Master thesis

Reducing the Total Travelling Distance of Order Picking in a Warehouse by introducing Class-Based Storage

Mark Sueters

UNIVERSITY OF TWENTE.

University of Twente
April 5, 2023

Colofon

Document:
Master thesis

Title:
Reducing the Total Travelling Distance of Order Picking in a Warehouse by Class-Based Location Assignment of SKUs

Author:
M.L. (Mark) Sueters

Institution:	Educational institution University of Twente	Organizational institution -
Faculty:	Faculty of behavioural, Management and Social sciences (BMS)	-
Programme:	Industrial Engineering and Management (IEM)	-
Specialization:	Production Logistics & Management (PLM)	-
Website:	www.utwente.nl	-
Supervisors:	Dr. M.C. van der Heijden Dr. B.A. Beirigo	C. Haak

Preface

Dear reader,

Right in front of you, my master's thesis is present. During the final phase of the master Production & Logistic Management at the University of Twente, I had the opportunity to conduct research at a 3pl company in a very dynamic environment. I experienced this research as very pleasant, which is also the result of some people involved. Special thanks to Matthieu van der Heijden and Breno Alves Beirigo for their supervision on behalf of the University of Twente. I also would like to thank Colin Haak, who supervised me on behalf of the company. When needed, he provided me with the required information and data to conduct this research. Besides, it was a pleasant environment to work in.

Completing this research also finishes my time as a student. During this time, I developed lots of social and academic skills, which I cannot wait to apply in the future. Without a doubt, my family and friends enabled me to enjoy this time.

Chapter 1 of this thesis contains the problem context. Thereafter, I provide insights into the current situation of the company (Chapter 2). In chapter 3, I conduct a literature review, where relevant topics to this research are reviewed. Then, in Chapter 4, I elaborate on the solution design, which will be analyzed in Chapter 5. Implementation requirements of the solution method are provided in Chapter 6. In the end, I conclude and provide recommendations in Chapter 7.

Lastly, I would like to wish you enjoyable reading.

Mark Sueters
Enschede, April 2023

Summary

This research is executed at the warehouse of a 3PL service provider, providing the service to take over logistics of other companies from receiving goods, warehousing these and delivering them to end-users. Order picking is the most costly activity in a typical warehouse, where travelling is the biggest part of the working time of order pickers. Several aspects are influencing the travelling distance of order picking. For this research, inefficient placement of SKUs in the warehouse and lack of SKU classification are chosen as problems to solve. The goal is to reduce the total travelling distance of order picking in the warehouse and the corresponding research question is:

What improvements regarding SKU-location assignment in a warehouse can be implemented to decrease the total travelling distance of order picking while the SKU assortment is constantly changing?

The travelling distance of order pickers mainly depends on the routes they make while performing picking tasks. The company picks according to zone picking, which means that they batch multiple orders and release picking tasks consisting of all items for multiple orders that are in the same zone. Currently, the company uses 16 different zones, which are all filled with items based on a Random Storage policy. During this research, I introduce using a Class-Based Storage policy instead of a Random Storage policy, which requires a classification method.

Classification: Historical demand patterns need to be taken into account during the classification. However, the constantly changing assortment of SKUs makes it challenging to detect underlying demand patterns on the SKU level. A suitable classification for Class-Based Storage is based on the FSN analysis, which labels SKUs as fast-, slow- and non-moving based on their Average Length of Stay in the warehouse. Random Forest can be used as an algorithm to classify SKUs into fast-, slow- and non-moving SKUs, also when SKU-specific history is not available. Random Forest uses data from SKUs with similar features to predict the class. For the company, I use the features "*brand*", "*product type*", "*gender*", "*size*", "*colour*", and "*season code*".

The effectiveness of the classification can be measured by comparing the class predictions with the actual class an SKU belongs to. Using Random Forest to classify into 2 classes (fast- and not-fast-moving) performs better in the companies case compared to classifying into 3 classes (fast-, slow- and non-moving) regarding classification accuracy. Therefore, classifying into 2 classes is considered as well during this research.

I propose to build 50 trees in the Random Forest model while selecting 2 features randomly. With these settings, the classification achieves an accuracy of 61.7% when classifying into 2 classes and 42.4% when classifying into 3 classes. The classification method provides us with the required space per class and the expected picks per class.

Slotting: Instead of using zones as currently done, I propose 2 zone layouts: a 2-class layout and a 3-class layout. For both layouts, the number of aisles (required space) per class is obtained by using the classification method. The resulting zone layouts are as visualised in Figure 1. Note that the layouts are disproportionate.

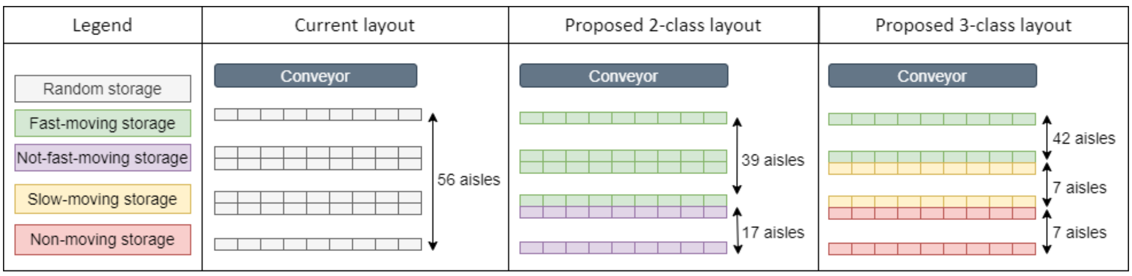

Figure 1: Current and proposed zone layouts

By concentrating SKUs that are likely to sell fast in aisles closer to the conveyor, a reduction in total travelling distance can be expected. Likewise, concentrating SKUs that are not likely to sell or sell very slowly to aisles far from the conveyor also has a positive impact on reducing the total travelling distance. Picks end up closer to each other, reducing the expected number of aisles to visit. Besides, the probability of visiting aisles furthest away from the conveyor becomes lower. In addition, pickers are expected to carry less weight on average.

Evaluation: Using the expected number of picks per class and the zone layouts enables me to approximate the total travelling distance by order pickers based on a method proposed by Hall. With this method, I estimate the expected travelling distance of a picking task. This, combined with the number of picking tasks a day, can then be used to calculate the expected travelling distance by order pickers for a given zone layout.

When classifying into 2 classes, the highest achieved reduction in total travelling distance is 3.2%. The highest improvement potential equals a reduction of 17.2%. Using the classification settings with the highest improvement potential, a reduction of 1.9% is achieved. When classifying into 3 classes, the highest achieved reduction in total travelling distance is 4.2%. The highest improvement potential equals a reduction of 20.4%. Using the classification settings with the highest improvement potential, a reduction of 3.6% is achieved.

So classifying into 3 classes results in a larger reduction in total travelling distance by order pickers and the improvement potential is also higher. However, using 3 classes in a zone layout requires exactly knowing where to put away items. Instead, by using 2 classes, fast-moving SKUs can be put away in aisles closer to the conveyor and not-fast-moving SKUs in aisles further away from the conveyor, without the need of leaving aisles empty for the middle class.

Implementation: Implementation of the classification model requires the availability of the inbound data, pull data and SKU feature information data, which then can be transformed into a specific format. Thereafter, this data can be used in an R script that automatically adds class labels to inbound. The output of this method is an inbound list with SKUs to store in the fast- and not-fast-moving aisles (2 classes) or in the fast-, slow- and non-moving aisles (3 classes). Labelling SKUs enable receivers to put away items in trolleys for their corresponding class. In this way, classes end up stored in their dedicated aisles.

Recommendations:

- Log selling price of items, which requires the retail price and the discount at the moment of selling. The selling price can then be used as a feature when training a Random Forest model.
- Data gathering: add SKU number to pull data, update feature information regularly and update training data regularly. This reduces the efforts of preparing data for the training of Random Forest models.
- Classify into 2 classes, because of less complexity during put away process.
- Find out how to label SKUs in the Warehouse Management System to enable using the classes.
- Decide what to do with items in the fast-moving area not being fast-moving: reallocate or leave in the area.
- Consider combining picking tasks to reduce travel distance through the main aisle.

An implication for **further research** is considering other routing strategies while planning pick tasks.

Definitions

Customer: Companies for whom the company provides logistics services
End-user: Customers of the customer
F/S-definition: ALS value that differentiates between fast- and slow-moving items
F/NF-definition: ALS value that differentiates between fast- and not-fast-moving items
FSN analysis: Fast-, slow- and non-moving analysis
Manco: An item that is missing
Mezzanine: Mezzanines are used to construct extra floors within a warehouse for example
Put Wall: A put wall is a shelving system in which multiple orders can be sorted separately
S/N-definition: ALS value that differentiates between slow- and non-moving items
Tote: A tote is a box in which items for 1 Put Wall can be stored that are in the same zone

Abbreviations

ALS: Average Length of Stay
B2B: Business-to-Business
B2C: Business-to-Customer
CR: Consumption Rate
mtry: number of features to select for each tree in a Random Forest model
ntree: number of trees to build in a Random Forest model
SKU: Stock Keeping Unit
TPR: True Positive Rate
WMS: Warehouse Management System.

List of Figures

List of Tables

Contents

1 Introduction

During this chapter, I start off by describing the company. This is followed by an elaboration on the problem context for this research, after which a corresponding problem statement and research objective are declared. To maintain a feasible timeline during this graduating project, the research is also scoped in this chapter. Thereafter, I state the research questions and how to approach answering these questions.

1.1 Company Description

This research is performed at a 3PL service provider company, which means that they provide logistics services to other companies. One of them is involved in this research, which I refer to as customer X. The company receives orders directly from the end-users and processes the orders by taking the items out of the warehouse and transporting the order to their destination. Customer X is an online outlet that sells products from other brands, mainly discounted. In their industry, too many products remain unsold. Instead of producing products, they have contact with a variety of brands and let the company do the logistics: receive products from brands, put away the products in the warehouse, pick orders, transport the products to end-users and handle returns. Nowadays the customer also sells newly launched products of certain brands.

The company warehouses millions of items for customer X. The warehouse consists of large halls, where 2 halls are reserved for the logistics of customer X. By using mezzanines, these halls have both 4 floors, resulting in a total of 37,000 square meters. Every floor in both halls is divided into 3 zones. On each floor, there is a doorway between the halls, which means that every floor has 6 zones in total. At the back of each hall and floor, conveyors are placed for transportation reasons.

1.2 Problem context

Costs of warehousing can be divided into a variety of components such as receiving, put away, order picking, packing and shipping. In literature, order picking is mentioned as the most costly (around 55% of the total warehousing costs) activity in a typical warehouse [1] [2] [3]. Therefore, it is interesting to look at order picking and processes influencing order picking more closely. To be able to place the order picking process in context, the process of handling orders is briefly described below.

The entire process of handling customer orders begins with incoming customer orders. An order consists of one or more items. The company uses put walls to sort the orders. An example of a put wall can be found in Figure 2. At the company, they set up put walls in such a way that the person that stores the items in the correct bin is surrounded by 3 put walls.

Figure 2: Example of put wall

However, before sorting the orders in the put wall, the items are required to be in front of the put wall. When the put wall is fully or partly reserved for a wave, picking tasks can be generated. When the picking task is ready, the picking process can begin. Pickers must register for a floor after which they receive a picking task for a zone on that floor (either Hall 4 or 5). A picking task consists of items for up to 6 put walls that are in the same zone. For each put wall, the picker carries one tote. The picker is equipped with an electronic device that indicates where to pick and in which tote the picks need to be placed. At the end of a run, the totes are placed on the conveyor which transports the totes to the put walls. So, on each floor, one or multiple order pickers work at the same time, as long as there are picking tasks for zones on the corresponding floor. When this is done for each floor, all the required items arrive at the put wall. Now they need to be placed in the correct bins of the put wall, resulting in sorted orders. Then it is time for the packing process. Missing or added items can be identified before the order is packed. After that, the order is ready to be packed. After the orders are packed, the packages are sorted for the correct carrier.

According to Frazelle, a typical distribution of an order picker's working time consists of travelling (55.4%), searching (18.4%), extracting (14.9%) and others (11.3%) [4]. The company sets a target of 90 picks regarding the number of piece picks per hour and can log the times that order pickers are searching and extracting items or travelling. Analysing these times show that the order picker's working time at the company corresponds to the percentages mentioned by Frazelle.

Current performance is not entirely in line with the target. The number of picks per hour strongly depends on the zone and type of order. Certain product types are for example more time-consuming when picking. On top of that, picking Business-to-Business (B2B) orders often results in more picks per hour compared to Business-to-Customer (B2C), due to picking more pieces in the same location. However, overall the number of picks should average around 90 picks per hour. The company wants to offer competitive pricing while achieving the promised services to end-users. As order picking is responsible for the biggest part of the total warehousing costs, and travelling is the biggest part of the working time of order pickers, the motivation for this research is as follows:

Investigate in what way the total travelling distance of order pickers can be reduced.

1.3 Problem Statement & Research Objective

As travelling is responsible for around 60% of the order pickers' time, I assume the potential for improvement during this component of order picking to be present. The motivation of this research can be translated into the following problem statement:

Currently, inefficiencies before and during the order picking process are affecting the total travelling time of order picking, which in turn affects the total warehousing costs.

In Figure 3, a visualization of the problem context is pictured using a problem cluster.

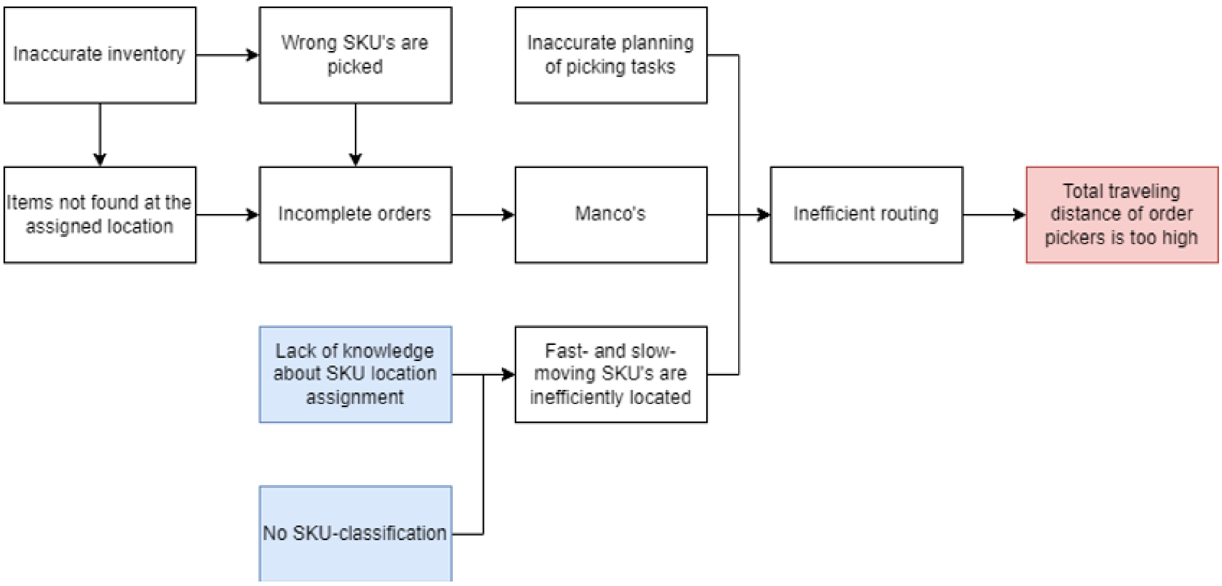

Figure 3: Problem Cluster

First of all, the way picking tasks are planned influences the total distance walked by order pickers. The closer the consecutive picks of a task are, the less total distance will be travelled. At this moment, picking tasks are manually planned. All items for orders for at most 6 put walls need to be picked from different zones. For each zone, the planner creates several picking tasks, depending on the number of items that need to be picked from the corresponding zone. So planning tasks is not just planning the closest picks in one task, because planning picking task is restricted to consist of items of at most 6 put walls. Planning this as efficiently as possible is therefore challenging. This aspect has quite some impact on the total travelling distance of order pickers.

Another inefficiency in the travelling distance of order pickers can be caused by incomplete orders at the put wall. SKUs are missing at the put wall if SKUs cannot be found by the order picker or if the order picker scans the item but forgets to put it in a tote. If the picker cannot find an item, the picker skips this item during the picking task, but the system will know that the item will be missing at the put wall. Missing items (manco's) will be checked once more by a person and if it is not there, an alternative item of the same SKU will be picked up somewhere else in the warehouse. If an item is scanned during the picking task and not present at the put wall, this becomes clear during packing. Also, in this case, the item will be picked during an extra picking route for manco's. I consider the potential for improving this aspect to be low because this is mainly caused by human errors.

Lastly, locating SKUs in the warehouse influences the order picker's travelling distance. However, it is hard to define the "right" places for SKUs. Besides, it is unknown what SKUs deserve to be stored in the "right" places. Some SKUs might be popular and likely to sell fast, while others are less likely to sell fast. Preferably, SKUs that are likely to sell fast should be stored in the "right" places. The determination of locations for SKUs based on some policy is called slotting. At this moment, the company does not classify SKUs, resulting in randomly chosen SKU locations within the warehouse. However, the size and type of the SKU are taken into account when assigning a zone of the warehouse. SKUs can be stored in 3 different storage types, depending on their size and type. Currently, the number of zones used for each storage type is as described in Table 1.

Storage type	**Zones used**
Storage type 1	12
Storage type 2	2
Storage type 3	2

Table 1: Number of zones used for each storage type

If an SKU should be stored in Storage type 1 for example, it is randomly assigned to 1 of the 12 Storage type 1 zones. It can be the case that different items with the same SKU number are located in different zones because they might be received at different moments in time. On top of that, the complexity regarding SKU classification is the constantly changing assortment of SKUs. Sold-out SKUs are not replenished by definition and new SKUs keep arriving, which means there is no historical data available for those SKUs.

As currently no classification of SKUs exists at the company, figuring out the potential for improvement in this aspect is worthwhile. Although I also consider the planning of picking tasks as an aspect with a lot of impact on the total travelling distance of order picking, the company is considering a zone for fast-moving SKUs. Therefore, for this research, inefficient placement of SKUs in the warehouse and lack of SKU classification are chosen as problems to solve (marked blue in Figure 3). To measure the impact of this research, I estimate the total travelling distance for the current warehouse layout and compare this with the estimated total travelling distance of a warehouse layout that is based on the proposed solution. The objective of this research is:

Investigate in what way the total travelling distance by order pickers can be reduced by classifying SKUs.

