me was longer than expected but having completed the mandatory military service and been fully at work for the past two years have given perspective on the things. I've had great time at LUT and gained extremely important friends, teachers, experiences, knowledge and contacts to shine and keep growing beyond university. Thanks especially to my friends from LUT and before for making the journey eventful and always, always supporting when needed. Someone said that the destination isn't the purpose – it's the journey itself which I can relate fully now completing the journey and seeing all the great memories and times when I've had the opportunity learn and grow.

Regarding the thesis, I would like to thank Neste Engineering Solutions Oy and Neste Oyj for giving me the opportunity to investigate interesting topic and I would like to thank all my ex-colleagues there for knowledge towards the thesis. Thanks, Beatrice Bilcu, for your wise words and comments in the beginning and especially Maiju Siekkinen who made all this possible, always trusted in me and let me take responsibilities in professional life. Furthermore, I would like to thank Pasi Luukka and Christoph Lohrmann for giving guidance during the project and providing me with in-depth comments helping me improve the thesis.

I wouldn't be here without my family to which I owe the biggest thanks. Thank you, mom, dad, Nuutti and Camilla. You've all always supported me during ups and downs. Thanks dad and Nuutti for being the backbone always to trust. Thanks mom, you've been the force pushing me forward academically and on other fields always (let's see with that PhD…). Finally, thank you, my fiancé Camilla for being there for me and sharing life together and coping with me from day to day always supporting. This is for you and our newborn baby boy.

Helsinki, 2.5.2021

Joel Lahtinen

TABLE OF CONTENTS

1 Introduction 1
1.1 Background 1
1.2 Research objectives and scope 2
1.3 Structure of the thesis 4
2 Supplier relationship management 5
3 Theoretical framework for machine learning 11
3.1 Data preprocessing 13
3.2 Dimensionality reduction & feature selection 17
3.2.1 Principal component analysis 19
3.2.2 Pearson correlation coefficient 22
3.3 Unsupervised learning 23
3.3.1 K-means clustering 27
3.3.2 Defining the correct quantity of clusters 28
3.4 Supervised learning 31
3.4.1 Artificial neural networks 34
3.4.2 Random forests 38
3.4.3 Evaluating supervised algorithm performance 40
4 Predictive supplier empirical analysis 43
4.1 Data collection 44
4.2 Data preprocessing 46
4.3 Feature selection 48
4.4 Data modeling 59
4.4.1 K-means clustering results 59
4.4.2 Classification results 68
5 Summary and conclusions 74
5.1 Answering the research questions 75
5.2 Limitations of the models and implementation 78
5.3 Future research opportunities 79
References 81
Appendices 92

FIGURES

Figure 1. Paradigm of the thesis. 4
Figure 2. Supplier relationship management process, modified from SRM framework of Park et al. (2010, p. 499). 5
Figure 3. Traditional supplier categorization model with general principles of supplier characteristics (Kraljic, 1983, p. 112-115). 6
Figure 4. Supplier assessment framework and approach in two-dimensional graph, extended from Park et al. (2010, p. 505) and Schuh et al. (2014, p. 30). 9
Figure 5. Taxonomy of machine learning categories (Badillo et al., 2020, p.872). 12
Figure 6. Cluster analysis methods classification. 24
Figure 7. An example of hierarchical clustering result as dendrogram. 25
Figure 8. An example of density-based clustering method based on R-package “factoextra” multishapes dataset (Kassambra, 2020, p. 80). 26
Figure 9. An example and idea of 4-fold cross-validation. 34
Figure 10. A presentation of neural network with three inputs, one hidden layer with four neurons and three outputs. 36
Figure 11. Illustration of operations performed in single neuron, adapted from Russell & Norvig (2010, p. 728). 37
Figure 12. Example of simple binary decision tree structure for defining whether stove is hot or not. 39
Figure 13. Illustration of a confusion matrix. 41
Figure 14. The process conducting the analysis adapting the steps of Garciá et al. (2015, p.1-2). 43
Figure 15. Scree plot of the original dataset. 50
Figure 16. Contribution of variables in the three principal components in the original dataset. 51
Figure 17. Variable contribution towards two main principal components and correlation directions between variables of the original dataset. 52
Figure 18. Pearson correlation coefficient matrix for the original dataset and its variables. 53
Figure 19. Scree plot of the reduced dataset. 55
Figure 20. Contribution of variables in the three primary dimensions in the reduced dataset. 56
Figure 21. Variable contribution towards two main principal components and correlation directions between variables of the reduced dataset. 57
Figure 22. Pearson correlation coefficient matrix for the reduced dataset and its variables. 58

Figure 23. Cluster number determination methods and their results. 61
Figure 24. Clustering result according to first two principal components with 5 clusters. 63
Figure 25. Total spend, number of orders, quantity of meters and purchase time window for clusters three and four. 64
Figure 26. Unit price mean, supplier lead time, total spend and number of orders of all clusters. 65
Figure 27. Number of orders and purchase time window and total spend, and unit price mean for clusters one and five. 66
Figure 28. Example of one tree in the best performing tuned Random forest algorithm. 74

TABLES

Table 1. Research questions and objectives. 3
Table 2. Names of preprocessing areas, goal of their activities and example actions as interpret by Garciá et al. (2015, p. 11-13). 14
Table 3. Example interpretations from Pearson correlation coefficient values adapting Schober et al. (2018, p. 1765). 23
Table 4. Qualification requirements for purchase order datapoints. 45
Table 5. Cluster centers of each variable of each group and number of suppliers per group. 62
Table 6. Linguistic descriptions of variables on each cluster. 68
Table 7. The results of the neural network classifier in the terms of average accuracy and TPR. 71
Table 8. The results of the random forest classifier in the terms of average accuracy and TPR. 72
Table 9. The results of tuned models trained with the best performing hyperparameters with test data set. 73

ABBREVIATIONS

AI	Artificial intelligence
ANN	Artificial neural network
BSS	Between sum of squares
CART	Classification and regression tree
CH	Calinski-Harabasz
ERM	Enterprise resource management
ERP	Enterprise resource planning
FN	False negative
FP	False positive
MAR	Missing at random
MCAR	Missing completely at random
ML	Machine learning
MLP	Multilayer perceptron
MSE	Mean squared error
NES	Neste Engineering Solutions Oy
MNAR	Missing not at random
OOB	Out-of-bag
PCA	Principal component analysis
PCC	Pearson correlation coefficient
PPV	Positive predictive value
RF	Random forest
RMSE	Root mean squared error
SRM	Supplier relationship management
TN	True negative
TP	True positive
TPR	True positive rate
TSS	Total sum of squares
WSS	Within sum of squares

1 INTRODUCTION

The first chapter introduction contains the background to the study, the key trends underlining the matter studied and motivation for the research problems which are sought to be solved. The objectives and scope of the study are defined and set. The structure of the thesis is introduced before diving into the core research.

1.1 Background

Advances in digital technologies has led into critical transform in businesses and underlining digitalization as a strategic priority in order to maintain company's position and market share (Legner et al., 2017, p. 301-302). Digitalization is one of the key elements in organizations and functions to support profitability, effectiveness and especially efficiency in long run perspective (Bienhaus & Haddud, 2017, p. 966). This change has forced companies to address their digital capabilities in order not to fall behind of the competition and maybe even drive more efficient value creation. It has led to investing more resources and money into data and its effective usage in terms of availability, analysis and decision support (Ritter & Pedersen, 2020, p. 181-182). One of the key elements enabling better usage and analysis of data are advanced analytics applications, such as *Artificial Intelligence* (AI) and *Machine Learning* (ML) concepts, which have gained popularity on enabling better data driven decision making by creating reliable decision support based on data, rather than the human interpretations of the data.

Neste Oyj, a Finnish renewable energy company, has also taken a step into this journey by setting one its three strategic priorities to "Drive efficiency in operations" where advanced analytics is one of its dimensions (Neste, 2021a). As Nguyen et al. (2018, p. 254-255) state, the usage of advanced analytics enables to excel in operations and drive efficiency from the data. Neste has been ramping up its capabilities to extract and use the data in its operative units but now the interest has been coming more and more also from the supportive units such as procurement. Procurement is a part of key operational excellence chain at Neste where it guards the safety, quality, compliance and efficiency of Neste's suppliers towards Neste and is committed operating with integrity towards the suppliers (Neste, 2021b). Guarding Neste's reputation and treating suppliers with integrity requires decision making based on objective factors and variables, where data and analysis of it is essential part of measuring the compliance, effectivity and importance of a supplier towards Neste.

Neste Engineering Solutions Oy (NES) is a subsidiary of Neste specialized to be a “technology, engineering and project management partner of choice” in Neste’s business area where it executes the complex investment projects and serves as technology R&D unit (Neste, 2021c). NES procurement function had set its targets to enable data driven decision making in its core operations utilizing as much objective knowledge as possible in decision situations. Supplier lifecycle management is an essential part of guarding the owner’s reputation and integrity of suppliers, and an initiative was risen to utilize advanced analytics applications to enhance the decisions made during the process. Bringing an objective point of view into different kinds of supplier groups to be utilized, and automatically finding a group for a new or existing supplier with changing behavior was seen a target to potentially enable enhancements and efficiencies inside this process.

1.2 Research objectives and scope

The first and foremost reason for this thesis is to serve the Neste Engineering Solutions Oy procurement function as a feasibility study in order to understand how advanced analytics applications could help decision making process in their own environment. The aim of the thesis is to enhance supplier relationship management process with analyzing NES existing internal order data of the suppliers with data analytics and machine learning solutions. Objectives of the thesis in more detail are to categorize technical material suppliers in coherent clusters and then evaluating two different machine learning algorithms to find out which one is more effective to classify suppliers with the data available assigning a new supplier into right cluster or re-evaluate the cluster of existing supplier. The findings of the study are used for decision making support for sourcing managers and sourcing specialists in their work maintaining relationships and contracts with suppliers.

Three research objectives were defined and formed to accomplish the purpose of the thesis and they are presented in **table 1** with corresponsive objectives. The *first* research question intends to clarify the needed variables to further phases of the study and to identify possible redundant variables. The *second* research question aims to assign every supplier to its respective cluster where the clusters would be as similar inside as possible and samples outside its own cluster would be as different as possible. In addition, the *second* research question should give the insight of the feature characteristics making that cluster unique. The *third* research question compares chosen classifying algorithms and evaluates their performance to find the most effective one to be used in classifying this type of data.

Table 1. Research questions and objectives.

Research questions	Objectives
1. What are the most important features of internal data available in grouping suppliers?	• Identify features which enable the evaluation of different suppliers with machine learning algorithms
2. What is the suitable quantity of clusters for the data and what are the key features of each cluster?	• Find out proper amount of clusters and describe the relevant factors of the cluster
3. Which of the algorithms is the best for classifying suppliers and should be used?	• Evaluate chosen classifying algorithms by their performance and suggest the best one to use

The scope of the study is very specific. Because of the nature of Neste and NES procurement integration the two main categories which NES procures are technical materials and technical services. This study concentrates on the former because there are more data of different variables available and the data is in easily accessible form such as databases and direct user interface extracts. Also, because the technical material supplier data is mainly available in a structured form, the results are straightforward to implement and there's a potential to quick value realization. Suppliers with just one purchase order are left out of scope because there's not enough information to evaluate them.

When looking into supplier relationship management process the thesis strictly focuses on describing the process steps and what is expected in each one of them and where machine learning algorithms bring value. The thesis doesn't take in account how the different steps should be performed or how the process itself creates value in a broader picture. Also, defining clear variables for comprehensive supplier relationship management process is out of scope since the process must be conducted with data available seek out potential utilization and value creation for requesting organization.

From NES point of view one critical material supplier is left out of the analysis which is the NES internal material supplier. The reason to leave that supplier out of scope is that their operating logic and way to create value to NES is totally different from external material suppliers. In addition, actions and procedures to manage with internal supplier differs significantly from external suppliers.

The timeframe of the research scope is adjusted between 1.1.2016 and 31.5.2019. This signifies that the supplier is included in study if it has material marked as received during this time period in NES systems. The reason why this timeframe was chosen was that there would be enough data to analyze in order to increase the reliability of the study. Chosen timeframe helps also maintaining the data quality with leaving out the time period of system integration in the past where lots of data was migrated manually thus having potentially manual mistakes.

1.3 Structure of the thesis

The thesis builds on five chapters. Each of the chapter has a brief introduction in the beginning. Chapter one presents the motivation behind the thesis along with the background, research objectives and structure of the thesis. The second chapter focuses on explaining a business process where machine learning algorithms can be applied. The third chapter presents the academic research and mathematical implications for the machine learning application process. In chapter four the empirical research is conducted, and its results are evaluated. The final chapter, number five, sums up the content of the thesis, draws conclusions for the research questions, explains the limitations of the research and its implementation potential and finally suggests future research possibilities. In **figure 1** the paradigm of the thesis is illustrated.

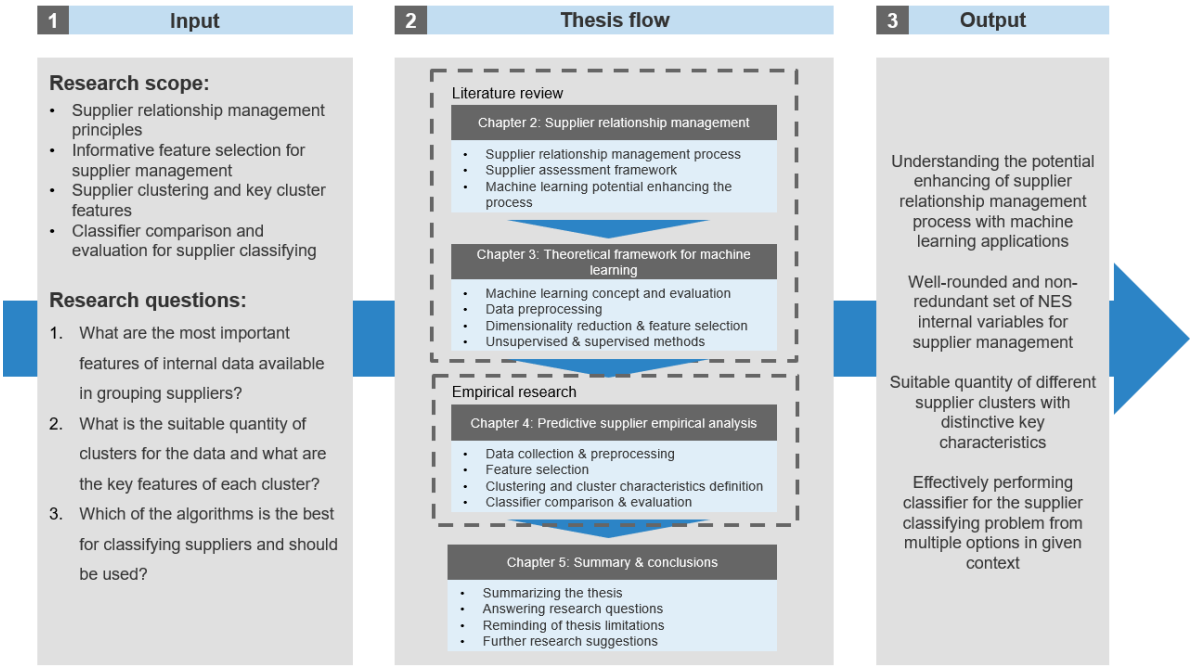

Figure 1. Paradigm of the thesis.

2 SUPPLIER RELATIONSHIP MANAGEMENT

In this chapter the theoretical principles of a supplier relationship management (SRM) concept are presented and importance of the ability to efficiently cluster and classify the suppliers into categories is explained. Focus of the chapter lies in presenting the different steps of SRM process and how machine learning supported clustering and classification affects the different steps of the process generating advantages. This chapter lays basis on the business value creation via machine learning applications in the NES organization.

Supplier relationship management can be seen as a process where the organization operating with suppliers needs to differentiate suppliers into different categories, have a concept of collaboration or partnering with certain suppliers in their network, assess the different suppliers based on their performance and continuously track this process (Schuh et al., 2014, p. 8). A mature SRM process enables the organization to create more value by being able to produce its own goods or service more efficiently (Park et al., 2010, p. 496). Park et al. (2010, p. 499) propose the SRM process to be conducted in five different steps being a continuous (looping) process over the time, like illustrated in **figure 2**, and the steps are:

1) Defining the purchasing strategy and criteria
2) Supplier selection
3) Supplier collaboration
4) Supplier assessment
5) Continuous development.

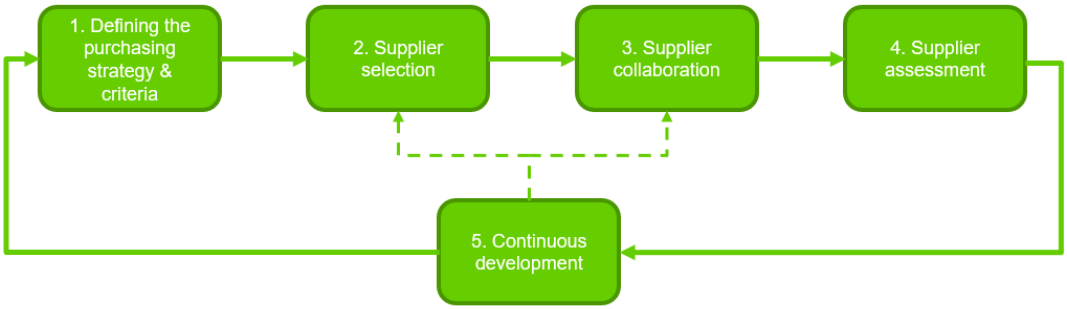

Figure 2. Supplier relationship management process, modified from SRM framework of Park et al. (2010, p. 499).

Defining the purchasing strategy and criteria is the first step in creating a SRM process and it traditionally has two stages: clustering of suppliers into profit impact and supply risk matrix based on different factors and developing action plans for chosen suppliers (Kraljic, 1983, p. 112-115). The purpose of this first step is to enable the purchasing organization to think strategically and leverage data in decision making (Krause et al., 2009, p. 19). In the first stage of the first step the different variables affecting supplier clustering are identified and suppliers are clustered (Kraljic, 1983, p. 112). Profit impact variables consist of factors like competence of the organization to procure the item at hand, economic factors like total spend of purchases, value add of the procured item and volume of purchases and image related factors, like co-operating with companies with certain brand or sustainability status (Kraljic, 1983, p. 111; Olsen & Ellram, 1997, p. 104-106). Sustainability as part of profit impact has become more and more important in the brand of many companies and the procuring organization can only be as sustainable as its suppliers are (Carter & Easton, 2011, p. 47-49). Supply risk variables consist of factors like product complexity and other product characteristics, suppliers' power, market factors and supplier specific risk and capabilities (Kraljic, 1983, p. 111; Olsen & Ellram, 1997, p. 104-106). In traditional classification the supplier pool is divided into four categories created by Kraljic (1983, p. 112) and presented in **figure 3:** Strategic suppliers, Bottleneck suppliers, Leverage suppliers and Noncritical suppliers.

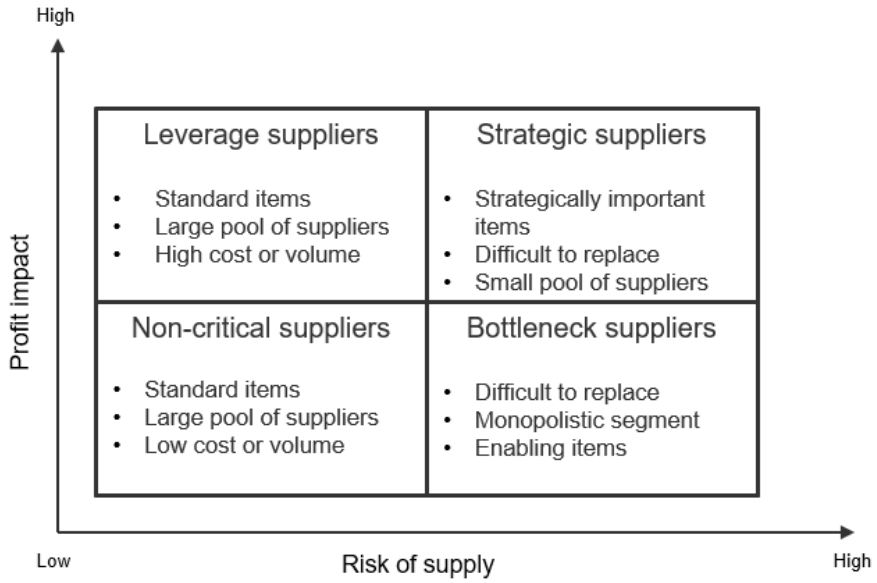

Figure 3. Traditional supplier categorization model with general principles of supplier characteristics (Kraljic, 1983, p. 112-115).

However, this approach takes only a limited amount of available data in account from existing suppliers and is an excellent tool for defining strategical focus of procurement function but isn't enough for SRM process (Park et al., 2010, p. 500). In traditional approach proposed by Kraljic the supplier groups weren't assessed by their own performance but rather with overall market analysis which lead to generic strategic positions based purely on market positions instead of accounting also supplier performance (Kraljic, 1983, p. 113-115). Therefore, the suppliers are analyzed with specified performance criteria to help to understand the different suppliers and their characteristics available. Chakraborty (2015, p. 29) lists as the most significant criteria the delivery performance (delivery time compliance and lead time), vendor replacement possibility, supplier relationship, quality performance (Goods are as in perfect condition and quantity of goods received is correct) and price. Olsen & Ellram (1997, p. 103-107) expand the idea of strict segments based on profit impact & supply risk into portfolio management idea where the supplier relationships are maintained with respect to their portfolio conducting from the profit impact, supply risk, supplier collaboration and supplier performance. To efficiently produce these required new portfolios, an algorithm could be more capable than human to distinct suppliers from each other and find patterns.

The second stage of the first step is to create the action plans for different segments, or now portfolios, where the required activities and target states towards different portfolios are defined (Park et al., 2010, p. 501). The ability to have portfolios which take on account also parts of performance enable more specific and accurate plans for different supplier portfolios (Olsen & Ellram, 1997, p. 108-110). In traditional approach the plans for low-risk material suppliers were commonly the same and some high-risk suppliers might've had their own plans which has led into too general approaches (Park et al., 2010, p. 501). Distinction between "good" and "bad" performance forces procurement as a function into different strategies approaching the suppliers which will lead into more efficient relationships and therefore towards higher value creation in the value chain (Hald & Ellegaard, 2010, p. 890-891).

Supplier selection is the second step of the SRM process where the suppliers are chosen into supplier pool to be used or left out from that pool due high risks, minimal profit impact, poor collaboration or bad performance (Park et al., 2010, p. 502). In supplier selection step the required performance levels of the suppliers for the variables formulated in first step are decided (Zimmer et al., 2015, p. 1413-1414). Supplier selection step might also include weighting of the criteria with respect to the strategy of the procurement organization (Lee et al., 2001, p. 310). With this variable evaluation even some full portfolios of suppliers could be

taken into hold-mode if the required variable performance levels are not met. Efficient supplier selection is always dependent on comprehensive and clear purchasing strategy & criteria where the variables and overall focus of the procurement are created and defined (Lee et al., 2001, p. 310-311).

The 3rd step of the SRM process is supplier collaboration where the chosen strategies are applied to the suppliers and the expectations are communicated both ways around (Park et al., 2010, p. 502). Oh & Rhee (2008, p. 492) classify the different parts of supplier collaboration into five main categories of:

1) Collaborative communication
2) Collaboration in product development
3) Collaborative problem solving
4) Strategic purchasing
5) Supplier development.

Collaborative communication means exchanging sensitive information between suppliers and procuring organization for the mutual benefit, mutual understanding and possible problem prevention. Collaboration in product development enables supplier to be involved in development of the final product to maybe adjust their offering or materials into required definitions beforehand in order to set the quality standards and other requirements correct in early stages of the product creation. Collaborative problem solving refers to activities performed in day-to-day operations between supplier and procuring organization regarding costs, deliveries and qualities of goods. Strategic purchasing in this dimension is related to communication between the supplier and procuring organization about the supplier performance on the chosen criteria and opportunity of procuring organization to change the supplier into more prominent one, if required. Supplier development refers to longer-term tasks agreed mutually between procuring organization and supplier to develop supplier capabilities and operations in co-operation towards more efficient relationship. Supplier collaboration is often measured via surveys targeted for the procuring organization resources who are in interface with the suppliers. Arranging these surveys periodically enables the procuring organization to see how the supplier collaboration evolves with suppliers. Typical questions in such surveys are, for example, the ability to co-operate, problem solving willingness and supplier flexibility in schedule or requirement changes (Oh & Rhee, 2008, p. 492-495, 498-507).

The fourth step of SRM process is the supplier assessment where the supplier is evaluated with its strategic dimensions of profit impact and supply risk, and with its supplier specific qualities of performance and collaboration relationship (Park et al., 2010, p. 505). This evaluation is based on the criteria set up in the first step of the SRM process and each supplier is evaluated over time so their position can change over time from a portfolio to another portfolio. Schuh et al. (2014, p. 30) and Park et al. (2010, p. 505) define the portfolios into which the suppliers are re-evaluated, after the initial clustering and classification made in step 1, based on their strategic- and supplier specific criteria development through the inspection cycle. **Figure 4** presents a framework with sequence and the different possible portfolios of mitigate, improvement, maintenance, collaboration and integrate with their illustrative relationship to all the strategic- and supplier specific criteria truncated into illustrative two-dimensional graph. The portfolios presented are directional, as some of the dimensions might not be relevant for all organizations, as the total quantity and definitions of portfolios are essentially derived from the supplier data and the procurement strategy of the organization at hand (Park et al., 2010, p. 505-507; Schuh et al., 2014, p. 29-34). Classification algorithms play a key role in this process of assigning new suppliers and re-assigning existing suppliers, as they develop continuously, into correct portfolios in technically mature organizations (Hudnurkar et al., 2015, p. 624-626).

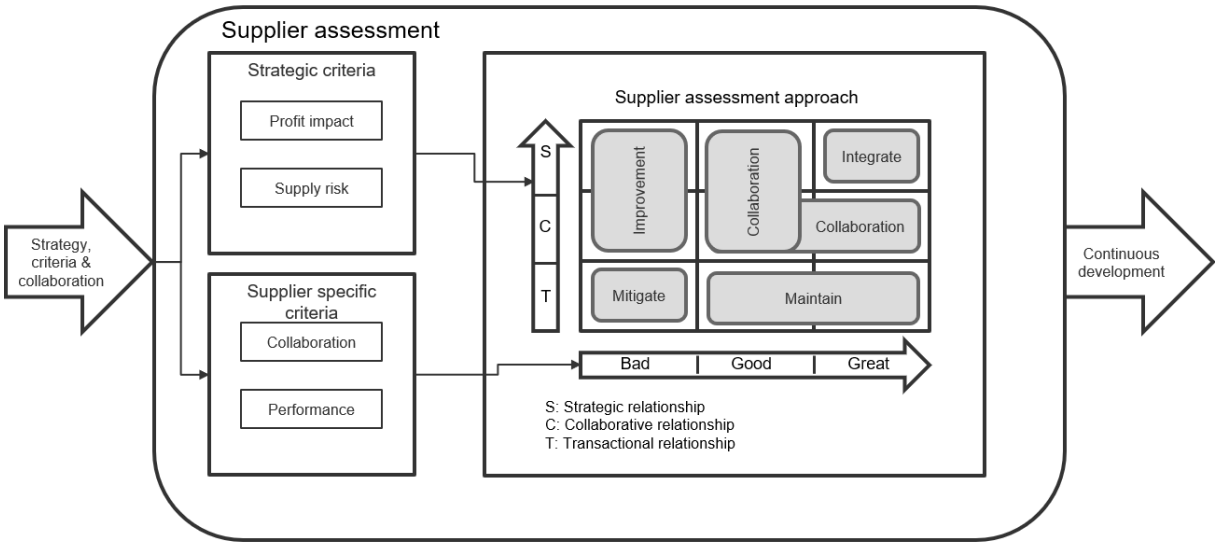

Figure 4. Supplier assessment framework and approach in two-dimensional graph, extended from Park et al. (2010, p. 505) and Schuh et al. (2014, p. 30).

