UNIVERSITY OF TWENTE.

Faculty of Electrical Engineering,
Mathematics & Computer Science

Revenue maximizing assignment of products within a physical store layout by Integer Linear Programming

Joren Kreuzberg
M.Sc. Thesis
Januari 2019

Graduation committee:
Prof. Dr. M.J. Uetz
N. de Brabander
Prof. Dr. M.N.M. van Lieshout

Discrete Mathematics and Mathematical Programming
Department of Applied Mathematics
Faculty of Electrical Engineering,
Mathematics and Computer Science
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

Abstract

This research aims to solve an adapted version of the general assignment problem. This assignment problem is the representation of finding a revenue maximizing assignment of products of a store to the physical locations within this store, where the assignment should satisfy additional feasibility requirements that induce a store layout which is intuitive from a customers' point of view. The main focus in this intuitive store layout, is assigning products of the same kind to physical locations within the store that are in close proximity to each other. It appears that the adapted assignment problem with this requirement can be modelled as the maximum weight connected subgraph problem, which turns out to be NP-hard.

This research consists of two parts. First the expected weekly revenue of every product on every physical location within the shop will be estimated by an intuitive statistical method using the available weekly sales data. These expected weekly revenues are used as input to determine the revenue maximizing assignment of the store.

The second part is the main focus of this research and consists of a NP-hardness proof of the considered adapted assignment problem and an Integer Linear Program to find the revenue maximizing assignment of the store, including a penalty in the objective function in case there is no feasible solution possible.

Keywords: Assignment problem, maximum weight connected subgraph, NP-hard, reduction, ILP.

Preface

This thesis is the graduation project for my master's program in Applied Mathematics, followed at the University of Twente. I conducted this project in collaboration between the University of Twente and the company IG&H, where I was challenged to solve a practical problem which turns out to be a familiar problem in Applied Mathematics.

In this preface I would like to thank several people who guided me through this graduation project and supported me during this period of hard work.
On the one hand, I would like to thank my supervisor from the University of Twente, Prof. Dr. Marc Uetz, who gave me the right guidance in terms of mathematical content and feedback. Besides the feedback he provided, he also motivated me in the more difficult periods of my thesis and therefore I am very glad that he has been my supervisor for this thesis.

On the other hand, I would like to thank Niels de Brabander and the rest of the analytics team of the company IG&H for the possibilities they gave me to improve my understanding and skills in data science topics. The workshops I was allowed to attend and the meetings with Niels gave me more insight in the impact data science and statistics can create within a company. Furthermore I would like to thank IG&H as a company for the integration of the interns among the employees and the memorable events I was allowed to be part of.

Regarding the support during the period of my thesis, I would like to thank my family, girlfriend, brother-in-law and close friends for the help and encouragement they gave me when I needed it.

As a result, I am proud to deliver this thesis, which is the last step in graduating for my master's degree in Applied Mathematics.

Joren Kreuzberg,
Januari 2019, Koekange.

Contents

1 Sets and variable definitions

Below the sets and variables are stated that are used frequently during this report:

P	The set of products that is sold in the store.
L	The set of physical locations within the store.
W	The weeks that products P have been sold in the store.
C	The set of product categories.
R	The set of product groups.
S	The set of product subgroups.
$P(C_x)$	The set of products belonging to category C_x.
$S(C_x)$	The set of subgroups belonging to category C_x.
$R(C_x)$	The set of groups belonging to category C_x.
$L(C_x)$	The set of locations belonging to category C_x.
$L_i(C_x)$	The set of locations a product $p_i \in P(C_x)$ was assigned to in the past.
$W(p_i, \ell_j)$	The weeks a product p_i was positioned on location ℓ_j.
$Q_t(p_i, \ell_j)$	The quantity of product p_i that was sold on location ℓ_j in week w_t.
$X(p_i, \ell_j)$	The expected weekly sales of product p_i on location ℓ_j.
$Y(p_i, \ell_j)$	The expected weekly revenue of product p_i on location ℓ_j.
$g(p_i)$	The gross profit of a product p_i.
$A(\ell_j)$	The capacity of a location $\ell_j \in L(C_x)$.