1.4 Scope

To maximize the probability of improvement, scoping the research is necessary. The first selection for the problem instance is regarding the processes within the warehouse. Only put away and storage will be considered in detail. Within the put away and storage process, I focus on:

- Classifying SKUs: both existing and newly introduced SKUs.
- Locating SKU classes in the warehouse.
- Determination of when and how often to classify SKUs within the process. Although receiving is not in the scope of this research, there is a possibility that SKU classification should be done right after receiving.

I will not focus on the following aspects during the research:

- Receiving(both inbound and returns), picking, packing and shipping.
- Incoming item flow of returns.
- Improving the inventory system.
- Storage types for items in the warehouse.
- Planning of picking tasks.
- Forecasting the number of order pickers.
- Alternative routing strategies for order picking.
- Human behaviour within the order picking process.
- Detailed operational implementation.

1.5 Research Questions

The research objective can be converted into the following research question:

What improvements regarding SKU-location assignment in a warehouse can be implemented to reduce the total travelling distance while the SKU assortment is constantly changing?

To answer this question, I complete the following phases: current situation, literature review, solution methods, solution analysis, implementation and conclusions and recommendations. For each phase, a sub-question is drafted below.

Phase 1: Current situation
First of all, it is important to properly map out the current situation. Knowledge about the current situation provides a starting point for this research, enabling us to know what information is needed to come to solutions for the core problem. Therefore, the first sub-question is formulated as follows:

---

1) What is the current situation at the company regarding processes within inbound/returns and outbound, performances and stock?

Phase 2: Literature review
After mapping out the current situation, I focus on the literature research. The main goal of this part is to gain insights into improvement possibilities. During the literature review, I want to gain knowledge about SKU classification, demand patterns, SKU allocation in a warehouse and performance measures during these processes. Also the classification of newly introduced SKUs, without historical data, will be reviewed during this phase.

---

2) What improvement methods regarding SKU classification and location assignment exist to reduce travelling distance based on literature?

Phase 3: Solution method
After obtaining relevant knowledge during the literature review, it is now time to compose solutions to reduce the total travelling distance. Again, I consider SKU classification and SKU location assignment when looking for solution methods. Experiments will be designed to test these solutions in a later phase.

3.1) What classification methods are most suitable for the company to classify SKUs and how can their effectiveness be tested?
3.2) In what way can these classes be used when designing zones within the warehouse?

Phase 4: Solution analysis
After composing suitable solutions, it is time to analyze these in a practical setting. During this phase, experiments are run to gain quantitative insight into the impacts of the solution method. The impact will be measured by comparing the total travelling distance based on the current warehouse layout and a warehouse layout that follows from the proposed solution method. Also, the sensitivity of the proposed solutions will be taken into account during this phase.

---

4) In what way does the proposed solution method influence the total travelling distance?

Phase 5: Implementation requirements
To make use of the proposed solution method, this phase ensures that the requirements for implementation of the most suitable SKU classification method are provided.

---

5) What are the requirements for implementation of the proposed solution method?

Phase 6: Conclusions and recommendations
Finally, after completing all previous phases of the research, conclusions and recommendations can be provided.

1.6 Approach

For each sub-question, the main activities for answering the question are mentioned in Table 2.

Phase	Activity	Description
1	Field Research	By interacting within the process, I want to quickly gain knowledge about how order picking is done at the company. During this field research, I ask questions to people to understand the processes properly, as well as asking questions to the production planner about target performance.
	Data collection and analysis	A wide variety of reports is available, such as historical data of received items and sales, picking reports and the item master. These data sets are extensive. By cleaning and transforming these data sets, I gain insight into what is currently in the warehouse, what typical orders consist of and the performance of the operations over a 2-year period.
	Reporting	Report findings to be able to conduct final report.
2	Desk Research	Search for literature that supports decisions regarding the topics SKU classification, demand forecasting, location assignment and warehousing performance measures.
	Reporting	Provide a brief review of topics related to this research.
3	Desk Research	Use the most suitable classification and storage assignment method to generate a solution. Also knowing how to test this solution and where to find the required data. This results in experiment designs to test the effectiveness of the solution.
	Reporting	Report findings to be able to conduct final report.
	Discuss findings	With production planner, inbound coordinator, and outbound coordinator.
4	Experiments	To evaluate the impacts of the candidate solution. The solution will be used to provide a new location assignment of the SKUs. For this newly provided location assignment, an estimation will be made of the corresponding overall picking rate.
	Data analysis	Evaluating the data output of the experiments, to conclude the impact of the candidate solution.
	Reporting	Report findings to be able to conduct final report.
5	Discuss findings	With production planner, inbound coordinator, and outbound coordinator. During this phase, it is important to make sure where in the process the classification of SKUs is going to be performed.
	Reporting	Report findings to be able to conduct final report.

Table 2: Activities per phase

2 Current situation

During this chapter, I provide insight into the current situation of the company. This consists of process descriptions of inbound, returns and outbound, performances of inbound, returns and outbound and stock behaviour. After this chapter, I can answer the question:

What is the current situation at the company regarding processes within inbound, returns and outbound, performances and stock?

In general, the processes at the company can be divided into 3 categories: Inbound, outbound and returns. Inbound consists of receiving items from the customer's suppliers and putting away these items somewhere in the warehouse, which I refer to as storing items from now on. The returns process is also about receiving and storing items in the warehouse. However, these items are returned by end-users. Eventually, in this way the warehouse is filled with items. This is where the outbound processes start: picking items, packing orders and shipping orders. The connection between all these processes is visualized in Figure 4.

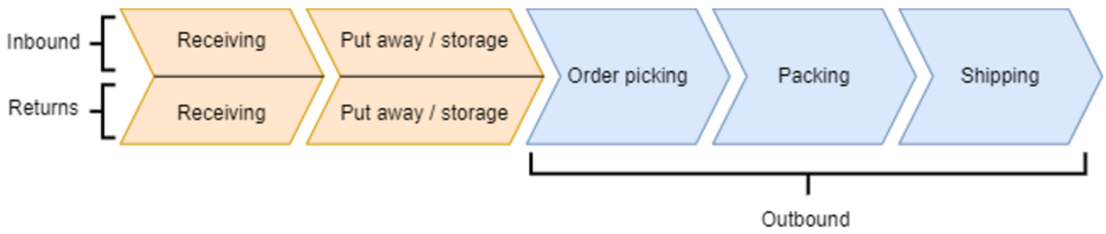

Figure 4: Warehouse processes

The company uses a Warehouse Management System (WMS) that requires the use of an electronic wearable device that is used during receiving, storage, order picking, packing and shipping. This device is equipped with a barcode scanner, enabling to add a status to items in the warehouse. In this way, the company has real-time insights into the overall status of the warehouse. Besides, a lot of data is gathered during a large variety of activities and this device prevents making errors during the processes.

2.1 Inbound

Inbound starts with unloading containers filled with items out of trucks. Every received shipment is digitally traceable through an Advanced Shipment Notification (ASN), which consists of various SKUs. The customer of the company receives this information a few days before it arrives at the warehouse. Currently, the SKU characteristics are not attached to this document.

After the containers of an ASN got unloaded, each specific item needs to be registered to the WMS. When registering an item, the SKU number is known and assigned to a unique case number, which is attached to the item with a barcode. In this way, every single item is labelled with a unique case number. So items can have the same SKU number, but not the same case number. After receiving a case number, the item is ready for storage. They are either stored in a zone for storage type 1, 2 or 3. For these storage types, respectively 12, 2 and 2 zones are available.

Given the required storage type, items are randomly stored in the warehouse. For example, if an item needs to be stored in storage type 1, it is randomly stored in 1 of the 12 storage type 1 zones. By randomly is meant that a zone is chosen where capacity is left. There is an online file that indicates the available capacity for each zone, enabling to decide where to store the next batch of items. As a result, items with the same SKU number may end up in different zones. When storing items in a zone, the case numbers of items are scanned as well as the locations in the warehouse. In this way, the WMS knows where each case number is located in the warehouse.

Inbound processes are only performed during the week. Since the company receives items from its customer's suppliers (brand owners), the quantities of SKUs and items are gathered. Over a 1-year period, the Coefficient of Variation (CV) for inbound equals 0.4. So inbound amounts vary from week to week, which means that the company has challenges regarding its use of capacity in its inbound logistics. The main varying capacity challenges are regarding the number of people to use for receiving and storage. Some capacity that is always taken into account is using supervisors and team leaders.

At this moment, the company doesn't need to look at the number of different SKUs that arrive each week because storage is done according to a Random Storage policy. However, for this research, it is interesting to give insight into the number of different SKUs that arrive, because it might influence potential classification methods. Over the 1-year period, on average around 6 items per SKU are received. The variety of SKUs arriving is a lot, which might result in challenges when handling different classes during inbound.

It is important to mention that the amount of inbound is not changeable for the company. The customer decides what items are entering the warehouse, so the amounts to receive are given. It is then up to the company to receive and store the items in the warehouse as efficiently as possible.

Another aspect to mention is the percentage of incoming SKUs for which historical data is available. History is available for SKUs that are received before, which enables forecasting methods that require historical data. For SKUs without history, other forecasting methods need to be looked for. Therefore, I provide monthly percentages of SKUs with and without history available in Figure 5.

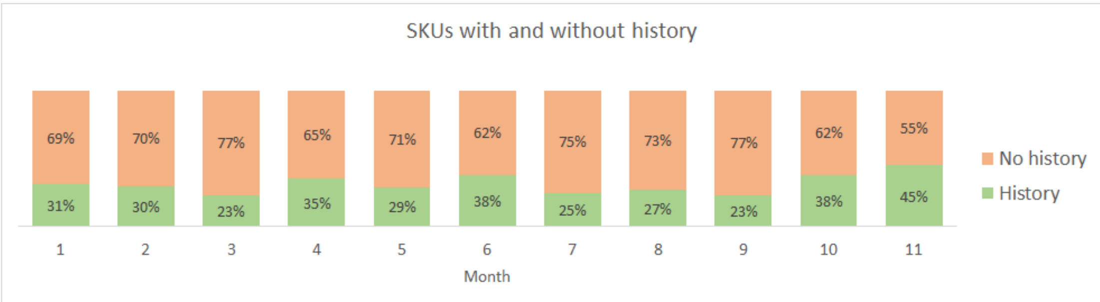

Figure 5: Inbound SKUs with and without history

For each month, I checked what SKUs were received and compared this with the SKUs that were received before this month. On average, around 32% of the SKUs were received before. Note that this also includes SKUs that were received only a few days before the month of inbound, which means that less than 32% of the SKUs have useful history to enable forecasting methods.

The storage process is part of the solution area for this research. So, the current performance regarding this process is relevant to mention. The CV of the storage productivity was 0.15 over the 1-year period. By storage productivity, I mean the weekly average number of items that are stored in an hour, which can be measured by dividing the number of stored items by the number of hours that were used for storage. Generally speaking, Random Storage has a positive impact on storage productivity, because there is no restriction for items regarding their location in the warehouse. Empty warehouse spaces can be filled quickly. However, after some time, randomly filled zones have more and more empty locations within the zone, which slows down the storage rates again. The WMS should be able to indicate the available space within each zone, but the company prefers putting away items in empty zones.

2.2 Returns

Returns start, just as inbound, with unloading containers filled with items out of trucks. However, these items are not delivered from customers' suppliers but are returned by end-users. Although these items already had been registered in the WMS before, they are registered to a new case number.

As the items are returned, they might need some recovery and a new package. Eventually, the returns are ready to store again in the warehouse. Whereas inbound items are randomly stored in zones with the correct storage types, returned products are dedicated to a specific zone: the C zone. The reasoning behind dedicated storage for returns is the assumption of before-sold items being more likely to sell fast again. In this zone, storage locations are set up more widely, with the idea of decreasing searching time for order pickers.

Summarized, incoming items are either called inbound (when received from the customer) or returns (if received from the end-user). The main differences in handling returns compared to normal inbound are:

- Returns might need some preparation before they can be packed again.
- For returns, a dedicated zone has been reserved.

Storage is also part of the returns process. The number of returns is mainly related to the number of sales, as a percentage of weekly sales is returned. On average, the weekly returns are around 39% of the weekly sales with a CV of 0.16. The storage productivity of returns (CV = 0.23) is slightly higher that the storage productivity of inbound. However, the CV is also higher, meaning that performance is less stable over the weeks. This can be the result of the restriction that returns are stored in a predetermined zone. Sometimes, this zone will be close to reaching its capacity, causing empty spaces that are further apart from each other to store.

2.3 Outbound

Outbound consists of the following processes: generate picking tasks, perform picking tasks, complete orders in the put wall, packing and shipping.

Generate picking tasks
At the company, they make use of zone picking. As described in the inbound section, items end up in different zones in the warehouse. The idea of zone-picking is to batch up to 100 customer orders and check for each zone in the warehouse what items of the batch are in that zone. This is done for a maximum of 6 batches per picking task.

Perform picking tasks
The order picker signs in for a floor, after which he/she receives a picking task for a zone on that floor. The order picker takes a trolley with 6 empty totes and starts at the south side of the zone. Having 6 totes enables the order picker to store the items for each batch separately. The wearable device of the picker tells what aisle and location within the aisle the next pick is. Arrived at this location, the picker scans a barcode of the location, after which the device tells what items to pick from the location. Thereafter, the picker scans the item's code and places it in the tote that corresponds to the item's batch. Thereafter, the location of the next pick pops up again. If items to be picked are not available in the location, the picker should mark the item as missing and receives the location of the next pick. After collecting all items for a maximum of 6 batches, the order picker can place the totes on a conveyor at the end of the zone.

An example of a route of an order picker on the 3rd floor is visualized in Figure 6. The picker starts a new task when he/she is next to the conveyor because this is where the previous task ended (only the first task of the day starts at the packing station). There, he/she receives a new task for either zone V, W, X, J, K or L. In the Figure, the zone of the pick task is K. The pickers start by walking to the south side of the zone, after which he/she visits the aisles in which items are located that need to be picked. In this way, the picker moves in an S-shape back to the north side of the warehouse, where all items of a picking task can be placed on a conveyor.

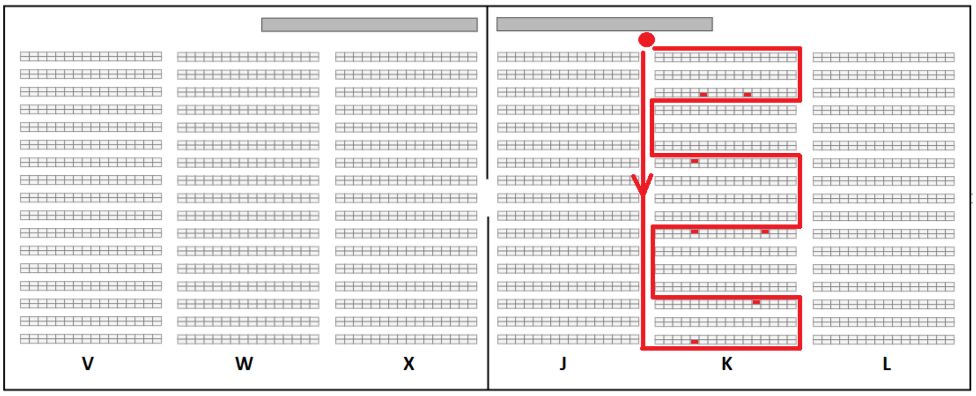

Figure 6: Example route of order picker

This conveyor transports the totes to their destination: the put walls. For each floor, one or multiple order pickers are working at the same time as long as picking tasks exist for zones on the corresponding floor. If this is done for every floor, all items of a batch are picked.

Complete orders in the put wall
From all zones, the totes are now available in front of the put walls. The put wall is responsible for sorting all orders of 1 batch and has the same number of bins available as the maximum number of orders in 1 batch. The wearable device at the put wall tells the person in what bin an item should be placed. In the end, all orders of a batch are sorted.

Packing and shipping
When orders are completed, there is someone else on the other side of the put wall, picking complete orders up and packing them. Before packing the order, the completeness of the order is checked. In case of missing items, the manco process starts, which essentially means another trip through the warehouse to collect the missing item. After packing the items of an order, the package is placed on another conveyor, which transports the packages to the shipping department. Arrived at the shipping department, packages are sorted to end up in a place for the correct carrier.

Just as for inbound, the measures for outbound processes are logged by the company. For each date, the pulled case numbers are available. These numbers can be matched with the correct SKU number, enabling to give insights into the characteristics of all pulled items, such as SKU number, brand, product type, gender, size, colour, retail price, season and storage type. Looking into these data, the following findings came up.

On average, the CV value of weekly demand was 0.25. This weekly variation in the number of sales is mainly the result of promos by the customer. Regarding the variety of SKUs sold, on average between 1 and 2 items per SKU are sold, which means a large variety of SKUs is sold each week. Outbound activities are performed each day, so not only from Monday till Friday.

As mentioned earlier, SKUs are either stored in storage types 1, 2 or 3. For the majority of weeks, around 90%, 5% and 5% of all sales are picked from zones for storage types 1, 2 and 3 respectively. The season of an SKU can be divided into Fall/Winter, Spring/Summer or No season. The season an SKU is designed for tends to correspond to the actual seasons. Regarding product types, items are classified into 61 different product types. The top 5 bestselling product types were responsible for around half of the total sales. The company warehouses items of many brands for their customer. Some brands are more represented in the sales data than others. To indicate how important the most-selling brands are, I looked at the contribution of the top 5 brands to the total sales. The top 5 brands do contribute a lot to the total sales per week. However, almost every week these brands do not cover half of the total sales. When looking at gender, on average the following division is visible: around 61% female, 27% male, 9% kids and 3% unisex. Regarding sizes of items, it is difficult to indicate some general remarks, as sizes are measured differently for categories. Therefore, a method to cluster similar sizes is required later on.

2.4 Stock

Combining the knowledge about what went in and out of the warehouse, the number of items and SKUs in stock over the weeks is known. In a 2-year period, the number of items stored increased by around 100%, while the variety in SKUs increased by around 68%.

Some general remarks on the stock characteristics:

- Season: 40% fall or winter, 51% spring or summer and 9% has no season.
- Gender: around 66% female, 23% male, 5% kids and 6% unisex.
- Storage type: around 87%, 7% and 6% for storage type 1, 2 and 3 respectively.
- Brands: top 5 is responsible for almost 26% of all stored items.
- Categories: top 5 is responsible for around 53% of all stored items.