Suppliers in “mitigate” portfolio, with transactional relationship and bad performance, are suppliers whom with the relationship isn’t properly working and the impact probably isn’t huge for procuring organization and the relationship with supplier should be disengaged but with openness and respect for possible future collaboration if the supplier develops (Schuh et al., 2014, p. 38). The opposite end, with strategic relationship and great performance, is the “integrate” portfolio suppliers where the supplier relationship is special and is strategically important and the supplier performs extremely well. Supplier relationships in this portfolio should be nurtured with extra communications and level of co-operation and these suppliers should be considered as key suppliers (Schuh et al., 2014, p. 36). The suppliers in portfolio of “collaboration”, with either strategic relationship and good performance, collaborative relationship and good performance or collaborative relationship and great performance, are the ones where the relationship should be deepened on co-operation and find common goals to develop (Park et al., 2010, p. 506). “Maintain” portfolio suppliers, with transactional relationship and good or great performance, might be the easiest for procuring organization to cope with as with them everything seems to work well and the relationship with them doesn’t require to be deepened as their strategic qualities are not vital for the procuring organization (Park et al., 2010, p. 506). The last portfolio of “improvement” suppliers, with bad performance but either collaborative or strategic relationship, is complex as these suppliers are important in terms of their strategic variables, but their performance or collaboration is poor. Here the keys for the relationship is to intensively engage on development actions and immediate reactions for arising issues to steer the relationship into wanted direction (Schuh et al., 2014, p. 38).

The fifth and the last part of the SRM process is the continuous development of the process and procurement strategy and criteria. To enable this kind of process to work and create value, the procuring organization is required to have suitable systems and tools where they can update the supplier information and data as well as follow their market- and supplier structure related risks (Park et al., 2010, p. 503). If the organization wants to be able to react to changes in their environment quickly, the continuous development is the key area as the supplier pool and content of discussions are derived from the supplier assessment and procurement strategy (Hald & Ellegaard, 2010, p. 888-891). Also, when the suppliers change and the environment changes, the procuring organization needs to review and maybe develop their strategy and criteria as new products might be needed or more data can be gathered from the suppliers (Lee et al., 2001, p. 316-317). As Manupati et al. (2018, p. 236-238) state the efficient classification and categorization algorithms are required for organizations to efficiently classify and categorize the suppliers into coherent portfolios based on data rather than human

interpretations of it. Doing the categorization and classification with data-based algorithm enables the procurement function to move towards data-driven decision making.

Having introduced of what is supplier relationship management and why it is important for procurement function to be able to categorize and classify the suppliers properly into relevant portfolios, the next part will focus on theory on how the technical approach can be achieved. This chapter described the optimal situation of the process and research which might not be fully utilized during the analysis of the data due some datapoints might not be available.

3 THEORETICAL FRAMEWORK FOR MACHINE LEARNING

This chapter introduces the theory on conducting the used analysis of the thesis. It focuses mainly on the process of conducting machine learning study with steps such as data pre-processing, feature selection, data modeling in both supervised and unsupervised methods. In addition, this chapter will define what is machine learning and how to evaluate the results of the supervised machine learning algorithms.

There are many overlapping definitions for machine learning (ML) but ML can be defined as algorithms and statistical models which can learn abstract concepts through examining a set of examples representing a concept (Badillo et al., 2020, p.871; Kubat, 2015, p.1). Learning can be interpreted as ability to improve the performance solving a problem with making observations of the environment and its behavior (Russell & Norvig, 2010, p. 693). An algorithm has learned when it performs its objective more efficiently or accurately than with the previous iteration. Humans learn also with same technique assigning attributes to a concept and in some utilizations machine learning is used for mimicking humanlike decision making in classification problems (Mello & Ponti, 2018, p. 1). For example, a small child would associate a beak, feathers and wings to birds whereas scales, fins and ability to live under the water would be associated with fish, hence having the ability to differentiate and classify these concepts of bird and fish. The other characteristic for machine learning models is that they can learn from the data without being clearly programmed with specific tasks to classify or regress (Liu et al., 2017). This enables the possibility to find relations on variables which were unknown prior applying machine learning algorithm. Discovering new relations on data enables the utilization of this new information to create new applications and enhancing processes.

Machine learning applications are traditionally classified in two different categories: Supervised learning algorithms and unsupervised learning algorithms (Clarke et al., 2009, p.231, 405). The main difference between those two different techniques is that in supervised learning the data is labelled and the goal is to predict the output whereas in unsupervised learning the data is unlabeled and the goal is to explore the data (Joshi, 2010, p. 10). Badillo et al. (2020, p. 872) breaks unsupervised and supervised learning further into two subgroups where unsupervised learning is split into clustering and dimensionality reduction purposes and supervised learning is split into regression and classification purposes as is shown in **figure 5**. Some scholars acknowledge also a third machine learning main category, reinforced learning, where the datapoints are missing labels but the model interacts with the environment when trying to achieve the desired outcome, so it's not totally unsupervised (Joshi, 2020, p.11).

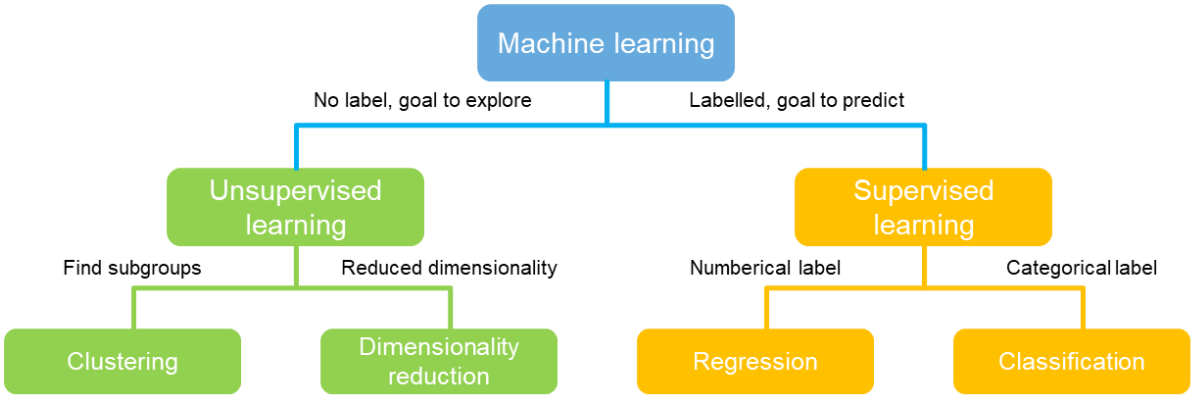

Figure 5. Taxonomy of machine learning categories (Badillo et al., 2020, p.872).

In order to understand machine learning applications, it is vital to understand the main terminology used. The algorithm itself is function (or classifier in supervised learning specialize to classification) which performs the mathematical steps to data e.g. assigning instances into clusters or mapping typical attributes into different labels. Instance is one observation of interest and in datasets observations are presented in each row e.g. supplier X with feature A, B and C. A feature or an attribute is a variable which characterizes the observation e.g. supplier X has a feature A which is size with value "big" and it's presented usually on the dataset as column. Term dimensionality refers to the quantity of different features or variables. A high-dimensional dataset would have hundreds of features as exact amount of "high" isn't defined. (Mello & Ponti, 2018, p. 2-3; Gollapudi, 2016, p. 7).

Machine learning applications are usually a part of larger process like data analysis which consists of steps like data acquisition, data exploration, data preprocessing, dimensionality

reduction, data modeling and result evaluation (Tattar et al., 2017, p. 9-11; Garciá et al., 2015, p. 1-2). Machine learning is almost always conducted during data modeling part and it usually brings the most value from the data analysis process (Garciá et al., 2015, p. 2-6). The ordering of the data analysis process can't be changed as each process step output serves as input for the next one but some of the steps can be left out as, for example, Badillo et al. (2020, p. 873) interprets data exploration as a voluntary step to be done if needed, and mentions the possibility to skip the data preprocessing and dimensionality reduction if the data is already in suitable format to be inputted into machine learning model. Afterall as machine learning is conducted in order to extract knowledge from data, the data analysis process and context are crucial to formulate the correct model and interpretations of it.

3.1 Data preprocessing

The purpose of data preprocessing in data analysis process is to prepare all the data to be ready for data modeling part as datasets in real world are rarely in a format to be utilized without any processing or modifications (Garciá et al., 2015, p. 10-11). As data can be acquired from multiple sources and in multiple formats the data can be messy and might be impossible to interpret by machine learning models (Nelli, 2018, p. 9-10). To tackle these problems Garciá et al. (2015, p. 11-13) have defined six different data preprocessing areas listed in **table 2** with name of the area, goal of the area and examples of typical actions. The order of performing the activities isn't defined and some of the areas can be conducted multiple or zero times during the data preprocessing phase.

Table 2. Names of preprocessing areas, goal of their activities and example actions as interpret by Garciá et al. (2015, p. 11-13).

Preprocessing area	Goal of activities	Examples
Data Integration	Combining data from different sources, identifying duplicates and detection of conflicts in data from different sources.	Merging two tables and finding same instances.
Data Transformation	Converting and/or consolidating the data.	Checking the datatypes of data or calculating average multiple instances of one feature.
Data Cleaning	Erasing or modifying bad data, filter data or reduce unnecessary details.	Filtering dataset to leave unwanted instances out or deleting features which are unnecessary.
Noise Identification	Detecting errors and impossible feature values or variances.	Correcting item price from -5€ to 5€ or apple price from 2m€ to 2€.
Missing Data Imputation	Fill in missing values.	Mean imputation or hot deck imputation.
Data Normalization	Normalize the data to same scale or range to give equal weight on features.	Min-max normalization or normalizing to unit interval.

Data integration and data transformation are very common first steps for starting to gather data from different sources. Data integration is an easy pitfall spot where redundancies and inconsistencies could be accidentally made with poor integration methods (Garciá et al., 2015, p. 12). When merging data from two different tables the best way to ensure that merge accomplishes is that both tables have the same unique identifier available, meaning that the identifiers have 1:1 relationship (Wilton & Colby, 2005, p. 93). A well and precisely performed data integration eases the other parts of data preprocessing.

Data transformation is essential since performing calculations and aggregations to the data requires the same data type of original values in R and other programming solutions (Venables et al., 2020, p. 7-8, 13). Most of the machine learning algorithms accept only numeric values as inputs when some of the raw data might have categorical values such as “male” or “female” (Potdar et al., 2017, p. 7). Potdar et al. (2017, p. 7) have presented ordinal coding as solution for categorical values where each of the category can be assigned with an integer and examples “male” and “female” are transformed into integers 1 representing “male” and 2 representing “female”. One of problems with ordinal coding is if the integers aren’t transformed back to original values, the interpreter might have difficulties to understand the results.

Consolidations and aggregations of the data are vital for machine learning applications since original data may have multiple different observations of one investigated object. So, if there are multiple observations for one investigated object, the data needs to be consolidated into one observation for each investigated object with mathematical methods to enable comparison between these investigated objects (Clarke et al., 2009, p. 409). Most of the machine learning algorithms can not interpret three different rows from an imaginary company A to be one observation even if the purpose would be to study company A and not its different, for example, orders hence the requirement to consolidate these rows as one.

Since most of the machine learning algorithms assume that the data is accurate, the data inputted should be as accurate as possible, but unfortunately real-world data gathered from all around is rarely accurate and complete (Luengo et al., 2020, p. 101). Data cleansing is the area in preprocessing where most of inaccurate of data is corrected, filtered or deleted (Garciá et al., 2015, p. 11). In real-life datasets most of the incorrect data come from human errors and mistakes which occur when inputting the data into user interfaces of different systems (Bai, 2019, p. 3-4). Some datasets consist of diverse different categories of observations or variables which are not necessary in analysis so they can be just deleted during data cleansing. Cleansing should be done also to “dirty data”, meaning values which make no sense at all such as “elephant” on weight variable (Garciá et al., 2015, p. 11). When data cleansing is done the dataset should have only the needed data and abnormalities should have been erased.

Noise identification can also be seen as part of data cleansing but it has its own characteristic: the noisy datapoint looks correct by datatype but the variance is disturbingly high, value is particularly extreme (single variable noise) or the combination of values are odd on multiple variables (multivariate noise) (Ciaburro, 2017, p. 89). Defining the “extreme value” is often complicated and if the noise is multivariate, then the problem gets even more complex but noise can be observed with rules-of-thumb and some spatial characteristics, such as small clusters when plotting the data (Luengo et al., 2020, p.102). For single value noise with numerical data one good rule-of-thumb would be examining the values with more than three times the mean absolute deviation as these tend to be outliers (Ciaburro, 2017, p. 90). Multivariate noise detection is especially difficult when the variables on their own would seem viable, but the combination doesn’t make sense and therefore detection often needs human supervising (Luengo et al., 2020, p. 103; Ciaburro, 2017, p. 89). Noise identification is extremely important in machine learning applications since high quantity of noisy variables

create bias inside algorithm logic and may cause a totally different result than without the noise (Garciá et al., 2015, p. 107).

Missing data imputation is an important part of preprocessing as missing data is regularly encountered by numerous people working with real-world datasets (Cheema, 2014, p. 487). As a good example, Honeywell corporation, a worldwide operating company specialized in databases and data-led solutions, has more than 50 % of its values missing inside the databases they are maintaining (Zhu et al., 2010, p. 61). The problem with missing data is that it highly affects the results derived from the analysis and it is a major issue especially in datasets gathered with questionnaires (Peyre et al., 2010, p. 287-288). One way to deal with incomplete data is just to delete the observations having incomplete values but often that approach is not applicable due to data sparsity and a need of larger dataset and therefore many approaches to fill the incomplete data have been designed (Merkle, 2011, p. 257-258).

One of the most used method categories to complete the incomplete data is imputation in which the aim is to replace the missing values with estimated ones (Mohamed et al., 2014, p.35). There are multiple different imputation methods and finding the most suitable one requires understanding of the nature of missing values as it affects the imputation methods (Wei et al., 2018, p.1-2, 8). There are three types of missing value occurrences: missing completely at random (MCAR), missing at random (MAR) and missing not at random (MNAR) where the relation of missing data is different towards the rest of the data (Di Guida et al., 2016, p. 2-3). MCAR data is not related to any of the variables (predictors) or responses in the dataset making it independent from all the other values, MAR data is dependent on some other or more variables than itself, for example as gender affects the probable height, meaning other variables in the dataset may predict the missing value, and the last type, MNAR, reflects situations where the missing value is dependent on the values of the same variable (Brown & Kros, 2003, p. 3; Cheema, 2014, p. 490-492). A good working assumption is that if the data is MCAR the more simplistic methods are usable as predictive power doesn't help in those situations but on MNAR and MAR situations a more refined methods tend to be more powerful (Gorelick, 2004, p. 2-4).

Multiple imputation methods have been proposed by scholars but due to its simplicity and ease of use, a mean imputation is a popular method to handle missing data (Brown & Kros, 2003, p. 615). It is a quick way to impute all the missing values with mean of the observed instances inside one variable or with median if the data is known skewed (Brown & Kros, 2003, p. 615).

It has many drawbacks especially if the proportion of missing value is high since it decreases the standard error making variance estimates misleading and it diminishes perceived correlations because of repetitive value (Cheema, 2014, p. 493; Brown & Kros, 2003, p. 615). Mean imputation is recommended to use on MCAR data and only on lowly related MNAR or MAR data (Di Guida et al., 2016, p. 92). Some other imputation methods include hot-deck imputation where the missing values are estimated using observed values from the same dataset and it is particularly popular in surveys and applicable especially on MNAR data (Cheema, 2014, p. 494-495). MAR data imputation is effective with methods which utilize regression or classification modeling which take advantage on correlations disclosed between the different variables (Mohamed et al., 2014, p. 35). Even those models produce often the most accurate and least biased results they are mathematically complex, conducting such models takes knowledge and time and they're by far the most computationally expensive (Zhu et al., 2012, p. 62-65).

3.2 Dimensionality reduction & feature selection

Real-life datasets often contain a lot of data features on the specific matter which might not be at all important or necessary for the modeling problem at hand (Freeman et al., 2014, p. 1812). Li et al. (2010, p. 2) discuss on two different categories of reducing dimensionality: methods which maps the high dimensional features to a totally new feature space composed of lower dimensionality (also known as feature extraction) and then to more traditional feature selection methods where some original features might be dropped out from the dataset with specified criteria. In short feature selection is a process to find the optimal subset of features with certain rules based on data at hand (Garciá et al., 2015, p. 163). This optimal subset or feature space should describe the modeling problem properly and minimize the degeneration of performance (Bolón-Canedo et al., 2015, p. 14). In more practical approach feature selection would be choosing the suitable features from the dataset or creating totally new features capturing the characteristics of old features for the used machine learning algorithm. The goal of feature selection is to maximize the model performance, reduce the complexity and computational requirements of the dataset, prevent the model from learning false associations from data and achieving more understanding the underlying relations with less complex results (Jiang & Wang, 2015, p. 203; Cannas et al., 2013, p. 1446). The importance of feature selection process increases together with the amount of data and especially in big data applications feature selection is essential step ensuring the correct results and interpretations of machine learning applications (Luengo et al., 2020, p. 4). Since in this study the feature extraction method is

used to do feature selection (forming subsets instead of creating new variables) from this forward it will be addressed as one dimension of feature selection.

Feature selection methods can be divided into two different main classes which are the filter and wrapper methods (Kubat, 2015, p. 205). Filtering methods are the simpler of these two and the purpose for filtering methods is to select features on evaluation measures such as information, distance, dependency or consistency measures (Garciá et al., 2015, p. 173). The most common filtering method is a ranker algorithm which ranks and orders the features by calculating their usability for the model used with specific criteria (Freeman et al., 2014, p. 1813). Then the selection of needed variables is done according to some threshold, quantity of features or with cross-validation (Kubat, 2015, p. 205). Filtering methods are popular due to their low computational cost, ability to be generalized easily and since filter methods are totally independent from other parts of data analysis process it can be used with any ML algorithm (Bolón-Canedo, 2015, p. 16; Blum & Langley, 1997, p. 254-255). The main problem on filter methods is that it might keep variables which have clear relationships because they ignore those intervariable connections on multiple variables when focusing on a single two-variable connection alone (Freeman et al., 2014, p. 1814).

Wrapper methods utilize the specific machine learning algorithm to evaluate different combinations of subsets and chooses the subset performing with highest predicting accuracy (Garciá et al., 2015, p. 174-175). Wrapper methods give the best subset to be used but because wrapper acts as a black box the analyst can't know how the algorithm settle upon the result (Li et al., 2010, p. 3). As wrapper methods take in account single variables and all the relationships to other variables it is more comprehensive than filter method and it doesn't require specifying the correct quantity of variables (González et al., 2019, p. 408). Due their nature to utilize machine learning in the evaluation process the wrapper methods are more complex, both process and computational way, and when the dimensionality gets higher the more time and power the wrapper method requires (Solorio-Fernández et al., 2016, p. 866-867).

Even feature extraction methods are mainly used for deriving a new set of variables which are a combination of original dimensions they can be also used for feature selection due their ability to express the contribution and importance of original variables in the new combined variables (Alpaydin & Bach, 2014, p. 116, 124; Kassambra, 2017, p.56-58). Feature extraction methods can be performed in many ways but usually they are divided within supervised and

unsupervised methods or linear and non-linear combinations based on their attributes (Alpaydin & Bach, 2014, p. 116). Advantages in feature extraction methods are that you can compress information from a very high quantity of variables into significantly smaller quantity of variables by transforming the data and then maybe discarding some of the new dimensions with some criteria (Joshi, 2020, p. 25-26). If the original data is selected carefully and the variables aren't related with each other the feature extraction methods don't perform so well as they rely on finding similar information from multiple variables and then compressing it.

3.2.1 Principal component analysis

Principal component analysis, also known as PCA, is an algorithm which is commonly used for reducing the dimensionality by seeking projections that capture the most information of the data judging by its variance (Fernandez et al., 2016, p. 1397). PCA is utilizing linear transformations to generate new orthogonal variables known as principal components which are ordered by the quantity of variance they explain (Ivosev et al., 2008, p. 4934). Technically, the principal components are just normalized eigenvectors of the covariance or correlation matrix of the original data that are orthogonal or directionally uncorrelated (Ringnér, 2008, p. 303). PCA is highly used tool amongst many fields studying data because of its simplicity and ability to be extended further i.e. Fuzzy-PCA (Sârbu & Pop, 2005, p. 1215).

Abdi & Williams (2010, p. 434) state that the four main goals of PCA to effectively reduce the dimensionality are:

1. Capture from the original data its most important information
2. Condense the original data keeping only the important information
3. Make the description of the original data less complex
4. Analyze the structure of the data, both features and objects.

Especially for this study the first, second and the fourth parts recognized by Abdi & Williams (2010, p. 434) are extremely important since they enable the examination of the data and its structure in feature selection point of view rather than feature extraction point of view. The variable structure knowledge and behavior interpretation enables analyzing variable combinations and their significance towards explaining sample clusters (Ivosev et al., 2008, p. 4934).

Understanding PCA requires some understanding of linear algebra but for any linear combination $\boldsymbol{x^T y} = \sum_i \boldsymbol{x_i y_i}$ the variance can be computed with:

$$\boldsymbol{E}\left\{\left(\boldsymbol{x^T y}\right)^2\right\} = \boldsymbol{x^T C x} \quad (1)$$

where the **C** denotes the covariance matrix **E { yy^T }** and **x** and **y** are vectors. Since the first principal component is required to have maximum variance the PCA problem can be presented, assuming the mean to be zero, as:

$$\max_{\boldsymbol{x}:\|\boldsymbol{x}\|=1} \boldsymbol{x^T C x} \quad (2)$$

Because the nature of the covariance matrix requires symmetry $\boldsymbol{cov}(\boldsymbol{z_i}, \boldsymbol{z_j}) = \boldsymbol{cov}(\boldsymbol{z_j}, \boldsymbol{z_i})$ the covariance matrix can be formulated as a product of:

$$\boldsymbol{C} = \boldsymbol{ODO^T} \quad (3)$$

where **D** is diagonal matrix with **D** = diag(λ_1, . . .,λ_m) and **O** is an orthogonal matrix. The columns in matrix **O** are eigenvectors and the diagonal values **λ** are eigenvalues. Now when combining the formulas 2 and 3 and changing variable $\boldsymbol{w} = \boldsymbol{O^T x}$ we get:

$$\boldsymbol{x^T C x} = \boldsymbol{x^T ODO^T x} = \boldsymbol{w^T D w} = \sum_i \boldsymbol{w_i^2 \lambda_i} \quad (4)$$

Having **O** as orthogonal, ||**w**|| = ||**x**||, the constraint stays the same for **w** as it was for **x**.
Finally, the problem can be transformed into its final form:

$$\max_{\sum \boldsymbol{w_i^2}=1} \sum_i \boldsymbol{w_i^2 \lambda_i} \quad (5)$$

From this last formula we can easily find the first principal component when the $\mathbf{w_i^2}$ corresponding to the biggest $\boldsymbol{\lambda_i}$ equals to one and the others equal zero. The second principal component can be found the same way by making the new **x** orthogonal to the first eigenvector and when **x** equals the eigenvector corresponding the 2nd biggest eigenvalue and the same logic works with *n*th principal component. (Fernandez et al., 2016, p. 1397-1398; Hyvärinen et al., 2009, p. 117-119).

With knowledge obtained above the PCA transformation can be done for a dataset with a few simple steps:

1. Mean-centering the data
2. Calculate the covariance matrix
3. Calculate the eigenvalues and eigenvectors of the covariance matrix and organize them with decreasing eigenvalues
4. Select an optimal subset of eigenvectors to be used as new base vectors
5. Project the original data on the new base

where optimal subset should contain as few principal components with reasonable enough value of variance (Fernandez et al., 2016, p. 1397-1398). There is a trade-off between the number of variables used and information kept: the more variables are dropped away then more information is lost (Brusco et al., 2009, p. 705). Choosing the optimal size of the subset can be a bit difficult and Brusco et al. (2009, p. 705-706) show in their work that finding the actual optimum without sophisticated methods can be error prone and misleading. However, choosing the suitable subset to represent the original data well enough can be calculated with rule-of-thumb formula:

$$\frac{\sum_{k=1}^{L} \boldsymbol{O}[k,k]}{\sum_{k=1}^{n} \boldsymbol{O}[k,k]} \cong \mathbf{0,9} \tag{6}$$

where **n** denotes the original amount of variables and **L** is $\mathbf{1} \leq \boldsymbol{L} \leq \boldsymbol{n}$ and value *0,9* indicates just approximate starting value for evaluating the sufficient variation explained as that value is highly dependent on the characteristics of the original data (Fernandez et al., 2016, p. 1398; Cangelosi & Goriely, 2007, p. 2-3). Practically this means that one should choose as few variables as possible while maintaining required variation explained. In this work the projection of the data into a new plane isn't the goal but understanding the most important principal components which explain enough variance are interesting as they bring the understanding of important features.

In order to understand the generated principal components and which original variables are the most important towards variance, a concept of variable contribution of component has been made. Variable contribution of component explains how much each of the original variables contribute to the principal component with value between 0 and 1 and where sum of all variable contributions is 1. The contribution of variable **j** to component **h** can be calculated as:

$$\boldsymbol{cont_{j,h}} = \frac{\boldsymbol{f_{j,h}^2}}{\boldsymbol{\lambda_h}} \quad (7)$$

where $\boldsymbol{\lambda_h}$ is the $\mathbf{h^{th}}$ components corresponding eigenvalue and $\boldsymbol{f_{j,h}}$ is the factor score value for corresponding eigenvalue and principal component. With contribution of variables the most important original variables, in terms of variance explained, can be found and in feature selection process those can be maintained into dataset and less explaining variables can be erased from the dataset. (Abdi & Williams, 2010, p. 3-5).

3.2.2 Pearson correlation coefficient

One simple filter-based approach to feature selection is Pearson correlation coefficient (PCC) which is simple, yet efficient, method to determine redundant features (García et al, 2015, p. 42). PCC is evaluating the connectivity and relationship between two vectors and it bases on the covariance matrix of the data (Mu et al., 2018, p. 42). Correlation based methods, like PCC, are the most common filter methods used in feature selection and PCC is probably the most used of them (DeepaLakshmi & Velmurugan, 2016, p. 2).

The formula for calculating PCC is a very simple one:

$$\boldsymbol{PCC(x_i, x_j)} = \frac{\boldsymbol{cov(x_i, x_j)}}{\sqrt{\boldsymbol{var(x_i) \times var(x_j)}}} \quad (8)$$

where $\boldsymbol{x_i}$ and $\boldsymbol{x_j}$ are vectors (or variables), *var* denotes for variance and *cov* denotes for the covariance between those vectors (Xu et al., 2017, p. 1975). Notable from the formula is that PCC is symmetric due characteristics of covariance matrix, meaning that $PCC(x_i, x_j) = PCC(x_j, x_i)$ (Mu et al., 2018, p. 42). Results from the PCC are also simple as the values range from $-1 \leq PCC \leq 1$ where *-1* means a perfect negative correlation and *1* means a perfect positive correlation (García et al., 2015, p. 42). If two variables are positively correlated then when the one increases the other increases too and if they are negatively correlated then increasing the one decreases the other (García et al., 2015, p. 42). Ideally the features which are left into original dataset are not correlated or are very little correlated with each other so the ideal value for PCC between two features is 0 (García et al., 2015, p. 42).

Even the results of PCC are simple the interpretation could be a bit more ambiguous. Defining “weak” and “strong” relationships is again highly based on the original data and its characteristics. Every relationship value provided by PCC should be examined in the context of the data and the scientific question at hand. Some rules-of-thumb are made and in **table 3** can be found the steps and interpretation for one example how to interpret different correlations. From there can be said that a PCC over 0,7 or under -0,7 indicates strong or very strong correlation. (Schober et al., 2018, p. 1765).