2 Introduction

This research is conducted in collaboration between the University of Twente and the company IG&H Consulting. IG&H Consulting provides advice to their clients in the retailing sector. For one of their retail clients, they analyzed which size of each category of products to sell in each store of this client. This resulted in determining which selection of products of each category should be sold within a store.
A question which remained unanswered in this project is where to place the selected products of each category in the layout of the store. This research aims to answer this question for a particular store.

2.1 Global description of the problem

For the store of this client, the products of this store have to be assigned to the locations within the store layout. Part of creating this layout is the consideration on which shelf each product should be positioned, i.e. to which physical location within the store a product should be assigned to. From the customers' point of view, the store layout should be intuitive and reasonable in order to find the products one is searching for. Therefore, in the positioning it is taken into account that products of the same kind, should also be positioned on locations within the store that are in close proximity to each other. But is it also considered on which location each product contributes to the highest revenue of the store?

There is much research dedicated to answering this question, mainly on the topic of shelf-space allocation [1, 2, 3, 4, 5, 6, 7]. In many of this literature, deterministic models are set up to maximize the revenue of a store. As far as we know, there is no research that also models the close proximity of locations based on the floor plan of the store.
Therefore this research aims to answer the question above by considering the problem of finding the revenue maximizing assignment of products of a particular store to the physical locations within this store. This assignment should meet the requirement that products of the same kind should be assigned to locations that are in close proximity to each other.

2.2 Global approach to solve the problem

In order to find this revenue maximizing assignment, this research is separated into two aspects. At first we need to obtain the expected weekly revenue of every product on each location within the store. Secondly, using these expected weekly revenues, the revenue maximizing assignment of the store need to be found.

First notice that the revenue of a product is actually its gross profit, the difference between the sales price and the cost price. For simplicity, we will call this revenue throughout this report.
For predicting the expected weekly revenues we use weekly sales data of this store from October 2017 up to February 2018. Hereby, we know how products have been sold on locations they were assigned to during this period. To consider which location results in the highest expected weekly revenue of a product, an estimation should be made for the expected weekly revenue of products on locations they never stood before.
Regarding the revenue maximizing assignment, the assignment should satisfy some feasibility requirements that illustrate a store layout which is convenient and intuitive for the customer. This mainly consist of the positioning of products of the same kind to locations that are in close proximity to each other.

2.3 Outline of this report

After this section, the report will continue with Section 3 which includes the problem description formulated as the assignment problem with additional feasibility requirements. After that, Section 4.1 will provide the method which is used to predict the expected weekly revenues and Section 4.2.2 will explain which model is used to determine the revenue maximizing assignment of the store.
Section 5 contains the computational comparison between the historical weekly revenue of the store and the expected weekly revenue of the revenue maximizing assignment of the store. The conclusions which can be drawn from these results are formulated in Section 6 and we will end with the topics which are applicable for discussion and further research in Section 7.

3 The problem description formulated as the assignment problem with additional feasibility requirements

This section outlines an elaborate description of the revenue maximizing assignment problem of the retail store we are considering. Section 3.1 will start with a clarification of the available sales data together with the notation we will use throughout this report. Subsequently, Section 3.2.1 will explain that the assignment of the store can be decomposed into an assignment for each category of products and locations. The additional feasibility requirements of the assignment problem will be explained in Section 3.2.2. At the end, Section 3.2.3 will finish with the formal statement of the revenue maximizing categorized assignment problem, for which this research aims to find the optimal solution.

3.1 Notation and available data

As described in section 2, we use the sales data of the store to find its revenue maximizing assignment. This data contains the weekly sales of each product of this store from October 2017 up to February 2018 together with the physical location the product was positioned on. Hereafter we will give some overload of notation, but this will be convenient throughout this report.