Note that these values are not the same as the values mentioned in the demand characteristics. However, the values are in line with each other, which is expected because you can only sell what you have in stock.

Another interesting aspect to look into is the capacity usage within the warehouse. In Figure 7, the capacity of each zone is pictured, as well as the capacity that is currently used. To indicate that zones are randomly filled with items, it is interesting to look at the number of different SKUs in each zone as well.

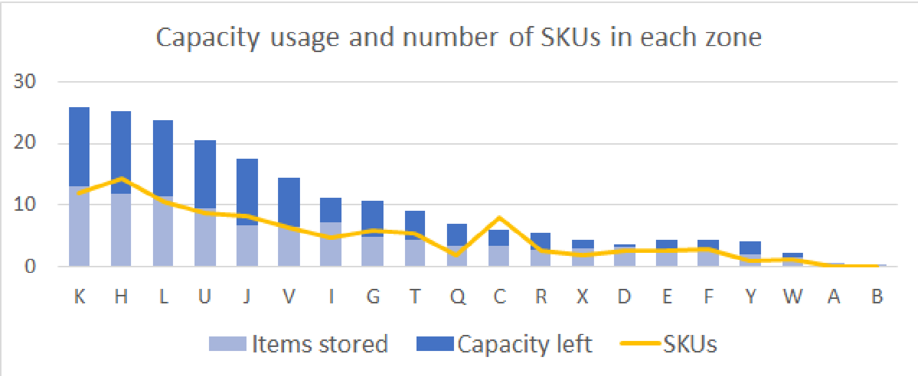

Figure 7: Capacity usage

Note that this Figure is based on factorized values. The main purpose of this Figure is to validate the variety of SKUs in each zone and indicate that some zones have more capacity than others. Most zones still have the capacity left to store items. By looking at the capacity left for each zone, the company can determine where to store new items. Compared to zones that are reserved for inbound, the returns zone (zone C) has a lot of different SKUs stored. When looking at the ratio number of SKUs stored to the capacity used for each inbound zone, this is in line with the Random Storage policy. Zone H, J, G and T have relatively more SKUs stored. For this ratio, looking at A and B is not relevant as these zones are not used for inbound storage.

2.5 Conclusion

Weekly **inbound** vary from week to week regarding the number of items (CV = 0.4). On average, around 6 items per SKU arrive in the warehouse. Remarkable is the fraction of SKUs that are received before, which is only 32%. Inbound is randomly allocated in the warehouse. Weekly **outbound** also varies from week to week regarding the number of items (CV = 0.25). On average, between 1 and 2 items per SKU are sold each week, which means a large variety of sold SKUs. Weekly **returns** are on average around 39% of the number of sales with a CV value of 0.16. Returned items are stored in a dedicated zone for returns. In a 2-year period, the number of items in **stock** doubled, while the number of SKUs increased by around 68%. This indicates that it becomes even more important to manage the current capacity correctly.

Where items are located in the warehouse has a great impact on the order picking performance. However, it is challenging to add some logic to the storage assignment of inbound. Especially the following aspects should be taken into account for the storage assignment:

- Both inbound and outbound are widely distributed regarding the variety of SKUs.
- The company does not influence what SKUs come into the warehouse and how many.
- Only a small fraction of SKUs has SKU-specific history available.

These characteristics of inbound and outbound are taken into account for the literature review in the next chapter.

3 Literature review

Warehouses have a variety of essential functions to fulfil. The basic requirements for warehousing activities are receiving SKUs, storing SKUs, receiving orders from customers, retrieving SKUs and completing orders and shipping those to the customers [5]. The main focus of this research is to reduce travelling distance by classifying SKUs within classes and determining a policy for these classes. Storage location assignment is a decision to be made on a tactical level, whereas the impact is big on the effectiveness of order picking [6]. During this chapter, I review a number of components related to this research, starting with the reference of the problem in Section 3.1. Thereafter, Section 3.2 covers a variety of storage policies. Then, in Section 3.3, I elaborate on demand forecasting. Lastly, in Section 3.4, I include a way of estimating travelling distance.

3.1 Reference of problem

Improving the efficiency of order picking can be achieved in different manners. One way is by optimizing the distribution of items in the warehouse [7]. In literature, the problem of allocating items to locations in the warehouse is called a Storage Assignment Problem (SAP). The goal of SAP is to improve order picking's efficiency by effectively locating products in the warehouse [8]. So, the solution to this problem is a strategy for the allocation of incoming goods in a warehouse. When SKUs to be picked are located closer to each other, they can be picked more quickly as the travel distance between picks becomes less [9].

Other researchers in the literature mention the problem as Storage Location Assignment Problem (SLAP). The goal of the SLAP is the determination of the optimal allocation of SKUs to slots [10], where slots are the different locations within the warehouse. The S(L)AP includes a variety of parameters, such as storage area design, storage space availability, warehouse storage capacity, physical characteristics of the products, arrival times, and demand behaviour [11].

In literature, proposed solutions are either based on exact methods, storage policies & rules, heuristics, metaheuristics, simulation, other trends and support tools or multi-criteria [11]. Depending on the warehouse characteristics, certain methods are more suitable than others. The problem is classified as NP-Hard [12] and is usually harder to solve when the numbers of locations and products involved increase [6]. A logistics centre can have lots of different SKUs for example, causing impossible calculation times for exact methods. Instead, strategies for such warehouses are often heuristically developed [13]. To gain insight into existing storage policies, those are reviewed below.

3.2 Storage policies

The subject of storage policies is described in many ways. For this literature review, the following terms were considered: SKU allocation, slotting, storage policies, and SKU location assignment. Most of the time, deciding where to locate SKUs individually is too complex or time-consuming due to the large number of SKUs that are stored in a warehouse [14]. Storage policies are used to support the decision of assigning warehouse locations to items on a larger scale compared to the SKU level. Eventually, this ends up in a warehouse filled with items, enabling it to handle customer orders. According to [15], storage policies can be divided into 3 main strategies: Random Storage, Dedicated Storage and Class-Based Storage. Each of those is briefly described below.

3.2.1 Random Storage

For the Random Storage policy, items can be located anywhere in the warehouse, as long there is space. Putting away items randomly causes effective use of space in the warehouse, as every space in the warehouse can be allocated without restrictions. Since the locations of stored items vary, it is necessary to keep the order pickers well-informed about where the products are stored. Besides, the travel times of the Random Storage policy are generally higher than those of other storage policies. For products with unpredictable or highly variable demand, where any form of dedicated slotting can lead to high-value slots closest to the outbound blocked with slow-moving products, random dynamic assignment is preferred to average out the risk [16].

3.2.2 Dedicated Storage

In contrast to Random Storage, Dedicated Storage allows only specific SKUs at specific locations. As a result, this policy consumes more space in the warehouse. For products with highly accurate forecasts, a dedicated measure-based assignment strategy will perform better than a Random Storage policy [16].

3.2.3 Class-Based Storage

Class-Based Storage is the trade-off between Dedicated and Random Storage policies [5]. SKUs are grouped into different classes based on some criteria and can be stored in available areas for these classes. Storage within an area is random. Class-Based Storage is especially useful because of its simplicity of implementation, manageable maintenance and ability to cope with product mix and demand variations [3].

However, Class-Based Storage requires a method to determine the number of classes to use, as well as what SKUs belong to what classes. According to [17], the travel times for storage and picking items becomes longer if the number of classes is large. This explains why a small number of classes are commonly used in practice. Their research concludes using between 2 and 5 classes is most of the time optimal. Example methods to divide items into classes are the ABC analysis and FSN analysis, which both divide items into 3 different classes.

ABC analysis
The ABC analysis is a method that divides items into classes based on the Pareto principle in combination with criteria, such as profit contribution per product, volume value per item or purchase value per supplier over a while. The items responsible for around 80% of the criteria value are put in class A, the next 15% in class B and the remaining number of items in class C.

FSN analysis
The FSN analysis is a method to classify items in warehouses as Non-Moving (70%), Slow-Moving (20%) or Fast-Moving (10%) [18]. As a result, warehouses can be divided into a fast-moving area, a slow-moving area and a dead stock area. Classes are assigned by calculating the *Average Length of Stay* (ALS) and/or the *Consumption Rate* (CR) of each SKU in the warehouse. Those values are calculated with the formulas **(1)** and **(2)**.

$$ALS = \frac{Total\ number\ of\ days\ in\ warehouse}{Number\ of\ received\ items} \quad \textbf{(1)}$$

$$CR = \frac{Total\ number\ of\ items\ sold}{Time\ period} \quad \textbf{(2)}$$

For example, 6 months of data is available and an SKU is received 10 times in total within these 6 months (no need to be received in 1 batch). 4 items were sold on day 10 after receiving, 5 items on day 20 after receiving and 1 item is not sold after day 40 after receiving. Then the total number of days in the warehouse is 4*10 + 5*20 + 1*40 = 180 days, which results in $ALS = \frac{180}{10} = 18$ days and $CR = \frac{9}{6} = 1.5$ per month. For the CR value, demand is assumed to be constant over time. In this way, SKUs that are consumed more on average or stayed on average less time in the warehouse are classified as fast-moving.

According to Frazelle, the area for fast-moving items needs to be easier accessible than the other areas [4]. When designing an area for fast-moving items, you need to determine where to locate the area, which items to include and how many of each item to include to minimize the travelling and searching times during order picking [19]. It may also be advantageous to group some products in the pick area, either for ease of replenishment or picking efficiency [16]. Concluding, using the class-based policy improves order picking efficiency as a result of storing highly turned-over items in more accessible locations in the warehouse [6].

3.3 Demand forecasting

Most of the time, a Class-Based Storage policy requires demand knowledge. By looking at demand patterns, future demand can be forecasted. How difficult demand forecasting might be, it is important in the current business environment, which is characterized as dynamic, as it supports the goal of many companies to reduce operational costs and improve sales and customer satisfaction [20]. Different products with different characteristics have different demand patterns, which should be taken into account when forecasting. Therefore, companies should categorize SKUs and use appropriate forecasting methods for the different categories [21].

In the fashion retail industry, the level of forecasting accuracy plays a crucial role in retailers' profit [22]. A complexity regarding demand forecasting, especially In the fashion industry, is due to the large variety of products, the short life cycle of the products, the short selling season and constantly changing trends in fashion, which in turn is often accompanied by a lack of historical data of specific products [23], [24], [25]. This environment describes the company's case as well. On top of that, the company cannot use forecasting information to restock their inventory, as they do not influence the SKUs that enter the warehouse. Therefore, the forecast doesn't need to be as detailed as possible. Just before a batch of different SKUs arrive, they need to know what SKUs belong to what classes. According to van Kampen and van Donk, periodic reclassification is needed to either increase the competitive strength of the company or to reduce risks [26].

Current forecasting techniques are generally divided into two groups: classical methods based on mathematical and statistical models and modern heuristic methods using artificial intelligence techniques [22].

3.3.1 Classical methods

According to [27], demand patterns are usually divided into intermittent, lumpy, smooth and erratic demand. When variation in quantity and times between demand occurrence is both low, demand can be considered as smooth. If the variation in quantity is high and the variation in times between demand occurrences is low, this is called erratic demand. Intermittent demand is characterized by low variation in demand quantity, but high variation in times between demand occurrences. When variation in demand quantity is also high, demand is characterized as lumpy.

Different demand patterns require different methods. When historical data is available, there are several demand forecasting methods, such as using simple moving average, weighted moving average, exponential smoothing, linear regression, double moving average, Holt method, Winters method and Croston method [28]. Currently, Warehousing Management Systems are commonly used, enabling to record products' order frequency, which in turn can be used during the storage assignment process for better warehousing performance.

Forecasting the demand for SKUs daily might be challenging, especially when demand is intermittent or erratic. Nevertheless, forecasting accuracy is an important input for inventory holding and replenishment decisions [29]. Treating each SKU individually when forecasting is often too time-consuming. To reduce managerial efforts, SKUs can be clustered. SKUs that have attributes such as name, size and colour in common can be clustered when forecasting [30].

Besides, the availability of historical data on specific SKUs is not always the case. Sometimes, demand forecasting for new products is required: products that are newly launched or products that are just not sold yet by the company. As mentioned in Chapter 2, around 32% of incoming SKUs are received before. No historical data on specific SKUs does not mean that there is no data available. Data on similar products might be interesting as well.

3.3.2 Modern methods

The emergence of big data, cloud computing and improved computing storage and processing capabilities have led to increased availability and accessibility to large volumes of data, making Machine Learning techniques a viable option for demand forecasting in the industry. Machine Learning based forecasting combines learning algorithms to identify underlying demand drivers and uncover insights by processing an excessive number of predictor variables and determining the ones that are significant [23]. Machine Learning can be used for either regression problems or classification problems, where a regression problem usually is about predicting a quantity (continuous) and a classification problem is about predicting a binary variable (categorical) [31].

Instead of forecasting exact amounts of sales, it is also possible to approach forecasting as a classification problem, where a product is either likely to sell (fast), or not. This is especially relevant for this case, where the company has no impact on what products come into the warehouse. They cannot use the forecasting information to refill their inventory with products that are forecasted to sell a lot. Information about if products will sell or not is then more relevant. Classification algorithms have in common that they can predict binary variables based on characteristic variables. A well-known classification or regression algorithm that proved itself in several cases is Random Forest. For example, [32] used Random Forest in their demand forecasting method to forecast demand profiles of new products, which is comparable to our case.

Random Forest

Random Forest is based on decision trees. Decision trees develop an anticipating model for a target variable based on characteristic variables, where the target variable is referred to as a classifier and the characteristic variables are often called features. A Random Forest model constructs decision trees and connects these with feature values, which are the nodes in a decision tree. The branches are the values that the node can have. By taking the mean of the responses that have the same characteristic variables, the prediction for an observation is generated [33]. Random Forest builds decision trees based on random samples and takes the majority of the sample outcomes as a classifier. As a result of the randomly selected features, a reduction of correlation between trees can be expected [34]. 2 important parameters for building a Random Forest are the number of decision trees and the number of features to randomly select for each decision tree. Using more trees in a Random Forest model improves prediction performance, but increases computational time as well. Therefore, it is a trade-off between performance and computational time.

Random Forest can be used for the development of a prediction model. First, the input data requires preparation. This includes handling missing data points and outliers and making sure that columns are categorical. After data preparation, the data can be divided into a train and test set. Then, a model can be trained with the train set. This set includes the features and the outcome of the binary variables that are going to be predicted. Eventually, the model can be used to predict the classifier in the test set [33].

There are several measurements discussed in the literature to evaluate the performance of the prediction models, which can be called performance metrics. An important metric is the confusion matrix, which illustrates the performance of a prediction model based on predictions and actual values of the classifier. Besides, this matrix indicates how many predictions are correct (either positive or negative), false positive or false negative, enabling measuring the predictive power of the model. Important measures out of the matrix are the accuracy, sensitivity, specificity, True Negative Rate (TNR) and True Positive Rate (TPR) [35]. Their formulas can be found in Table 3.

Accuracy	**Sensitivity**	**Specificity**	**TNR**	**TPR**
$\frac{TN+TP}{TN+FN+FP+TP}$	$\frac{TN}{TN+FP}$	$\frac{TP}{FN+TP}$	$\frac{TN}{TN+FN}$	$\frac{TP}{FP+TP}$

Table 3: Performance measures confusion matrix

The accuracy tells us what the ratio is between correct predictions and all predictions. The sensitivity indicates the ratio between the number of true negatives and all actual negatives, whereas the specificity indicates the ratio between the true positives and all actual positives. Lastly, the TNR indicates the ratio between true negatives and all predicted negatives, whereas the TPR indicates the ratio between true positives and all predicted positives.

3.4 Total travelling distance

In the company's case, they perform picking according to the zone picking principle, where multiple orders are batched. For each batch, they cluster the items that are in the same zone. Doing this for at most 6 batches results in the final picking tasks, where each task consists of picks from the same zone. Eventually, the total travelling distance by order pickers mainly consists of the sum of the travelling distance for each picking task.

Pickers can travel in different ways through zones to complete a picking task. The main options for routing are the Traversal strategy, the Midpoint strategy, the Largest Gap strategy or a hybrid strategy[36], which are visualized in Figure 8.

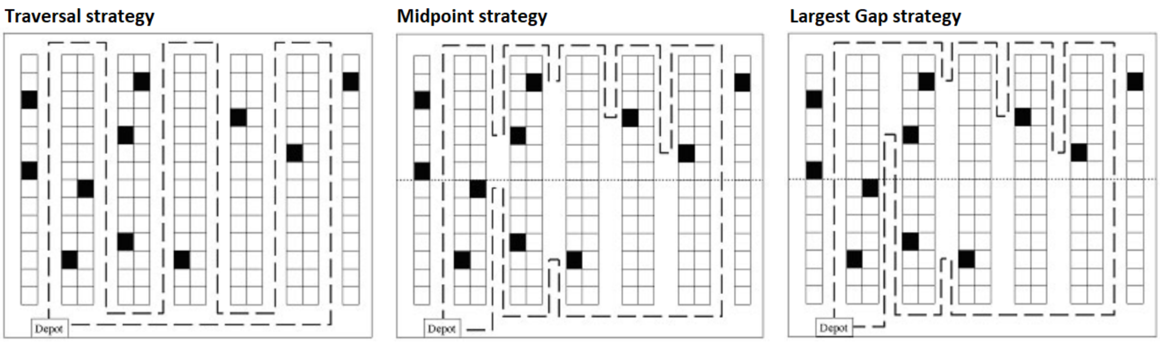

Figure 8: Routing strategies

At the company, the sequence of the items to pick in a task is based on the traversal strategy, where order pickers are expected to traverse through the complete aisle if a pick is present in that aisle. As a result, the picker moves through the zone in an S-shape.

Hall [36] also proposed a way to estimate the travelling distance by an order picker to complete a picking task when picking according to the traversal strategy. To complete a picking task, an order picker starts at the depot, picks up all items from a zone and returns to the depot again. Hall's method estimates the expected travelling distance in the main aisle and the expected travelling distance through the aisles. The required parameters for calculating the expected distance in the y and x directions are the length of x and y, the number of picks from the zone(N) and the number of aisles in the zone(M). In Figure 9, M equals 4. To calculate the expected distances in the y and x directions, the formulas **(3)** and **(4)** can be used.