Table 3. Example interpretations from Pearson correlation coefficient values adapting Schober et al. (2018, p. 1765).

PCC value	Interpretation
+- 0.00 – 0.10	Non-existent correlation
+- 0.10 – 0.39	Weak correlation
+- 0.40 – 0.69	Moderate correlation
+- 0.70 – 0.89	Strong correlation
+- 0.90 – 1.00	Very strong correlation

3.3 Unsupervised learning

As stated before, the unsupervised learning is one type of machine learning application where the data does not have labels. The goal of the unsupervised learning is to find connections, patterns and similarities from the data (Badillo et al., 2020, p. 873). Understanding the structure of the data might help to choose a better algorithm for supervised learning or choosing a more adequate set of initial variables to use (Joshi, 2020, p. 133). Unsupervised learning is also valuable indeed in situations where the data is required to label for further processing or for supervised algorithm to use (Joshi, 2020, p. 133). As the data in this study is unlabeled, the unsupervised learning techniques enable efficient labeling and give opportunity for the user to enhance their process with unsupervised learning.

To understand how dimensionality reduction can be considered part of unsupervised learning is simple: dimensionality reduction addresses datasets descriptive statistics patterns, connections and similarities to find lower-dimensional layers of the data space that are representing the highest data density (Hastie et al., 2017, p. 486). Multiple dimensionality reduction algorithms, like PCA, were developed before the term “Machine Learning” even was founded but multiple other dimensionality reduction algorithms are developed recently to address new problems in data analysis and of some which combine successfully supervised and supervised learning to reduce dimensionality (Badillo et al., 2020, p. 874-875).

Clustering is part of unsupervised learning and the objective of cluster analysis is to provide knowledge from the data by dividing the observations into different groups of observations (Wu, 2012, p. 2). The goal of defining these clusters from the data is that the intra-cluster similarity is being maximized and inter-cluster similarity is being minimized (Sharma & Yadav, 2013, p. 207). Wu (2012, p. 2) identifies two major purposes for clustering analysis to exist: understanding and utilization. Understanding refers to ability to find and extract meaningful clusters of observations that are similar and have the same characteristics and utilization refers to ability to understand what does a representative example of single cluster look like according to statistical values (Wu, 2012, p. 2). The cluster analysis methods can be divided roughly into three different main categories: hierarchical clustering, partitional clustering and density-based clustering as presented in **figure 6** (Wierzchon & Klopotek, 2018, p. 29, 34, 51; Wu, 2012, p.4-5).

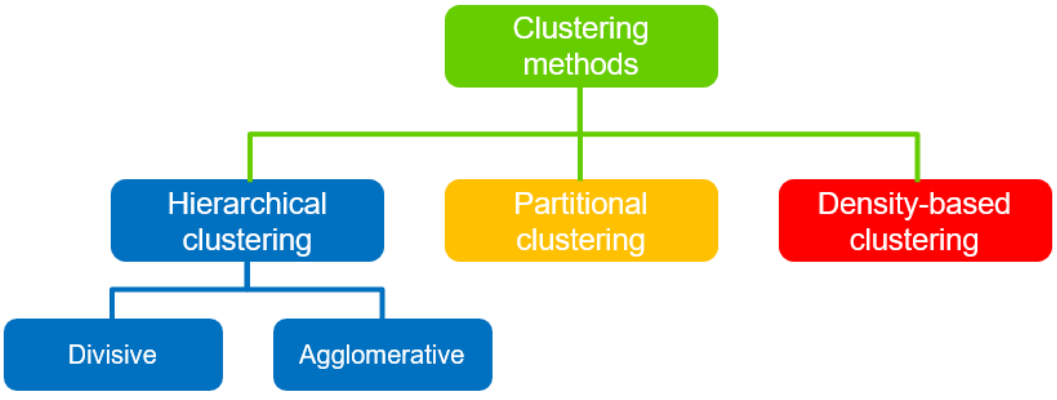

Figure 6. Cluster analysis methods classification.

Hierarchical clustering methods create a cluster hierarchy where each pair of observations or clusters are progressively nested into a bigger cluster until everything is in one cluster (Revathy et al., 2017, p. 164). They produce tree-like structures which are called dendrograms like in **figure 7** (Wierzchon & Klopotek, 2018, p. 29). Hierarchical clustering is based on distances between observations and linkage criteria between new clusters and existing clusters (Yang et al., 219, p. 12-13). Hierarchical clustering can be split into two main approaches: divisive and agglomerative (Chormunge et al., 2014, p. 92). The difference is that agglomerative approach is “bottom-up” and starts from individual observations merging them based on distances and criteria until one cluster is left, where the divisive approach is “top-down” and starts from a single cluster and splits it based on distances and criteria until there are as many clusters as there are observations or if some kind of stopping criteria is met e.g. number of clusters (Chormunge et al., 2014, p. 92).

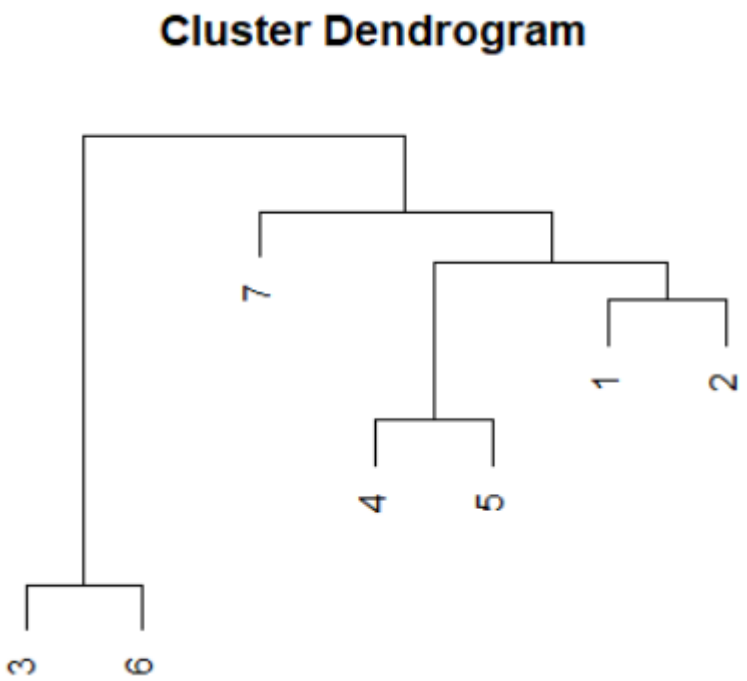

Figure 7. An example of hierarchical clustering result as dendrogram.

Partitional clustering methods divide the data into clusters which are disjointed by partitioning the data into K partitions which represent a cluster (Yue et al., 2016, p. 1127-1130). Partitional methods require the quantity of clusters to be predefined in order to function and the algorithm then clusters the data into these clusters by satisfying two rules: each cluster must have at least one observation and each observation can be in one cluster only (Wierzchon & Klopotek, 2018, p. 34). Partitional methods are based on distances and iterative rounds to minimize an

objective function to satisfy the condition, for example, in terms of WSS (within sum of squares) of similar clusters (Yue et al., 2016, p. 1130). The most known partitional clustering algorithm is a K-means algorithm with its all variations e.g. fuzzy K-means, kernel K-means and genetic K-means (Bagirov et al., 2020, p. 10).

Density-based clustering methods takes the cluster as a dense region of observations which is neighbored by regions which have low densities (Wu, 2012, p. 4). These clustering methods comes especially convenient when the clusters are in non-convex shapes and the data contains a lot of noise and outliers as can be seen from **figure 8** (Wierzchon & Klopotek, 2018, p. 51). Density-based methods base on distances but rather than comparing the distance to centrum it takes in account the inter-observation distances (Chormunge et al., 2014, p. 93). A few known density-based clustering algorithms are DBSCAN and DENCLUE (Wu, 2012, p. 4). Density-based methods tend to have two drawbacks: they might lack a bit of interpretability and if a cluster is having one more dense area and one lower density area it might be less informative than some other clustering algorithm (Chormunge et al., 2014, p. 93).

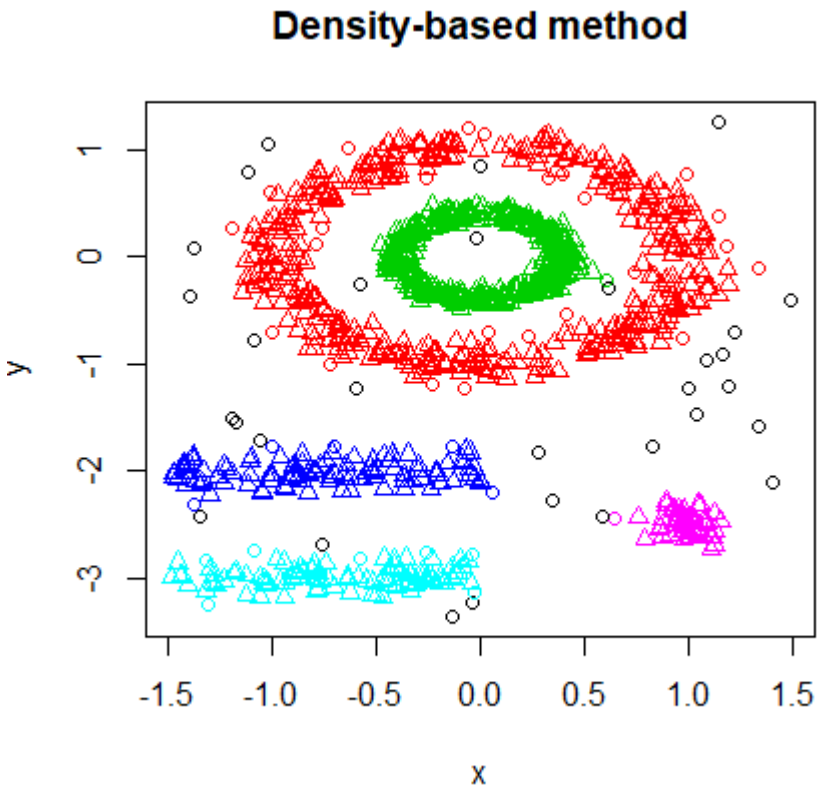

Figure 8. An example of density-based clustering method based on R-package “factoextra” multishapes dataset (Kassambra, 2020, p. 80).

3.3.1 K-means clustering

K-means clustering algorithm is a partitional clustering algorithm which tries iteratively to minimize the total WSS (Mouton et al., 2020, p. 2). K-means algorithm divides the observations into nearest clusters seeking the local optimum solutions where the observations will not move into new clusters whenever a new iteration round is applied or the change of cluster will not improve the total WSS (Hartigan & Wong, 1979, p. 100-101). Since K-means is based on distance metrics such as Euclidean distance or Manhattan distance it is especially efficient on numerical datasets (Thakare & Bagal, 2015, p. 13-14). Sometimes K-means algorithm is also known as Lloyd's algorithm as it was introduced by Stuart Lloyd in 1957 (Thakare & Bagal, 2015, p. 13).

The objective function for K-means algorithm can be formulated as following:

$$\min_{\{b_i\},1\leq i\leq k} \sum_{i=1}^{k} \sum_{j=1}^{m_i} dist(x_j, b_i) \tag{9}$$

where $\mathbf{b_i}$ is the centroid of the corresponding cluster, $\mathbf{m_i}$ is the number of observations belonging to cluster **i**, $\mathbf{x_j}$ is the observation belonging to cluster **i** and ***dist*** refers to distance function used (Wu, 2012, p. 7-8). According to Thakare & Bagal (2015, p.13-15) the Euclidean distance is the most common one and they conclude that the performance of K-means clustering algorithm highly depends on the data and distance measure metric used. If the data is not numerical and is, for example, categorical then the choosing the distance metric is extremely important as categorical data can't express statistical measures and distance measure will affect the result (Chen & Yin, 2018, p. 160).

The K-means clustering algorithm is straightforward when the quantity of clusters is determined:

Step 1. Choose randomly the initial k cluster centers from the observation space
Step 2. Assign each of the observations into the nearest cluster with chosen distance metric
Step 3. Update the new cluster center based on observations assigned into that specific cluster

Step 4. Repeat steps 2 and 3 until no observation changes cluster improving total WSS or until some specific defined stopping criteria fulfills (Revathy et al., 2017, p. 165).

The advantages of K-means clustering algorithm is that it is simple and sufficiently robust along with high efficiency and ability to cope with most of the data types. If the dataset consists only of plain numerical variables, then the K-means algorithm is always a good option. Data with complex properties such as high-dimensionality, class imbalances and poor standardization might require enhancing K-means algorithm with different techniques, yet the principle of the algorithm will remain the same. The problems related to K-means are in its ability to perform with non-globular clusters and it is prone to outliers but either way the advantages are more severe than problems. (Wu, 2012, p. 8).

3.3.2 Defining the correct quantity of clusters

The number of the clusters is sometimes unknown and many clustering algorithms, especially partitional algorithms, require the quantity of clusters predefined in order to work (Hartigan & Wong, 1979, p. 100-101). Choosing the right quantity of clusters is important since excessive quantity of clusters makes the result almost impossible to understand and too low quantity of clusters will result loss of information and possibly incorrect decisions (Celebi & Aydin, 2016, p. 73-74). This means that a trade-off exists between suitable quantity of clusters and “perfect” unambiguous clusters. To establish the appropriate value of clusters with “k” representing the quantity of clusters is done by generating a partition quality measure of q(k) and then calculating a value of k which optimizes the measure (Wierzchon & Klopotek, 2018, p. 165). These quality measure values are highly dependent of the data itself (Chiang & Mirkin, 2010, p. 4-5). Depending on the data there might be simpler ways to determine the quantity of clusters and therefore approaches deriving the k can be roughly divided into three main categories and their sub-categories according to Wierzchon & Klopotek (2012, p. 165) and Chiang & Mirkin (2010, p. 8):

1. Data visualization approaches
2. Heuristic approaches
3. Quality measure approaches
    a. Variance-based methods: optimizing variance led functions by changing the value of k

b. Structural methods: comparison of intra-cluster similarity and inter-cluster separation with different values of k
c. Consensus distribution methods: Utilizing consensus matrix for different sets of clusterings on different k value
d. Hierarchical methods: Choosing the value of k by judging the result of divisive or agglomerative clustering process
e. Resampling methods: Evaluating the value of k by clustering results of sampled data

Data plotting for visualization purposes is very simple and always recommended to perform especially on low-dimensional data to obtain understanding of data structure (Miller, 2017, p. 7-9). For example: a simple scatter plot might reveal clear clusters from the data space. For high-dimensional data, the data visualization can be done by projecting the data into bi- or tri-dimensional space (Wierzchon & Klopotek, 2018, p. 165). However sometimes data visualization for cluster seeking isn't even an option due to inability to compress the high-dimensional data information into 2d- or 3d plane.

Heuristic approaches include rules-of-thumb, like $\boldsymbol{k \approx \sqrt{v/2}}$, where *v* is the quantity of observations, or the elbow method (Wierzchon & Klopotek, 2018, p. 167). Elbow method requires running the clustering algorithm with different value of k and then gathering the total within sum of squares (total WSS) for each values of k in line graph (Fattah et al., 2016, p. 15). Then the object is to find the quantity of clusters after which the information add is minimal meaning finding the value of k after which the curve flattens, creating an angle of "elbow" (Paternina et al., 2018, p. 173). The total WSS can be calculated using Euclidean distance with formula:

$$\boldsymbol{tWSS = \sum_{i=1}^{k} \sum_{j=1}^{m_i} \left\| x_j - b_i \right\|^2} \quad (10)$$

where there is $\mathbf{m_i}$ points $\mathbf{x_1, \ldots, x_{mi}}$ in cluster **i,** a set of clusters $\mathbf{C = (C_1, \ldots, C_k)}$ and the $\mathbf{b_i}$ is the cluster centroid calculated as:

$$\boldsymbol{b = \frac{1}{m} \sum_{j=1}^{m} x_j} \quad (11)$$

where **b** represents one cluster centroid with **m** points (Bhargavi & Gowda, 2015, p. 3675). One downside of the elbow metric is that it only takes on account the cohesion inside clusters but doesn't address the similarity or dissimilarity between the clusters (Fattah et al., 2016, p. 15). The second downside of elbow method is the ambiguousness since it reaches the optimal value when there are as many clusters as there are observations and finding a coherent "elbow" can be extremely difficult (Paternina et al., 2018, p. 173).

One of the popular variance-based quality measures is called Calinski-Harabasz index (CH index) (Chiang & Mirkin, 2010, p. 9). The CH index is calculated with formula:

$$\boldsymbol{CH(k)} = \frac{\boldsymbol{tBSS}}{\boldsymbol{tWSS}} \times \frac{\boldsymbol{m-k}}{\boldsymbol{k-1}} \tag{12}$$

where **m** is the amount of observations and total between cluster sum of squares *tBSS* = *TSS* – *tWSS*, where TSS is equal to:

$$\boldsymbol{TSS} = \sum_{\boldsymbol{i=1}}^{\boldsymbol{m}} (\boldsymbol{x_i} - \bar{\boldsymbol{x}})^{\boldsymbol{2}} \tag{13}$$

where TSS means total sum of squares and $\bar{\boldsymbol{x}}$ is vectorized notation of the mean of **m** points (Celebi & Aydin, 2016, p. 81). With CH index the optimal value for k is when the k maximizes the function (Wierzchon & Klopotek, 2018, p. 170). Chiang & Mirkin (2010, p. 9) conclude that multiple researches have conducted the best results with CH index on different datasets and CH index was utilized by many authors for the cluster quantity choosing.

Silhouette index or silhouette width is a structural method which utilizes the quality scores of separate points of clustering and uses point-wise approximations for average quality estimation (Celebi & Aydin, 2016, p. 79). It measures inside-cluster cohesion and clusters separation from the rest of the clusters (Chiang & Mirkin, 2010, p. 11). The formula for calculating the silhouette width for one point is:

$$\boldsymbol{S(i)} = \frac{\boldsymbol{b(i)-a(i)}}{\boldsymbol{\max\,(a(i),b(i))}} \tag{14}$$

where **a(i)** equals the average distance between the point and other points inside the same cluster and **b(i)** equals the minimum of the average distance between the point and every other point which are in each of the other clusters (Wierzchon & Klopotek, 2018, p. 166). The values

are between -1 and 1 in silhouette, where 1 means that assignment to cluster is perfect, 0 means that the observation could be as well assigned in the other cluster and -1 means that assignment to cluster is totally incorrect (Chiang & Mirkin, 2010, p. 11). The total silhouette value can be calculated as:

$$\boldsymbol{S} = \frac{\mathbf{1}}{\boldsymbol{m}} \sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{m}} \boldsymbol{S}(\boldsymbol{i}) \tag{15}$$

where the higher the silhouette value is, the better the clustering (Celebi & Aydin, 2016, p. 79).

3.4 Supervised learning

Supervised machine learning can be understood as an algorithm learning to regress or classify data based on given labels on observations, as stated before. To be more precise, the supervised learning can be understood as a learning type with capability from observing exemplary pairs of inputs and outputs and learning the connection mapping inputs into outputs (Russell & Norvig, 2010, p. 695). The inputs and outputs are numbers and explaining values (Russell & Norvig, 2010, p. 695). The goal of supervised learning is to predict the correct output as accurately as possible while being able to generalize the prediction ability on new observations with the same variables (Clarke et al., 2009, p. 231-232).

Supervised machine learning applications can be categorized into regression problems and classifications problems (Badillo et al., 2020, p. 872). The regression problem tries to predict outputs for each of the inputs where the classification problem assigns each input into a finite discrete set of numbers or factors (Bishop, 2006, p. 4). A practical explanation is that regression problem tries to predict the correct value and classification problem tries to predict the correct category or class. Although there may be some overlaps on algorithm usage and few algorithms can be used for both problems they are evaluated differently as classification problem can be evaluated for instance on its accuracy and regression problem can be evaluated for instance with root mean square error (Russell & Norvig, 2010, p. 695-696). As this study focuses on classifying observations, we will focus more on that area.

In order to learn and perform its purpose the supervised learning algorithm needs at least two different datasets: training and testing sets (Fernandes de Mello & Ponti, 2018, p. 2). Training dataset is used for algorithm to learn these connections between the inputs and outputs and

tuning the hyperparameters (Yang & Shami, 2020, p. 295-296). To tune the hyperparameters efficiently in more complex supervised learning problems the training set is split into training and validation set where the validation set is used for measuring the performance of different settings of hyperparameters (Russell & Norvig, 2010, p. 709). The testing set is used after the algorithm is trained to measure its performance on new and unseen data, and validate its final usability (Kubat, 2015, p. 11-12). For this split into different sets to be feasible the stationarity assumption must be true which means that training, validation and testing sets from come from the same distribution having approximately the same mean, variance and autocorrelation (Russell & Norvig, 2010, p. 708).

The hyperparameter tuning is the process to optimize the hyperparameters used for configuring the model before running it. ANN models, for example, have two different types of parameters: model parameters, such as neuron weights, and hyperparameters which define the architecture of the whole model. Having the correct model and optimized hyperparameters very often make the distinction between a very poor and general model and model with high accuracy and high enough level generalization abilities which makes the hyperparameter tuning extremely important. The tuning happens manually by following rules-of-thumb and measuring the performance on validation set with different hyperparameter values or by using automatized searching techniques such as grid search or Bayesian optimization. (Yang & Shami, 2020, p. 295-296; Wistuba et al., 2018, p. 43-45).

One of the most important aspects of supervised machine learning is fitting the model (Clarke et al., 2009, p. 4-5). Concepts of overfitting and underfitting possess a real danger for either performance or generalization of the model and these problems derive two sources of error existing naturally in classification problems: variance and bias errors (Fernandes de Mello & Ponti, 2018, p. 98-99). Variance error exhibits the distance of the classifier performance from optimal solution and it derives from the variance of the data as training sets are never able to capture the full variance of the whole dataset (Kubat, 2015, p. 193-194). Bias error exhibits the distance from chosen classifier to optimal classifier in space of all classifiers and it derives from the chosen algorithm and its ability to cope the given dataset (Kubat, 2015, p. 194). A good example from high bias error would be using distance-based algorithm for categorical data which by its nature can’t understand that difference of classes 1 and 2 is equal to difference of classes 1 and 3 even though their “distance” is higher. In total variance error and bias error generate the total error of the model which should be minimized (Fernandes de Mello & Ponti, 2018, p. 99).

The problem within model fitting is that a trade-off exists between bias and variance errors (Russell & Norvig, 2010, p. 696). Choosing a larger training dataset leads to smaller bias related errors but it increases the variance related errors as the classifier specializes more on training data and loses gradually its ability to generalize and correctly classify new observations from observation space (Kubat, 2015, p. 193-194). Thinking of classifiers, choosing classifier with stronger bias will lead to smaller variance error as the classifier is able to predict with broader set of observations but loses gradually its ability to predict the specific task at hand (Kubat, 2015, p. 193-194). Overfitted model has small bias related error and high variance related error and underfitted model has small variance related error and high bias related error. The optimum lies somewhere between these errors and its highly related on the data at hand and the level of specialization the classification problem requires (Blanc & Setzer, 2019, p. 5271).

The objectives for splitting the data are following: the algorithm to-be-trained should be fed as much training data as possible and the test set should have enough samples to have explaining statistical confidence (Joshi, 2020, p. 169). Basic solution would be holdout method where the original dataset would be divided into training and test sets with division of 70-80 % into training set and 20-30 % into test set depending on data and its abilities (Joshi, 2020, p.169). Then the training set would be again split with same proportions into training and validation sets. This approach can be quite problematic, especially with sparse dataset, since excluding both testing and validation datasets from the training phase might not leave enough training data to model be able to specialize to the problem at hand (Bishop, 2006, p. 32).

One classic method to address this problem is k-fold cross-validation procedure (Clarke et al., 2009, p. 27). In k-fold cross-validation the training data is split into k subsets of similar size (folds) and then for each of the subsets the model is trained on the remaining subsets and evaluated on subset chosen for validation (Chollet, 2018, p. 113). The k-fold cross-validation score is the mean of validation scores of each k (Chollet, 2018, p. 113). Typically, the k is between 5 and 15 depending on the data but it might be even lower on especially sparse data (Clarke et al., 2009, p. 28). The k-fold cross-validation is beneficial when the chosen model exhibits great variance between different training set splits since it enables the change of validation set (Chollet, 2018, p. 113). Exemplary 4-fold cross-validation is presented in **figure 9**.

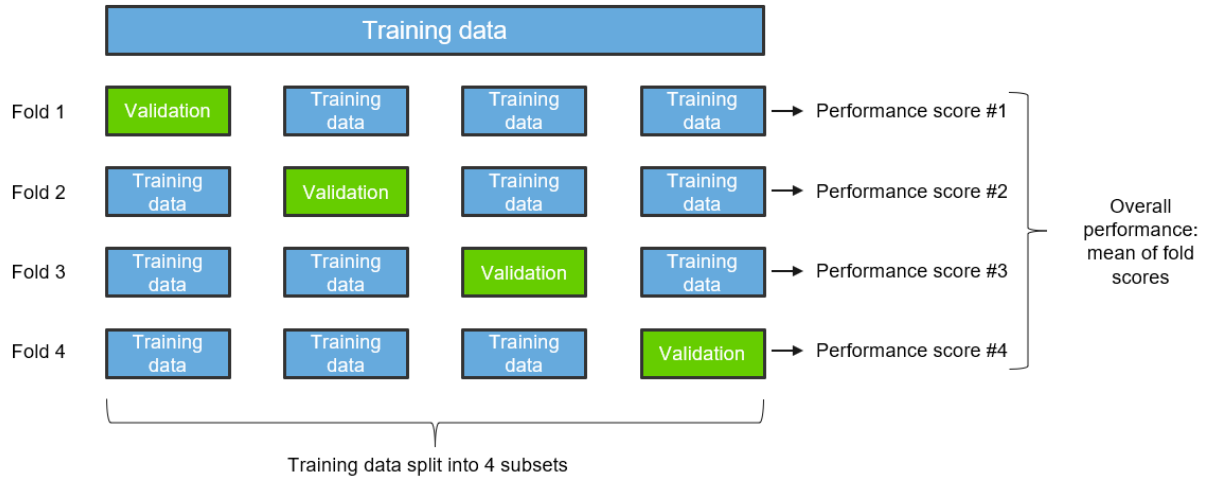

Figure 9. An example and idea of 4-fold cross-validation.

3.4.1 Artificial neural networks

The artificial neural networks (ANN's) might be the most known modern machine learning applications and there's a lot of buzz around the concept and its possibilities. ANN's are networks of nodes or so-called neurons which are interconnected with directional links (Russell & Norvig, 2010, p. 728). As the term *Artificial* suggests the ANN's mimic the structure of human brain which are built on electronic signals going from neuron to neuron with axioms and synapses (links or edges in ANN's) without biological weaknesses, such as cell degenerations, leading to errors (Russell & Norvig, 2010, p. 10-11; 728; Lahtinen et al., 1996, p. 213). In ANN's the neurons act as individual processors for inputs to generate outputs sent to other neurons via links which are weighted relating their correlation activity (Schmidhuber, 2015, p. 86). The general purpose of ANN's is to map a set of inputs (variables or features of observation) into a value (regression) or into a finite predefined set of classes (classification) (Rocha et al., 2007, p. 2810-2811).