First of all, the weeks of sales data are defined as $W = \{w_1, \dots, w_{n_w}\}$, where n_w denotes the number of weeks of the sales data. The products that were sold during these weeks are denoted by the set $P = \{p_1, \dots, p_{n_p}\}$ and the physical locations within the store as $L = \{\ell_1, \dots, \ell_{n_\ell}\}$.
In addition, every product is part of some subgroup, group and category, whereby each subgroup belongs to some group, and every group belongs to a certain category. The set of product subgroups is defined as $S = \{S_1, \dots, S_{n_s}\}$, product groups as $R = \{R_1, \dots, R_{n_r}\}$ and the product categories as $C = \{C_1, \dots, C_{n_c}\}$. The connection between these product groups can be illustrated with the following example. Consider an electronic shaver as a product which belongs to the subgroup shavers. This subgroup belongs to the group electronic personal care. Finally, the group electronic personal care is part of the product category electronic equipment.
Each product $p_i \in P$ belongs to precisely one subgroup of S. The same holds for each subgroup of S to a group of R, and for each group of R to a category of C.
Therefore we can denote the subgroup, group and category a product p_i belongs to, as $S(p_i) \in S$, $R(p_i) \in R$ and $C(p_i) \in C$ respectively. Since each product p_i belongs to exactly one category C_x, we can define the set of products that belong to category C_x as $P(C_x)$, where $P(C_x) \subseteq P$. Besides the products, we can denote the subgroups and groups which belong to category C_x by $S(C_x)$ and $R(C_x)$ respectively.

Besides products, the set of physical locations L can also be categorized into their corresponding category C_x since each location is predetermined to a category. The locations that belong to category C_x are denoted by $L(C_x)$. Since we can separate the products and locations of the store into the categories $\{C_1, \dots, C_{n_c}\}$, we can decompose the problem to find the revenue maximizing assignment of the store into the problem of finding the revenue maximizing assignment for each category C_x, which we will call the categorized assignment problem. By combining these assignments, the assignment of the store will be obtained. Section 3.2.1 will explain this decomposition with some visualizations and introduces the categorized assignment problem.

Before clarifying this decomposition, we need some last notation to indicate information about a product $p_i \in P(C_x)$.
A product $p_i \in P(C_x)$ was only positioned on a part of the locations $L(C_x)$ during the weeks $\{w_1, \dots, w_{n_w}\}$. The locations a product $p_i \in P(C_x)$ was positioned on in the past will be mentioned as historical locations and denoted by $L_i(C_x)$, which is a subset of $L(C_x)$: that is, $L_i(C_x) \subseteq L(C_x)$. Therefore the non-historical locations of a product p_i consist of the set $L(C_x) \setminus L_i(C_x)$. The weeks a product p_i was positioned on its historical location $\ell_j \in L_i(C_x)$, is denoted by $W(p_i, \ell_j)$.
Furthermore, the revenue of a product is given by its gross profit, denoted by $g(p_i)$. These are necessary to obtain the expected weekly revenues of products on locations.
At last, we define the sales of a product $p_i \in P(C_x)$ on location $\ell_j \in L_i(C_x)$ in week $w_t \in W$ by $Q_t(p_i, \ell_j)$.

Now the notation is defined, we can give an overview of the available sales data. Table 1 illustrates the sales data of all products P over the weeks of W:

Product	Week	Weekly sales	Subgroup	Group	Category
p_1	w_1	$Q_1(p_1, \ell_j)$	$S(p_1)$	$R(p_1)$	$C(p_1)$
p_1	w_2	$Q_2(p_1, \ell_j)$	$S(p_1)$	$R(p_1)$	$C(p_1)$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
p_1	w_{n_w}	$Q_{n_w}(p_1, \ell_j)$	$S(p_1)$	$R(p_1)$	$C(p_1)$
p_2	w_1	$Q_1(p_2, \ell_j)$	$S(p_2)$	$R(p_2)$	$C(p_2)$
p_2	w_2	$Q_2(p_2, \ell_j)$	$S(p_2)$	$R(p_2)$	$C(p_2)$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
p_2	w_{n_w}	$Q_{n_w}(p_2, \ell_j)$	$S(p_2)$	$R(p_2)$	$C(p_2)$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
p_{n_p}	w_1	$Q_1(p_{n_p}, \ell_j)$	$S(p_{n_p})$	$R(p_{n_p})$	$C(p_{n_p})$
p_{n_p}	w_2	$Q_2(p_{n_p}, \ell_j)$	$S(p_{n_p})$	$R(p_{n_p})$	$C(p_{n_p})$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
p_{n_p}	w_{n_w}	$Q_{n_w}(p_{n_p}, \ell_j)$	$S(p_{n_p})$	$R(p_{n_p})$	$C(p_{n_p})$

Table 1: Available sales data of all products P.

As input for the revenue maximizing assignment problem, we need the expected weekly revenues $Y(p_i, \ell_j)$, obtained from the expected weekly sales $X(p_i, \ell_j)$ of a product $p_i \in P(C_x)$ on all locations $\ell_j \in L(C_x)$. Section 4.1 clarifies the method which is used to obtain the expected weekly revenues $Y(p_i, \ell_j)\ \forall p_i \in P(C_x), \ell_j \in L(C_x)$.

3.2 The categorized assignment problem with additional feasibility requirements

The previous section described the notation which we will use throughout this report and the data which is available to predict the expected weekly revenues.
Section 3.2.1 starts with the explanation how the assignment problem of the entire store is decomposed in the assignment problem per category. After that, Section 3.2.2 will describe the additional feasibility requirements that are applicable to the categorized assignment problem and Section 3.2.3 will provide the final statement of the categorized assignment problem.

3.2.1 Decomposition of the assignment problem of the store into the categorized assignment problem

A categorized assignment problem aims to find the revenue maximizing assignment of products $P(C_x)$ to locations $L(C_x)$ for a category C_x which satisfies some additional feasibility requirements. The representation of products $P(C_x)$ and locations $L(C_x)$ can be visualized as a bipartite graph $G_x = (P(C_x) \cup L(C_x), E)$ as in Figure 1:

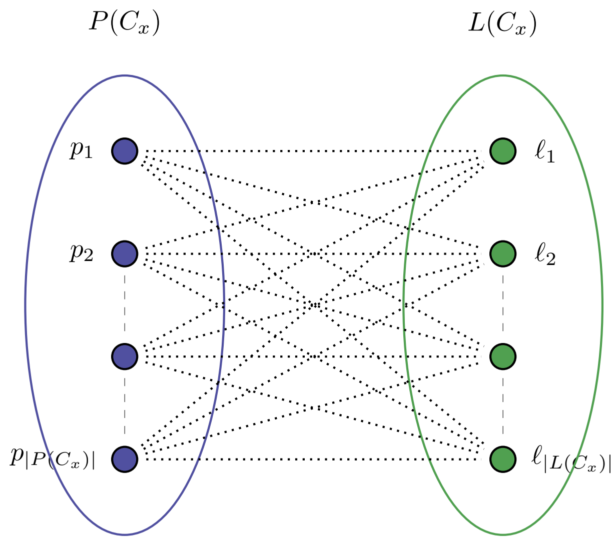

Figure 1: Categorized assignment problem: bipartite graph $G_x = (P(C_x) \cup L(C_x), E)$, $|E| = |P(C_x)| \cdot |L(C_x)|$.

The weight of every edge $(p_i, \ell_j) \in E$ represents the expected weekly revenue of product p_i on location ℓ_j. In the categorized assignment problem it has to be determined which edge to choose for each product, i.e. to which location $\ell_j \in L(C_x)$ each product $p_i \in P(C_x)$ will be assigned to. The products and locations of each category can be visualized in this way. Only the size and weights of the graph differ per category.

To visualize the decomposition of the assignment problem of the store into the categorized assignment problems, consider Figure 2.