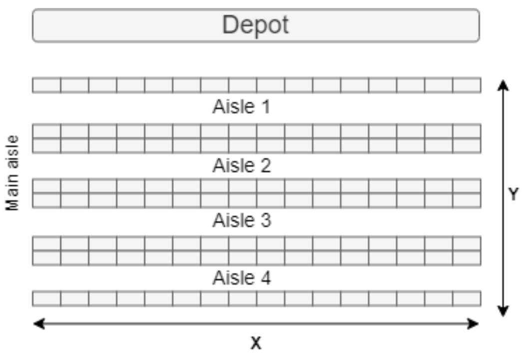

Figure 9: Length of x and y

$E[Dy] = 2y\ (\frac{N}{N+1})$ **(3)**
$E[Dx] = 0.5x + Mx\ (1 - (\frac{M-1}{M})^N)$ **(4)**

Summing both values results in the total expected travelling distance for a picking task.

3.5 Conclusion

- The problem can be defined as a Storage (Location) Assignment Problem, where the goal is to reduce total travelling distance of order picking.
- Instead of using a Random Storage policy, introducing Class-Based Storage supports the company in deciding where to put away incoming items. Dedicated Storage is not suitable for the company, because of the large variety of SKUs and the constantly changing SKU assortment.
- The introduction of Class-Based Storage requires a method to classify SKUs. A suitable class division for the company can be based on an FSN analysis, where the *Average Length of Stay* is calculated for each SKU and used to classify an SKU as fast-, slow-, or non-moving. Because only a few items per SKU are sold, taking the *Consumption Rate* into account is not recommendable. In the company's case, SKU-specific historical data is limited. Random Forest enables us to find demand patterns within large data sets without relying on SKU-specific historical data.
- As a result, every incoming SKU can be classified as fast-moving, slow-moving or non-moving, which can then be used to put away SKUs in dedicated areas. Using a dedicated area for fast-, slow- and non-moving items is an effective way to improve picking efficiency.
- To approximate the total travelling distance of order pickers for a given zone layout, Hall [36] provides formulas to approximate the distances in the x and y direction during a picking task. These formulas can be used for a zone layout where multiple classes have their own area.

4 Solution design

During this research, the purpose is to provide the company with a solution regarding location assignment for incoming goods based on SKU classification. In this way, a reduction in the overall travelling distance in the warehouse by order pickers should be achieved. Based on the current situation and the Literature review, I provide a solution design in this chapter.

In the current situation, incoming SKUs are randomly allocated in the warehouse. Instead, I add logic in the process of allocating incoming SKUs in the warehouse in the form of a Class-Based Storage policy. The following 3 aspects will be elaborated on by me:

1) **Classification**: a way to classify SKUs
2) **Slotting**: What to do with those classes
3) **Evaluation**: Measuring the total travelling distance by order pickers

4.1 Classification

A variety of class-based policies has been discussed in the literature review. During this review, it emerged that an FSN analysis is a suitable way to indicate whether an SKU is fast-moving, slow-moving, or non-moving.

4.1.1 FSN analysis

By performing an FSN analysis, incoming SKUs are labelled as fast-, slow- and non-moving. The FSN analysis requires 2 data sets: inbound data and pull data. This combination of data sets provides the Length of Stay for every incoming item, which enables me to calculate the Average Length of Stay (ALS) for each SKU:

$$ALS_s = \frac{\sum_{i=1}^{max(i)} Length\ of\ Stay_{s,i}}{\sum_{i=1}^{max(i)} 1} \quad \textbf{(5)}$$

In this formula, s is the SKU, while i corresponds to the item number of the SKU. For example, 5 items ($max(i)$) of the same SKU are received (no need to be received at the same time). For each of those items, the length of stay in the warehouse is known. The sum of those lengths of stays divided by 5 is the ALS value for the SKU.

After calculating the ALS values for each SKU, I determine the definitions for fast-, slow- and non-moving items. Around the first 10% of the cumulative ALS is labelled as F; So the SKUs with the lowest ALS values that together are responsible for 10% of the sum of all ALS values. The next 20% of the cumulative ALS is classified as S. The remaining 70% is labelled as N. This 10/20/70 division is based on literature, but I will change these later on when performing a sensitivity analysis. For the remainder of this report, I use the term F/S definition as ALS value that differentiates between fast- and slow-moving SKUs and S/N definition as the ALS value that differentiates between slow- and non-moving SKUs.

Concluding, the output of this analysis consists of the *Average Length of Stay* for each SKU, which can be translated into a class label that is either fast-moving, slow-moving, or non-moving. When new inbound arrives and the SKU is labelled by the FSN analysis, these labels can be used to classify the SKU accordingly. Nevertheless, in the company's case, SKU-specific history is not always available. As mentioned in Chapter 2, around 32% of the incoming SKUs are SKUs that have historical data available. Therefore, I need some method to use the fast-, slow- or non-moving labels to classify SKUs without requiring SKU-specific history. In Literature, Random Forest has proven itself a suitable algorithm for this classification.

4.1.2 Random Forest Classification

By using Random Forest, SKUs can be classified as fast-, slow-, and non-moving based on similar SKUs where history is available. By similar SKUs, I mean SKUs with similar values for their features. The development of a Random Forest classification model has, in general, the following phases: data gathering, data modification, model training, model testing, and evaluating the model, where some phases are recurring. Based on the evaluation, the Random Forest model might be trained, tested, and evaluated on improved conditions.

Data gathering
Development of the classification method requires a clean data set, consisting of SKUs with their features and class label. To obtain this data set, I combine inbound data, pull data and SKU information data. The **SKU information data** consists of rows of data (entries), each representing an SKU, and columns that provide information about the SKU (features). Important features for the clean data set are "*Brand*", "*Product type*", "*Gender*", "*Size*", "*Colour*", and "*Season*", for which I provide a description in Table 4. The **inbound data** contains data entries each representing a single incoming item. So the number of entries in this data tells us the number of items that came into the warehouse. Important information from this data source is the number of items per SKU that come into the warehouse and the inbound date. The **pull data** contains data entries for each picked item in the warehouse and provides the outbound date.

Feature	**Description**
Brand	The brand name of the SKU.
Product type	The product type is about the function of the SKU.
Gender	The SKU is designed for kids, females, men or unisex.
Size	The size of the SKU.
Colour	The colour of the SKU.
Season	The season the SKU is designed for. Either Spring/Summer, Fall/Winter or No Season.

Table 4: Description of features

All data sets have overlapping columns, so can be merged. The combination of the inbound data and pull data enables us to determine the number of days that each item of inbound stayed in the warehouse. This information, together with the class definitions, is required to determine whether the response variable "Class" is fast-moving (F), slow-moving (S) or non-moving (N). Adding features of the SKU to the data results in a data set where each entry represents an SKU and is provided with its features and the number of days the SKU was on average in the warehouse.

Data modification
The Random Forest algorithm package (randomForest) I use in R requires features to have at most 53 different categorical values. In the initial cleaned data, the features *Brand* and *Size* have more than 53 different values. The main reason for sizes having more than 53 distinct values is that the UK, US and Europe have different sizing tables for equal sizes. Besides, different categories have different ways of sizing as well. Nevertheless, similar sizes can be clustered in more generic forms, such as small, medium or large. For brands, clustering is not possible. Besides, I consider the brand being one of the most important features regarding its impact on being fast-, slow or non-moving. Therefore, I do not modify this feature. To be able to use Random Forest anyway, I clustered all non-top 52 brands (monthly) as the same value. On average, this contains around 10% of the SKUs. The top 52 brands are based on the number of SKUs inbound in that month. Another essential step in cleaning the data for the classification model is combining similar values of categorical values. Some feature values are described differently, while the same value is meant. After combining similar values, the number of categorical values for each feature is described in Table 5.

Feature	Brand	Product type	Gender	Size	Colour	Season
# values	53	30	4	15	15	4

Table 5: Number of categorical values for each feature

Feature-wise, the data is clean. Entry-wise, however, more preparation is necessary. I removed entries that are picked in B2B orders because these orders do not represent actual demand at the moment of selling. Instead, other businesses restocked their inventory, or the brand decided to take back their unsold items. Also returned items are not taken into account, as there is already a dedicated zone in use for returns. Lastly, complete rows of data are needed. The SKU information data does not contain feature information of all SKUs in the warehouse, which is the reason for incomplete data rows. Rows in which the values 0 or empty exist need preparation. When information for all features is missing, I deleted the data entry, because classifying SKUs with no feature information available would be random. However, if only 1 feature value is missing, I use Random Forest to impute missing values based on similar SKUs.

After this last modification, the data is clean and ready to use for further model development. To develop a classification model with this data, I determine what part of the data can be used for the training phase.

Training model
After obtaining the S/N definition, we know what data to use when training a Random Forest model. Only training data for which is known whether it stayed on average longer in the warehouse than the S/N definition should be used. Based on the training set, Random Forest is used as an algorithm to train a classification model. During this training phase, the model detects causal connections between features and the response variables. The Random Forest model predicts classes for the test data. These predictions are either the label F, S or N.

Testing model
The test set consists of inbound data to be classified as fast-, slow, or non-moving. Only features of the test set are used as input for the trained model, resulting in classified test data.

Evaluating model
For the evaluation of the model, I only test on data for which is known whether the SKUs were actually fast-, slow- or non-moving. For example, I do not know yet for inbound of the previous month whether SKUs are sold within 2 months. When the final classification model is ready, it can be used on data without knowing the actual classes of SKUs. To measure the performance of the model, the confusion matrix provides a lot of insights. Based on this matrix, the classification method can be evaluated on the accuracy, specificity per class and the True Positive Rate (TPR) per class. When the TPR of a class is close to 1, it means that most predictions of this class are correct. The specificity tells us something about detecting a class. If this value is close to 1, it means that the model detects most of the classes with its predictions.

Limitations for the classification model
Not for all SKUs on stock, feature information is up to date, which reduces the percentage of representing input data for the classification model. Besides, response variables of new SKUs can be classified as long as feature values are used in the training model. As a result, response variables of SKUs with newly introduced features cannot be estimated yet. To classify SKUs of brands that are not available in the training data, I train a Random Forest Model without the "*Brand*" feature. Lastly, due to the size of the training data, I limit the number of trees to build in the Random Forest model.

4.1.3 Final classification model

To classify inbound, a Random Forest model is essential. The development of a Random Forest model should be performed periodically as visualized in the flowchart in Figure 10, and is offline available for new inbound to arrive afterwards.

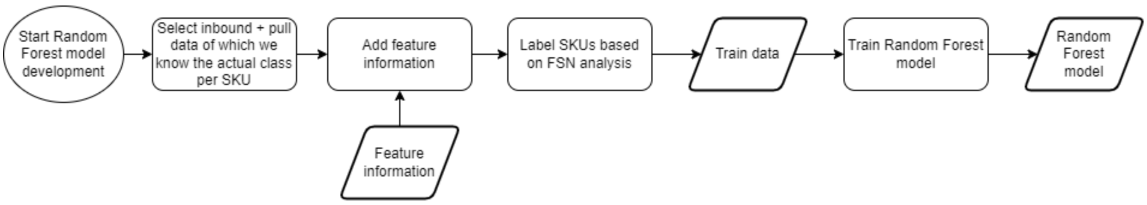

Figure 10: Development of Random Forest model

I select training data from prior inbound (historical data) of which is known whether SKUs are fast-, slow- or non-moving. This depends on the S/N definition. For example, if the S/N definition turns out to be 100, I should only train with data of which is known whether SKUs stayed on average more than 100 days in the warehouse. However, excluding too much data before the moment of classification is not desirable, because this also excludes valuable information for the classification. Thereafter, the FSN analysis adds class labels to the training data, resulting in the training data. Each entry consists of an SKU number, the features and the class. When new inbound data becomes available, the Random Forest model classifies SKUs as visualized in Figure 11.

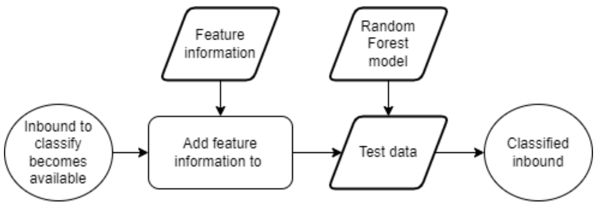

Figure 11: Classification process

Newly arrived inbound data consists of data entries, each representing an SKU. Adding feature information to every SKU result in the test data. The latest developed Random Forest model can then be used on the test data to label SKUs as fast-, slow- or non-moving, which is essentially classified inbound.

The performance of the prediction can be evaluated by calculating the performance measures from the confusion matrix, such as the True Positive Rates per class, the specificity per class and the accuracy of the model. Using the classification method on inbound data of a 1-year period provides:

- $Inbound_{class,day}$: the number of items selected per class per day
- $Outbound_{class,day}$: the number of picks per class per day

Combining these enables us to have an estimation of the inventory level for each class on each day within the 1-year period. The inventory level for each class at the end of each day ($InventoryLevel_{class,day}$) is calculated as follows:

for $day = 1$
$InventoryLevel_{class,day} = Inbound_{class,day} - Outbound_{class,day}$ **(5)**

for $day \geq 2$
$InventoryLevel_{class,day} = InventoryLevel_{class,day-1} + Inbound_{class,day} - Outbound_{class,day}$ **(6)**

In this way, I obtain for each day what percentage of items should be stored as F, S or N, which can be translated to the **required space** per class. This calculation also gives insights about the **number of picks** per class per day.

4.2 Slotting

The classification results in an indication of what percentage of space should be used for storage of fast- slow and non-moving items. Now I need to determine how to divide this space in the proposed warehouse design. The main goal is to reduce the total travelling distance of order pickers. Because of the company's zoning strategy, the travelling distance by pickers mainly consists of their routes during a picking task. These routes start at the conveyor and end at the conveyor. In between, the pickers travel through a zone and visit the aisles where they need to pick at least 1 item.

The company stores its items in 3 different storage types. It is not recommendable to mix picks from different storage types in the same picking task, because they have different sizes. For example, an order picker can take 100 items from storage type 1, but 100 items from storage type 2 or 3 would be impossible due to the capacity of the picker. Therefore, I advise to keep using different zones for different storage types. Nevertheless, the classification of SKUs can be used to divide classes within zones, as pictured in Figure 12. Note that this Figure visualizes only 1 zone. The idea is to provide a design for all zones in the warehouse. In this way, the company can use this zoning division for zones of all storage types.

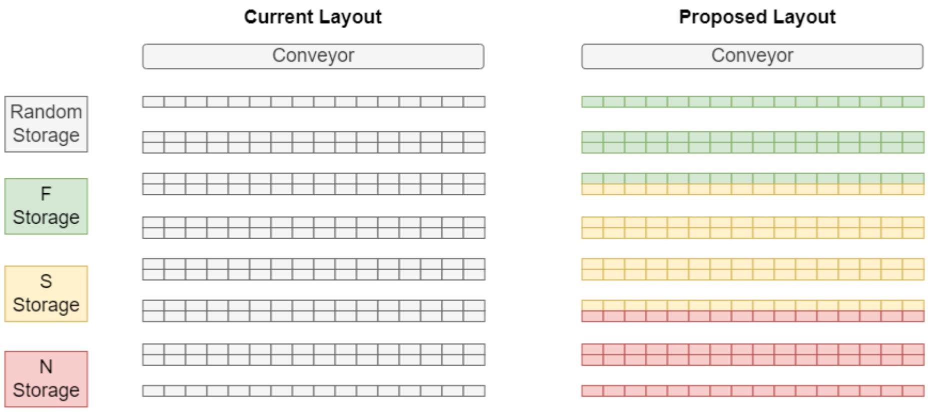

Figure 12: Current and proposed zone layout

In both layouts, the same number of aisles are used. In the company's warehouse, they use around 56 aisles in a zone. The classification output provides the number of aisles needed for fast-, slow- and non-moving items.

Putting away fast-moving items closest to the conveyor has a variety of advantages. More picks can be expected per aisle in aisles closer to the conveyor, which results in more picks per distance travelled compared to randomly filled zones. On the other hand, putting non-moving items in a place far from the conveyor eliminates travelled distances, as fewer aisles are expected to be visited in this area. This also reduces the probability of needing to visit aisles on the south side of the warehouse. Besides, putting away fast-moving items closer to the conveyor also reduces the total weight an order picker is carrying on average. Most picks are expected from fast-moving aisles, which are visited at the end of a picking task and are closer to the conveyor.

Filling the zone according to the layout can be performed as follows. When items at inbound are scanned at the receiving department, the system should indicate whether the item is classified as fast-, slow- or non-moving. Accordingly, the item can be placed in a trolley for the corresponding class. In this way, trolleys are filled with items for specific classes. There is a heat map available for each floor, in which we can check where capacity is available to store. For putting away fast-moving items, aisles with capacity left closest to the conveyor should be selected for put away. Putting away non-moving items should be done in aisles furthest away from the conveyor. In between, slow-moving items can be placed.

4.3 Evaluation

The zone layouts can be evaluated based on formulas proposed by Hall, where I estimate the expected travelling distance in the y and x directions during 1 picking task in a given area. The following information is required to approximate the expected travelling distances:

- N_{Class}: Average number of picks per picking task from class
- M_{Class}: Number of aisles dedicated to class
- Y_{Class}: Vertical length of an area for a class (from first to last aisle)
- X_{Class}: Horizontal length of an area for a class
- Routing according to Traversal Strategy (S-shape)

In Figure 13, a visualisation of what is meant by Y_{Class} and X_{Class} can be found for both the current and proposed zone layouts. For both cases, the sum of all N_{Class} is equal as well as the sum of all M_{Class}

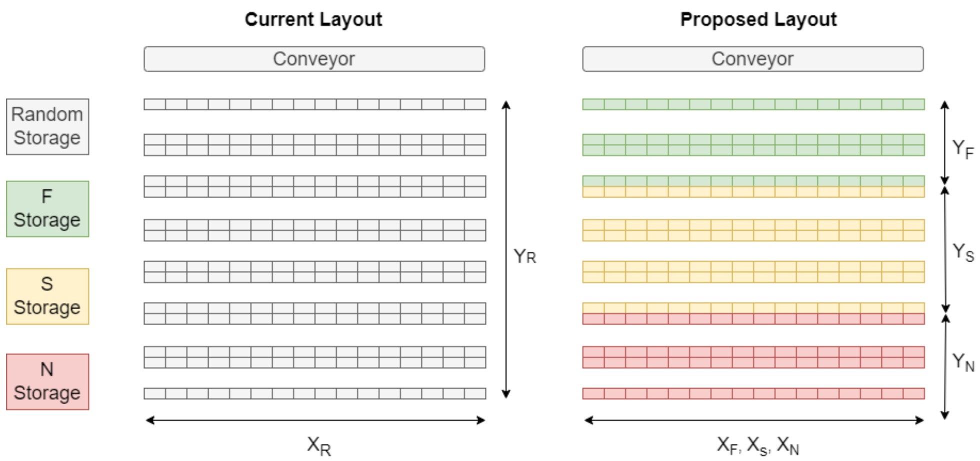

Figure 13: Distance estimation parameters

For the current zone layout, using the method by Hall would end up in one estimation of the travelling distance in the y and x directions, because this layout considers 1 class: Random (R). This results in 1 value for N and M because all aisles are used for the same class and all picks come from the same class. For this estimation, the expected distance in the y direction is calculated as follows:

$E[D_Y] = 2Y_R\ [\frac{N_R}{N_R+1}]$ **(7)**

$[\frac{N_R}{N_R+1}]$ is an estimation of the last aisle to visit, where picks are assumed to be uniformly distributed over the area. This estimation is multiplied by 2Y, which is twice the maximum y direction of the area. The formula of the expected distance in the x direction is:

$E[D_X] = M_R X_R\ [1 - [\frac{M_R-1}{M_R})^{N_R}]]$ **(8)**

In this formula, $[1 - [\frac{M_R-1}{M_R})^{N_R}]]$ is the probability that at least 1 pick is available in an aisle. This essentially tells the probability of traversing an aisle. By multiplying this by the number of aisles (M_R) and the length of an aisle (X_R), we arrive at the expected travelling distance in the x direction.