Terms artificial neural network and multilayer perceptron (MLP) as often used interchangeably but to be precise multilayer perceptron is just one kind of ANN architecture where as others, such as recurrent networks, do exist (Rocha et al., 2007, p. 2809). Russell & Norvig (2010, p. 729) go even further on the categorization and claim that if activation function has a hard threshold then it is a perceptron and if the activation function is logistic, which is the preferred and more common option, the correct term would be sigmoid perceptron (Kubat, 2015, p. 91). The difference between single- and multilayer perceptron is that in single-layer perceptron the

inputs are directly connected into the output neurons (Russell & Norvig, 2010, p.729). On the other hand, multilayer perceptron might have multiple, but at least one, layer(s) between the input and output layers and they're called hidden layers (Nelli, 2018, p. 356). The concept of model depth explains the quantity of layers in the ANN and the deeper the network is the more complex problems it is generally able to solve (Chollet, 2018, p. 8-12). The basic architecture of ANN's are based also on the quantity of inputs and outputs, so a dataset with seven variables and three possible classes will have seven inputs and three output neurons (Dash & Behera, 2019, p. 6-7).

Multilayer perceptron is a feed-forward ANN which means that the data inputted into system propagates through each layer of the network towards the output which means that there are no loops and no layers can be skipped (Joshi, 2020, p. 44). Furthermore, neurons inside one layer are not connected with each other but neurons are fully connected to every single neuron on previous and next layers (Hastie et al., 2017, p. 393). When compared to the recurrent network which has loops and can even support short term memory, the feed-forward network is just a function of its inputs and has no internal state excluding the weights of links (Russell & Norvig, 2010, p. 729). The prediction of MLP is based on the values the output neuron obtains and each of the output neurons exhibits one possible class in classification problems (Kubat, 2015, p. 95). The values of the output neurons might not be explicitly expressed but they explain how likely the observation belongs to each of the classes and the class is determined with the highest output value (Kubat, 2015, p. 95). In **figure 10** is a presentation of an ANN with three inputs, one hidden layer with four neurons and output layer for three-class classification problem.

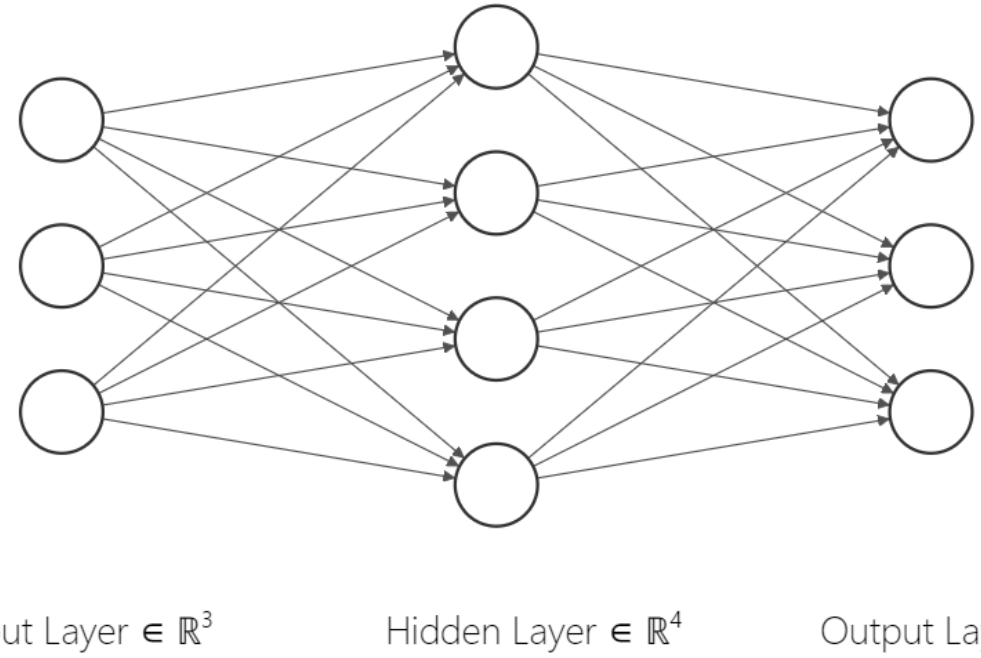

Figure 10. A presentation of neural network with three inputs, one hidden layer with four neurons and three outputs.

To understand more of the feed-forward propagation the basic unit of MLP, neuron, is essential. Each neuron receives n inputs from previous neurons (input layer receives the corresponsive values of observation) with a weight w associated with input. Each neuron also has a bias input, a value of 1 with its own weight, to serve as an intercept in the linear equation and help the neuron to evaluate its values without being fully affected by previous layer. The values from links are multiplied by the weights of the links and then summarized inside the neuron before being fed to activation function. Activation function transforms the value and feeds it forward to all the succeeding neurons. There are multiple different activation functions available but sigmoid functions are the most suitable for multiclass classification problems due to their ability to learn non-linear states and map inputs into probabilities and hard thresholds perform on binary classifications. So, the operations performed by neuron can be summed into mathematical formula:

$$\boldsymbol{a_k = f(\sum_{i=0}^{n} w_i a_i)} \tag{16}$$

where, $\mathbf{a_k}$ is the value neuron feeds forward, **f** is the activation function of neuron, $\mathbf{w_i}$ and $\mathbf{a_i}$ are the corresponsive weight of link and value from previous neuron and $\mathbf{a_0}$ is the bias input. The operations of single neuron are visually presented in **figure 11.** (Russell & Norvig, 2010, p. 727-732; Costarelli & Spigler, 2013, p. 102-103).

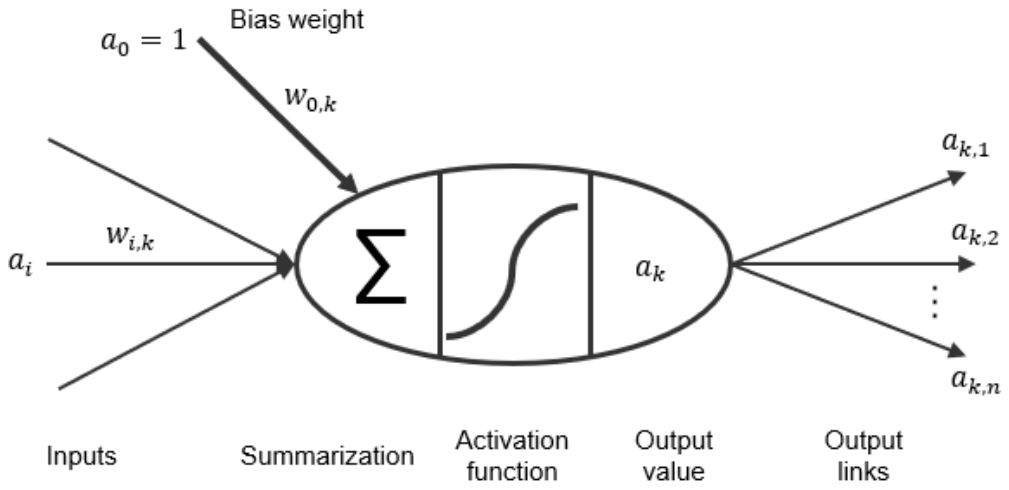

Figure 11. Illustration of operations performed in single neuron, adapted from Russell & Norvig (2010, p. 728).

The feed-forward neural networks learn with backpropagation process, thus the name feed-forward backpropagation neural networks (Hastie et al., 2017, p. 395). In backpropagation process the outputs of ANN are evaluated against the actual classes of the examples and then a cost function is used to calculate the difference between predictions and actual classes and this delta value is used for updating the weights of the model (Russell & Norvig, 2010, p. 733-734). Difference can be formulated for neuron *j* with i^{th} weight w_{ji} as following:

$$\Delta \boldsymbol{w_{ji}} = \boldsymbol{\alpha}(\boldsymbol{r_j} - \boldsymbol{y_j})\boldsymbol{x_i} \tag{17}$$

where **α** is so-called learning rate, which is small constant, **r** is the target output, **y** is the actual output and **x** is the input towards neuron (Russell & Norvig, 2010, p. 733-735). In a nutshell, ANN algorithm tries to minimize the error rate on prediction by updating the weights which are initialized as small values when starting to train the ANN (Hastie et al., 2017, p. 394). Especially on multiple output and hidden layer neural networks it is not so straightforward to understand which weights are causing the errors so the process updating the weights requires propagating the error value to previous layer, updating weights between the two layers and proceed this way backwards the network until the first hidden layer weights have been upgraded (Russell & Norvig, 2010, p. 734).

At the end, training an ANN has six concrete steps:

1) Define the number of hidden layers and number of neurons and used activation function
2) Initialize the weights and biases
3) Perform the forward propagation process and obtain outputs
4) Calculate the cost function and delta from optimal
5) Backpropagate the delta to update weights
6) Repeat previous steps until cost function delta is minimized (Badillo et al., 2020, p. 882-883).

3.4.2 Random forests

Random forests are extensions of decision tree -algorithm so in particular a random forest is a combination of tree predictors from same sampled data with similar distribution (Breiman, 2001, p. 5). To understand random forests some initial understanding of decision trees is required. Decision trees are supervised machine learning algorithms which try to classify or regress the data by splitting it with some variables or rules (Joshi, 2020, p. 53, 55). Decision trees are extremely popular due their simple usage and very visual and easy interpretation (Ahmad et al., 2017, p. 80).

As the decision tree tries to formulate the correct rules and their quantity, number of variables to use and when to terminate the tree it requires optimization based on some impurity measure (Joshi, 2020, p.55). The most known impurity measures used are the Gini index and cross-entropy measures for classification and mean-squared error (MSE) for regression problems (Clarke et al., 2009, p. 252). Impurity measures are used to determine splits and when to stop splitting and minimizing those measures creates distinct subsets sequentially ordered by size (Clarke et al., 2009, p. 252-253). Gini index is the most used measure since classification and regression tree (CART) -process created by Breiman et al. (1984, p. 31) utilizes Gini index. The Gini index explains the probability of random selection data sample input being misclassified when it belongs to the distribution of classes in the node and formulated as:

$$\boldsymbol{G} = \sum_{\boldsymbol{m=1}}^{\boldsymbol{m=k}} \boldsymbol{p_{mi}}(\mathbf{1} - \boldsymbol{p_{mi}}) \tag{18}$$

where the **m** represents the class, **k** the total number of classes, **i** represents each node and $\mathbf{p_{mi}}$ represents the fractions of classes predicted as **m** in node **i** (Joshi, 2020, p. 57).

Decision tree consists of a root node, internal nodes, leaf nodes and branches connecting different nodes. The root node refers to the first split of the tree where the data will be split into mutually exclusive subsets, internal nodes are possible choices to be made after the parent node so they divide the data into smaller subsets and leaf nodes are exhibiting the final results of classification or regression. The branches represent the rule of how the data is further split. In visualization of decision tree -models the root node is usually the topmost and the leaf nodes are found from bottom like **figure 12** is illustrating. (Song & Lu, 2015, p. 132).

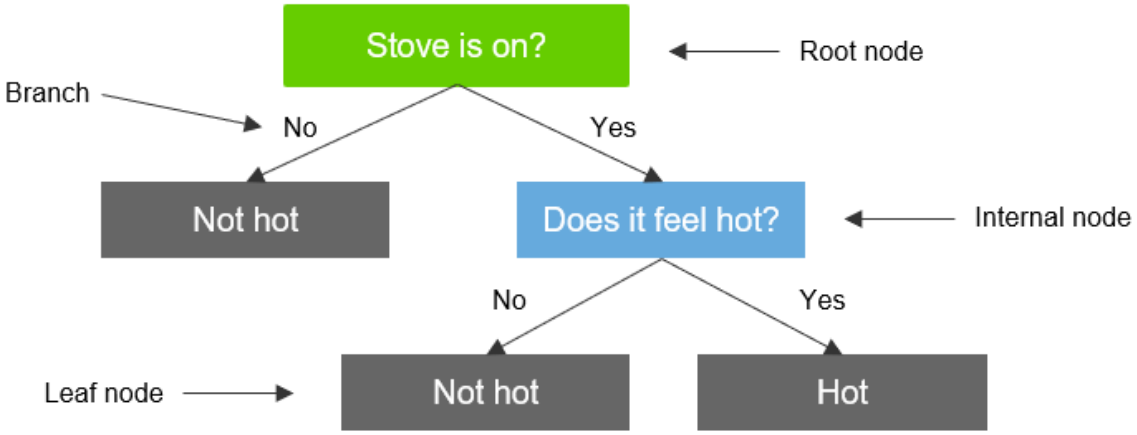

Figure 12. Example of simple binary decision tree structure for defining whether stove is hot or not.

The extension into random forests is necessary in some applications since decision trees may be sub-optimizing the performance or may lack the required robustness of the solution and are constantly overfitted (Ahmad et al., 2017, p. 80). To be able to extend the decision tree model into a larger random forest, a concept of ensemble learning is required. In ensemble learning a group of so-called weak learners (individual classifiers) are combined to find out the most popular result via voting scheme where majority of weak learners lean towards the most popular solution (Joshi, 2020, p. 60). Bauer & Kohavi (1999, p. 108-110) split ensemble methods into two main categories with their extensions: bagging, also known as bootstrap aggregation, and boosting. In bagging additional data is generated via random sampling where an individual observation can be utilized multiple times in so-called bootstrap samples leaving a 1/3 out of the samples as out-of-bag (OOB) samples which could be used for model evaluation if required (Breiman, 1996, p. 123-124). After creating the bootstrap samples, different models (trees) are trained and classification is done utilizing voting scheme aggregation on ensemble model values (Breiman, 1996, p. 123-124). Boosting on the other

hand is a sequential method based on iterative updating of weights of incorrect classifications creating different kinds of predictors and then combining these results from individual classifiers built on top of each other via voting scheme (Freund & Schapire, 1996, p. 120). In a nutshell can be said that bagging is a parallel method and therefore computationally cheaper and it decreases variance where boosting is sequential method and decreases bias (Bauer & Kohave, 1999, p. 108-110).

Random forests combine bagging and random feature selection to create more efficient algorithm compared to decision trees (Gray et al., 2013, p. 169). The random forest algorithm can be applied in both regression and classification problems, like decision trees, and we will focus on the latter. Random feature selection in random forests is applied in the splitting of nodes. In decision tree all the possible p variables were available to determine the best split but in random forests this quantity is used as a hyperparameter being usually $\sqrt{p}$ and called as the number of variables available for splitting (Clarke et al., 2009, p. 256). Combining bagging and random feature selection the random forest can generate low-correlating trees which enhances the robustness and resiliency to noisy data of the model (Hastie et al., 2009, p. 588-589). Random forests are effective at predicting especially classifiers and their ability to produce highly different trees lead to ability of rarely overfitting (Breiman, 2001, p. 29).

The general process for training a random forest includes these steps:

1. Create a bootstrap sample from the original dataset.
2. Train a tree on bootstrap sample with choosing the best split with a subset of variables p.
3. Repeat the first two steps until the selected quantity of trees are grown (Ahmad et al., 2017, p. 81).

3.4.3 Evaluating supervised algorithm performance

After training the algorithm with training data some measures are required to validate the performance while tuning the hyperparameters and in the end evaluating the performance of the whole algorithm with the unseen test data. This performance evaluation differs in regression and classification problems where in regression the performance is evaluated via difference of the actual value and predicted value, such as in root mean squared error (RMSE), whereas in classification problems the logic must be built upon the correct and incorrect classifications (Joshi, 2020, p. 171). In regression problems the goal is to minimize the error

and in classification the goal is usually to minimize the number of misclassifications. The evaluation phase is important since in the evaluation phase the usage of the chosen algorithm is justified which has an impact on the whole application and issue at hand (Bradley, 1997, p. 1145, 1158).

When focusing on classifier evaluation a concept of confusion matrix is essential for understanding the performance calculations built upon classifications (Fawcett, 2006, p. 862). The typical classifier performance metrics are derived from confusion matrix, such as accuracy (Kumar et al., 2020, p. 2076). A confusion matrix is a contingency table showing the related actual classes and their predictions (Bradley, 1997, p. 1145-1146). Confusion matrix is usually represented as 2x2 matrix for binary classification, but it can be quite easily to be extended to multi-class classification problem where the correct classifications would be on diagonal elements (Hand & Till, 2001, p. 176). A simple 2x2 confusion matrix is illustrated in **figure 13** with four possible outcomes.

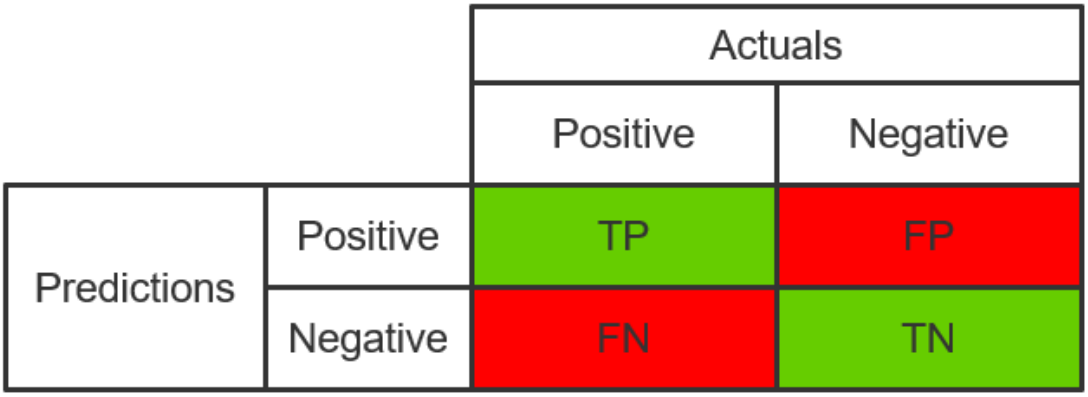

Figure 13. Illustration of a confusion matrix.

The four possible outcomes are:

1. True positives (TP): observations being positive and correctly predicted as positive by the classifier.
2. True negatives (TN): observations being negative and correctly predicted as negative by the classifier.
3. False positives (FP): observations being negative and incorrectly predicted as positive by the classifier.

4. False negatives (FN): observations being positive and incorrectly predicted as negative by the classifier. (Kubat, 2015, p. 214).

Since there are only four categories available, all the observations from the dataset are distributed into these categories. These four possible outcomes serve as base for the metrics and are the calculation blocks for them (Fawcett, 2006, p. 862). Maybe the most used metric for classifier performance is the accuracy which explains the percentage of a classifier to correctly classify the observations (Kubat, 2015, p. 214). Accuracy is calculated as following:

$$\boldsymbol{Acc} = \frac{\boldsymbol{TP+TN}}{\boldsymbol{TP+TN+FP+FN}} \tag{19}$$

Even accuracy describes well the overall performance of the classifier it still has one drawback: when the dataset has some minor classes where the classification error happens, the accuracy may look fine even some cases are totally incorrectly classified (Powers, 2011, p. 38-39). In datasets where majority of observations belong to few classes and marginal quantity of observations belong to rest, this creates a problem. Real-life datasets tend to possess unevenly distributed observations on classes and it requires another measure to focus on class represented by just a couple of observations (Kubat, 2015, p. 217). Introducing the true positive rate (TPR) and Positive predictive value (PPV), also known as Recall and Precision, which take in account also the incorrect classifications (Joshi, 2020, p. 172). TPR and PPV are calculated as following:

$$\boldsymbol{TPR} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FN}} \tag{20}$$

$$\boldsymbol{PPV} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FP}} \tag{21}$$

The only difference between these two values is the different nominator where TPR has false negatives and PPV has false positives added there. Still the difference is clear when TPR measures how well the class is predicted from all actual observations belonging to class where PPV measures how correct the classifier is when predicting observation to belong to class (Joshi, 2020, p. 172). The TPR is used in this work since TPR portraits if something is false negatively classified to see the efficiency of finding correct clusters for suppliers. It is important for the application to detect all the correct classification cases rather than positively classified ones.

4 PREDICTIVE SUPPLIER EMPIRICAL ANALYSIS

This chapter presents the process steps and results of the empirical analysis of this study. The purpose is to categorize suppliers into clusters with unsupervised machine learning algorithm and then experiment with supervised machine learning algorithms in order to create way to assign new suppliers into coherent clusters. The process and its steps conducting the analysis is illustrated in **Figure 14**.

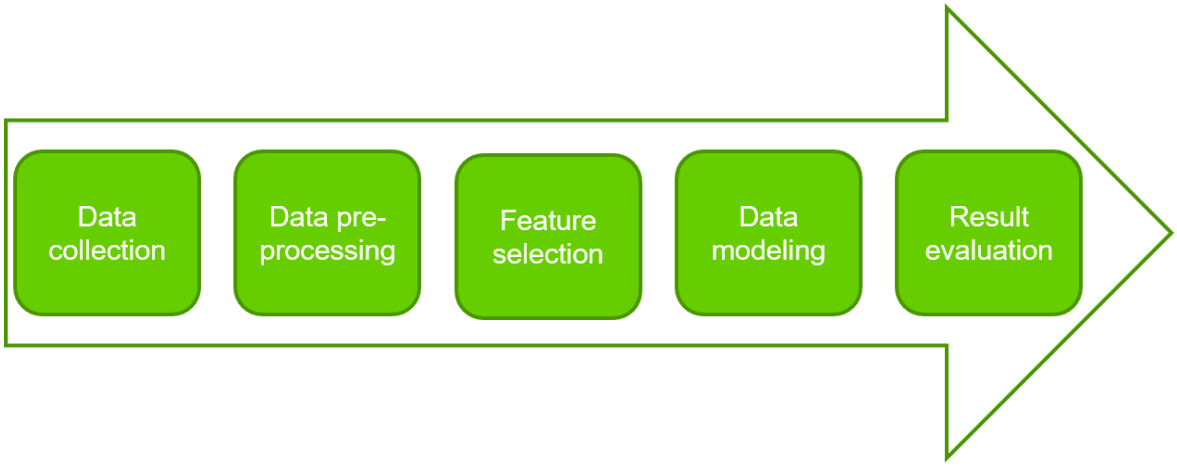

Figure 14. The process conducting the analysis adapting the steps of Garciá et al. (2015, p.1-2).

The following subchapters will describe the process more in detail. In the first subchapter the starting point of the data collection is described along with identification of requirements for the data and different data sources. The next subchapter focuses on the transformations and preliminary processing steps such as outlier and missing value handling which had to be done to the dataset prior moving on. These processing steps conduct of combining the data, transforming datatypes, choosing and conducting the variables to be used in further phases of the analysis, handling outliers and clear quality errors, aggregations of the data and handling the missing values in the dataset. In the third subchapter principal component analysis (PCA) and Pearson correlation algorithms are applied into the data to reduce the dimensionality and choose the most appropriate variables for the data modeling phase. In this study the term feature selection is used instead of dimensionality reduction since the feature extraction method is used to complement feature selection analysis rather than creating new variables. First the algorithms are applied and based on the results the data set dimensionality is reduced and then the algorithms are applied again to see if the new dataset has preserved its diversity

and unnecessary variables are eliminated. The fourth subchapter is divided in two parts. In the first part the data is modeled with unsupervised K-means clustering machine learning algorithm in order to find correct amount of cluster and assigning each of the suppliers into their own clusters. The average variables of the clusters are also calculated. In the second part of the fourth subchapter the now cluster wise labeled dataset is divided into training and testing datasets with 4-fold cross validation method. Then ANN and random forest machine learning algorithms are trained with training data and then this trained algorithm is tested with test data to find out which of the machine learning algorithm performs better assigning suppliers into correct clusters. In the last subchapter the results from the analysis are evaluated and the more effective of the two supervised machine learning algorithms is identified.

4.1 Data collection

The landscape of NES operating IT-systems has been highly volatile the last few years because of the merger between Neste Oyj and NES. This means that the data about the suppliers has been spread widely in different systems. The other problem of history as different legal entities in perspective of data is that some of the legacy systems have a lot of duplicate data since the same orders have been in NES project management system and in Neste ERP systems. As there is no one coherent place to look for supplier data it meant that data had to be fetched from multiple systems and storages and combined afterwards.

Purchase order data was the main content of data for the whole thesis. Purchase order to qualify into the analysis had to fulfill multiple requirements. Requirements are following: the status of the purchase order must be finished to contain the full purchase delivery life cycle, the finishing date must be between the set boundaries of 1.1.2016 and 31.5.2019, the activity code of the purchase must be set to technical material and the supplier can’t be internal warehouse or similar supplier since the analysis is concentrating on external suppliers. The qualification requirements for purchase order datapoints is compiled in **table 4**. To be able to apply these filters into the data in systems the extraction must be done in purchase order row level.

Table 4. Qualification requirements for purchase order datapoints.

Requirement	Reason
1. The status of the purchase order must be finished.	Include data from full purchase delivery cycle.
2. The finished date of the purchase order must be between 1.1.2016 and 31.5.2019.	Validate the datapoint for the set boundaries.
3. The activity code of the purchase must be set to technical material.	To filter out other purchases than material purchases.
4. The supplier must be external.	Analysis focuses on external suppliers.

The problem with the order data was that it was duplicated in both legal entities (NES and Neste) systems so a decision was made to use only NES systems to obtain this order data of different suppliers since the metadata inside the order had higher quality and deeper insights. Inside NES two main source systems for order data were identified: Lean 6.2 Enterprise Systems and AVEVA ERM. Lean 6.2 is a legacy ERP system where previously the material management process was handled. Lean 6.2 was also split into two separate operating units of same system when NES was still Neste Jacobs Oy to keep the main customer Neste's data in different storage than the other customers data. The internal names for these two systems were RePro for the Neste data and NoPro for the other client's data which will be referred hereafter. AVEVA ERM is the current system for material and project management where all the current material and services purchase orders are.

Since RePro and NoPro are basically the same system with different storage and access application the data extraction was identical from them. The extract was done from user interface from purchase order row -window with filters described in **table 4**. Extracting purchase order row data from AVEVA ERM was executed from Oracle database with SQL-query with one additional filter added which was that finishing date must be after 1.4.2019 because then we know that there is no duplicate data between legacy and current systems. The total size of purchase order row data table after combining is 23091 rows and 16 columns where the columns contain metadata of each purchase order row such as supplier, quantity, project and total cost.

In addition to purchase order row data NES wanted to enhance the data with two more datasets: list of key suppliers and supplier performance feedback -questionnaire. The key

supplier list was extracted from sourcing platform called Scanmarket and it contained a list of suppliers which have been identified as key suppliers in Neste supplier management process. The only filter applied there was the category of the supplier which was set to technical material. Supplier performance feedback questionnaire is a part of Neste's supplier management -process and it's conducted bi-quarterly via Google forms. In the questionnaire all sourcing and strategic purchasing personnel give feedback of selected suppliers with whom they have made business in the questionnaire period. The questionnaire consists of six statements which are answered on scale from one to six where one means strongly agree and six means strongly disagree. The statements are:

1. Supplier fulfills defined tasks
2. Supplier's key persons are easy to contact
3. Supplier's employees are interested in solving requests or issues
4. Supplier is flexible to deliver changes
5. Supplier's commodities (i.e. products and services) are satisfactory as a whole
6. The co-operation with supplier is on good track

Data collected consists of three different datasets from five different sources: purchase order row table, key suppliers list and supplier performance feedback questionnaire. Total of 38 key suppliers are identified in the key supplier list. The questionnaire includes 42 suppliers and their received averages on each of the statements and their total amount of feedbacks received.

4.2 Data preprocessing

After all the data was gathered, the first action was to combine the data into a single data table. The three different datasets had one common nominator which was the supplier name which made it possible to link the correct supplier in purchase order dataset to corresponding results of the questionnaire and the key supplier list. The datatypes were also checked at this point of the study. All the dates had to be transformed from general text type data into date fields and quantities and values from general text type into numerical data types.

Next part of the preprocessing was cleansing the dataset from data outliers and quality errors which are in the dataset due to human errors. The dataset had many errors regarding the monetary, quantity and date related information which is cause of manual work and multiple

updates in the system. From the monetary side all unit prices below 0,1€ can be set as outliers since standard material list doesn't contain anything that cheap and purchase requisition material is always more expensive. Unit quantities of empty or 0 were removed from the set since there's no information of their correct quantities and without quantities the unit prices are incorrect. Date related outliers such as negative lead times are removed from the set because it is sure that they are data quality errors which shouldn't be in the analysis.