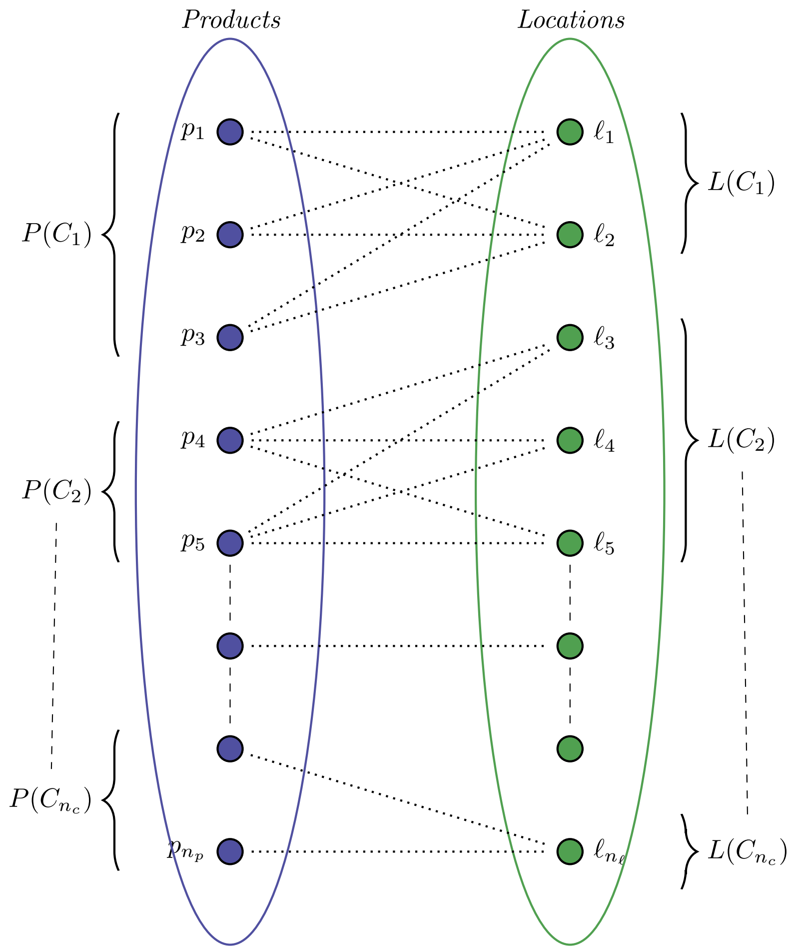

Figure 2: Assignment problem of the entire store: bipartite graph $G = (P \cup L, E)$, with $|P| = n_p, |L| = n_\ell, |E| = \sum_{x=1}^{n_c} |P(C_x)| \cdot |L(C_x)|$.

In Figure 2 an example is given of a decomposition of the assignment problem of the store into n_c categorized assignment problems. The upper three products and upper two locations belong to the first category. In determining the revenue maximizing solution of the store, the solution for each categorized assignment problem can be computed individually.

The next section will describe the additional requirements an assignment has to satisfy for the categorized assignment problem of category C_x. We will explain this for some category $C_x \in C$, where C_x could be any of the $\{C_1, \dots, C_{n_c}\}$ categories.

3.2.2 Feasibility constraints

In determining the revenue maximizing assignment, we have to take into account the requirements for an assignment to be feasible. This section will describe each of those requirements and explains why they are necessary. Section 4 will derive the mathematical formulation of these requirements.

First of all, each location $\ell_j \in L$ has a certain capacity $A(\ell_j)$ which indicates the maximum number of different products that can be assigned to this location. This capacity $A(\ell_j)$ is equal to the highest number of different products of category C_x that have been positioned on location ℓ_j in a week.
Within this research, we only focus on which product to assign to which location. The quantity of such a product is outside of the scope of this research. Section 7 will discuss whether it is relevant to include the quantity of an assigned product into the assignment problem.