However, in the proposed layout I use 3 areas, for 3 classes. Essentially, this means that I need to estimate travelling distances in the x and y directions for 3 areas. We have an indication of how many aisles are dedicated to fast-, slow- and non-moving each day (M_F, M_S and M_N). Besides, the classification method provides us with information about the expected number of picks from each class per day (N_F, N_S and N_N). Combining this information, I design distance approximation formulas for order pickers routing based on Hall (1992) for the proposed zone layouts.

$$E[D_Y] = 2Y_F + 2Y_S + 2Y_N \ (\frac{N_N}{N_N+1}) \ \textbf{(9)}$$

In formula **(9)**, I assumed at least 1 pick from N Storage, which results in travelling twice through the F and S areas in the y direction. Expecting at least 1 pick from the N area is reasonable, as a picking task consists of around 100 picks. The total travelling distance in the x direction is approximated by formula **(10)**:

$$E[D_X] = M_F X \ (1 - (\frac{M_F-1}{M_F})^{N_F}) + M_S X \ (1 - (\frac{M_S-1}{M_S})^{N_S}) + M_N X \ (1 - (\frac{M_N-1}{M_N})^{N_N}) \ \textbf{(10)}$$

Using this way of estimating for both the current zone layout and the proposed layout, performances regarding their approximated total travelling distance by order pickers can be compared.

4.4 Conclusion

The goal of the solution design is to reduce the total travelling distance by order pickers in a warehouse. The travelling distance of order pickers mainly consists of order pickers completing picking tasks, where picking tasks start and end at the conveyor while picking around 100 items in a zone. To achieve this goal, the proposed solution design is to introduce a classification method, enabling the storage of incoming SKUs according to a Class-Based Storage policy. For the company's case, I propose to classify SKUs according to the FSN analysis. During this analysis, SKUs are classified based on their Average Length of Stay.

As a small number of SKUs have history available, only using class labels as the outcome from the FSN analysis is not suitable for the company. Instead, I use these FSN outcomes to label training data and develop a Random Forest model. It is recommended to develop a Random Forest model periodically. The Random Forest model predicts labels on the test data, resulting in classified inbound. The input parameters for this classification require a sensitivity analysis in the next chapter, to check how dependent the performance is on the given parameters and what parameters settings should be used in the final classification method.

This classification can be used to store fast-, slow- and non-movers in different aisles of a zone. By concentrating items that are likely to sell fast in aisles closer to the conveyor, we can expect a reduction in total travelling distance. Likewise, concentrating items that are not likely to sell or sell very slowly to aisles far from the conveyor also has a positive impact on reducing the total travelling distance.

By using the classification method on inbound data of a 1-year period, we gain information about the required space in a zone for each of the classes F, S and N. Besides, the expected number of picks out of each class becomes available. This information is important when designing the new zone layout and indicating the improvement potential compared to the current layout.

5 Solution analysis

In this chapter, I provide insights regarding the proposed solution. The proposed policy is to classify incoming SKUs and use the aisles closest to the conveyors for fast-moving SKUs, the aisles in the middle for slow-moving SKUs and the aisles furthest away from the conveyor for non-moving items.

Therefore, the first section of this chapter will provide insight into the performance of the classification method. Thereafter, I use the classification method to determine the new zone layout and measure the performance of this layout by using an experimental setup. In the end, the improvement regarding the total travelling distance as a result of the proposed policy compared to the current policy will be covered. Therefore, I will use the experimental setup for the current policy as well. To indicate how dependent the solution method is on various parameters, I also provide a sensitivity analysis.

5.1 Classification

The proposed classification method requires a variety of parameters. First, an FSN analysis is performed to label training data as F, S or N. Performing this analysis on data from the previous year, the resulting class definitions based on ALS are as follows: the F/S definition is 75 and the S/N definition is 150. I also start with using 100 trees in the Random Forest, while selecting 3 features randomly for every tree. In this way, the computational time is reasonable and the model trains well. Another important step in the development of a Random Forest model is to select only features that add value to the performance of the classification model.

5.1.1 Feature importance

In this section, I want to confirm that each feature contributes to the accuracy of the Random Forest classification model. The considered features at this moment are "*Brand*", "*Product type*", "*Gender*", "*Size*", "*Colour*", and "*Season*".

If a feature is not contributing to the performance of the classification, it will just be noise and make it harder for the Random Forest algorithm to detect underlying patterns. To get some idea of the importance of each feature for the classification performance, the mean decrease in the Gini index provides insights. This index essentially measures each feature's contribution to the homogeneity of the nodes and leaves in the resulting random forest. The higher this value, the higher the importance of the feature. The features ranked on their importance can be found in Table 6.

Feature	**Mean Decrease in Gini Index**
Brand	14.876
Colour	10.514
Product type	8.624
Size	8.000
Season	3.199
Gender	3.025

Table 6: Feature importance

Although some features are more important than others, all features have importance for the performance and should therefore be included in the final model.

As this classification method classifies SKUs based on their feature values, it is interesting to check what values per feature are more likely to end up being fast-moving. Therefore, I provide for each feature the values ranked on their Average Length of Stay in Figure 14.

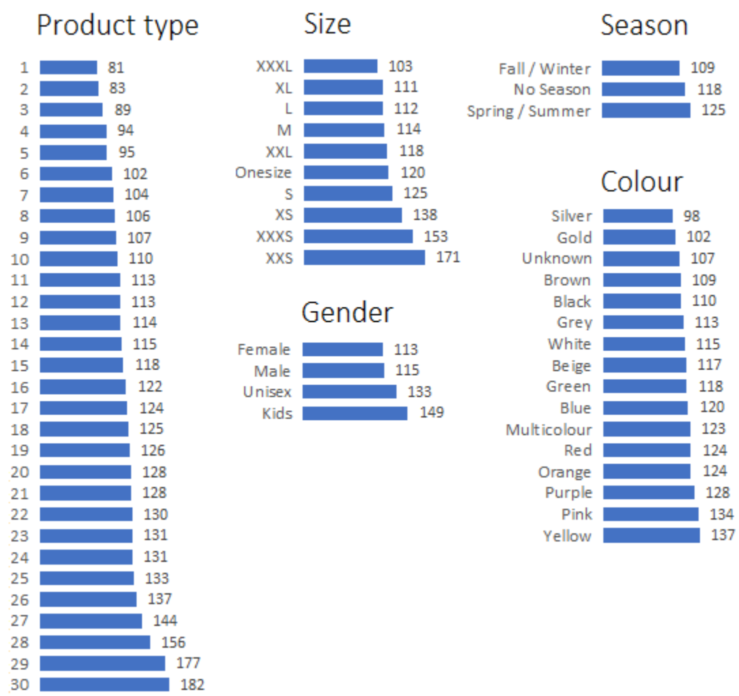

Figure 14: Feature values ranked on ALS value

On average, no feature value stays on average less time in the warehouse than the F/S definition. Regarding the colour of an item, silver items sell the fastest on average. Remarkable is that larger sizes sell faster on average compared to smaller sizes. Items designed for fall or winter stay less time in the warehouse compared to items designed for the summer or spring. Also, items designed for females and males sell faster on average than items for unisex or kids.

5.1.2 Performance metrics

Using the proposed classification method results in the confusion matrix in Table 7, where the percentages represent the fraction of the total inbound. Only inbound data is used for which is known whether items are sold within 150 days or not. This is mandatory to calculate the *ALS* value enabling the determination of the SKU's actual class.

		Actual class		
		F	**S**	**N**
Prediction	**F**	16.1%	10.7%	20.6%
	S	3.0%	2.3%	5.1%
	N	10.3%	7.9%	24.0%

Table 7: Confusion matrix as result of proposed classification method

We can now use these numbers to provide the summary in Table 8. In conclusion, 34 per cent of the predicted fast-moving SKUs are correct, while 55 per cent of all actual fast-moving SKUs are detected by the classification method. Regarding slow-moving SKUs, 22 per cent are predicted correctly, while 11 per cent is detected. For the non-moving SKUs, 57 per cent is predicted correctly, while 48 per cent is detected.

Class	**Actual**	**Predicted**	**TPR**	**Specificity**
F	29.4%	47.4%	0.34	0.55
S	20.9%	10.4%	0.22	0.11
N	49.7%	42.2%	0.57	0.48

Table 8: Performance summary

It is always a trade-off between predicting correctly as much as possible (TPR) and detecting as many as possible out of the actual classes (specificity). Predicting accurately is helpful, but if that prediction affects only a small number of items, the desired effect is not achieved. On the other hand, it is nice to detect a lot of the actual classes, but if you also select a lot of other classes, the effect of having an area for fast-, slow- and non-moving items vanish. The accuracy combines these metrics. Overall, the accuracy of the classification is 42.4 %.
However, the performances of classification models should not be compared only based on accuracy. When the majority of SKUs belong to the same class, the accuracy of a model can be very high by classifying all SKUs as that class. This phenomenon is referred to as the no information accuracy. To avoid this, also TPR values and specificities of each class should be considered.

The model is not performing well, especially regarding the classification of slow-movers. Remarkable is what classes SKUs actually belong to when classified as slow-moving. These fractions are in line with SKUs classified as non-moving. Apparently, the classification method is not great at distinguishing between slow- and non-movers. Therefore, it is interesting to check how the classification model performs when classifying into 2 classes: fast-moving (F) or not-fast-moving (NF).

For this classification, I use the **F/NF definition**, which is the *ALS* value to differentiate between fast- and not-fast-moving SKUs. Initially, I use the same value for the F/NF definition as for the F/S definition. Classifying into 2 classes enables me to use more data to train because it requires knowing whether SKU's *ALS* values are below or above 75. Instead of excluding 5 months of training data, 3 months can be excluded. The resulting confusion matrix is shown in 9.

		Actual class	
		F	**NF**
Prediction	**F**	11.7%	20.6%
	NF	17.7%	50.0%

Table 9: Confusion Matrix 2 classes

These numbers result in the summary in Table 10. In conclusion, 36 per cent of the predicted fast-moving SKUs are correct, while 40 per cent of all actual fast-moving SKUs are detected by the classification method. Regarding not-fast-moving SKUs, 74 per cent are predicted correctly, while 71 per cent is detected. Overall, the accuracy is 61.7%.

Class	**Actual**	**Predicted**	**TPR**	**Specificity**
F	29.4%	32.3%	0.36	0.40
NF	70.6%	67.7%	0.74	0.71

Table 10: Performance summary 2 classes

Classifying into 2 classes achieves higher accuracy than classifying into 3 classes. However, this does not indicate what the impact is on the reduction in total travelling distance. Therefore, I go on with classifying into 2 and 3 classes for the sensitivity checks in the next section, to eventually validate the impact on the total travelling distance.

5.1.3 Sensitivity

To gain insights into the importance of the parameter settings for this classification method, I perform a sensitivity analysis in this section. During this analysis, the classification method is used with different parameter settings compared to the settings in the previous section [37]. The analysis is a method to measure the impact of uncertainties of one or more input variables on uncertainties of output variables [38]. In the end, the analysis improves the prediction of the model, as it shows the interaction between different parameters and their responses.

Changing Random Forest parameter settings
The development of a Random forest model requires parameters such as the number of trees to use in the Random Forest, as well as the number of features to randomly select for each tree. Both have an influence on the performance of the model, where especially the number of trees in the forest influences the computational time. Using double the number of trees in the forest results in around doubling the computational time. Therefore, it is worthwhile to validate whether increasing or decreasing the number of trees has an impact on the prediction performance.
If for example using double the trees only improves the classification a little bit, a small value for the number of trees in the Random Forest model can be used instead. For this sensitivity analysis, I consider the values 50, 100 and 150 for the number of trees and the values 2, 3 and 4 for the number of features to select randomly. This results in 9 different combinations for these settings. For each scenario, the confusion matrices and their corresponding performance measures can be found in Appendix A and Appendix B.

2 classes: Using a value of 2 for the number of features to select randomly performs better regarding accuracy for all numbers of trees to use. As expected, using more trees in the forest increases the accuracy of the classification model. However, this is only a small improvement compared to the extra amount of computational time. On top of that, when the size of the training data slightly increases, the model is not even able to build 150 decision trees in the model, because a normal computer is not able to handle the size of such a model. Therefore, I go on with building a Random Forest model with 50 decision trees while selecting 2 features at random for each tree.

3 classes: Also when classifying into 3 classes, selecting 2 features for each tree in the Random Forest achieves the best results. Besides, using more trees in the model only increases the performance a little bit. So the robustness of the model enables us to use 50 trees while selecting 2 features randomly.

Changing *ALS* value(s) for the class differentiations
For the 2-class classification, I used an F/NF definition of 75. For the 3-class classification, I used an F/S definition of 75 and an S/N definition of 150. These values are based on the FSN analysis of the year before. To check how sensible the classification performance is regarding this value, I also performed the classification method with different *ALS* values to differentiate between classes.

2 classes: For this sensitivity analysis, I consider values of 55 to 325 in bins of 10 for the F/NF definition. Using different values for the F/NF also ends up with different fractions of SKUs being actually F and NF. Lower values result in few SKUs being actual F class while using higher values results in a large fraction of SKUs being actual F class. As a result, the accuracy for these cases can be high because of no information accuracy; When most SKUs belong to the same class, classifying all SKUs as this class results in high accuracy. In extreme cases, this would mean that classifying all SKUs the same achieves very high accuracy. Nevertheless, this would be the same as the current situation, where all SKUs are treated as 1 class.

Instead, the proposed classification method determines for each of the F/NF definitions what percentage of space to use for each class and how many items are picked from each class. In the end, the trade-off is to select fewer aisles for F with fewer picks from F versus selecting more aisles for F and having more picks from F. Somewhere in between, there is a number of aisles to use for each class where the expected total travelling distance is lowest.

For each of the F/NF definitions between 55 and 325, I use the classification method with 50 decision trees in the Random Forest, while selecting 2 random features for each decision tree. When classifying into 2 classes (F and NF), the required spaces based on the proposed classification method are visualized in Figure 15. The underlying numbers can be found in Appendix C

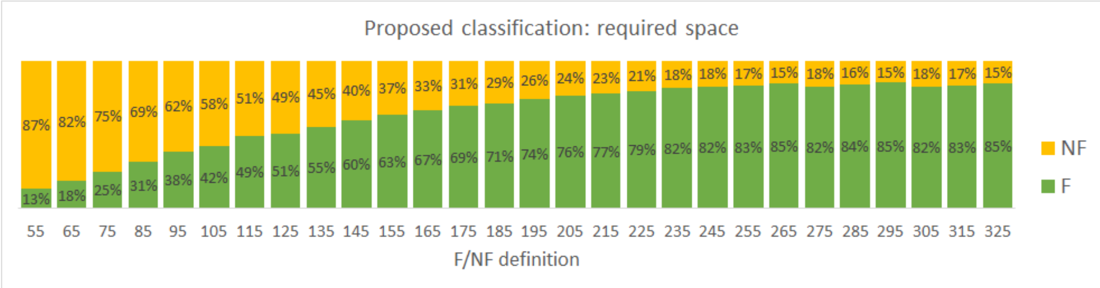

Figure 15: Required spaces for proposed 2-class classification method

As expected, the higher the F/NF definition, the larger the required space for the F class. We can also observe that the required space for this class reaches around 80% when using an F/NF definition of 235 and stabilizes around this value when using higher values for the F/NF definition. As using a higher value for the F/NF definition results in more SKUs being F class, the same pattern for the number of picks from each class can be expected. The average daily picks per class based on the proposed classification method can be found in Figure 16.

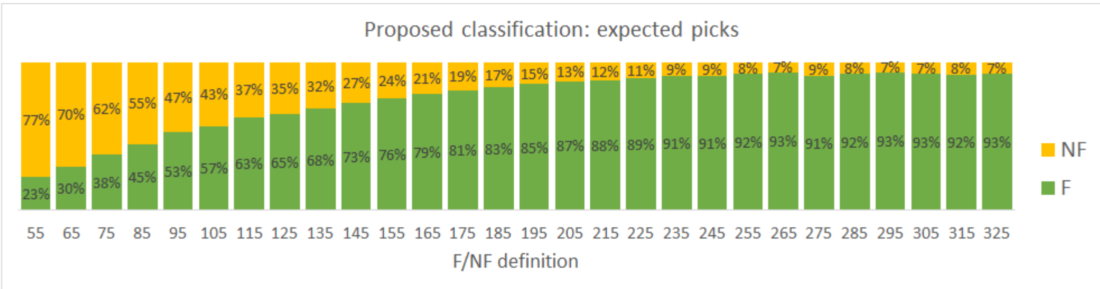

Figure 16: Expected picks (proposed 2-class classification method)

To check the maximum improvement potential of using Class-Based Storage within each zone, I provide the required zone spaces and the number of picks for each class based on perfect classification in Figures 17 and 18.