After the data was cleansed it required aggregation in the parts which were fetched from ERP and ERM systems. The aggregation was done from order row level to supplier level while trying to keep all the different order characteristics from one supplier. To enable profit impact point of view investigations the orders, order rows, quantities of procured items, participations to different projects and total spend were summed. Indications for profit impact related variables were calculated as following: the supplier relationship lifespan was calculated as difference between the first and most recent purchase order and purchase frequency was calculated by dividing the supplier relationship length with number of purchase orders. Performance related variables were calculated as averages, medians or standard deviations. Cost related items were calculated in all those three statistical metrics but delivery deviations were taken only as averages and supplier lead time was also taken as average on difference between purchase order release date and receiving date.

The variables for the thesis were gained from three original tables and variables related to supplier profit impact criteria were nro_of_orders, nro_of_projects, nro_of_rows, quan_m, quan_pieces and tot_spend which have straight connection to profit impact. Supply risk variables were related to knowledge of the supplier explained with relationship status and length (key_suppl and purch_time_window) and utilization of supplier (purch_freq) hence decreasing or increasing supplier risk and supplier's negotiation power based on those variables. Variables related to supplier collaboration all came from the supplier performance feedback questionnaire: Fullfills_tasks, Easy_to_contact, Solving_req_or_issues, Flexible_to_changes, Satisfaction_to_changes and Co_op_direction. Fully supplier performance related variables were unit_price_mean, unit_price_median, unit_price_sd, supp_contr_dev, supp_agr_dev and supp_LT. The explanation for each of the variable are provided in **appendix A**. At this point it was finally realized that SRM process and framework couldn't be taken fully into scope of the analysis as there were no variables from external supply risk like suppliers economical or geographical factors. Also, on performance related criteria the quality part was totally missing from the original data, even the questionnaire

questions imply the quality to some extent but are unfortunately just human interpretation of quality and not always the reality.

The last part of preprocessing is handling the missing data. Discarding data with missing values is not a working solution at this point since the amount of missing values in the dataset is relatively high due to nature of the data and the variables lacking values are important to analysis. There are two different kinds of missing values in the dataset left and they must be treated separately. The easier kinds of missing values are just values which aggregated suppliers don't have. These are easy to replace with value of zero and this kind of variables are nro_of_feedbacks, quan_m and quan_pieces.

The other kind of missing values are missing values which indicate performance such as delivery deviation from the contractual date or the questionnaire values. Replacing these missing values with just zero would make the whole analysis biased and false. Inspecting the missing values an assumption can be made that missing questionnaire values are missing completely at random (MCAR) because the different topics on questionnaire at least should be independent from each other and the decision makers giving the questionnaire answers are somewhat unaware of the other data used in study. Even the questionnaire values wouldn't be completely MCAR, the effect from other variables can be assumed low due to nature of decision makers and their disconnection to other data. With this piece of the information the decision to use mean imputation in questionnaire values was made. On missing date-related values the mean imputation was seen fit. The reason to apply mean imputation in all these cases was that mean imputation is simple and computationally cheap to apply. The other reason was that mean imputation performs relatively well also on missing at random (MAR) and missing not at random (MNAR) data when the dependency is low to other inspected or uninspected values. For the suppliers which do not have any recorded variable observations aren't punished for missing information but rather given the mediocre assumption of performance. Well and badly performing suppliers stand out of the data mass with relatively higher or lower scores.

4.3 Feature selection

The goal of the feature selection phase was to find a way to reduce dimensionality of the dataset and eliminate those variables which are too much alike. Many of the variables in the dataset were derived and aggregated from the same values in the original collected dataset

which might result variables with very similar information content. The purpose is to keep all the meaningful variables which represent the variance inside the dataset and hold valuable information to separate different kinds of suppliers from each other. The first part of the feature selection phase was to scale the variables in the dataset to prevent the magnitude of some of the variables to dominate the associations between all variables. The scaling was done with unit variance scaling to scale all the variables to same standard deviation of one so the different variances wouldn't mess the principal component analysis and unit variance scaling doesn't affect negatively to Pearson correlation coefficient (PCC). The reason to choose unit variance scaling was that it's computationally and mathematically simple and was prebuilt inside the used R package.

After scaling the dataset, the feature selection was done using two different techniques together. PCA was chosen for the other technique since it is a common and effective way to reduce the dimensionality of the dataset. PCA is also straightforward and gives valuable information about the variance inside the dataset which helps to pick the correct variables into the final dataset to be imported into machine learning algorithms. A filter-based method for other technique was chosen because when supported by PCA a simpler and less heavy method than wrapper approach was suitable. Filter-based approach also helps scaling the feature selection for much larger datasets in the future since its significantly lighter method than wrapper method computationally. Pearson correlation coefficient was chosen as filter-based approach to choose the variables. Pearson correlation coefficient is efficient and simple to interpret when business-oriented persons such as sourcing personnel try to figure out the reasons some of the initial variables were dropped out. The combination of PCA and PCC is easy and effective way to subtract variables which are highly correlated with each other and preserve the overall variance inside the dataset.

For PCA the starting point was to check that which proportion of variance each principal component explains in the dataset. In **Figure 15** is illustrated Scree plot of each of the principal components and the percentage of the variances they explain. From the plot can be seen that the first three principal components are dominant on the variance explained. The total sum of variance explained is 57,6 % from the first three principal components which isn't enough to describe the data but explains majority of it. After the principal component a clear "elbow" can be seen in plot which indicates that the first three principal components are the most significant in terms of explaining variance. The remaining principal components steadily decrease from 4th principal components' 6,4 % explained to almost 0 % in the 21st principal component.

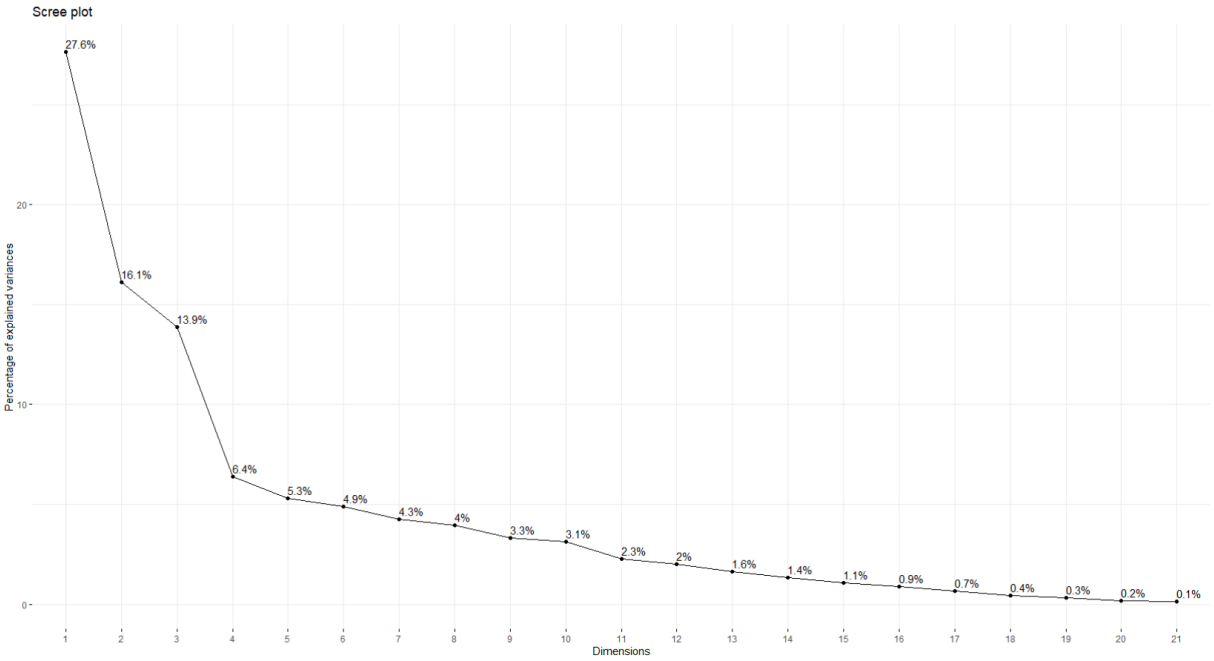

Figure 15. Scree plot of the original dataset.

When trying to find out which variables affect to the primary three principal components it is essential to see how much variables contribute to principal components. Only the three first principal components were investigated even though they account 57,9 % of the variation and not 90 %, as was mentioned in literature review, since the goal was to identify the most dominant ones. In **Figure 16** is presented the proportion of each variable contributing to the three primary principal components. The first principal component mainly consists of the results of the supplier questionnaire as the six major variables are the six questionnaire statements contributing about 10-12,5 % each. The amount of orders, rows, projects and ordered pieces contribute about 5 % each to first principal component. The second principal component consists of more balanced group of variables when nine variables contribute five or more percentages. The major variables here are the numbers of projects, orders, feedbacks and rows with purchase time window, key supplier status, total spend and few questionnaire statements. The second principal component mainly captures the volatility between long lasting and much used relationships versus new and/or minor usage relationships. The third principal component consists mainly of two types of variables: monetary and lead time. The five significant variables contribute over 90 % of the principal component and the unit price mean is the most significant with a bit over 25 % of contribution. The supplier lead time and the unit price standard deviation both are contributing around 17 %, and total spending and the unit price median are contributing around 15 % of the principal component. So over half of

the dataset's variance is explained by the questionnaire variables, usage and order variables and monetary and lead time variables.

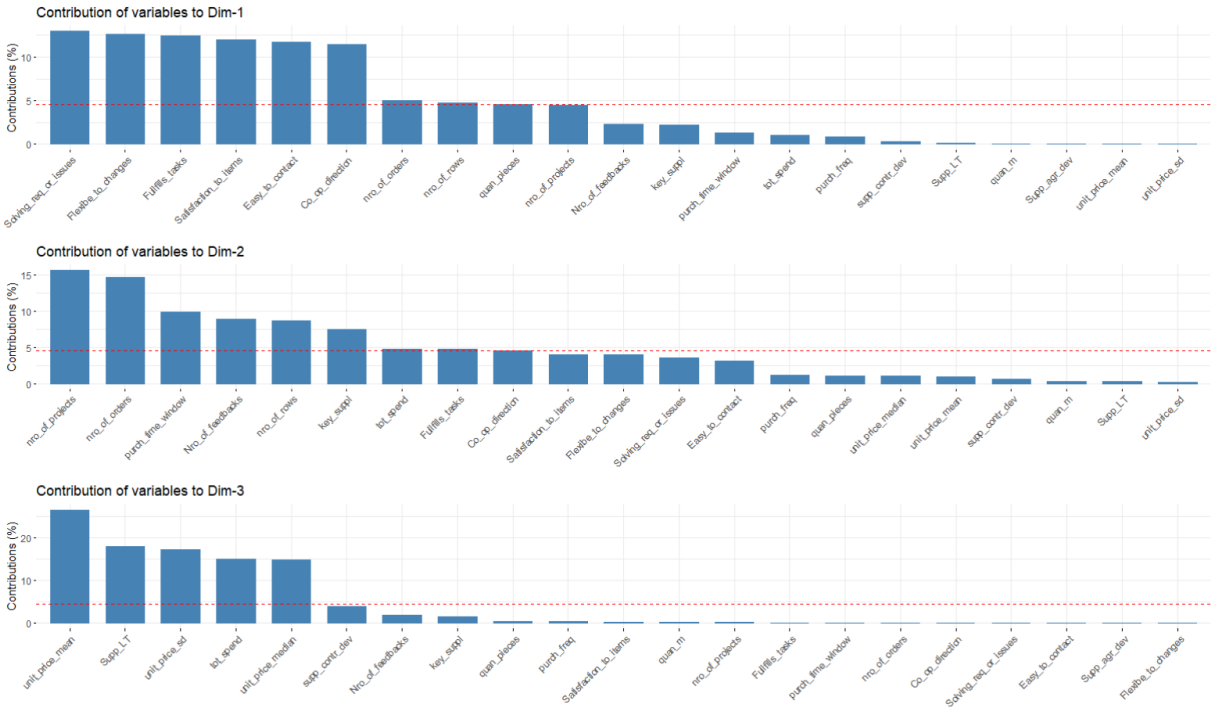

Figure 16. Contribution of variables in the three principal components in the original dataset.

The last part of the PCA was to examine the graphical illustration where one could see the contribution of variables towards the three primary principal components and see approximate correlation between the variables. In **Figure 17** is illustrated the contribution of each variable to the first three principal component axis with color coding and length of arrow and correlation towards other variables. The first observation which can be seen from the plot is that the questionnaire variables all seem highly correlated towards each other. Basically, they tell the same thing of the supplier if the correlation is high. Remarkable is also that there are no negatively correlating variables in the opposite quartile from the questionnaire variables. Almost all the variables correlate a bit with each other but lead time and quantity pieces variables seem to be independent from others. Many of the variables related to order amounts, projects and feedback amounts are clearly correlated with each other which is logical since more orders means more transactions with each other. The major contributors to third principal component can be recognized from the picture as their own little "cluster". These unit price variables and lead time are negatively correlated from order amount related variables and are internally towards same direction. Exception here is the total spending on supplier which is

highly correlated with number of orders and is logical since the total spend grows over time with coming orders.

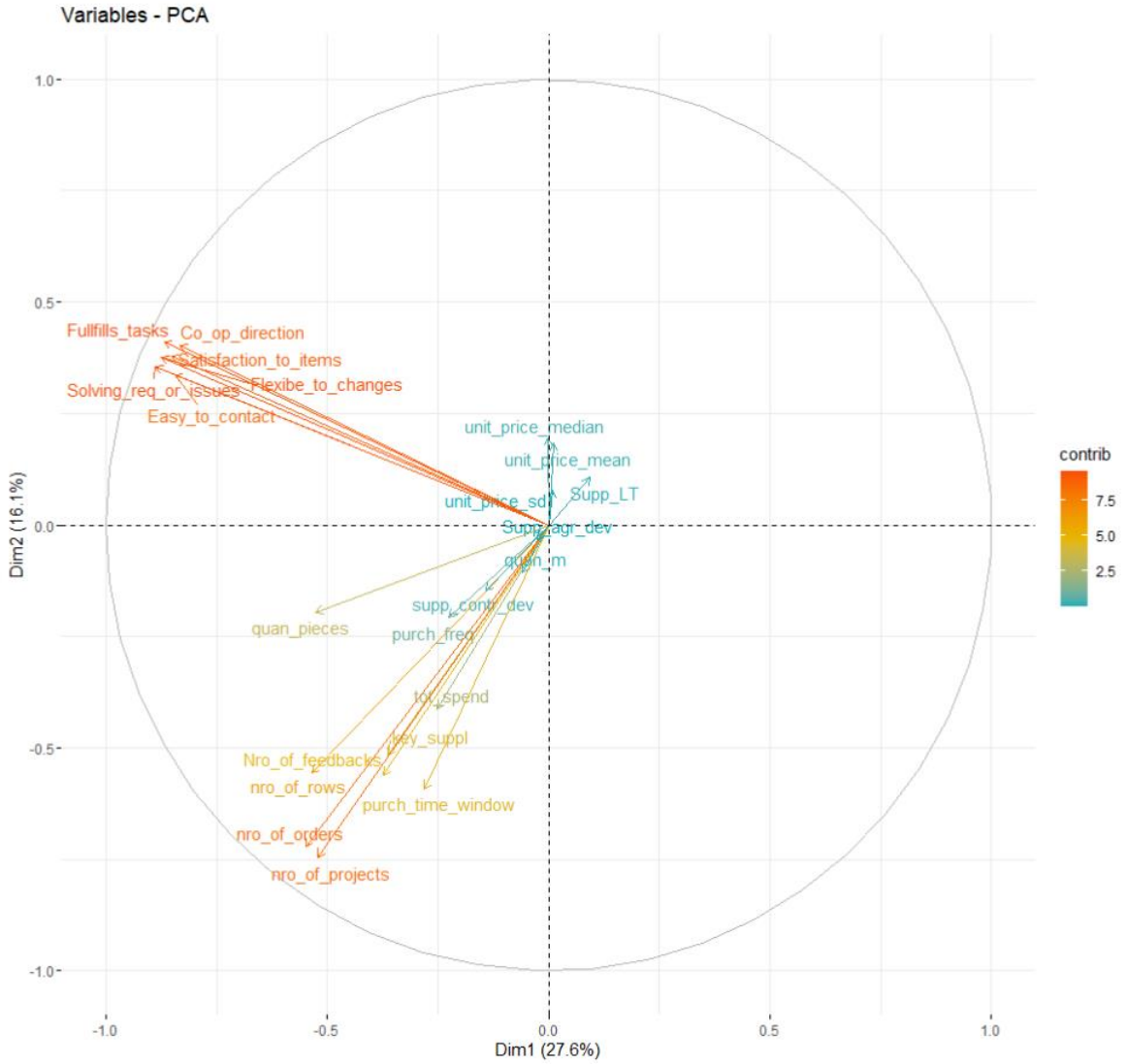

Figure 17. Variable contribution towards two main principal components and correlation directions between variables of the original dataset.

When these approximate correlations were found the PCC fits very well into the analysis. To inspect all the correlations between the variables a correlation matrix was built where the correlation direction was color coded and the value shown. In **Figure 18** is presented PCC matrix for the dataset and all its variables. As inspected in PCA the questionnaire variables are indeed highly correlated when the correlation coefficients for all the occasions are 0,8 or higher. Questionnaire values possess the almost the same information in this dataset based on these highly correlated coefficients. The number of orders and the number of projects variables have almost correlation coefficient of one with value of 0,96 which means that the information value

is almost the same on these two variables. The unit price mean and the unit price median have a correlation coefficient value of 0,8 which isn't too surprising since both variables were derived from the same original variables of the data and calculating these statistical values contain a lot of the same elements. The key supplier variable has a correlation coefficient value of 0,74 with the number of feedbacks variable since the key suppliers are evaluated more often than other suppliers. The number of rows variable correlates highly with the number of orders and the number of projects logically since more orders and projects mean more rows and vice versa. Either way these three variables aren't effective in making distinction with different suppliers together.

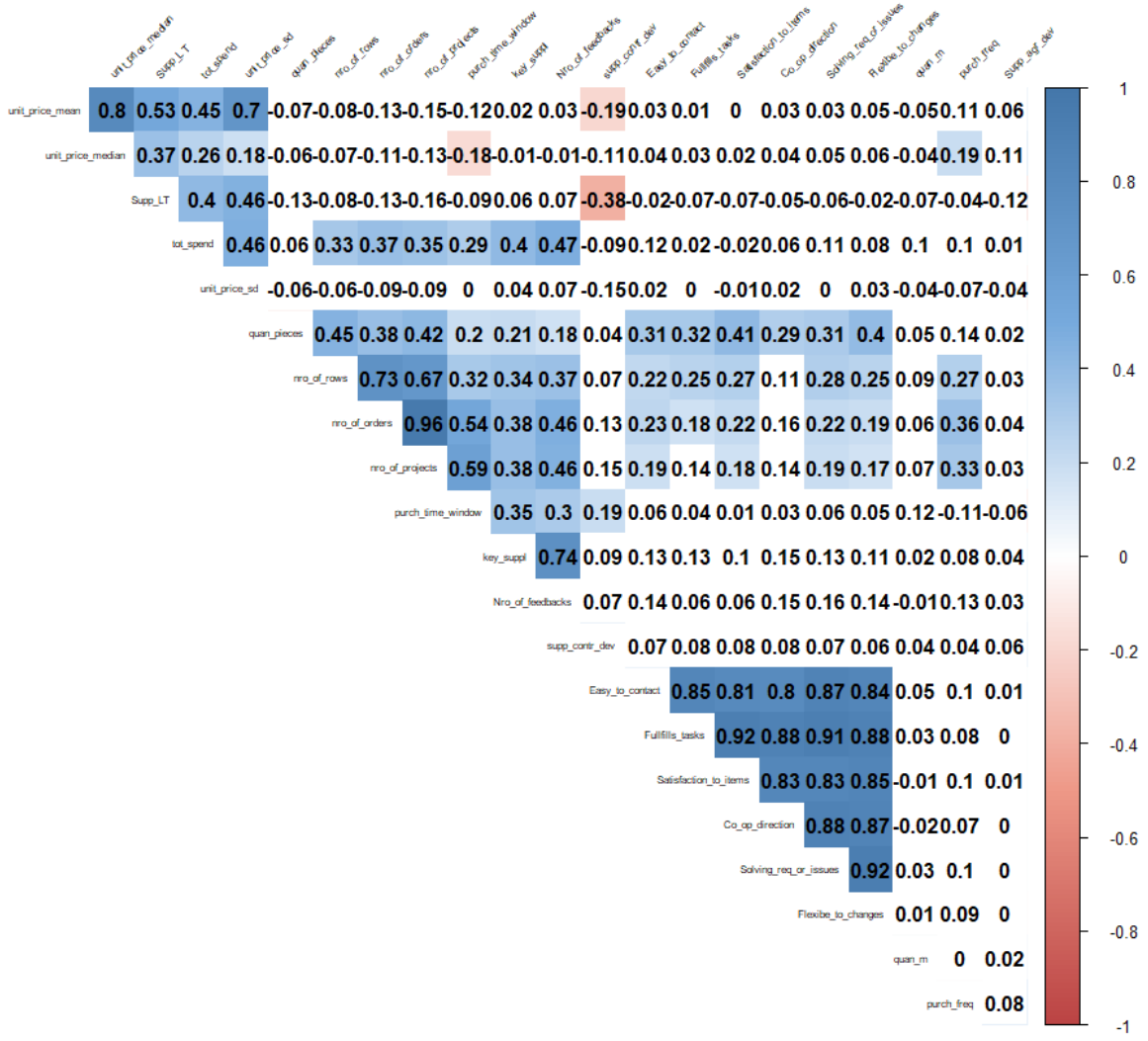

Figure 18. Pearson correlation coefficient matrix for the original dataset and its variables.

In the light of the findings from PCA and PCC some variables were chosen to be either aggregated into one, modified into a new one or just discarded from the dataset. Decision was

made that all the questionnaire results were aggregated into one variable called survey mean which is the mean of statement results. This way the dataset was reduced and multiple variables containing much of the same information could be presented as one. Dealing with variables of the number of orders, number of projects and number of rows was different since just one variable couldn't display all the information the analysis would need. A new variable was chosen to be made and the new variable is supplier order size which is calculated as supplier order size = number of rows / number of orders. This new variable helps to distinct suppliers which have more order rows such as bulk material suppliers from suppliers which deliver smaller batches such as pumps or compressors. After creating the supplier order size variable, the variables of number of rows and number of projects were discarded since the number of rows was now unnecessary and the number of orders display the importance of the supplier better than the number of projects. From the unit price variables, the unit price mean was kept in the dataset and the unit price median discarded. The last two variables discarded from the dataset were the number of feedbacks and the supplier agreed delivery deviation. The reason to discard these variables from the dataset was made with souring specialists since both variables have too much NES internal effect on them. Those internal affects rise from the supplier and order management processes and comparing and judging suppliers with these variables measure more of NES internal process than supplier performance. After this the size of dataset was reduced to 13 variables where two of them are new: the survey mean and the supplier order size.

To check how the dataset changed with reductions, the PCA and PCC are performed again to the reduced dataset to see actual changes. In **figure 19** is illustrated scree plot but now for the reduced dataset. The first "elbow" can be seen after two principal components already. Now the three most significant principal components explain only 49,7 % of the dataset's variance which is almost 8 percentage points lower than in the original dataset. Principal components from four to seven explain from 7 % to 8,3 % of the variance each compared to original datasets range of 4,3 % to 6,4 %. From these inspections it can be said that the dataset is now more balanced than the original dataset and smaller explaining principal components still explain more variance and the dominant explain less variance than in the original dataset. This indicates that reducing the variables has been a good thing for the dataset and variables are more precise.

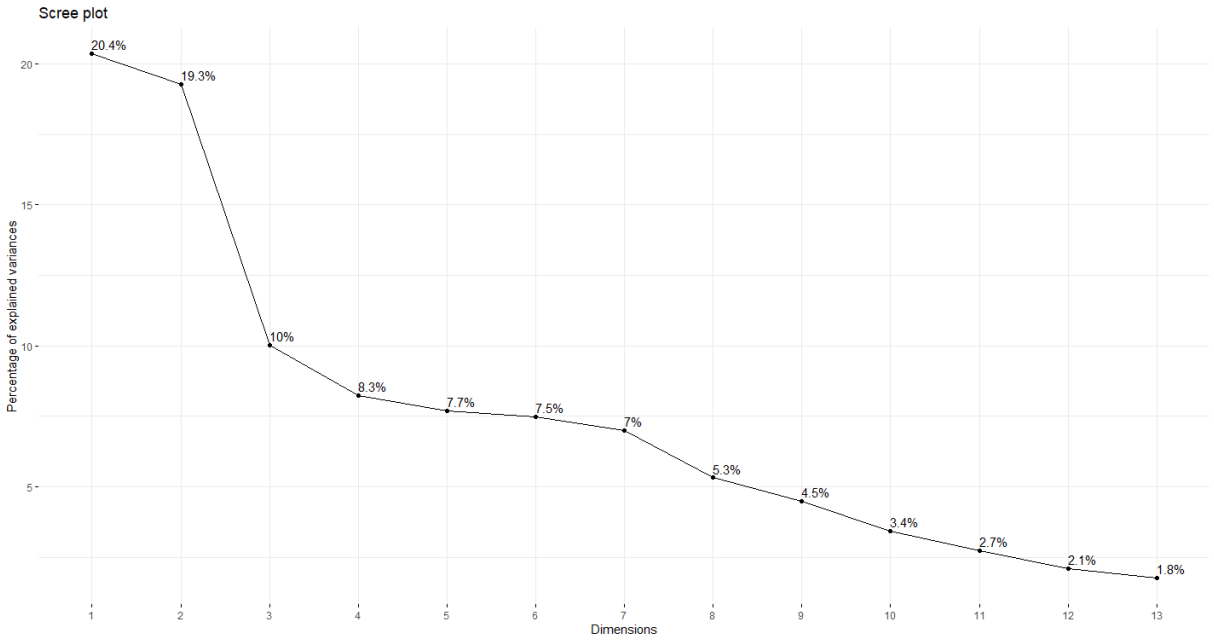

Figure 19. Scree plot of the reduced dataset.

Changes in the three primary principal components are shown in **figure 20** where is illustrated the contribution of each variable for the three primary principal components of the reduced dataset. Now the primary principal components have totally different variables in them. The first principal component consists now mainly from the monetary and lead time variables with each of them contributing over 20 % of the principal component. The second principal component consists of order amounts, purchase time window meaning supplier age as supplier and their key supplier status variables mostly. The number of orders here is clearly the most dominant with proportion over 25 %. The other three significant variables contribute around 15 % each. The third principal component consists now from multiple different variables. The two most contributing variables are the purchase time window (~22,5 %) and the purchase frequency (~20 %) followed by the survey mean (~18 %), quantity pieces (~15,5 %) and supplier order size (12,5 %). The third principal component consists of different types of variables which is good since gaining more explanation value for every variable doesn't make the analysis skewed to the direction of just some variables. A positive change from the original dataset is that now the most explaining principal component is based on monetary variables instead of questionnaire statements because the questionnaire results might be a bit biased since they are generated by humans instead of raw monetary data which tells how much we use the suppliers. The monetary element is also one of the most significant elements in the supplier relationship management process since the suppliers which have a lot of spend tend to be more important suppliers and they must be managed differently. Those monetary

elements combined with the number of orders from the second principal component gives already a good view about the usage and money spent on different suppliers.