With the above mentioned in consideration, the first feasibility requirement can be formulated as follows:

Definition 1. *Within a feasible solution to the categorized assignment problem, the number of different products to a location* $\ell_j \in L(C_x)$ *may not exceed its capacity* $A(\ell_j)$*.*

The second requirement concerns the hierarchy of products, subgroups and groups within a category C_x. For the convenience of the customer, products of of a certain subgroup or group should be assigned to locations of $L(C_x)$ that are in close proximity to each other. In this way a store layout becomes reasonable and intuitive for the customers of the store.
However, close proximity can be interpreted differently regarding a subgroup or a group. To illustrate this difference, consider the following example:

Example 1. *Consider the example of a store layout in Figure* 3 *together with its schematic representation in Figure* 4*.*

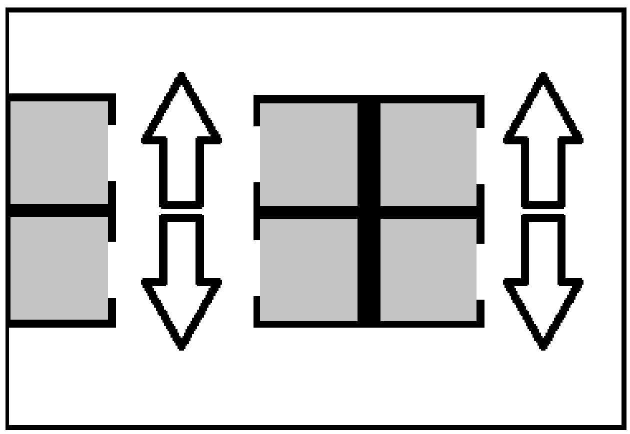

Figure 3: A physical store layout with six locations illustrated as grey boxes and walkways illustrated as arrows.

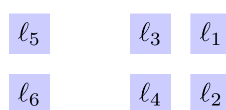

Figure 4: Schematic representation of Figure 3.

Within Figure 3*, observe that the facing of each location is directed towards the walkways. Location* ℓ_1 *is positioned next to location* ℓ_2 *and they are facing the same walkway. Location* ℓ_3 *is standing with its back to location* ℓ_1*, as well as for location* ℓ_4 *and* ℓ_2*.*
Now consider the situation in which the subgroup shavers, from the example of Section 3.1*, would be assigned to location* ℓ_1 *and* ℓ_3*. A customer could experience this assignment as unreasonable because the customer has to walk around the locations to the other walkway to compare products of the subgroup shavers. So products of a certain subgroup should be positioned on the same location or locations that are next to each other and facing in the same direction. On the other hand, it might be more reasonable to assign different subgroups of a certain group to location* ℓ_1 *and* ℓ_3 *because products of different subgroups may be positioned more widely from each other and the locations a group is assigned to, cover a larger area of the store layout than the locations a subgroup is assigned to.*

As Example 1 illustrates, there is a difference in close proximity regarding subgroups and groups.
For this reason we introduce the connectivity graphs $B_x = (L(C_x), \widetilde{E})$ and $U_x = (L(C_x), E)$ for each category, that represent the locations $L(C_x)$ within the store layout together with the edges that represent which

locations are in close proximity to each other regarding subgroups and groups respectively. The edges $\widetilde{E}$ illustrate which locations are in close proximity to each other regarding subgroups, i.e. which locations are adjacent regarding subgroups. The same applies for the edges E within U_x regarding groups, i.e. which locations are adjacent regarding groups.

As suggested in Example 1, an edge $e = (\ell_j, \ell_k) \in \widetilde{E}$ only exists if the locations ℓ_j and ℓ_k are positioned next to each other within the store layout and if they are facing in the same direction.
Regarding groups, an edge $e \in E$ exists if one of the following scenario's is applicable:

1. two locations are positioned next to each other and they are facing in the same direction.
2. two locations are facing in the opposite direction, with a walkway in between.
3. two locations are facing in a perpendicular direction, with a walkway in between.
4. two locations are positioned back to back.