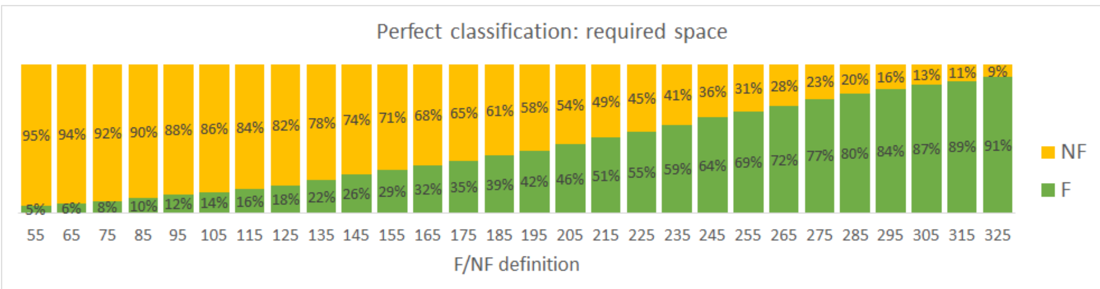

Figure 17: Required spaces for perfect classification

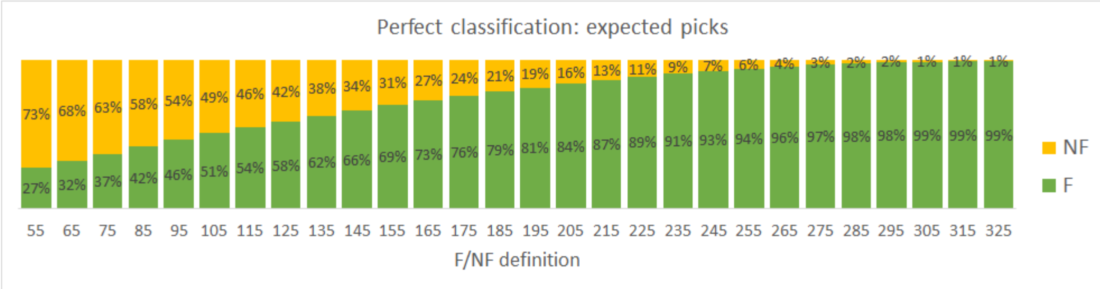

Figure 18: Expected picks (perfect classification)

Compared to the required space and expected number of picks as a result of the proposed classification method, I observe a smooth increase in the fraction of space to use for F when increasing the F/NF definition. Also, the expected number of picks from the F class increases smoothly when increasing the F/NF definition. In the next section, I combine the required space and number of picks for each class for each F/NF definition to validate what values achieve the largest reduction in total travelling distance of order picking. I perform this analysis for both the proposed classification and a perfect classification to indicate what reduction in travelling distance is achieved and could be achieved if classification is done perfectly.

3 classes: For this sensitivity analysis, I consider values of 55, 105, 155, 205 and 255 for the F/S definition and values 105, 155, 205, 255 and 305 for the S/N definition. Combining these settings, where the S/N definition is required to be larger than the F/S definition, results in 15 scenarios. Also for the 3-class classification, the classification method determines for each scenario what percentage of space to use for each class (Figure 19) and how many items are picked from each class (Figure 20). The underlying numbers can be found in Appendix D.

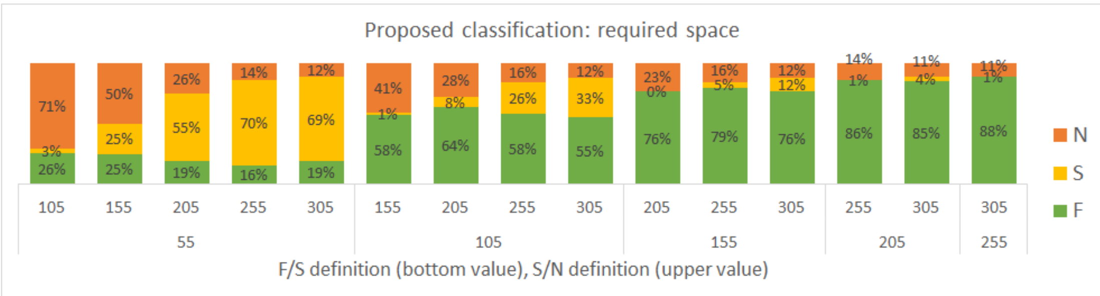

Figure 19: Required spaces for proposed 3-class classification method

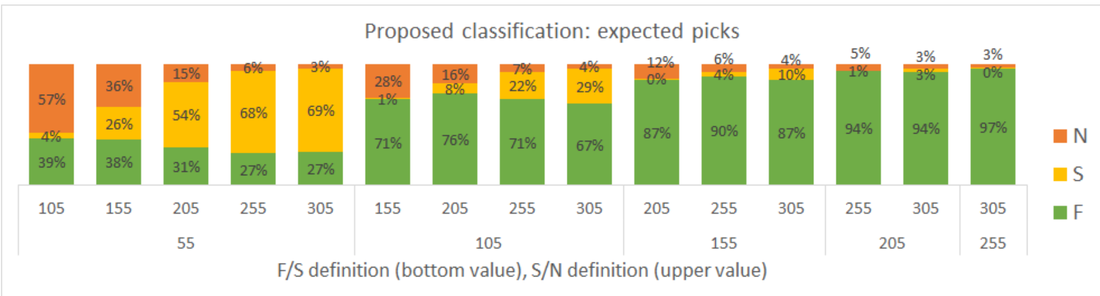

Figure 20: Expected picks (proposed 3-class classification method)

Based on the perfect classification for each of the F/S and S/N definitions, the required space and number of picks for each class can be found in Figures 21 and 22.

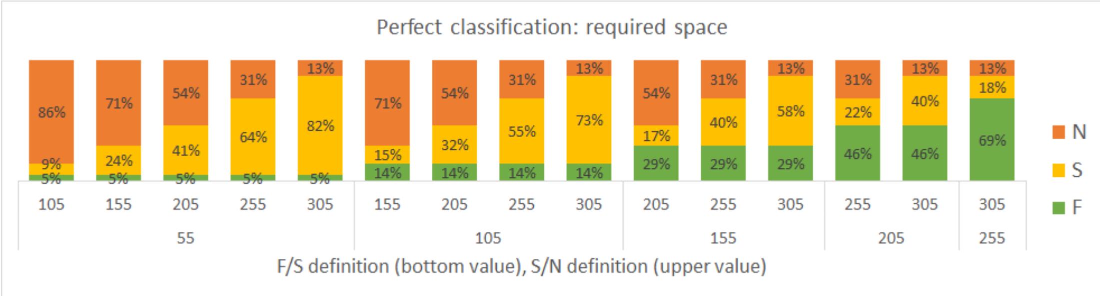

Figure 21: Required spaces for perfect 3-class classification

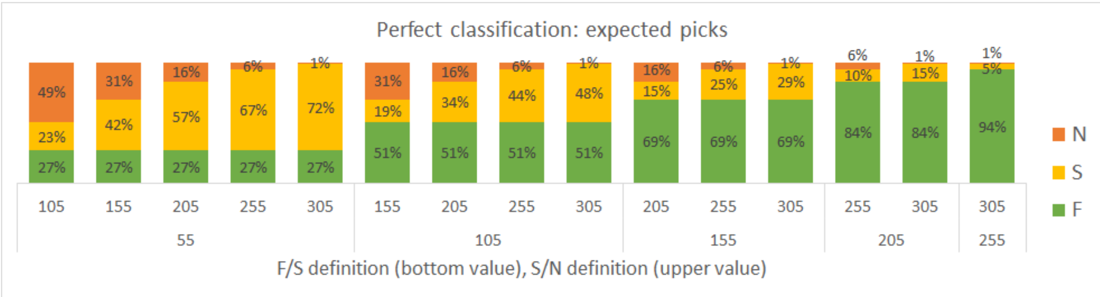

Figure 22: Expected picks (perfect 3-class classification)

Also for the 3-class classification, I combine the required space and number of picks for each scenario in the next section to validate what values achieve the largest reduction in total travelling distance of order picking.

5.2 Zone layout performance

I propose to introduce Class-Based Storage as an allocation policy instead of allocating all SKUs randomly in the warehouse. The goal of this change in policy is to reduce the total travelling distance by order pickers. By using the classification method as proposed in the previous chapter, classified SKUs can now be allocated to dedicated areas within zones for the corresponding class. This information can be used to design a new zone layout. To measure the reduction of the total travelling distance by order pickers, I first elaborate on an experimental setup to estimate the travelling distance by order pickers in a given zone layout. Thereafter, I measure the performance of zone layouts with this experimental setup: the layout based on Random Storage (current situation), the layouts based on the proposed Class-Based Storage (proposed situation) and the layout based on perfect Class-Based Storage. In the end, a comparison of these performances can be provided.

5.2.1 Experimental setup

The structure of the experimental setup is as follows: The warehouse starts with empty zones. Then, zones are filled by using a storage policy. The storage policy provides the parameters to generate a zone layout: The number of classes and required aisles for each class in each zone. As a result, I design a zone layout. By using the expected number of picks from each class in the warehouse within a given time window, the total travelling distance for a given zone layout can be approximated. For this experimental setup, the time window equals a 1-year period.

The travelling distance by order pickers mainly depends on how picking is planned. Optimizing picking routes is not the goal of this research. According to the company, most of the time a picking task is released when around 100 picks are available in a zone. Pickers start their task at the conveyor and walk to the south side of the zone. Then they move in an S-shape through the zone, while picking all items of the task (traversal routing strategy). In this way, they end at the conveyor again, ready to start a new task. To estimate the length of 1 picking task, I use the following input parameters:

- M_{Class}: For each class, the number of aisles.
- N_{Class}: For each class, the number of picks.
- Y_{Class}: For each class area within the zone, the length of the area.
- X_{Class}: For each class, the width of the aisles.

M_{Class} and N_{Class} follow from the classification output. Y_{Class} depends on M_{Class}, while X_{Class} is the same for every class. Next to these input parameters, the following assumptions are made when estimating the travelled distances.

- Picking tasks are released if 100 picks are available in a zone. When multiple classes are considered in a zone layout, the sum of all N_{Class} equals 100.
- The number of picking tasks per day equals daily demand divided by 100.
- Each picking task starts and ends at the conveyor.
- As all aisles dedicated to the same class are filled with items of the same class, I assume a uniform distribution of the daily picks for the aisles of a class.

For each day, the fraction of aisles used for each class is known, as well as the number of picks from each of the classes. Note that picks from returns, B2B or stock received before the start of the 1-year time period are not taken into account. So for each day, the required parameters to approximate the travelling distance by order pickers for a given zone layout are known. This approximation is performed for the current zone layout, the proposed zone layout and a perfect zone layout. This last layout is to check the upper limit of the improvement potential of using Class-Based Storage within zones.

5.2.2 Current layout

Currently, the company is using a random allocation policy. Essentially, this means that all items are handled the same way, which can be seen as using 1 class. A visualisation of the current layout of zones can be found in Figure 23.

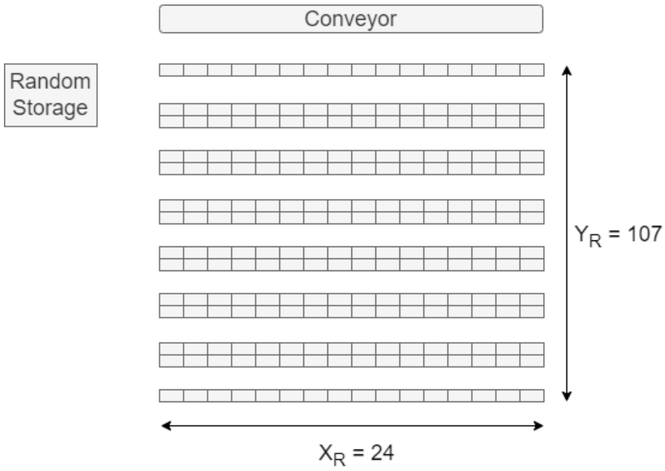

Figure 23: Current zone layout

Using this layout provides us with the parameters

- $M_R = 56$ (number of aisles used for Random(R) Storage)
- $N_R = 100$ (number of picks from Random(R) Storage)
- $Y_R = 107$ (maximum distance in y direction for Random(R) Storage)
- $X_R = 24$ (distance in x direction for Random(R) Storage)

These parameter settings, as well as the daily picks, enable us to approximate the total travelling distance of order pickers within zones each day (EDT_{Day}) with the formula:

$EDT_{Day} = (E[D_Y] + E[D_X]) * \frac{Daily\ demand}{100}$ **(11)**

where

$E[D_Y] = 2 * 107\ [\frac{100}{100+1}]$ **(12)**

$E[D_X] = 50 * 24\ [1 - (\frac{50-1}{50})^{100}]$ **(13)**

Calculating this estimation for every day in the 1-year time window, the total distance approximation equals **14,479 kilometres**.

5.2.3 Proposed layout: 2 classes

The 2-class zone layout stores fast-moving (F) and not-fast-moving (NF) SKUs in different areas within the zone. The classification method provides the required sizes for F and NF for a variety of scenarios. In these scenarios, F/NF definitions from 55 to 325 are used. Those required sizes per class can be used in the zone layout as pictured below.

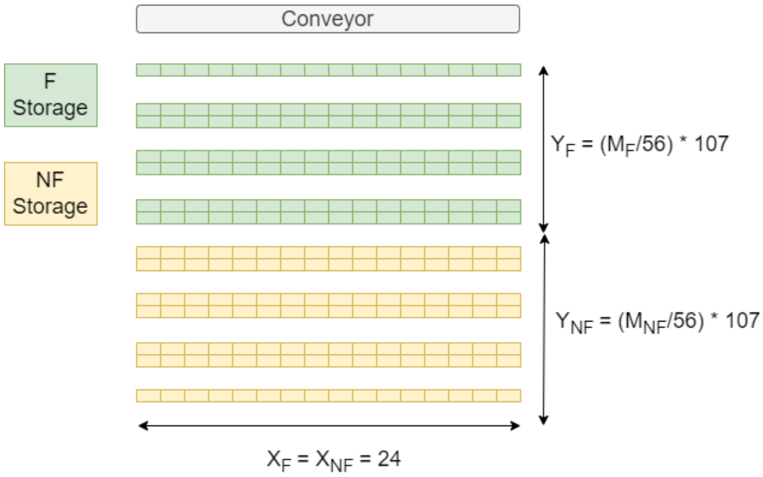

Figure 24: Proposed 2-class zone layout

For all scenarios, X_F and X_{NF} are 24. However, the values for Y_F, Y_{NF}, N_F, N_{NF}, M_F and M_{NF} depend on the outcomes of a specific scenario. For all scenarios, $Y_F + Y_{NF} = 107$, $N_F + N_{NF} = 100$ and $M_F + M_{NF} = 56$. Now the total travelling distance by order pickers within zones on a day (EDT_{Day}) can be approximated by using the following formula:

$$EDT_{Day} = (E[D_Y] + E[D_X]) * \frac{Daily\ demand}{100} \quad \textbf{(14)}$$

where:

$$E[D_Y] = 2Y_F + 2Y_{NF}\ [\frac{N_{NF}}{N_{NF}+1}] \quad \textbf{(15)}$$
$$E[D_X] = M_F * 24\ [1 - (\frac{M_F - 1}{M_F})^{N_F}] + M_{NF} * 24\ [1 - (\frac{M_{NF}-1}{M_{NF}})^{N_{NF}}] \quad \textbf{(16)}$$

Calculating this estimation for every day for every scenario, the approximated reduced travelling distances are as visualized in Figure 25.

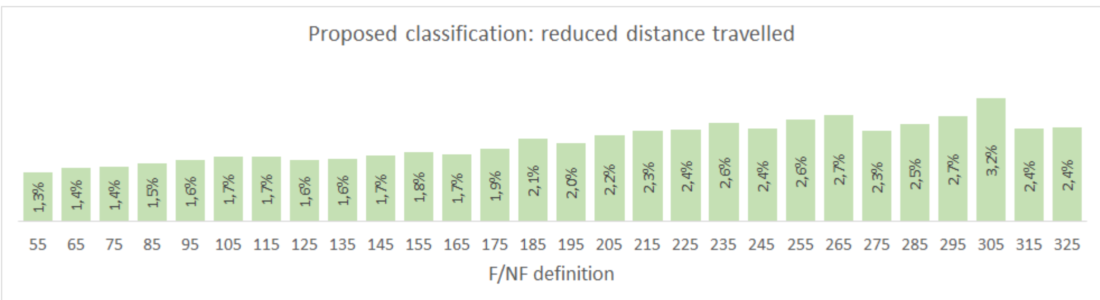

Figure 25: Reduced total travelling distance of proposed 2-class classification

In conclusion, using an F/NF definition of 305 results in the largest reduction of the estimated total travelling distance. This would result in a zone layout where 46 aisles are dedicated to fast-moving items and 10 aisles to not-fast-moving items, where the approximation is to travel **14,019 kilometres** in total. Compared to the current zone layout, this is a reduction of around 461 kilometres (3.2%).

Although this is the highest achieved reduction based on the proposed classification method, I also want to check what F/NF definition to use if classification is performed perfectly. Also for the perfect layout, the zone can be divided as in Figure 24. Calculating the total travelling distance approximation for each scenario based on perfect classification results in reduced travelling distances as pictured in Figure 26.

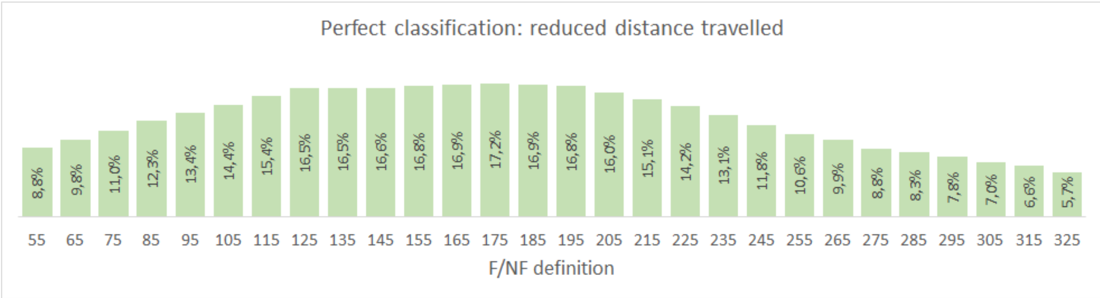

Figure 26: Reduced total travelling distance of perfect 2-class classification

In conclusion, using an F/NF definition of 175 results in the largest reduction of the estimated total travelling distance when classification is performed perfectly. Using 175 for the proposed classification method would result in a zone layout where 39 aisles are dedicated to fast-moving items and 17 aisles to not-fast-moving items, where a total travelling distance of **14,210 kilometres** is expected. Compared to the current zone layout, this is a reduction of 270 kilometres (1.9%). Although this reduction in distance is less compared to using an F/NF definition of 305, the improvement potential is higher.