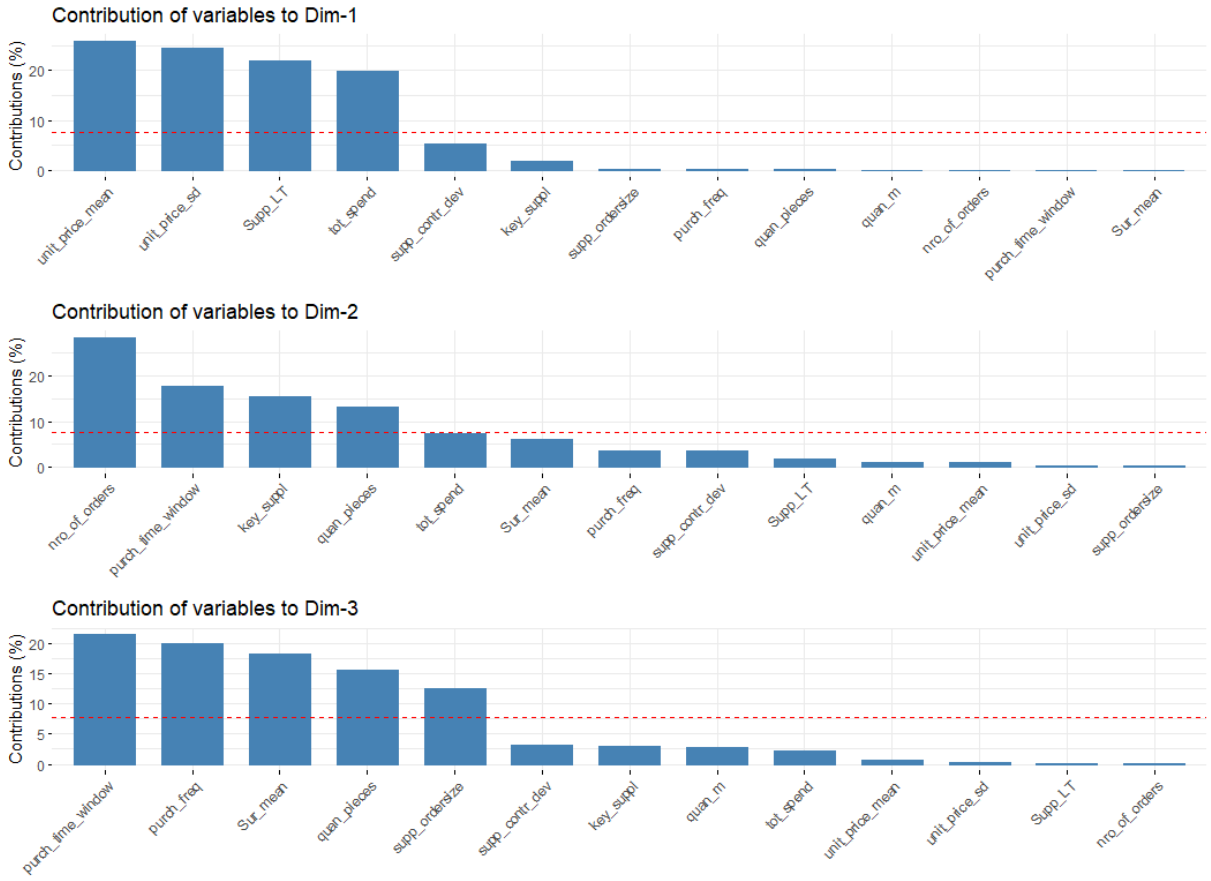

Figure 20. Contribution of variables in the three primary dimensions in the reduced dataset.

Next the correlations are inspected with the reduced dataset on how they have changed and is there something still to correct. In **Figure 21** is illustrated the contribution of each variable to two primary principal components with color coding and length of arrow and correlation towards other variables in the reduced dataset. The change to the original illustration is clear. Now there can be seen again two main directions but there are a lot of variation and variables which aren't so clearly into the main two directions. The monetary and lead time variables still create a small cluster, but their correlations aren't so strong anymore. The other main direction consists of multiple variables with the number of orders as the most significant, but the others have visible deviations from its direction. Overall, this presentation is more balanced than the original even though it could be even more distributed.

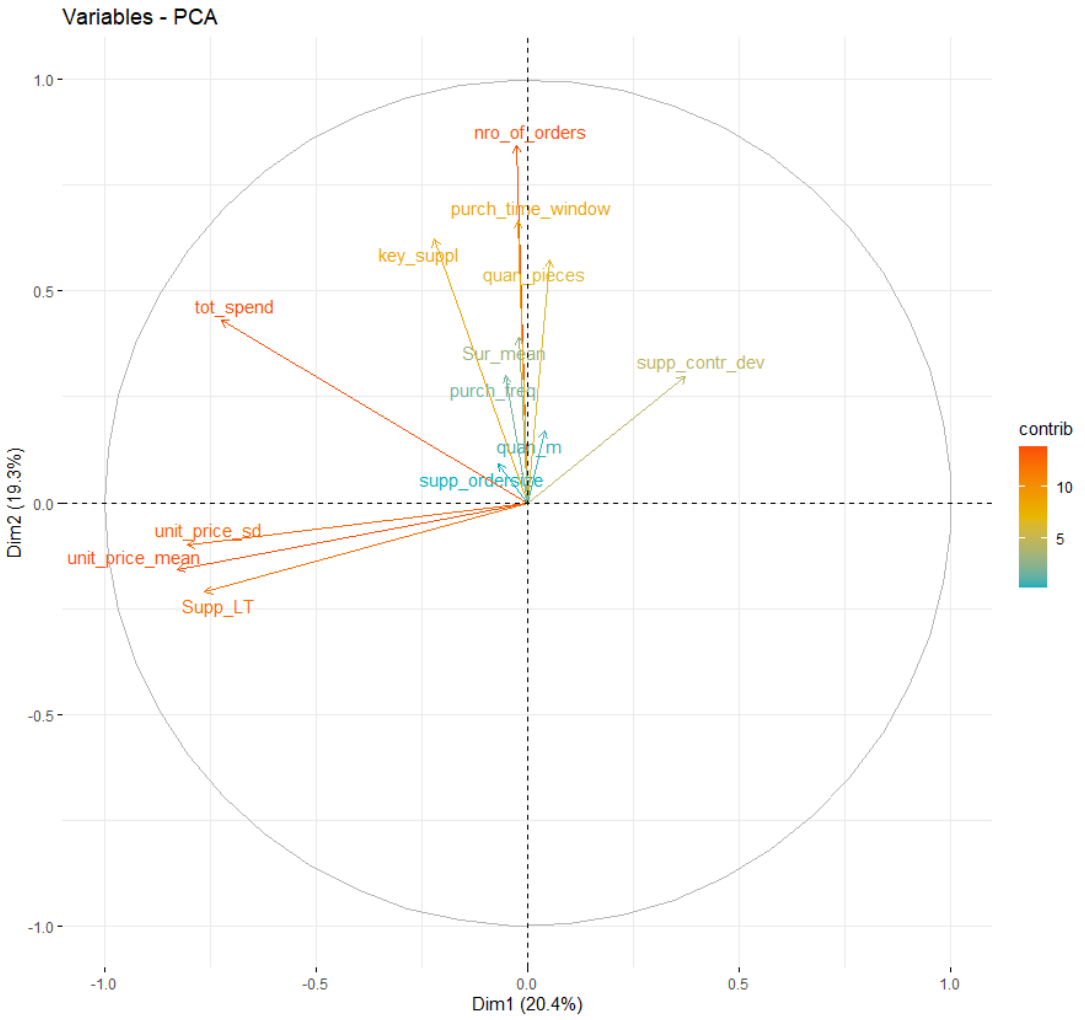

Figure 21. Variable contribution towards two main principal components and correlation directions between variables of the reduced dataset.

The last thing was to check that no high correlation coefficients remain in the dataset. In **figure 22** is presented PCC matrix for the reduced dataset and all its variables. Looking into the matrix all the high correlations have taken care of except the correlation coefficient of 0,7 between the unit price mean and unit price standard deviation. These two variables must still be preserved since the combination of these two variables have the information about the prices and price volatility of the orders. No other variable correlation coefficient exceeds 0,55 mark which is a good result compared to the early expectations. It is also great to see the variables which are basically independent from all the other variables in the dataset such as the supplier contractual delivery deviation and the supplier order size. The correlation directions also seem very logical. A good example is supplier contractual delivery deviation which is the only variable having stronger negative correlations which are explained by the nature of the variables. If the lead time grows then the bigger is the negative value of days delivered early. The same logic

applies with the unit price mean as more expensive and complex the product and delivery are, the less likely it is delivered early.

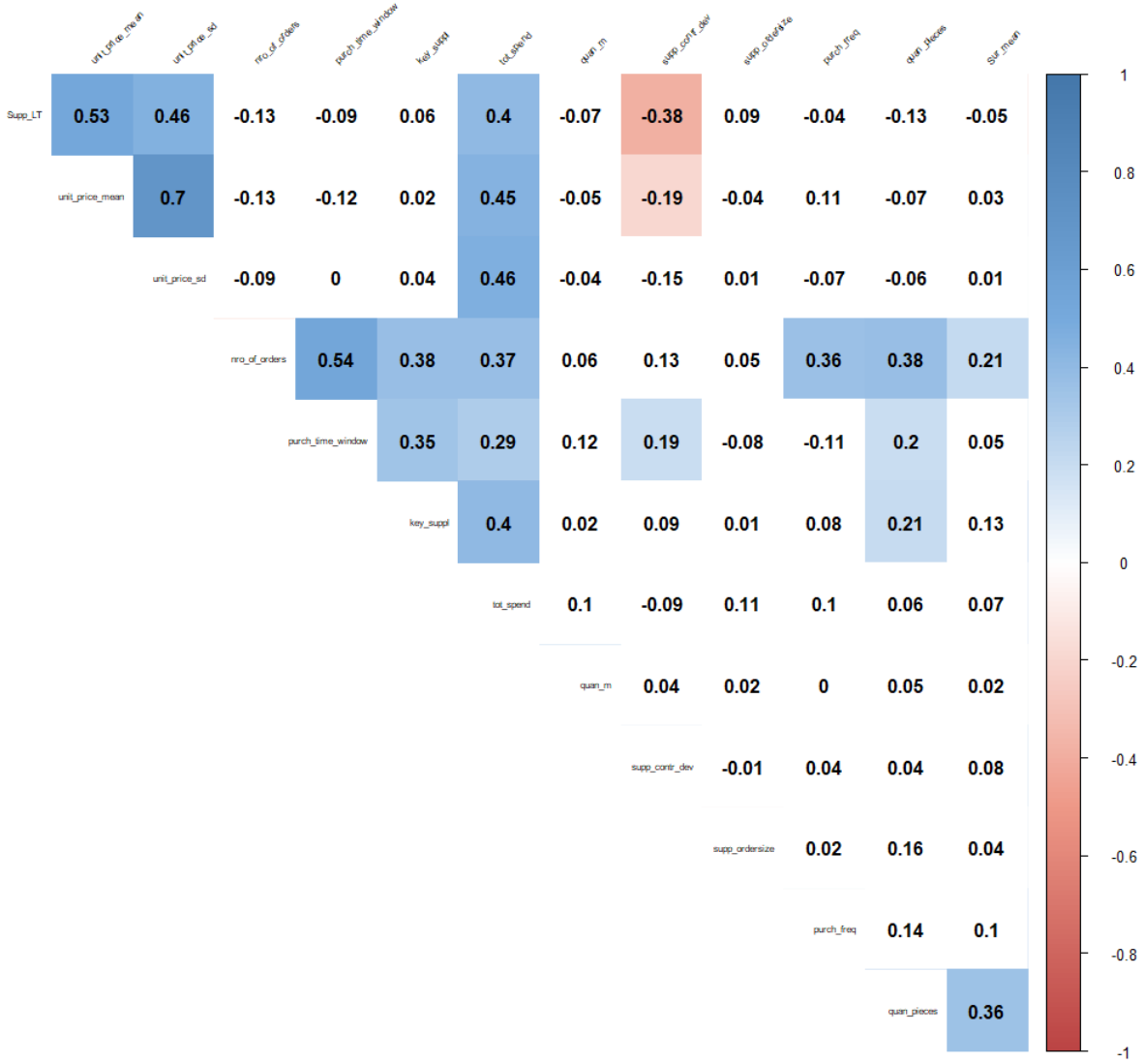

Figure 22. Pearson correlation coefficient matrix for the reduced dataset and its variables.

The reduced dataset is now ready for the data modeling phase. It has no considerable correlations between the variables and the variance explained in the dataset is distributed well enough through the variables. Reducing the size from the original 22 variables into 13 variables makes the machine learning algorithms easier to conduct decreasing the need of computational power and time.

4.4 Data modeling

In the data modeling phase two primary tasks are performed: divide the suppliers into coherent clusters and teach predictive algorithms to predict correct cluster for test suppliers and find out the most effective algorithm to perform the predictive clustering. The first part, unsupervised method, tries to extrapolate algorithmic relationships of the variables in the dataset and find groups which have characteristic values on variables creating a group of similar suppliers. The second part focuses on validating the correct hyperparameters for chosen supervised machine learning algorithms and then evaluating the results on chosen algorithms in order to choose the most effective method to predict supplier classes.

4.4.1 K-means clustering results

Choosing the unsupervised algorithm for the task at hand was straightforward. K-Means approach was chosen for its simplicity, availability in different packages in R and illustration purposes. The data is all numerical in the dataset and no categorical data exists which makes the usage of distance-based measures effective in clustering which K-Means utilizes. However, distance-based approach requires dataset standardization or normalization because without normalizing the values the weights on different variables may be biased since bigger values have more variance. To prepare the data for the K-Means algorithm it was scaled from 0 to 1 scale to avoid dominance of variables with high values.

Problem with K-Means clustering is determining the quantity of clusters and finding out cluster centers. Prior using the algorithm itself the analyst must give the number of clusters used. To determine the correct quantity of clusters three different methods were used: Elbow method, Silhouette width and Calinski-Harabasz index. The objective in K-Means algorithm is to find clusters similar inside and dissimilar with each other, and these methods assess those criteria. Elbow method presents the total within sum of squares (total WSS) of each cluster which means it assesses similarity of intra-cluster variation and the smaller the total WSS is the better the clustering result in theory is. Problem with elbow method is that it doesn't take in account inter-cluster variation and theoretically "the best" total WSS would require as many clusters as there are objects to cluster. Silhouette value and Calinski-Harabasz index were chosen because they also assess inter-cluster variation as part of their criteria and object is to maximize the values to obtain the best result. Silhouette value and Calinski-Harabasz index have an intra-cluster and inter-cluster components in the algorithm.

The analysis was conducted with calculating relevant values for each of the method in cluster quantities between two and ten. Ten clusters were considered from the NES side the absolute maximum for different clusters to be handled as supplier portfolios. The results of the cluster number determination methods are illustrated in **figure 23**. Usage of two clusters is also prohibited since the practical need for clustering suppliers must have more than two clusters. Having two clusters as supplier management portfolios would lead into too generic approach for diverse suppliers which would lead into inefficiencies hence using two clusters would be unwise. Cluster quantities of three, four and five are then left. The results are ambiguous since each of the method propose different quantity of clusters to be mathematically "the best" one. In elbow method there is no clear "elbow" to be seen and the line itself is close to linear. Moving to four clusters the change is notable but after that the changes are minor. There is a slight enhancement from six clusters to seven, but it is almost the same as from four to five clusters and nowhere near to first two enhancements. If there's an "elbow" to be found, then it's between clusters three and four. The silhouette width tops on two and three clusters and then drops significantly although five clusters have some improvement compared to four, but the others have low widths. Calinski-Harabasz index tops also on two clusters but then dives deeply in three clusters and gives relatively good values on four and five clusters before dropping again. Even though elbow method gives mathematically the best results on the high number of clusters with observations from these methods the clusters from six to ten can be ruled out. Silhouette method has a high value on three clusters, but elbow method shows significant improvement from three to four clusters and Calinski-Harabasz performs poorly with only three clusters. For practical use also three clusters would be too general to enable accurate enough approaches towards the suppliers. Four clusters have improvement in elbow method but Silhouette and Calinski-Harabasz are higher but not as high as with five clusters. The chosen quantity of clusters is five since both Silhouette and Calinski-Harabasz have relatively high value and elbow method has slight improvement also there.

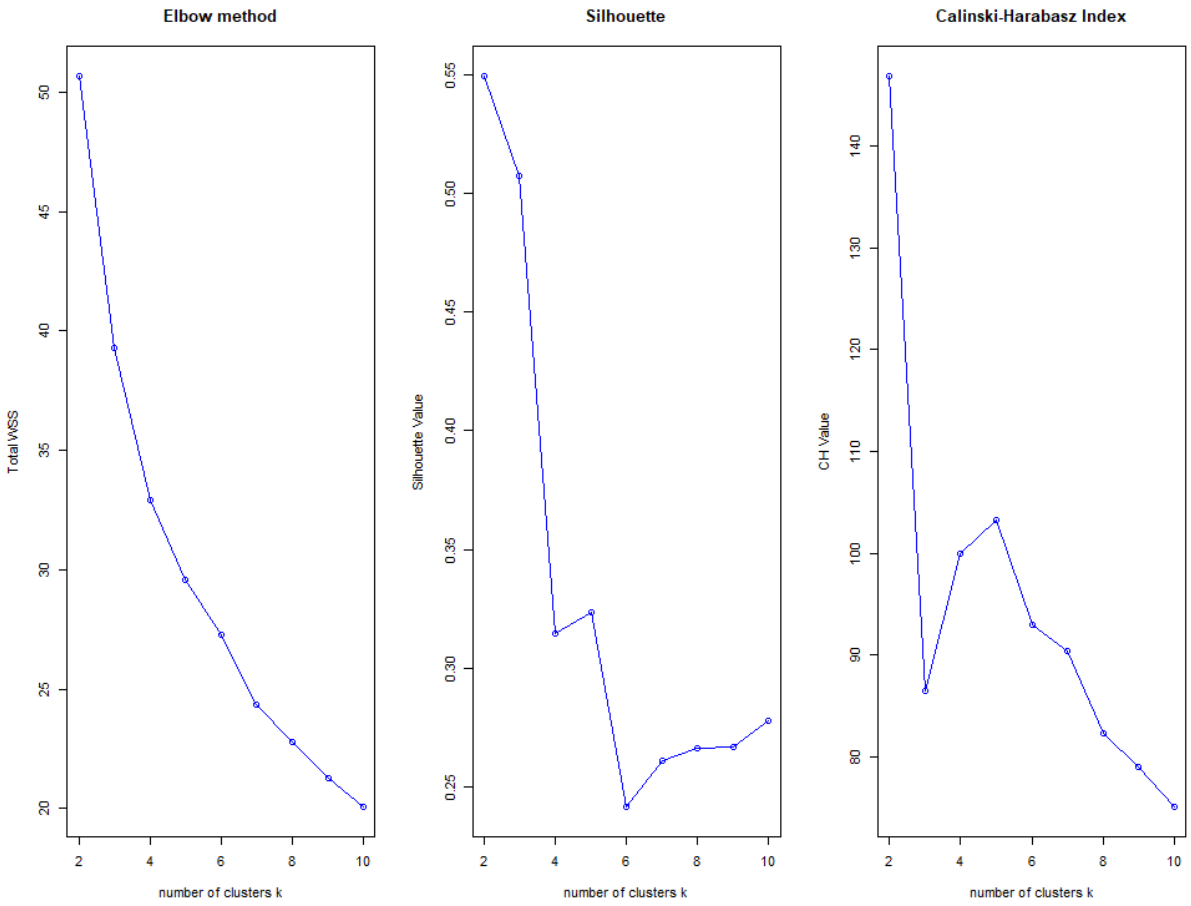

Figure 23. Cluster number determination methods and their results.

With the number of clusters decided the K-Means algorithm was performed with scaled data, five clusters and 25 random starting sets. The characteristics of each group are presented in **table 5** where mean of each variable for corresponding group is calculated and total number of suppliers of each group is presented. From the **table 5** can be seen that all 32 key suppliers have been divided strictly into clusters three and four since their mean is one and mean of other clusters is zero. The key supplier status has been one of crucial elements determining the clusters since the clusters with key suppliers are significantly smaller than the clusters without key suppliers. Clusters one and five are the biggest clusters supplier wise with 86 suppliers in cluster one and 89 suppliers in cluster five. The diversity between different suppliers inside these clusters might then vary more. The 2nd cluster is very dissimilar from the others. It has very high unit price values, considerable long lead times and no items which would be measured in metrical terms meaning these suppliers supply no pipes or cables at all. Problem is that the clusters except for 2nd cluster are looking somehow similar. In **figure 24** can be seen that in the first two principal components almost all clusters except the 2nd cluster

are highly overlapping. Visualizing in first two principal components is problematic because they explain only 39,7 % of the variance so much of the information is left out. This is problem for high dimension datasets where the variance of the data is almost evenly split. For these reasons it is reasonable to examine each of the clusters alone with comparisons to chosen clusters to figure out strict characteristics and explanations for each cluster.

Table 5. Cluster centers of each variable of each group and number of suppliers per group.

Variable name	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
Key_suppl	0	0	1	1	0
Nro_of_orders	3,83	2,43	93,17	18,9	22,96
Quan_m	721,06	0	5839,51	537,43	3563,23
Quan_pieces	236,29	236,79	2337,37	2342,84	695,61
Tot_spend	120629,3	1581957	3942148	662591,1	450483,2
Unit_price_mean	7585,34	137012,3	40267,15	23229,68	9179,52
Unit_price_sd	7884,35	142827,2	58847,47	34556,8	13980,46
Purch_time_window	293,19	298,5	1251,42	874,05	978,64
Purch_freq	0,03	0,04	0,07	0,02	0,02
Supp_contr_dev	-12,16	-70,56	-9,6	-8,96	-14,86
Supp_LT	65,69	249,84	124,43	104,14	80,64
Sur_mean	4,54	4,57	4,71	4,59	4,57
Supp_ordersize	4,98	8,61	6,56	4	3,46
Nro_of_suppliers	86	28	12	20	89

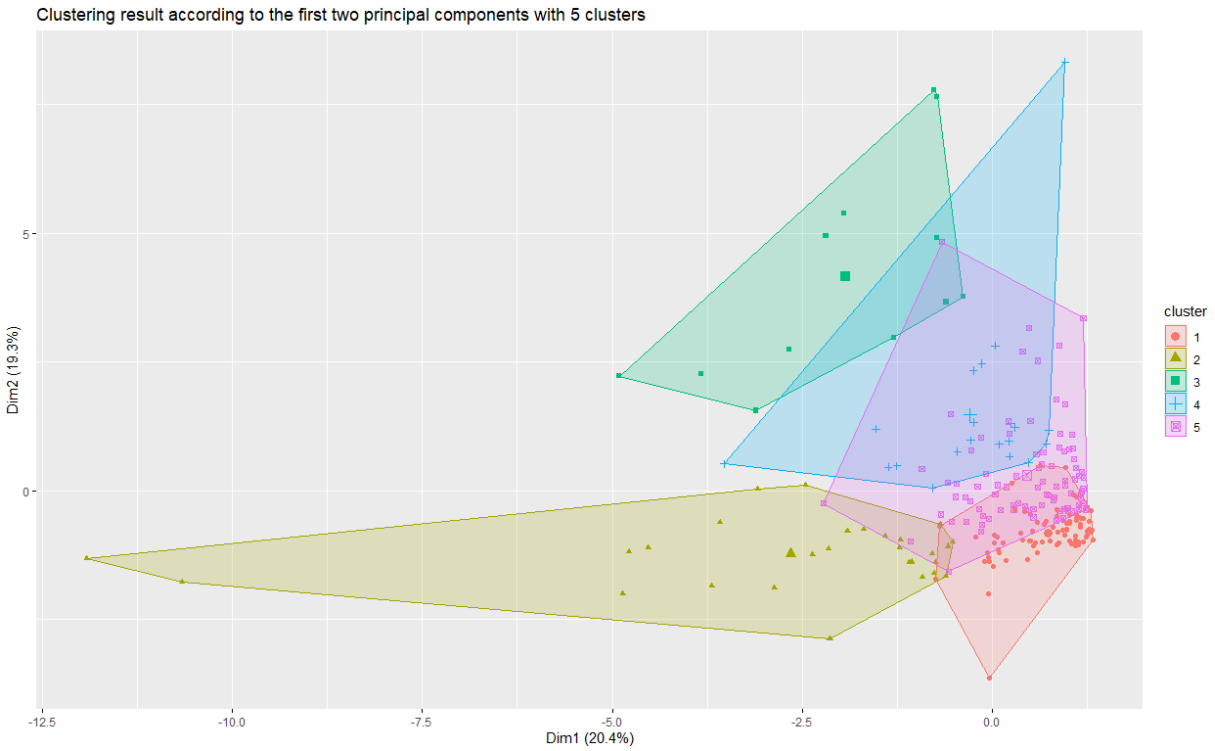

Figure 24. Clustering result according to first two principal components with 5 clusters.

Starting with the key supplier clusters three and four, the biggest differences can be found from the total spend and number of orders, quantity of meters and purchase time window. In **figure 25** is one scatter plot illustrating the total spend on x-axis and the number of orders in y-axis and another scatter plot illustrating the total quantity of meters on x-axis and the purchase time window on y-axis for clusters three and four. The first plot of the figure shows undeniably that key suppliers with high total spend and high total amount of orders belong to cluster three and all cluster four suppliers have grouped into the lower left corner of the plot. Within these two variables the cluster four is very similar group having lower amount of orders and total spend but cluster three has two kinds of key suppliers: ones with high amount of orders and total spend and ones with low amount of orders but still notably high total spends. The second plot shows that in cluster three all the suppliers have longer lasting relationship with NES. In the fourth cluster the purchase time window length isn't similar within all the suppliers, but it separates it from third cluster since the third cluster has always long-lasting relationships where in fourth cluster it doesn't matter. The difference in the mean metrical quantities basically comes from two cluster three key suppliers which supply massive amounts of items which are calculated in meters such as pipes or cables. One interesting notation is that only few key suppliers are supplying cables or pipes since they are a fundamental part of creating the production plants. Another interesting metric is that there are couple of key suppliers in the

fourth cluster which have surprisingly low total spend. Idea of having a key supplier which isn't used is odd but there might be change of suppliers' name behind this kind of oddity.

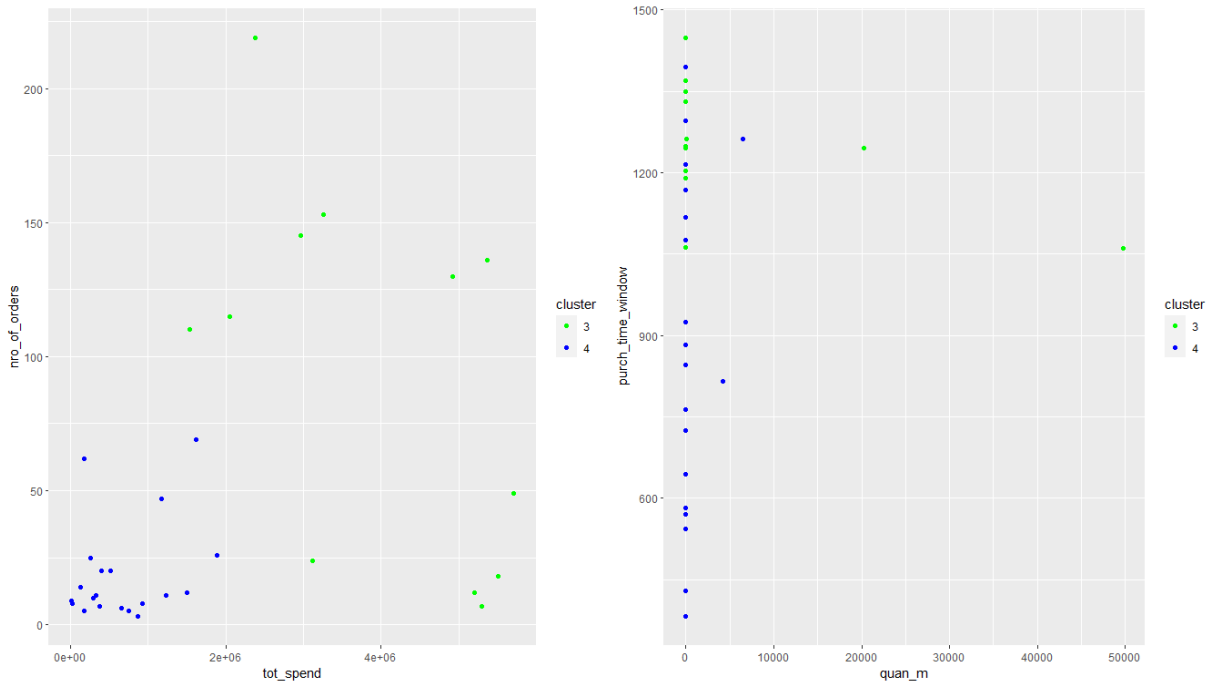

Figure 25. Total spend, number of orders, quantity of meters and purchase time window for clusters three and four.