5.2.4 Proposed layout: 3 classes

The 3-class zone layout stores fast-moving (F), slow-moving (S) and non-moving (N) SKUs in separate areas within the zone (Figure 27). The classification method provides the required sizes for F, S and N for a variety of scenarios.

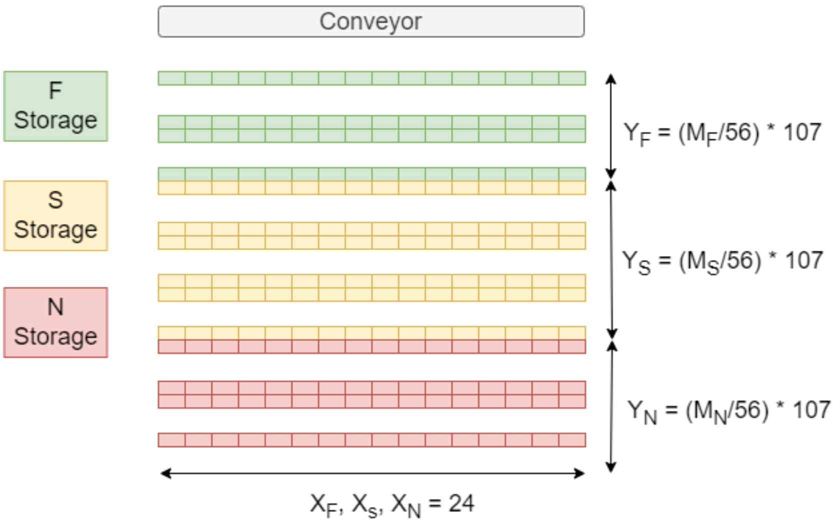

Figure 27: Proposed 3-class zone layout

For all scenarios, X_F, X_S and X_N are 24. However, the values for Y_F, Y_S, Y_N, N_F, N_S, N_N, M_F, M_S and M_N depend on the outcomes of a specific scenario. For all scenarios, $Y_F + Y_S + Y_N = 107$, $N_F + N_S + N_N = 100$ and $M_F + M_S + M_N = 56$

Now the total travelling distance by order pickers within zones on a day (EDT_{Day}) can be calculated by using the following formula:

$EDT_{Day} = (E[D_Y] + E[D_X]) * \frac{Daily\ demand}{100}$ **(17)**

where:

$E[D_Y] = 2Y_F + 2Y_S + 2Y_N\ [\frac{N_N}{N_N+1}]$ **(18)**

$E[D_X] = M_F * 24\ [1 - (\frac{M_F-1}{M_F})^{N_F}] + M_S * 24\ [1 - (\frac{M_S-1}{M_S})^{N_S}] + M_N * 24\ [1 - (\frac{M_N-1}{M_N})^{N_N}]$ **(19)**

Calculating this estimation for every day for every scenario, the estimated reduced travelling distances are as visualized in Figure 28.

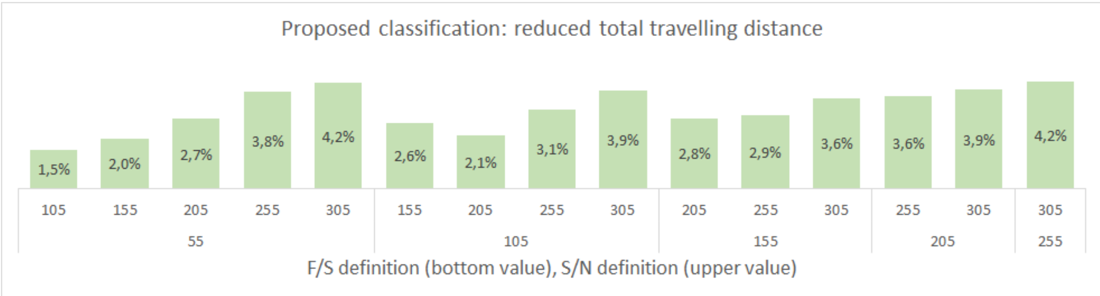

Figure 28: Reduced total travelling distance of proposed 3-class classification

In conclusion, 2 settings achieve a reduction of around 602 kilometres (4.2 %) in total travelling distance, which corresponds to a total travelling distance of around 13,868 kilometres:

- The F/S definition should be 55 while the S/N definition is 305, which results in 11 dedicated aisles to F items, 38 aisles to S and 7 aisles to N.
- The F/S definition should be 255 while the S/N definition is 305, which results in 49 dedicated aisles to F items, 1 aisle to S and 6 aisles to N.

Also for the perfect layout, the zone can be divided as in Figure 27. Calculating the total travelling distance estimation for each scenario based on perfect classification, I arrive at the estimated reduced travelling distances as pictured in Figure 29.

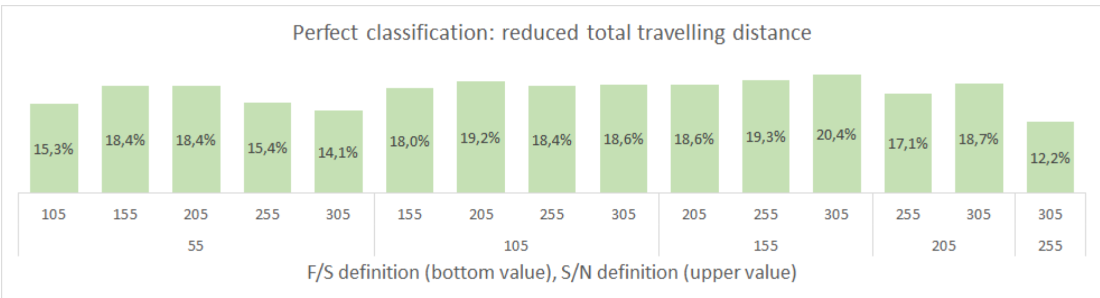

Figure 29: Reduced total travelling distance of perfect 3-class classification

Using an F/S definition of 155 and an S/N definition of 305 results in the largest reduction of the estimated total travelling distance when classification is performed perfectly (20.4 %). When applying these definitions to the proposed classification method, the company requires a zone layout where 42 aisles are dedicated to F items, 7 aisles to S and 7 aisles to N. The total travelling distance is approximately **13,964 kilometres**. Compared to the current zone layout, this is a reduction of 515 kilometres (3.6 %) in total.

5.3 Conclusion

During this chapter, it became clear that each of the features "*Brand*", "*Product type*", "*Gender*", "*Size*", "*Colour*", and "*Season*" contribute to the performance of the classification. Regarding the number of trees to build and the number of features to select for each tree, I use the values 50 and 2 respectively for the Random Forest model. The accuracy of the proposed classification method is higher for 2 classes (62%) than for 3 classes (42.4%). However, this does not correlate with the expected reduction in total travelling distance.

Classifying into 2 classes enables us to provide a zone layout that reduces the total travelling distance by 3.2%, while 17.2% is achievable with perfect classification. The resulting zone layout dedicates 39 aisles to F items and 17 aisles to NF items. When considering 3 classes in the proposed zone layout, the highest achieved reduction in travelling distance is 4.2% and would be 24.4% if the classification was perfect. This would result in a zone layout where 42 aisles are dedicated to F 7 aisles to S and 7 aisles to N.

Although 3 classes perform better, this also introduces more complexity to the solution. Most of the complexity is regarding the implementation. Using 2 classes enables one to put away closest to the conveyor or furthest away from the conveyor, while 3 classes require exactly knowing where to put away the middle class.

Concluding, the potential of introducing CBS is high for both using 2 and 3 classes. Improving the classification method is something to focus on from now on, as only a 4.2% reduction is achieved while 20.4% is achievable.

6 Implementation requirements

In this chapter, I elaborate on the requirements for the implementation of the proposed classification method. For each required step from receiving inbound data to acquiring an inbound list with class labels, this chapter will make clear what requirements are needed. This chapter describes not in detail who is responsible for every action but is more of a guide that supports knowing what is needed to classify incoming items and some remarks on what will be different operationally when using the method.

The goal of implementing this method is to support the company to put away inbound to the right areas within zones to reduce the total travelling distance of order picking. The process of classification can be structured as collecting data, preparing data, running the classification model and putting away classes. A variety of data sets needs to be collected and prepared, after which the classification model is ready to run. This results in a classified list of inbound, which enables us to put away classes in separate trolleys. In the end, aisles for classes are filled with corresponding classified SKUs. Performances of these zones should be monitored, to check whether the classification model has accurate predictions. For each phase, the requirements are described below.

6.1 Data collection

For the classification method, 2 data sets are required: data to train the classification model (= training data) and data to classify (= test data). To gain these data sets, the company needs to collect the following data:

- Historical Inbound data; every time the model is going to be trained, this file should be updated.
- Pull data; every time the model is going to be trained, this file should be updated.
- SKU feature information data; every time the model is going to be trained, the latest version of this document should be used.
- Inbound to arrive data.

If the company makes sure to store these data in a specific folder on the computer, the data can be automatically transformed as described in the next section.

6.2 Data preparation

Not all collected data is necessary for the classification. These data sets can now be transformed into the correct layout.

Training data

- For the training data, inbound data of up to 120 days before classification should be taken into account. Only keep relevant columns, such as the SKU number and date of inbound.
- The column *DateShipped* should be merged from Pull data, enabling to calculate the column *DaysInWarehouse*, which can later be used to label the data entry as a class.
- The column *InboundMonth* should be added to the data file, based on the *InboundDate*. *InboundMonth* can be used to attach the corresponding inbound season.
- Lastly, the SKU feature information should be merged into the data set. These columns are required when training a Random Forest model on the training data.

Test data

- For the test data, the company starts with the new inbound data file and should keep the same columns as kept for the training data.
- Thereafter, the SKU feature information data should be merged with the test data.

6.3 Running Model

During this research, an R file has been developed that can be used to classify a list of inbound. The company requires a computer with access to R and the Excel files from the previous steps. Also for running the model, the data sets need to be present in a specific location on the computer. It is recommended to train at least once every month a new model. In this way, as much historical data as possible is used when detecting causal connections between feature values and SKUs selling fast or not.

Input parameters

- Path to data files: this enables R to import files from the correct folder.
- In case of 2-class classification: F/NF-definition: to label training data as F or NF.
- In case of 3-class classification: F/S-definition and S/N-definition: to label training data as F, S or N.
- Number of trees: 50.
- Number of features to select for each tree: 2.
- MonthToClassify: to use training data of the corresponding season.

When these parameters are provided to the R file, the model is ready to run. As a result, the company obtains the new inbound where every SKU is labelled F or NF (2-class classification) or F, S or N (3-class classification).

6.4 Putting away classes

Instead of receiving all SKUs the same way, the company now receives SKUs that are classified. The system knows (based on the new inbound document) what SKUs to put into the aisles for a class. The receiver scans an item, checks the class label and puts the item in a trolley for the corresponding class label. This trolley should then be labelled according to the items that are inside.

The person that takes a trolley to put away should check what class is labelled to the trolley. Depending on the label, the person needs to check for space in the aisles for the corresponding class. On each floor of the warehouse, a heatmap is present in which for each zone is indicated what aisles have capacity left.

Lastly, it is recommended to keep track of the performance of the classification method. Although it is not possible to evaluate the performance right away, it is recommended to monitor the performance of the classification method. Especially the number of picks out of the zones. Monitoring the performance is important to improve the quality of the classification method. If the performance is not as desired, this should trigger looking for root causes.

7 Conclusions and recommendations

In this final chapter, all outcomes of the previous chapter are used to conduct conclusions and recommendations. I briefly conclude on the different sub-questions below, to eventually answer the research question. Thereafter, I provide recommendations regarding the solution and the improvement opportunities.

What is the current situation at the company regarding processes within inbound/returns and outbound, performances and stock?

- Inbound/returns: the variety of incoming SKUs is large, while the company does not influence what SKUs arrive and how many. Inbound is stored in the warehouse according to a Random Storage policy.
- Outbound: the variety of outgoing SKUs is large. Multiple orders are picked by a zone-picking strategy. The travelling distance of order pickers mainly depends on travelling through zones to complete picking tasks.
- Stock: The number of items in the warehouse is increasing, which indicates that it becomes even more important to manage storage capacity correctly. The SKU assortment is constantly changing; Only 32% of the incoming items are SKUs that are received before.

What classification methods are most suitable for the company to classify SKUs and how can their effectiveness be tested?

- Class labels based on FSN analysis: SKUs are either fast-(F), slow(S)- or non-moving(N), depending on their Average Length of Stay. Also classifying into fast-(F) and not-fast-moving(NF) is considered during this research.
- Training data can be labelled according to the FSN analysis, which enables us to train a Random Forest model.
- Classification performance can be evaluated by metrics from the confusion matrix, where class predictions are compared with actual classes. Considered metrics are the True Positive Rate, specificity and accuracy.

In what way can these classes be used when designing zones within the warehouse?

- Storing fast-moving items in an area closer to the conveyor and slower-moving items further away from the conveyor reduces the expected travelling distance.
- The classification method enables to label incoming SKUs. These SKUs can then be put away in an aisle that is dedicated to the corresponding class.
- The classification method provides us with the required space and expected picks per class for a variety of scenarios. The required space enables the designing of new zone layouts and the number of picks enables evaluating the layouts.

In what way does the proposed solution method influence the total travelling distance by order pickers?

- Current zone layout results in a total travelling distance approximation of 14,479 kilometres.
- Classifying into 2 classes achieves an accuracy of 61.7%. The final proposed 2-class layout has 39 aisles for F and 17 aisles for NF, resulting in a total travelling distance approximation of 14,019 kilometres.

- Classifying into 3 classes achieves an accuracy of 42.4%. The final proposed 3-class layout has 42 aisles for F, 7 aisles for S and 7 aisles for N, resulting in a total travelling distance approximation of 13,868 kilometres.
- The 3-class layout outperforms the 2-class layout, but is more complex to implement.

What are the requirements for implementation of the proposed solution method?

- Data collection of inbound data, pull data and SKU feature information.
- Preparation to format for R file.
- Train Random Forest model periodically.
- Use model on new inbound data to label incoming SKUs.
- Put labelled SKUs in corresponding trolleys and transport trolleys to aisles for corresponding classes.

What improvements regarding SKU-location assignment in a warehouse can be implemented to reduce the total travelling distance while the SKU assortment is constantly changing?

Travelling distance by order pickers mainly depends on routing through zones by order pickers. The distance of such routes can be approximated by a method based on Hall. Instead of assigning SKUs to random locations in the warehouse (Random Storage), I propose Class-Based Storage. As a result of this research, I propose 2 zone layouts: A layout considering 2 classes and a layout considering 3 classes (Figure 30). Because the SKU assortment is constantly changing, SKU-specific history is limited. To classify incoming SKUs anyway, I propose to use Random Forest classification in combination with an FSN analysis. By calculating the Average Length of Stay for SKUs with history, the Random Forest model labels new data based on similar SKUs regarding their features. With the proposed classification method, a reduction of 4.2% in total travelling distance is achieved, while the improvement potential is a reduction of 20.4%.

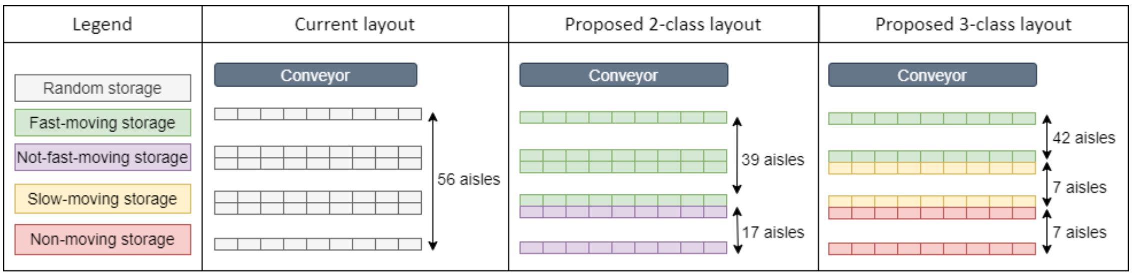

Figure 30: Current zone layout and proposed zone layouts

Recommendations

After conducting this research, there are several recommendations to be made. As introducing the Class-Based Storage policy is mainly based on the proposed classification method, I first provide recommendations to improve this classification method.

- Enable additional feature "price sold for". The price of an item is most likely a valuable feature regarding its impact on selling the item. The company has retail prices available. Therefore, the company only needs to log the discount percentages at the moment of selling items. If the actual selling price of each item is logged, the company can use the price feature when training Random Forest models for the classification method.
- Data gathering: Keep SKU number and inbound date attached in pull data. This reduces efforts when preparing data for the classification. Adjust SKU information document. Up-to-date feature information for SKUs is important information when building Random Forest models. Lastly, store training data to extend data to train with.

Besides recommendations to improve the classification method, I have additional recommendations:

- Classify into 2 classes, as it is less complex to implement. There is no need to know what exact rows are dedicated to the S class. F-labelled SKUs can be stored in the aisles closest to the conveyor and NF-labelled items in the aisles furthest away from the conveyor. The company has a heatmap available in which the capacity for each aisle is visualized. This enables us to see in what zones there is space left in the aisles.
- Find out how to label SKUs in the Warehouse Management System to enable using the classes.
- Decide what to do with items in the area for fast-moving SKUs that end up not being fast-moving: either reallocate or leave them in the area.
- Consider combining picking tasks to reduce travelling distance in the y direction. In this way, the picker can start picking on his/her way to the south side of the zone as well. The order picker's capacity (180 items) should be taken into account.

Implication for further research

Besides the placement of SKUs in the warehouse, the travelling distance of order picking also depends on **routing strategies** and **pick task planning**. The planning of picking tasks determines the order in which items are picked. Currently, the company plans the order of picks in such a way that the order picker can pick the items according to the traversal strategy (S-shape). Instead, research into other routing strategies might provide insights into reducing travelling distance even more.

References

[1] Tompkins James A et al. *Facilities planning*. John Wiley & amp; Sons, 2010.

[2] J. Temple Black. *The design of the factory with a future*. McGraw-Hill, 1992.

[3] René de Koster, Tho Le-Duc, and Kees Jan Roodbergen. "Design and control of Warehouse Order Picking: A Literature Review". In: *European Journal of Operational Research* 182.2 (2007), pp. 481–501. DOI: 10.1016/j.ejor.2006.07.009.

[4] Edward Frazelle. *World-class warehousing and material handling*. McGraw-Hill Education, 2016.

[5] Jinxiang Gu, Marc Goetschalckx, and Leon F. McGinnis. "Research on warehouse operation: A comprehensive review". In: *European Journal of Operational Research* 177.1 (2007), pp. 1–21. DOI: 10.1016/j.ejor.2006.02.025.

[6] David Ming-Huang Chiang, Chia-Ping Lin, and Mu-Chen Chen. "Data mining based storage assignment heuristics for travel distance reduction". In: *Expert Systems* 31.1 (2012), pp. 81–90. DOI: 10.1111/exsy.12006.

[7] Chenyang Xin, Xiaodong Liu, Yingjie Deng, and Qi Lang. "An optimization algorithm based on text clustering for warehouse storage location allocation". In: *2019 1st International Conference on Industrial Artificial Intelligence (IAI)* (2019). DOI: 10.1109/iciai.2019.8850832.

[8] Jason Chao-Hsien Pan, Po-Hsun Shih, Ming-Hung Wu, and Jyh-Horng Lin. "A storage assignment heuristic method based on genetic algorithm for a pick-and-pass warehousing system". In: *Computers amp; Industrial Engineering* 81 (2015), pp. 1–13. DOI: 10.1016/j.cie.2014.12.010.

[9] Kevin Gue and Onur Uludag. "A High-Density, Puzzle-Based Order Picking System". In: June 2012. DOI: 10.13140/2.1.4447.9360.

[10] Junwoo Kim, Francis Mendez, and Jesus Jimenez. "Storage location assignment heuristics based on slot selection and frequent itemset grouping for large distribution centers". In: *IEEE Access* 8 (2020), pp. 189025–189035. DOI: 10.1109/access.2020.3031585.

[11] Juan José Reyes, Elyn Lizeth Solano-Charris, and Jairo Rafael Montoya-Torres. "The storage location assignment problem: A literature review". In: *International Journal of Industrial Engineering Computations* (2019), pp. 199–224. DOI: 10.5267/j.ijiec.2018.8.001.

[12] E. A. Frazele and G. P. Sharp. "Correlated Assignment Strategy Can Improve Any Order-Picking Operation". In: *Ind. Eng.* 21.4 (Apr. 1989), pp. 33–37. ISSN: 0019-8234.

[13] Y. Zhang. "Correlated storage assignment strategy to reduce travel distance in order picking". In: *IFAC-PapersOnLine* 49.2 (2016), pp. 30–35. DOI: 10.1016/j.ifacol.2016.03.006.

[14] Geoff Buxey. "Reconstructing inventory management theory". In: *International Journal of Operations amp; Production Management* 26.9 (2006), pp. 996–1012. DOI: 10.1108/01443570610682607.

[15] Filippo Bindi, Riccardo Manzini, Arrigo Pareschi, and Alberto Regattieri. "Similarity coefficients and clustering techniques for the correlated assignment problem in warehousing systems". In: 2007.

[16] Lorcan Murphy. "Increasing e-commerce distribution center capacity through slotting strategy". PhD thesis. Jan. 2020.

[17] Yugang Yu, René B.M. de Koster, and Xiaolong Guo. "Class-Based Storage with a Finite Number of Items: Using More Classes is not Always Better". In: *Production and Operations Management* 24.8 (2015), pp. 1235–1247. DOI: https://doi.org/10.1111/poms.12334.

[18] Candrianto, Fitra Aulia, Mia Ayu Gusti, Mutiara Novenica, and Egy Juniardi. "Analysis of placement maximizing planning in warehouse using FSN analysis using class based storage method (Case study: Pt. XYZ)". In: *Proceedings of the 4th Padang International Conference on Education, Economics, Business and Accounting (PICEEBA-2 2019)* (2020). DOI: 10.2991/aebmr.k.200305.134.

[19] Riccardo Accorsi, Riccardo Manzini, and Marco Bortolini. "A hierarchical procedure for storage allocation and assignment within an order-picking system. A case study". In: *International Journal of Logistics Research and Applications* 15.6 (2012), pp. 351–364.

[20] Ankur Jain, Manghat Nitish Menon, and Saurabh Chandra. "Sales Forecasting for Retail Chains". In: 2015.

[21] J E Boylan, A A Syntetos, and G C Karakostas. "Classification for forecasting and stock control: A case study". In: *Journal of the Operational Research Society* 59.4 (2008), pp. 473–481. DOI: 10.1057/palgrave.jors.2602312.

[22] Min Xia and W.K. Wong. "A seasonal discrete grey forecasting model for fashion retailing". In: *Knowledge-Based Systems* 57 (2014), pp. 119–126. DOI: 10.1016/j.knosys.2013.12.014.

[23] Majd Kharfan, Vicky Wing Chan, and Tugba Firdolas Efendigil. "A data-driven forecasting approach for newly launched seasonal products by leveraging machine-learning approaches". In: *Annals of Operations Research* 303.1-2 (2020), pp. 159–174. DOI: 10.1007/s10479-020-03666-w.

[24] Sébastien Thomassey. "Sales forecasts in clothing industry: The key success factor of the Supply Chain Management". In: *International Journal of Production Economics* 128.2 (2010), pp. 470–483. DOI: 10.1016/j.ijpe.2010.07.018.

[25] I-Fei Chen and Chi-Jie Lu. "Demand forecasting for multichannel fashion retailers by integrating clustering and machine learning algorithms". In: *Processes* 9.9 (2021), p. 1578. DOI: 10.3390/pr9091578.

[26] Tim J. van Kampen and Dirk Pieter van Donk. "When is it time to revise your SKU classification: Setting and resetting the decoupling point in a dairy company". In: *Production Planning amp; Control* 25.16 (2013), pp. 1338–1350. DOI: 10.1080/09537287.2013.839063.

[27] A A Syntetos, J E Boylan, and J D Croston. "On the categorization of demand patterns". In: *Journal of the Operational Research Society* 56.5 (2005), pp. 495–503. DOI: 10.1057/palgrave.jors.2601841.

[28] Rob J. Hyndman and George Athanasopoulos. *Forecasting: Principles and practice.* OTexts, 2021.

[29] Evangelos Spiliotis, Spyros Makridakis, Artemios-Anargyros Semenoglou, and Vassilios Assimakopoulos. "Comparison of statistical and machine learning methods for daily SKU demand forecasting". In: *Operational Research* 22.3 (2020), pp. 3037–3061. DOI: 10.1007/s12351-020-00605-2.

[30] Qi Lang, Xuejun Pan, and Xiaodong Liu. "A text-granulation clustering approach with semantics for E-Commerce Intelligent Storage Allocation". In: *IEEE Access* 8 (2020), pp. 164282–164291. DOI: 10.1109/access.2020.3021421.

[31] Pratap Chandra Sen, Mahimarnab Hajra, and Mitadru Ghosh. "Supervised Classification Algorithms in Machine Learning: A Survey and Review". In: *Emerging Technology in Modelling and Graphics*. Ed. by Jyotsna Kumar Mandal and Debika Bhattacharya. Singapore: Springer Singapore, 2020, pp. 99–111.

[32] R.M. van Steenbergen and M.R.K. Mes. "Forecasting demand profiles of New Products". In: *Decision Support Systems* 139 (2020), p. 113401. DOI: 10.1016/j.dss.2020.113401.

[33] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An introduction to statistical learning: With applications in R.* Springer, 2022.

[34] Manish Varma Datla. "Bench marking of classification algorithms: Decision trees and random forests - a case study using R". In: *2015 International Conference on Trends in Automation, Communications and Computing Technology (I-TACT-15)* (2015). DOI: 10.1109/itact.2015.7492647.

[35] Deepak Kumar Sharma, Mayukh Chatterjee, Gurmehak Kaur, and Suchitra Vavilala. "Deep learning applications for disease diagnosis". In: *Deep Learning for Medical Applications with Unique Data* (2022), pp. 31–51. DOI: 10.1016/b978-0-12-824145-5.00005-8.

[36] RANDOLPH W. HALL. "Distance approximations for routing manual pickers in a Warehouse". In: *IIE Transactions* 25.4 (1993), pp. 76–87. DOI: 10.1080/07408179308964306.

[37] Richard Chin and Bruce Y. Lee. "Analysis of Data". In: *Principles and Practice of Clinical Trial Medicine* (2008), pp. 325–359. DOI: 10.1016/b978-0-12-373695-6.00015-6.

[38] C. Pichery. "Sensitivity analysis". In: *Encyclopedia of Toxicology* (2014), pp. 236–237. DOI: 10.1016/b978-0-12-386454-3.00431-0.

Appendix A

Sensitivity analysis confusion matrices for ntree and mtry values (2 classes)

			mtry = 2		mtry = 3		mtry = 4	
			Actual		Actual		Actual	
			F	NF	F	NF	F	NF
ntree = 50	Prediction	F	*11,1%*	*18,9%*	*11,6%*	*20,4%*	*11,7%*	*21,1%*
		NF	*18,3%*	*51,8%*	*17,8%*	*50,2%*	*17,7%*	*49,6%*
ntree = 100	Prediction	F	*11,0%*	*18,7%*	*11,5%*	*20,2%*	*11,7%*	*20,9%*
		NF	*18,4%*	*51,9%*	*17,9%*	*50,4%*	*17,7%*	*49,7%*
ntree = 150	Prediction	F	*11,0%*	*18,6%*	*11,5%*	*20,1%*	*11,7%*	*20,8%*
		NF	*18,4%*	*52,1%*	*17,9%*	*50,5%*	*17,7%*	*49,8%*

Figure 31: Confusion matrices sensitivity analysis: ntree and mtry values (2 classes)

Parameters		**F**		**NF**		
ntree	**mtry**	**TPR**	**Specificity**	**TPR**	**Specificity**	**Accuracy**
50	2	0,370	0,377	0,739	0,733	0,628
100	2	0,371	0,375	0,739	0,735	0,629
150	2	0,372	0,375	0,739	0,737	0,631
50	3	0,362	0,393	0,738	0,711	0,618
100	3	0,363	0,392	0,738	0,714	0,619
150	3	0,364	0,392	0,739	0,715	0,620
50	4	0,357	0,398	0,737	0,702	0,613
100	4	0,359	0,398	0,738	0,704	0,614
150	4	0,359	0,398	0,738	0,705	0,615

Table 11: Performance measures sensitivity analysis (2 classes)

Appendix B

Sensitivity analysis confusion matrices for ntree and mtry values (3 classes)

			mtry = 2			mtry = 3			mtry = 4		
			Actual			Actual			Actual		
			F	S	N	F	S	N	F	S	N
ntree = 50	Prediction	F	16,6%	11,1%	21,2%	16,0%	10,7%	20,4%	15,8%	10,6%	20,3%
		S	1,9%	1,5%	3,3%	3,0%	2,4%	5,2%	3,4%	2,6%	5,9%
		N	10,8%	8,3%	25,1%	10,3%	7,9%	24,1%	10,1%	7,7%	23,4%
ntree = 100	Prediction	F	16,7%	11,2%	21,3%	16,0%	10,7%	20,4%	15,9%	10,6%	20,3%
		S	1,9%	1,5%	3,2%	3,0%	2,4%	5,1%	3,4%	2,6%	5,8%
		N	10,8%	8,2%	25,2%	10,4%	7,9%	24,1%	10,1%	7,7%	23,6%
ntree = 150	Prediction	F	16,7%	11,2%	21,3%	16,1%	10,7%	20,4%	15,9%	10,6%	20,3%
		S	1,9%	1,5%	3,2%	3,0%	2,3%	5,1%	3,3%	2,6%	5,7%
		N	10,8%	8,2%	25,2%	10,4%	7,9%	24,2%	10,2%	7,7%	23,7%

Figure 32: Confusion matrices sensitivity analysis: ntree and mtry values (3 classes)

Parameters		F		S		N		
ntree	mtry	TPR	Specificity	TPR	Specificity	TPR	Specificity	Accuracy
50	2	0,339	0,566	0,228	0,074	0,568	0,506	0,433
100	2	0,340	0,569	0,229	0,073	0,570	0,507	0,434
150	2	0,340	0,570	0,230	0,072	0,570	0,507	0,434
50	3	0,340	0,546	0,223	0,113	0,570	0,484	0,425
100	3	0,340	0,546	0,225	0,113	0,570	0,486	0,425
150	3	0,340	0,547	0,225	0,112	0,570	0,486	0,426
50	4	0,338	0,539	0,220	0,125	0,568	0,471	0,419
100	4	0,339	0,540	0,222	0,125	0,569	0,474	0,420
150	4	0,339	0,540	0,222	0,124	0,569	0,476	0,421

Table 12: Sensitivity outcomes for ntree and mtry (3 classes)

Appendix C

Required space and expected picks (2 classes)

	Proposed classification				Perfect classification			
	Required space		Picks		Required space		Picks	
F/NF definition	**F**	**NF**	**F**	**NF**	**F**	**NF**	**F**	**NF**
55	13,3%	86,7%	22,7%	77,3%	4,92%	95,08%	27,29%	72,71%
65	18,4%	81,6%	29,5%	70,5%	6,42%	93,58%	32,30%	67,70%
75	24,9%	75,1%	37,6%	62,4%	7,97%	92,03%	36,82%	63,18%
85	31,5%	68,5%	44,9%	55,1%	9,75%	90,25%	41,84%	58,16%
95	38,4%	61,6%	52,7%	47,3%	11,83%	88,17%	46,46%	53,54%
105	42,2%	57,8%	56,9%	43,1%	13,98%	86,02%	50,57%	49,43%
115	48,8%	51,2%	62,9%	37,1%	16,03%	83,97%	54,41%	45,59%
125	50,8%	49,2%	64,7%	35,3%	18,21%	81,79%	58,25%	41,75%
135	55,0%	45,0%	68,4%	31,6%	22,01%	77,99%	62,30%	37,70%
145	59,8%	40,2%	72,7%	27,3%	25,75%	74,25%	66,09%	33,91%
155	63,2%	36,8%	76,0%	24,0%	28,94%	71,06%	69,47%	30,53%
165	66,6%	33,4%	78,6%	21,4%	32,09%	67,91%	72,67%	27,33%
175	69,1%	30,9%	81,0%	19,0%	35,19%	64,81%	75,82%	24,18%
185	70,9%	29,1%	83,3%	16,7%	38,66%	61,34%	78,72%	21,28%
195	74,2%	25,8%	85,3%	14,7%	41,99%	58,01%	81,49%	18,51%
205	76,1%	23,9%	87,0%	13,0%	46,22%	53,78%	84,15%	15,85%
215	77,4%	77,4%	88,0%	12,0%	50,63%	49,37%	86,68%	13,32%
225	79,5%	20,5%	89,5%	10,5%	54,95%	45,05%	88,99%	11,01%
235	81,6%	18,4%	91,0%	9,0%	59,33%	40,67%	91,00%	9,00%
245	82,1%	17,9%	90,8%	9,2%	64,09%	35,91%	92,86%	7,14%
255	83,2%	16,8%	92,0%	8,0%	68,51%	31,49%	94,36%	5,64%
265	84,6%	15,4%	93,0%	7,0%	72,15%	27,85%	95,61%	4,39%
275	82,3%	17,7%	91,0%	9,0%	76,87%	23,13%	96,84%	3,16%
285	84,1%	15,9%	92,4%	7,6%	80,17%	19,83%	97,76%	2,24%
295	85,1%	14,9%	93,3%	6,7%	83,54%	16,46%	98,50%	1,50%
305	82,4%	17,6%	92,5%	7,5%	86,69%	13,31%	98,95%	1,05%
315	83,2%	16,8%	91,7%	8,3%	88,86%	11,14%	99,17%	0,83%
325	84,6%	15,4%	92,5%	7,5%	91,48%	8,52%	99,27%	0,73%

Table 13: Proposed and perfect 2-class classification summary

Appendix D

Required space and expected picks (3 classes)

		Proposed classification					
Definition		Required space			Expected picks		
F/S	S/N	F	S	N	F	S	N
55	105	25,59%	3,34%	71,08%	38,79%	4,36%	56,85%
55	155	25,06%	25,04%	49,91%	37,96%	26,44%	35,60%
55	205	18,98%	55,34%	25,68%	31,17%	53,98%	14,84%
55	255	15,71%	70,10%	14,19%	26,51%	67,71%	5,77%
55	305	19,02%	69,45%	11,54%	27,35%	69,26%	3,39%
105	155	57,58%	1,21%	41,21%	71,11%	1,10%	27,79%
105	205	63,69%	8,34%	27,97%	75,75%	8,10%	16,15%
105	255	58,46%	25,74%	15,80%	71,12%	21,96%	6,92%
105	305	54,89%	32,73%	12,39%	67,34%	28,92%	3,74%
155	205	76,36%	0,48%	23,16%	87,35%	0,43%	12,22%
155	255	79,45%	5,04%	15,52%	89,76%	3,78%	6,46%
155	305	76,21%	11,85%	11,93%	86,85%	9,51%	3,64%
205	255	85,63%	0,60%	13,76%	94,29%	0,50%	5,20%
205	305	84,70%	3,98%	11,32%	93,74%	2,96%	3,29%
255	305	88,43%	0,64%	10,94%	96,71%	0,41%	2,88%

Table 14: Proposed 3-class classification summary

		Perfect classification					
Definition		Required space			Expected picks		
F/S	S/N	F	S	N	F	S	N
55	105	4,92%	9,06%	86,02%	27,29%	23,28%	49,43%
55	155	4,92%	24,03%	71,06%	27,29%	42,18%	30,53%
55	205	4,92%	41,30%	53,78%	27,29%	56,86%	15,85%
55	255	4,92%	63,59%	31,49%	27,29%	67,06%	5,64%
55	305	4,92%	81,77%	13,31%	27,29%	71,66%	1,05%
105	155	13,98%	14,96%	71,06%	50,57%	18,90%	30,53%
105	205	13,98%	32,24%	53,78%	50,57%	33,58%	15,85%
105	255	13,98%	54,53%	31,49%	50,57%	43,78%	5,64%
105	305	13,98%	72,70%	13,31%	50,57%	48,38%	1,05%
155	205	28,94%	17,27%	53,78%	69,47%	14,68%	15,85%
155	255	28,94%	39,57%	31,49%	69,47%	24,89%	5,64%
155	305	28,94%	57,74%	13,31%	69,47%	29,49%	1,05%
205	255	46,22%	22,29%	31,49%	84,15%	10,21%	5,64%
205	305	46,22%	40,47%	13,31%	84,15%	14,81%	1,05%
255	305	68,51%	18,17%	13,31%	94,36%	4,60%	1,05%

Table 15: Perfect 3-class classification summary