The 2nd cluster is the easiest one to differentiate from the others. 2nd cluster has with no doubt the most expensive unit prices and lead times among the biggest supplier contractual deviation and highest supplier order size. In **figure 26** in the first plot is visualized the unit price mean and the supplier lead time variables and in the second plot is visualized the total spend and the number of orders with all the clusters. From the visualization can be seen that the 2nd cluster has high lead times which usually lead to contractual deviations. The other typical characteristic of 2nd cluster is relatively high unit prices and the highest unit price suppliers can be found from this cluster. The only other suppliers which have a unit price over 100'000 are key suppliers meaning that all suppliers which aren't key suppliers having unit price over 100'000 belong to cluster two. Long lead times and high prices indicate to massive and critical items used in production plants such as processing columns, tanks, compressors and pumps. The other thing indicating towards these important items is that this cluster hasn't at all items measured in meters and a relatively high order size with low amount of orders. Usually suppliers supplying critical plant items have very few orders with lots of equipment inside them. None of the cluster two suppliers have many orders but there is a significant quantity of

suppliers with proportionally high total spends. This high value of total spends on these suppliers without them being a key supplier is questionable since some of these go even on top-10 suppliers in that category. This high spend should be managed with more care and concentration than just normal suppliers.

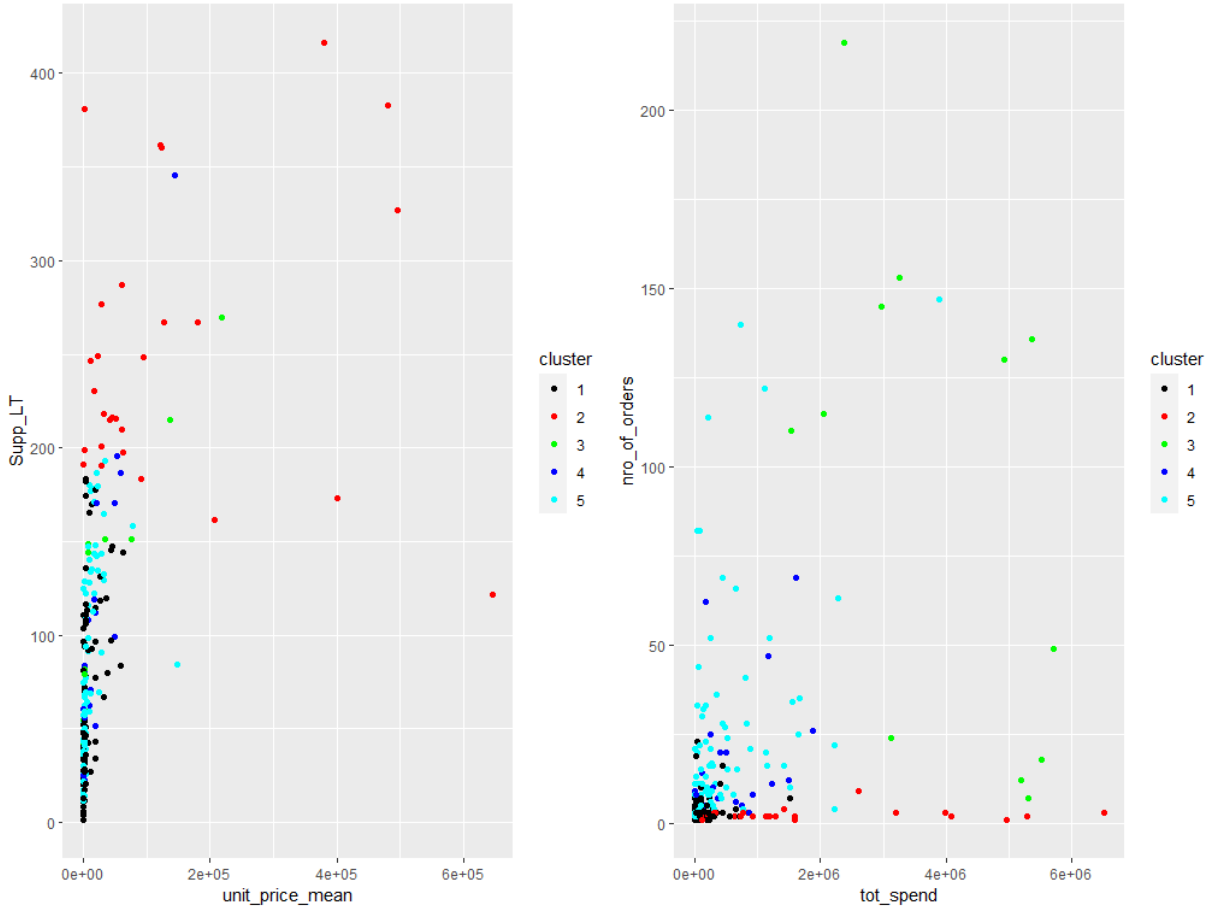

Figure 26. Unit price mean, supplier lead time, total spend and number of orders of all clusters.

The last two clusters, one and five, are the most difficult to differentiate from others and especially from each other. These two clusters are also the biggest clusters with 86 and 89 members in them. Between those two clusters one variable has immense power to create separation which is purchase time window. In **figure 27** is in the first plot is presented the purchase time window and the number of orders and in the second plot is presented the total spend and the unit price mean for clusters one and five. From the first scatter plot a clear division can be seen: when the purchase time window is around 650 there is clear change which separates the 1st cluster from the 5th. This is the only clear separation which shows in the plots. When looking the plots in **figure 27** and **figure 24** the clear second difference between clusters one and five can be drawn, as the 1st cluster has very low variance internally

compared to 5th cluster. The variation between different suppliers in the shown variables is notably small leading to high intra-cluster similarity. Key characteristics of the first cluster are small purchase time window, small total spend, small unit price mean and standard deviation. Cluster one represents small total spend suppliers which haven't been used for long. This kind of suppliers are the ones that should be eliminated from the supplier palette and replace their supply with those suppliers which supply similar items and are used more often, if it is possible, to leverage economies of scale. The fifth cluster is the most confusing of the clusters since it really doesn't have vivid characteristics to lean on. From **figure 26** can be seen that the number of orders and spend wise the fifth cluster is rather similar with fourth cluster. The thing is that fourth cluster has a lot higher unit price means and standard deviations where fifth cluster has small to mediocre values on those variables. Noteworthy is that the fifth cluster has proportionally higher average value on items measured on metrics rather than on pieces which means that suppliers supplying pipes and cables on greater quantities and are not key suppliers are in this cluster. The fifth cluster composes of suppliers which have a longer relationship with NES as suppliers and are not key suppliers.

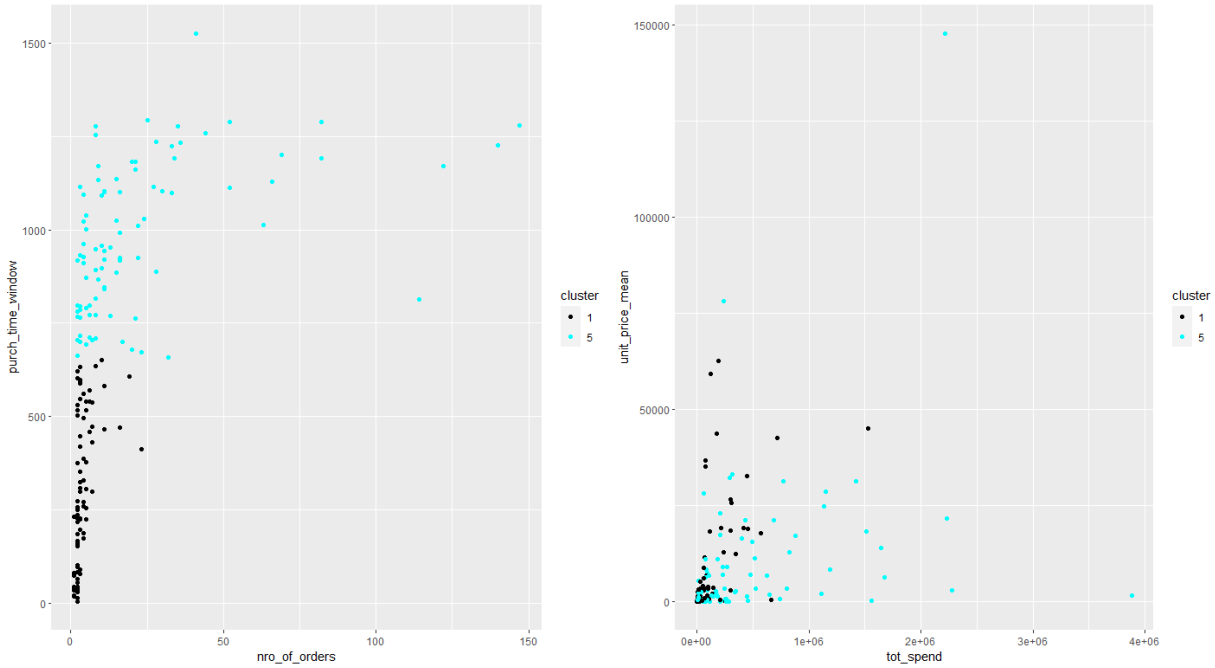

Figure 27. Number of orders and purchase time window and total spend, and unit price mean for clusters one and five.

To understand better the characteristics of each cluster a linguistic description was made where the different variables are evaluated. The key variables are evaluated with either binary or three step approach. Three step approach contains values low, mediocre and high. These

linguistic characteristic descriptions are compiled in **table 6.** The first cluster scores low on almost everything meaning it has suppliers with low importance also for NES. The goal for NES sourcing would be getting rid of small suppliers and develop longer lasting relationships with suppliers who can supply the same items but only with larger pool of items or greater volume. Second cluster has few orders but with high value of items and total spend. Suppliers in second cluster are important suppliers even though they are not key suppliers. They should be considered to add to key suppliers list since their total spend is high and they produce critical items for NES business. Third cluster is a key supplier, has high values of orders and quantities and mediocre values on prices and no low values. These are the key suppliers on basic and regularly needed items since they have a high number of orders and they supply continuously and with high frequency the business of NES. Relationships with these suppliers should be enhanced and more focus and mass on the orders should be added. Fourth cluster has mediocre values of orders, mediocre values of prices and spend and low values on purchase frequency and supplier order size. The fourth cluster contains few suppliers which are indeed remarkable, and the relationships should be maintained with them. Some of the suppliers in the fourth cluster should be checked and potentially rethought with their key supplier status since they have very minor impact on the whole supplier portfolio and shouldn't receive as much attention as the other suppliers greater in volume and spend. Cluster five has a high value of metrically measured item suppliers, mediocre quantity of orders, spend and purchase time window and low unit prices. This cluster contains many interesting suppliers and should be inspected by the sourcing organization to obtain the maximal benefit out of them. There are many suppliers which could be thought as key suppliers given their volume of orders and spend. Fifth cluster has unfortunately very low cohesion between the suppliers so it will require effort and time to pinpoint the correct suppliers.

Table 6. Linguistic descriptions of variables on each cluster.

Variable name	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
Key_suppl	No	No	Yes	Yes	No
Nro_of_orders	Low	Low	High	Mediocre	Mediocre
Quan_m	Mediocre	Low	High	Mediocre	High
Quan_pieces	Low	Low	High	High	Mediocre
Tot_spend	Low	High	High	Mediocre	Mediocre
Unit_price_mean	Low	High	Mediocre	Mediocre	Low
Unit_price_sd	Low	High	Mediocre	Mediocre	Low
Purch_time_window	Low	Low	High	Mediocre	Mediocre
Purch_freq	Low	Mediocre	High	Low	Low
Supp_contr_dev	Mediocre	High	Mediocre	Mediocre	Mediocre
Supp_LT	Low	High	Mediocre	Mediocre	Mediocre
Sur_mean	Mediocre	Mediocre	High	Mediocre	Mediocre
Supp_ordersize	Mediocre	High	Mediocre	Low	Low

4.4.2 Classification results

The goal of the supervised methods is to find as effective algorithm as possible to classify the suppliers into correct clusters. This way the NES procurement unit could enhance their supplier management process and make the classifying of new suppliers or old suppliers with changed supply behavior even faster. The machine learning algorithms chosen to do this task were artificial neural network (ANN) and random forest (RF). ANN was chosen for its increased popularity and high capabilities of even complex classifying problems. Even though it's a computationally heavy algorithm to use the renewed data managing and reporting pipeline of NES has capabilities to implement solutions like neural networks. Random forest is great for classification issues in visualization purposes when the user can pull out any of the trees inside

the model and see how for example the algorithm could choose the class of the instance. This way the random forest isn't a total black box and gives opportunities for sourcing organization to think and tune the key variables used in supplier evaluation.

First the dataset was split into training and test sets in order to teach the supervised algorithm. The training dataset was used to teach and select the machine learning algorithms and their hyperparameters, and the training set was used to select the last model to be chosen as the best from the chosen machine learning algorithms. Problem with this dataset was the low quantity of observations of some clusters. The training and testing datasets were formed by splitting the original dataset with 65 % of the data and 153 observations going into training set and 35 % of the data and 82 observations going into the test set. The reason splitting data like this was that the data had only a few observations from 3rd and 4th cluster, so this split ratio was done to ensure that also the test data has enough observations from those clusters to be classified. Training set had eight observations (5,2 % of all the data) from third cluster and 13 observations (8,5 %) from fourth cluster which was sufficient for evaluating purposes and the corresponsive values in test set were three (3,7 %) and seven (8,5 %) observations. Training set must have more observations from the smaller clusters so the model selection and validation could be done because in k-fold cross-validation method the training set itself is still split further and those sparse observations should be represented in every fold.

In model selection and validation, a 4-fold cross-validation method was used. Usually k-fold cross-validation is performed with at least five or even ten different folds but in with this data it was impossible due the scarcity of observations in cluster four. In this method the training data was split into four equal sized subsets where each of the subsets acted as validation data for each of the folds. This means that each of the subsets were used once as validation data. 4-fold cross-validation process was conducted for both of algorithms, artificial neural network and random forest, and for different hyperparameter options in both methods. Since the idea of the algorithms is to classify the suppliers into correct cluster accuracy was chosen as the most important measure. Accuracy defines how many instances the algorithm classifies correctly which is the reason behind choosing it. Another measure was taken in account which was true positive rate (TPR) to allow evaluating from other point of view also. TPR evaluates how well the algorithm classifies the corresponsive instance into correct cluster so the higher TPR is the probability of detection of the right cluster. Accuracy is the primary measure to lay decisions on but if accuracies are very close on different hyperparameters and there is a notable difference in TPR then it is taken into consideration in the decision. After the

hyperparameters of classifier algorithms are chosen and algorithms validated, the finalized results are evaluated on test data set and the most efficient classifier algorithm chosen.

For the neural network a neuralnet -named package of R was used. The default hyperparameters of the package were used except in the hidden layer architecture and activation function meaning that learning rate, error function threshold and maximum epochs were left as default. The activation function was set to hyperbolic tangent since it is zero centered. The hidden layer architecture varies from one to three layers and from three to seven neurons on each layer and nine neurons were tried in single layer. A total of 17 different setups of the network were evaluated in terms of number of layers and number of neurons in each layer. Single layer architectures perform relatively well in terms of accuracy except the neural network with just three neurons which scores almost 30 percentage points lower than the other single layer neural networks as shown in **table 7**. On the other hand, the neural networks with three hidden layers all perform poorly in terms of accuracy in classifying under 50 % of the suppliers correctly. Three hidden layer architecture achieves in terms of TPR with highest values of 94,5 % and 95,8 %. The highest values of accuracy can be found from architecture with two hidden layers. The clearly highest value is in architecture with 7 neurons in first hidden layer and 5 neurons in second hidden layer with accuracy of 86,6 %. This hidden layer architecture has also the highest TPR value of 85,8 % if the TPR values from three hidden layer architectures aren’t taken in account.

Table 7. The results of the neural network classifier in the terms of average accuracy and TPR.

Hidden layer architecture	Accuracy	TPR
3	0,453	0,648
5	0,747	0,792
7	0,755	0,792
9	0,774	0,840
3 3	0,429	0,719
3 5	0,721	0,797
3 7	0,804	0,853
5 3	0,474	0,631
5 5	0,753	0,798
5 7	0,839	0,840
7 3	0,442	0,676
7 5	0,866	0,858
7 7	0,816	0,806
3 3 3	0,487	0,794
5 3 3	0,463	0,945
7 3 3	0,487	0,958
7 5 3	0,447	0,641

Random forest evaluation was conducted using party -named package in R. In this package the default hyperparameters were used except the number of trees and number of variables available for splitting at each tree node and this hyperparameter is now referred as mtry parameter from now on. The number of trees is on default 500 so in the modeling the number of trees vary from 300 to 700 to find the optimal hyperparameter. In terms of mtry the default value is $\sqrt{p}$, where p denotes the number of predictor variables. The values of $\sqrt{p}$-2, $\sqrt{p}$ and $\sqrt{p}$+2 are tried during modeling. In **table 8** can be seen that random forest classifier endures brilliantly on this classification problem. Multiple hyperparameter settings result accuracy of over 90 % which can be taken as very satisfying result with this kind of data where some

instances are sparse. Mtry parameter of √2-2 performs badly when compared to others barely topping 75 % of accuracy. The top four accuracies are close with each other since the top four results fit in between 1,3 percentage points. Each of the mtry parameter values of √p+2 score high but then random forest with 300 trees and mtry parameter value of √p hits high also with accuracy of 92,1 %. The top two accuracies are 92,4 % and 92,1 % but the model with the highest accuracy doesn't perform so well on terms of TPR having only 81,9 % which is the fourth highest. So, the best performing model is the one with 300 trees and mtry parameter value of √p which has accuracy of 92,1 % and TPR of 83,4 % which is the highest of all models.

Table 8. The results of the random forest classifier in the terms of average accuracy and TPR.

Hyperparameters	Accuracy	TPR
Ntree = 300, mtry = √p -2	0,755	0,498
Ntree = 500, mtry = √p -2	0,774	0,503
Ntree = 700, mtry = √p -2	0,779	0,498
Ntree = 300, mtry = √p	0,921	0,834
Ntree = 500, mtry = √p	0,853	0,688
Ntree = 700, mtry = √p	0,879	0,753
Ntree = 300, mtry = √p +2	0,911	0,825
Ntree = 500, mtry = √p +2	0,918	0,826
Ntree = 700, mtry = √p +2	0,924	0,819

After hyperparameter tuning and choosing the best possible hyperparameters on both of machine learning algorithms the next step was to perform the evaluation on test set data. As none of the test set data was used in hyperparameter tuning all the observations are completely new for the algorithms. The results of the performance with the unseen test data for both the selected and tuned algorithms are shown in **table 9.** Accuracy value of ANN is 90,2 % and TPR is 81,2 % where for Random forest accuracy is 96,3 % and TPR is 88,0 %. With these results it is easy to say that Random forest performs better with this kind of clustered supplier data compared to ANN and it should be used when implementing this

solution to automatically evaluate correct group for suppliers. The accuracy was higher on test data with both models than with the validation rounds which might relate to similar samples on that set of testing data. Remarkable on that issue still is that the highest accuracy values on validation rounds are not lower compared to test data, so looks like the algorithms perform well in predicting the correct cluster overall.

Table 9. The results of tuned models trained with the best performing hyperparameters with test data set.

Machine learning algorithm	Accuracy	TPR
Artificial neural network	0,902	0,812
Random forest	0,963	0,880

As the solution will be used as decision support and aiding tool in supplier management process the accuracy rate of classifying instances with Random forest is more than enough to guide the decision maker into correct direction assigning supplier into correct group. A high TPR value indicates also that the random forest rarely classifies falsely suppliers, so it detects with ease the correct groups most of the time. When the clusters assigned in unsupervised phase are just directional instead of rules, the small percentage of mistakes the Random forest algorithm makes are even more diminished as the whole process is controlled by human in the end. The other major advantage Random forest has compared to the so-called black box algorithms, like ANN, is that the decision maker is able to pull out parts of the decision logic used in the algorithm. Even Random forest isn't as unambiguous as decision tree algorithm as it's a group of decision trees with different split points it still enables having a glimpse of possible decision logic when classifying supplier. Looking into the logic in couple of different trees the decision maker can obtain some knowledge how the classifier ended up in the results it made. In **figure 28** is presented a one single tree of 300 possible options in the chosen Random forest hyperparameters. Decision maker could extract some information of how one way would be to differentiate suppliers from each other. For example instances from cluster three and four falling into decision node with variable unit price mean can be divided into two groups: if the unit mean price is higher or equal than 49858,76 then it will surely belong to group four and if the unit mean price is lower than 49858,76 then it will belong to group three with possibility of 42,9 % and into group four with possibility of 57,1 %.

In **figure 28** the numbers in the branches represent the value in which the splitting decision with that variable is done. The number in white box shows the decision order. P-value inside decision node reflects the split quality with testing the chosen variables statistical significance. N value would represent the number of instances ending up into final leaves but in Random forest classified no single tree represents the final solution, so all these values are zero. The y-vector holds the possibility distribution of clusters where the instances falling into that leaf will be assigned starting from left with cluster one and ending on right with cluster five.

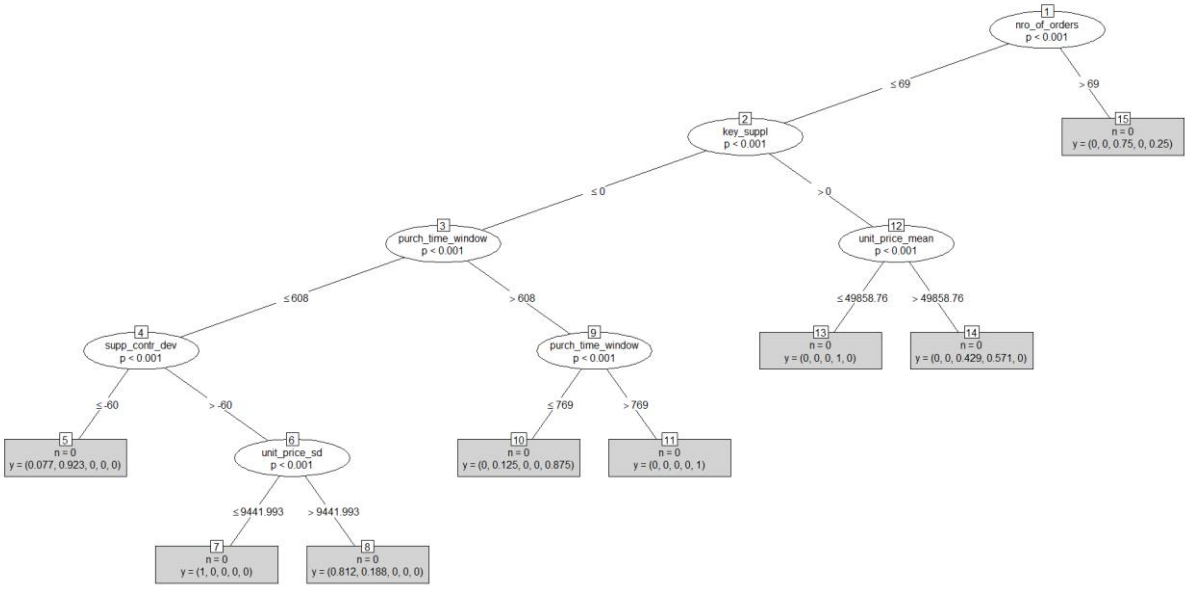

Figure 28. Example of one tree in the best performing tuned Random forest algorithm.

5 SUMMARY AND CONCLUSIONS

The thesis is built on three main distinctive parts. In the first part the process and framework for the supplier relationship management is introduced via literature review to bring the knowledge of the business application and need for the study. While describing the supplier relationship management process and its framework the advantages of capability to categorize and classify the supplier efficiently into their portfolios is rationalized hence bringing the reason to conduct this feasibility study for procurement organization. The first part builds the basis for the whole application in business environment and explains its potential to create value for NES when implemented to its daily operations.

The second part of the study comprises of the literature review and applications on machine learning area. The second part goes through the data analysis process which explains the whole flow from data extraction into model evaluation and result interpretation. In the second part of the thesis the chosen machine learning approach is justified and different algorithms, and how those are conducted, is presented. The second part also lays the mathematical basis on evaluating the algorithms and the results of the study.

The last main part of the thesis is the empirical machine learning application where the problem of trying to cluster and classify NES related supplier data is conducted. In this chapter the process of creating and running the application is presented and the reasons for choosing different machine learning and processing algorithms is explained. The main offering of the third part is the interpretation of the results derived from the data through different algorithms and understanding the given data and its potential in clustering and classification applications and the most suitable algorithms to perform those activities.

This last chapters compromises from short summary of the thesis and conclusions which can be taken from the takeaways of the literature reviews and results of the empirical application. This first part of the last chapter presented the summary of the thesis. Secondly, the presented research questions are answered within the scope of the study. Thirdly, the limitations of this empirical application are discussed and pointed out for the audience to understand the possible problems in implementing this empirical study into business environment without proper integration. Finally, the options and suggestions for future research are given to enhance this study.

5.1 Answering the research questions

The aim for the study was to find the most important and non-redundant variables from the given dataset for efficient clustering of the suppliers, understanding of what would be suitable amount of clusters or portfolios for this dataset and what kind of characteristics each cluster has, and evaluate between two different machine learning algorithms which one would be more suitable classifying given data into pre-determined clusters. Also, one of the purposes of the thesis was to explain the theoretical framework for possible implementation of studied machine learning application for procurement environment. To reach the objectives of the study, three

research questions were formulated in the beginning. Learning towards the literature review and empirical research, the answers are presented next.

1. *What are the most important features of internal data available in grouping suppliers?*

From the theoretical framework point of view the solution is simple: the important features are those which represent the problem thoroughly and are non-redundant which each other while keeping the dimensionality as low as possible. Purely mathematical point of view would be balancing of having high enough variance from the original set with as few variables as possible. From supplier relationship management point of view the variable space should in optimal situation have datapoints from supplier profit impact, supply risk, performance and collaboration relationship. To be able to cluster the supplier as effectively as possible and formulate correct strategy for supplier relationship management each area previously mentioned should have multiple different-angle measurements.

With the help of mathematical algorithms of principal component analysis and Pearson correlation coefficient the original set of variables was reduced by examining correlations and variances explained between different variables. Variables nro_of_proj, nro_of_rows and unit_price_median were discarded as redundant variables with some other variable remaining in the dataset explaining more of the total variance. At this point one new variable was derived from the original variables to represent the different batch sizes ordered from the suppliers: supp_ordersize. Variables nro_of_feedbacks and supp_agr_dev were discarded with decision of having too much NES internal affect in them rather than measuring the supplier characteristics. Survey questionnaire results were all highly correlating to each other, so a decision was made to aggregate the results into one number representing the whole survey.

All the variables or datapoints used to derive variables which were extracted from the systems in the beginning, were considered relevant for the research, so the variables which were not discarded or aggregated in feature selection part of the research can be considered as “important” for supplier grouping. The list includes these variables: key_suppl, nro_of_orders, quan_m, quan_pieces, tot_spend, unit_price_mean, unit_price_sd, purch_time_window, purch_freq, supp_contr_dev, supp_LT, sur_mean and supp_ordersize. Also, it was found in the research that some dimensions of SRM model were not properly covered, so probably some explanatory variables are still left to be researched but were not part of given internal data.

2. *What is the suitable quantity of clusters for the data and what are the key features of each cluster?*

As the dataset to be clustered was chosen, the suitable quantity of the clusters was defined in literature review with respect to inter- and intra-cluster similarities. The suitable quantity of clusters would be achieved in a trade-off between interpretable quantity of clusters and cluster purity. The rule is that intra-cluster similarity should be as high as possible and inter-cluster similarity should be as low as possible with retaining the quantity of clusters in interpretable amount with respect to the problem tried to be solved. Literature review presented algorithms to measure similarities of intra- and inter-clusters which help to optimize the correct quantity of clusters in given range of interpretable clusters.

The clustering quantity determination algorithms CH index and Silhouette width were tried in range from two to ten clusters as ten clusters was considered as maximum number for practical approach. Even quantity of two clusters performed very well in CH index and Silhouette value it was also ruled out since two supplier management portfolios would be too generic to be used. After these restrictions the best possible option was found from quantity of five clusters where both CH index and Silhouette value illustrated good performance with clear progress also in elbow index compared to four clusters. Therefore, it was concluded that for the dataset available five clusters would be the suitable quantity.

After conducting the clustering into five different clusters with K-means algorithm the first cluster had 86 suppliers with low or mediocre values in all variables, the second cluster had 28 suppliers with low value of orders but with high monetary values and lead times, the third cluster had 12 suppliers with high values on all variables except unit prices, lead time and order size, the fourth cluster had 20 suppliers with mediocre values in all variables except high quantity of pieces and low purchase frequency and order size and the fifth cluster had 89 suppliers with mediocre or low values in all variables except in metric quantity. From the five clusters key suppliers were divided into clusters three and four. The other key variables are presented in **table 6** which was also presented earlier.

3. *Which of the algorithms is the best for classifying suppliers and should be used?*

From the literature review the goal for classifying algorithms was found being able to classify the new observations as effectively as possible without losing the ability to generalize into the whole problem. There exists a trade-off between accuracy and generalization which is being tackled by training the classifier with part of the data and then tested with unseen data from the same dataset. Finally, the performance of the classifiers is evaluated by the ability of classifying testing data correctly which can be measured, for example, with classification accuracy and true positive rate. The classifier with the highest accuracy and TPR is best for the problem and should be used.

The chosen machine learning algorithms for the research were neural network and random forest due their popularity and usability. The data was split into training and testing sets with picking random instances from all the clusters. Further, to help the classifier to examine multiple aspects from the training data and decrease the bias, the training data was split into train and validation datasets with 4-fold cross-validation method. After training the algorithms with different hyperparameter architectures, the best performing architectures for the algorithms were chosen. Accuracy was the primary criterion for the problem at hand and TPR used as supportive criteria. The best performing neural network architecture with validation data was two hidden layer network with seven neurons in the first and five neurons in the second hidden layer with accuracy of 86,6 % and with TPR of 85,8 %. The best performing random forest architecture with validation data was with 300 grown trees and with mtry parameter value of $\sqrt{p}$ with accuracy of 92,1 % and with TPR of 83,4 %. After this the algorithms were exposed to the testing dataset and random forest was better classifier with 96,3 % accuracy and 88,0 % TPR compared to neural networks accuracy of 90,2 % and TPR of 81,2 %. From this we can conclude random forest is the better algorithm for classifying suppliers and should be utilized in the supplier relationship management process. Also, the random forest enables pulling single trees out of the model which helps the sourcing personnel to understand parts of the model logic compared to total black box model of neural network.

5.2 Limitations of the models and implementation

The research conducted in the thesis has two main issues. One is related to used data quality and scarcity and the other relates to implementation of the supplier relationship management process with machine learning capabilities using these variables. The first affects greatly to the results of the whole research and might lead to minor deviations of the reality and the second

probably hinders the utilization of the feasibility study as-is and probably requires a more refined variable gathering from different sources.

The quality of the data used in the research had problems which was known from the start. The main reason for this was the human error on manual inputs but also the change of ERM system at NES. The ERM system change led towards data extraction from two systems which in theory could have duplicate data and some migrated purchase orders from the old system towards new didn't have all the required datapoints for this research migrated. Another issue related to the used data was its scarcity. With scarcity is referred to the fact that there was a lot of purchase order data available but those many of those purchase orders especially from the old systems didn't have the required datapoints. This led to the fact that aggregating the data was done on ignoring the null fields and calculating the aggregations on the datapoints available. The other scarcity related issue was towards the survey about supplier collaboration. The survey had data only from a handful of suppliers of the supplier pool available from purchase orders. This fact probably affected the research creating minor biases.

The second main issue implementing the full supplier relationship management framework rises from the fact that few of the variables presenting the affecting factors of the supplier relationship framework were totally missing. It might be a bit dangerous to conduct the procurement strategy towards suppliers purely on basis of the research as no factors like goods quality or supplier specific risk or market environment are available. In order to utilize the supplier relationship management framework in its full potential, variables explaining these factors should be added into the dataset to be analyzed, clustered and classified. On the other hand, this thesis was supposed to be a feasibility study on utilizing advanced analytics in procurement function, so it lays the basis on future development.

5.3 Future research opportunities

This research aimed to utilize machine learning algorithms on procurement data in order to ease the procurement function work in supplier categorization and classification. The future research should focus on defining more precise variables required for optimal supplier relationship management process rather than just examples of overall areas like "profit impact" and "supply risk". Kant and Dalvi (2015, p. 360-365) list 151 supplier evaluation criteria and 52 supplier selection criteria part of which could be utilized as variables to gain more in-depth understanding the supplier dimension.

Relating to the variables themselves, weighting criteria should be assigned to each one of them. As Marinelli and Antoniou (2019, p. 92-94) present in their research that different factors, such as high competition in the market, affects more to monetary value achieved than others. This knowledge should be utilized in supplier relationship management and assign greater weights to variables which affect more one value creation and assign smaller weights to variables which have lesser impact. With this approach the focus in the data would be in the variables that matter rather than variables that differentiate the dataset and explain more variance.

Applying fuzzy logic would be extremely interesting on the variables and worthy the research as crisp boundaries between the clusters are too exclusive and most of the key attributes of different clusters could be explained linguistically on approximate terms with enough information still to utilize them. Osiro et al. (2014, p. 108-111) have in their research conducted that “fuzzy set theory is suitable for dealing with the vagueness intrinsic to qualitative factors of suppliers’ evaluation” which proves that fuzzification can be applied to supplier data to some extent and actually perform well. This fuzzification would enable creating and expressing the clusters with membership degree rather than binary “yes” or “no” option which would also give the sourcing professionals ability to choose which action plan to implement on single supplier having characteristics from two or more clusters.

REFERENCES

Ahmad, M. W., Mourshed, M. & Rezgui, Y. (2017). Trees vs Neurons: Comparison between random forest and ANN for high-resolution prediction of building energy consumption. *Energy and Buildings.* Vol. 147, 77-89.

Alpaydin, E. & Bach, F. (2014). Introduction to Machine Learning. 3rd edition. Cambridge: MIT Press.

Badillo, S., Balazs, B., Birzele, F., Davydov, I., Hutchinson, L., Kam-Thong, T., Siebourg-Polster, J., Steiert, B. & Zhang, J. (2020). An introduction to Machine Learning. *Clinical Pharmacology & Therapeutics,* Vol. 107(4), 871-885.

Bagirov, A., Karmitsa, N. & Taheri, S. (2020). Partitional Clustering via Nonsmooth Optimization. Cham, Switzerland: Springer Nature.

Bai, Y. (2019). Data cleansing method of talent management data in wireless sensor network based on data mining technology. *Journal on Wireless Communications and Networking.* Vol. 2019(19), 1-6.

Bauer, E. & Kohavi, R. (1999). An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants. *Machine Learning.* Vol. 36, 105-139.

Bhargavi, M. & Gowda, S. (2015). A novel validity index with dynamic cut-off for determining true clusters. *Pattern Recognition*. Vol. 48(11), 3673-3687.

Bienhaus, F. & Haddad, A. (2017). Procurement 4.0: factors influencing the digitalization of procurement and supply chains. *Business Process Management Journal,* Vol. 24(4), 965-984.

Bishop, C. (2006). Pattern Recognition and Machine Learning. New York: Springer Science+Business Media.

Blanc, S. & Setzer, T. (2019), Bias-Variance Trade-Off and Shrinkage of Weights in Forecast Combination. *Management Science.* Vol. 66(12), 5720-5737.

Blum, A. & Langley, P. (1997). Selection of relevant features and examples in machine learning. *Artificial Intelligence.* Vol. 97(1-2), 245-271.

Bolón-Canedo, V., Sánches-Maroño, N. & Alonso-Betanzos, A. (2015). Feature Selection for High-Dimensional Data. Cham: Springer International Publishing Switzerland.

Bradley, A. (1997). The use of the are under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition.* Vol. 30(7), 1145-1159.

Breiman, L. (1996). Bagging predictors. *Machine learning.* Vol. 24, 123-140.

Breiman, L. (2001). Random forests. *Machine learning.* Vol. 45, 5-32.

Breiman, L., Friedman, J., Olshen, R. & Stone, C. (1984). Classification and Regression Trees. Belmont, CA: Wadsworth international.

Brown, M. & Kros, J. (2003). Data mining and the impact of missing data. *Industrial Management & Data Systems.* Vol. 103(8), 611-621.

Brusco, M., Singh, R. & Steinley, D. (2009). Variable Neighborhood Search Heuristics for Selecting a Subset of Variables in Principal Component Analysis. *Psychometrika.* Vol. 74(4), 705-706.

Cangelosi, R. & Goriely, A. (2007). Component retention in principal component analysis with application to cDNA microarray data. *Biology Direct.* Vol. 2(2), 1-21.

Cannas, L., Dessì, N. & Pes, B. (2013). Assessing similarity of feature selection techniques in high-dimensional domains. *Pattern Recognition Letters.* Vol. 34(12), 1446-1453.

Carter, C. & Liane Easton, P. (2011). Sustainable supply chain management: evolution and future directions. *International Journal of Physical Distribution & Logistics Management.* Vol. 41, 46-62.

Celebi, M. & Aydin, K. (2016). Unsupervised Learning Algorithms. Cham, Switzerland: Springer International Publishing.

Chakraborty, S. (2015). Reduction Supplier Tale through Systematic Vendor Management: A Study on Purchase and Vendor Management. *Journal of Supply Chain Management Systems.* Vol. 4(4), 24-37.

Cheema, J. (2014). A Review of Missing Data Handling Methods in Education Research. *Review of Educational Research.* Vol. 84(4), 487-508.

Chen, B. & Yin, H. (2018). Learning category distance metric for data clustering. *Neurocomputing.* Vol. 306, 160-170.

Chiang, M. & Mirkin, B. (2010). Intelligent Choice of the Number of Clusters in K-Means Clustering: An Experimental Study with Different Cluster Spreads. *Journal of Classification.* Vol. 27, 3-40.

Chollet, F. (2018). Deep Learning with Python. Shelter Island, USA: Manning Publications Co.

Chormunge, S., Sanjay, C. & Sudarson, J. (2014). Performance Evaluation of Clustering Methods for Low and High Dimensional Data. *International Journal of Advanced Research in Computer Science.* Vol. 5(4), 91-95.

Ciaburro, G. (2017). MATLAB for Machine Learning. Birmingham, UK: Packt Publishing.

Clarke, B., Fokoué, E. & Zhang, H. (2009). Principles and Theory for Data Mining and Machine Learning. New York: Springer Science+Business Media.

Costarelli, D. & Spigler, R. (2013) Approximation results for neural network operators activated by sigmoidal functions. *Neural Networks.* Vol. 44, 101-106.

Dash, T. & Behera, H. (2019). A comprehensive study on evolutionary algorithm-based multilayer perceptron for real-world data classification under uncertainty. *Expert systems.* Vol. 36(1), 1-20.

DeepaLakshmi, S. & Velmurugan, T. (2016). Empirical study of feature selection methods for high dimensional data. *Indian Journal of Science and Technology*. Vol. 9(39), 1-6.

Di Guida, R., Engel, J., Allwood, J., Weber, R., Jones, M., Sommer, U., Viant, M. & Dunn, W. (2016). Non-targeted UHPLC-MS metabolomic data processing methods: a comparative investigation of normalization, missing value imputation, transformation and scaling. *Metabolomics.* Vol. 12(5), 1-14.

Fattah, N., Ghaleb, M. & Al Mahdy, O. (2016). Flow units delineation of multiple hydrocarbon reservoirs using hydraulic zonation technique via cluster analysis algorithm, Zeit Bay Field, Gulf of Suez, Egypt. *Arabian Journal of Geosciences.* Vol. 9(7), 1-22.

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters.* Vol. 27(8), 861-874.

Fernandes de Mello, R. & Antonelli Ponti, M. (2018). Machine Learning: A Practical Approach on the Statistical Learning Theory. Cham: Springer International Publishing Switzerland.

Fernandez, D., Gonzalez, C., Mozos, D. & Lopez, S. (2016). FPGA implementation of the principal component analysis algorithm for dimensionality reduction of hyperspectral images. *Journal of Real-Time Image Processing.* Vol. 16(5), 1395-1406.

Freeman, C., Kulić, D. & Otman, B. (2014). An evaluation of classifier-specific filter measure performance for feature selection. *Pattern Recognition.* Vol. 48, 1812-1826.

Freund, Y. & Schapire, R. (1996). A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences.* Vol. 55, 119-139.

Garciá, S., Herrera, F. & Luengo, J. (2015) Data Preprocessing in Data Mining. Cham: Springer International Publishing Switzerland.

Gollapudi, S. (2016). Practical machine learning: tackle the real-world complexities of modern machine learning with innovative and cutting-edge techniques. Birmingham, UK: Packt Publishing.

González, J., Ortega, J., Damas, M., Martín-Smith, P. & Gan, J. (2019). A new multi-objective wrapper method for feature selection – Accuracy and stability analysis for BCI. *Neurocomputing.* Vol. 333, 407-418.

Gorelick, M. (2006). Bias arising from missing data in predictive models. *Journal of Clinical Epidemiology.* Vol. 59, 1115-1123.

Gray, K., Aljabar, P., Heckemann, R., Hammers, A. & Rueckert, D. (2013). Random forest-based similarity measures for multi-modal classification of Alzheimer's disease. *NeuroImage.* Vol. 65, 167-175.

Hald, K. & Ellegaard, C. (2010). Supplier evaluation processes: the shaping and reshaping of supplier performance. *International Journal of Operations & Production Management.* Vol. 31(8), 888-910.

Hand, D. & Till, R. (2001). A Simple Generalisation of the Area Under the ROC Curve for Multiple Class Classification Problems. *Machine Learning.* Vol. 45(2), 171-186.

Hastie, T., Tibshirani, R. & Friedman, J. (2017). The Elements of Statistical Learning. 2nd Edition. New York, Springer Science + Business Media.

Hartigan, J. & Wong, M. (1979). Algorithm AS 136: A K-Means Clustering Algorithm. *Journal of Royal Statistical Society, Series C (Applied Statistics).* Vol. 28, 101-108.

Hyvärinen, A., Hurri, J. & Hoyer, P. (2009). Natural Image Statistics: A Probabilistic Approach to Early Computational Vision. London: Springer-Verlag Limited.

Hudnurkar, M., Rathod, U. & Jakhar, S.K. (2015). Multi-criteria decision framework for supplier classification in collaborative supply chains: Buyer's perspective. *International Journal of Productivity and Performance Management.* Vol. 65(5), 622-640.

Ivosev, G., Burton, L. & Bonner, R. (2008). Dimensionality Reduction and Visualization in Principal Component Analysis. *Analytical Chemistry.* Vol 80(13), 4933-4944.

Jiang, S. & Wang, L. (2015). Efficient feature selection based on correlation measure between continuous and discrete features. *Information Processing Letters.* Vol. 116(2), 203-215.

Joshi, A. (2020). Machine Learning and Artificial Intelligence. Cham: Springer Nature Switzerland.

Kant, R. & Dalvi, M. (2015). Development of questionnaire to assess the supplier evaluation criteria and supplier selection benefits. *Benchmarking: an international journal.* Vol. 24(2), 359-383.

Kassambra, A. (2020). Package 'factoextra'. [Online]. Available at: https://cran.r-project.org/web/packages/factoextra/factoextra.pdf. [Accessed 20.12.2020].

Kassambra, A. (2017). Practical Guide to Principal Component Methods in R. CreateSpace Independent Publishing Platform.

Kraljic, P. (1983). Purchasing must become supply management. *Harvard business review.* Vol. 61(5), 109-117.

Krause, D., Vachon, S. & Klassen, R. (2009). Special Topic Forum on Sustainable Supply Chain Management: Introduction and Reflections on the Role of Purchasing Management. *The journal of supply chain management.* Vol. 45(4), 18-25.

Kubat, M. (2015). An introduction to Machine learning. Cham: Springer International Publishing Switzerland.

Kumar, M., Jindal, M., Sharma, R. & Jindal, S. (2020). Performance evaluation of classifiers for the recognition of offline handwritten Gurmukhi characters and numerals: a study. *Artificial Intelligence Review.* Vol. 53, 2075-2097.

Lahtinen, H., Ylinen, A., Lukkarinen, U., Sirviö, J., Miettinen, R. & Riekkinen Sr., P. (1996). Failure of carbamazepine to prevent behavioural and histopathological sequels of experimentally induced status epilepticus. *European Journal of Pharmacology.* Vol. 297(3), 213-218.

Lee, E., Ha, S. & Kim, S. (2001). Supplier Selection and Management System Considering Relationships in Supply Chain Management. *IEEE transactions on engineering management.* Vol. 48(3), 307-318.

Legner, C., Eymann, T., Hess, T., Matt, C., Bähmann, T., Drews, P., Mädche, A., Urbach, N. & Ahlemann, F. (2017). Digitalization: Opportunity and Challenge for the Business and Information System Engineering Community. *Business information system engineering,* Vol. 59(4), 301-308.

Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R., Tang, J. & Liu, H. (2017). Feature Selection: A Data Perspective. *ACM Computing Surveys.* Vol. 50(6), 1-45.

Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y. & Alsaadi, F.E. (2017). A survey of deep neural net-work architectures and their applications. *Neurocomputing*, Vol. 234, 11-26.

Luengo, J., Gargiá-Gil, D., Ramírez-Gallego, S., Garciá, S. & Herrera, F. (2020). Big Data Preprocessing. Cham: Springer Nature Switzerland.

Merle, E. (2011). A Comparison of Imputation methods for Bayesian Factor Analysis Models. *Journal of Educational and Behavioral Statistics.* Vol. 36(2), 257-276.

Miller, J. (2017). Big Data Visualization. Birmingham, UK: Packt Publishing.

Mohamed, M., Hashem, A. & Abdelsamea, M. (2014). Scalable Algorithms for Missing Value Imputation. *International Journal of Computer Applications.* Vol. 87(11), 35-42.

Mouton, J., Ferreira, M. & Helberg, A. (2020). A comparison of clustering algorithms for automatic modulation classification. *Expert Systems with Applications.* Vol. 151, 1-10.

Mu, Y., Liu, X. & Wang, L. (2018). A Pearson's correlation coefficient based decision tree and its parallel implementation. *Information Sciences.* Vol. 435, 40-58.

Nelli, F. (2019) Python Data Analytics: With Pandas, NumPy and Matplotlib. Berkeley, CA: Apress L.P.

Neste. (2021a) Neste's strategy – Faster, Bolder and Together. [Online]. Available at: https://www.neste.com/about-neste/who-we-are/strategy#90c17277. [Accessed 27.3.2021].

Neste. (2021b). We value operational excellence. [Online]. Available at: https://www.neste.com/about-neste/suppliers/procurement. [Accessed 27.3.2021].

Neste (2021c). Neste Engineering Solutions. [Online]. Available at: https://www.neste.com/about-neste/who-we-are/neste-engineering-solutions#90c17277. [Accessed 27.3.2021].

Nguyen, T., Zhou, L., Spiegler, V., Ieromonachou, P. & Lin, Y. (2018). Big data analytics in supply chain management: A state-of-the-art literature review. *Computers and Operations Research.* Vol. 98, 254-264.

Oh, J. & Rhee, S. (2008). The influence of supplier capabilities and technology uncertainty on manufacturer-supplier collaboration: A study of the Korean automotive industry. *International Journal of Operations & Production Management.* Vol. 28(6), 490-517.

Olsen, R. & Ellram, L. (1997). A Portfolio Approach to Supplier Relationships. *Industrial Marketing Management.* Vol. 26(2), 101-113.

Osiro, L., Lima-Junior, F. R., Carpinetti, L. (2014). A fuzzy logic approach to supplier evaluation for development. *International journal of production economics.* Vol. 153, 95-112.

Park, J., Shin, K., Chang, T. & Park, J. (2010). An integrative framework for supplier relationship management. *Industrial Management & Data Systems.* Vol. 110(4), 495-515.

Paternina, M., Zamora-Mendez, A., Ortiz-Bejar, J., Chow, J. & Ramirez, J. (2018). Identification of coherent trajectories by modal characteristics and hierarchical agglomerative clustering. *Electric Power Systems Research.* Vol. 158, 170-183.

Peyre, H., Leplége, A. & Coste, J. (2010). Missing data methods for dealing with missing items in quality of life questionnaires. A comparison by simulation of personal mean score, full information maximum likelihood, multiple imputation, and hot deck techniques applied to the

SF-36 in the French 2003 decennial health survey. *Quality of life research.* Vol. 20(2), 287-300).

Potdar, K., Pardawala, T. & Pai, C. (2017). A Comparative Study of Categorical Variable Encoding Techniques for Neural Network Classifiers. *International Journal of Computer Applications.* Vol. 175(4), 7-9.

Powers, D.M.W. (2011). Evaluation: from Precision, recall and F-measure to ROC, Informedness, Markedness & Correlation. *Journal of Machine Learning Technologies.* Vol. 2(1), 37-63.

Revathy, N., Guhan, T. & Selvarajan, S. (2017). A Study on the Scalability of Classical Data Clustering K-means Algorithm. *International Journal of Advances in Engineering & Technology.* Vol. 10(2), 159-174.

Ringnér, M. (2008). What is principal component analysis? *Nature biotechnology.* Vol 26(3), 303-304.

Ritter, T. & Pedersen, C.L. (2020). Digitization capability and the digitalization of business models in business-to-business firms: Past, present, and future. *Industrial Marketing Management.* Vol. 86, 190-190.

Rocha, M., Cortez, P. & Neves, J. (2007). Evolution of neural networks for classification and regression. *Neurocomputing.* Vol. 70(16), 2809-2816.

Russell, S. & Norvig, P. (2010). Artificial Intelligence: A Modern Approach. 3rd edition. Harlow: Pearson Education.

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks.* Vol. 61, 85-117.

Schober, P., Boer, C. & Schwarte, L. (2018). Correlation Coefficients: Appropriate Use and Interpretation. *Anesthesia & Analgesia.* Vol. 126(5), 1763-1768.

Schuh, C., Strohmer, M., Easton, S., Hales, M. & Triplat, A. (2014). Supplier Relationship Management. Berkeley, CA: Appress.

Sharma, S. & Yadav, R. (2013). Comparative Study of K-means and Robust Clustering. *International Journal of Advanced Computer Research.* Vol. 3(3), 207-210.

Solorio-Fernández, S., Carrasco-Ochoa, J. & Martínez-Trinidad, J. (2016). A new hybrid filter–wrapper feature selection method for clustering based on ranking. *Neurocomputing.* Vol. 214, 866-880.

Song, Y. & Lu, Y. (2015). Decision tree methods: applications for classification and prediction. *Shanghai Archives of Psychiatry.* Vol. 27(2), 130-135.

Sârbu, C. & Pop, H. (2005). Principal component analysis versus fuzzy principal component analysis: A case study: the quality of Danube water (1985-1996). *Talanta.* Vol. 65(5), 1215–1220.

Tattar, P., Ojeda, T., Murphy, S., Bengfort, B. & Dasgupta, A. (2017). Practical data science cookbook: practical recipes on data pre-processing, analysis and visualization using R and Python. 2nd edition. Birmingham, UK: Packt Publishing.

Thakare, Y. & Bagal, S. (2015). Performance Evaluation of K-means Clustering Algorithm with Various Distance Metrics. *Journal of Computer Applications.* Vol. 110(11), 12-15.

Venables, W., Smith, D. & the R Core Team. (2020). An Introduction to R. [Online] Available at: https://cran.r-project.org/doc/manuals/r-release/R-intro.pdf. [Accessed 19.9.2020].

Wei, R., Wang, J., Mingming, S., Jia, E., Chen, S., Chen., T. & Ni, Y. (2018). Missing Value Imputation Approach for Mass Spectrometry-based Metabolomics Data. *Scientific Reports.* Vol. 8, 663-671.

Wilton, P. & Colby, J. (2005). Beginning SQL. Indiana, US: Wiley Publishing.

Wistuba, M., Schilling, N. & Schmidt-Thieme, L. (2018). Scalable Gaussian process-based transfer surrogates for hyperparameter optimization. *Machine Learning.* Vol. 107, 43-78.

Wu, J. (2012). Advances in K-means Clustering. New York, Springer-Verlag Berlin Heidelberg.

Xu, J., Tang, B., He, H. & Man, H. (2017). Semisupervised Feature Selection Based on Relevance and Redundancy Criteria. *IEEE transaction on neural networks and learning systems.* Vol. 28(9), 1974-1984.

Yang, J., Grunsky, E. & Cheng, Q. (2019). A novel hierarchical clustering analysis method based on Kullback–Leibler divergence and application on dalaimiao geochemical exploration data. *Computers and Geosciences.* Vol. 123, 10-19.

Yang, L. & Shami, A. (2020). On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing.* Vol. 415, 295-316.

Yue, S., Wang, J., Wang, J. & Bao, X. (2016). A new validity index for evaluating the clustering results by partitional clustering algorithms. *Soft Computing.* Vol. 20, 1127-1128.

Zhu, B., Changzheng, H. & Liatsis, P. (2010). A robust value imputation method for noisy data. *Applied Intelligence.* Vol. 36(1), 61-74.

Zimmer, K., Fröhling, M. & Schultmann, F. (2016). Sustainable supplier management – a review of models supporting sustainable supplier selection, monitoring and development. *International Journal of Production Research.* Vol. 54(5), 1412-1442.

APPENDICES

Appendix A: List of variables and their explanations and usage.

Name of variable	Description of variable	Used in clustering?
key_suppl	Key supplier indicator	Yes
fullfills_tasks	Supplier capability fulfilling defined tasks	No
easy_to_contact	How easy it is to contact supplier personnel	No
solving_req_or_issues	Supplier capacity solving requests or issues	No
flexible_to_changes	Supplier flexibility towards delivery changes	No
satisfaction_to_items	Supplier's commodities satisfactory	No
co_op_direction	Trend of co-operation	No
nro_of_feedbacks	Number of feedbacks	No
nro_of_orders	Number of different orders	Yes
nro_of_projects	Number of different projects	No
nro_of_rows	Number of different order rows	No
quan_m	Total quantity in meters	Yes
quan_pieces	Total quantity in pieces	Yes
tot_spend	Total spend	Yes
unit_price_mean	Average unit price	Yes
unit_price_median	Median unit price	No
unit_price_sd	Standard deviation of unit price	Yes
purch_time_window	Difference between the first and last recorded purchase	Yes
purch_freq	Purchase frequency	Yes
supp_contr_dev	Supplier's average delivery date deviation from contractual date	Yes
supp_agr_dev	Supplier's average delivery date deviation from separately agreed date	No
supp_LT	Supplier's average lead time	Yes
sur_mean	Average results of survey variables	Yes
supp_ordersize	Average supplier's order size	Yes