Thus, an edge $e \in \widetilde{E}$ only exists if the first scenario is applicable and an edge $e \in E$ if one of the four scenario's is applicable. For this reason $\widetilde{E}$ is always a subset of E: that is, $\widetilde{E} \subseteq E$.
Observe that the graph B_x only consists of paths since locations in B_x are adjacent if the first scenario is applicable.

To illustrate how a graph B_x and U_x is created, observe Figure 5 and Figure 6. They illustrate the graphs B_x and U_x respectively, generated by the store layout of Figure 3.

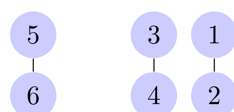

Figure 5: The graph $B_x = (L(C_x), \widetilde{E})$ which is induced by the layout of Figure 3.

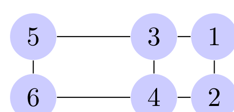

Figure 6: The graph $U_x = (L(C_x), E)$ which is induced by the layout of Figure 3.

Observe that within U_x, the locations ℓ_1 and ℓ_2, ℓ_3 and ℓ_4, and ℓ_5 and ℓ_6 are adjacent by scenario 1. Location ℓ_3 and ℓ_5, and ℓ_4 and ℓ_6 are adjacent by scenario 2. At last, location ℓ_1 and ℓ_3, and ℓ_2 and ℓ_4 are adjacent by scenario 4. To illustrate when scenario 3 would be applicable, consider you are facing a location ℓ_j and you have to rotate 45 degrees and face location ℓ_k with a walkway in between.

In order to check whether the assigned locations of a subgroup are positioned in close proximity to each other, the assigned locations of each subgroup must induce a connected subgraph in B_x, for each subgroup. Similarly, the assigned locations of each group must induce a connected subgraph in U_x, for each group.

Definition 2. *Within a feasible solution to the categorized assignment problem, the allocated locations of every subgroup should induce a connected subgraph in graph B_x, for each subgraph, and the allocated locations of every group should induce a connected subgraph in graph U_x, for each group.*

To give insight how the graphs B_x and U_x play a role in the feasibility of the solution, consider Example 2.

Example 2. *Consider the case of assigning the four products of Table 2 to the locations $\{\ell_1, \ell_2, \ell_3\}$ of Figure 4.*

Product	Subgroup	Group
p_1	S_1	R_1
p_2	S_1	R_1
p_3	S_2	R_1
p_4	S_2	R_1

Table 2: Example of the properties of the products $P(C_x)$.

The edges between the locations $\{\ell_1, \ell_2, \ell_3\}$ of graph B_x in Figure 5 and of graph U_x in Figure 6 can be inserted into the bipartite graph of products and locations as in Figure 7 and Figure 8 respectively. Furthermore products of the same subgroup are circled together in Figure 7 and products of the same group are circled together in Figure 8.

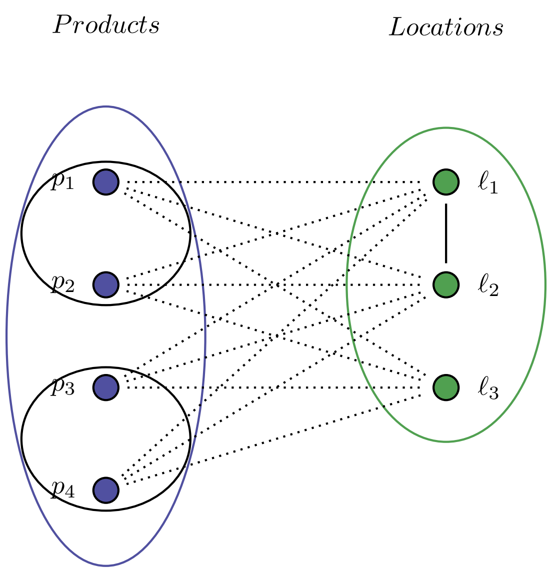

Figure 7: The bipartite graph $G_x = (P(C_x) \cup L(C_x), E)$ with graph B_x inserted.

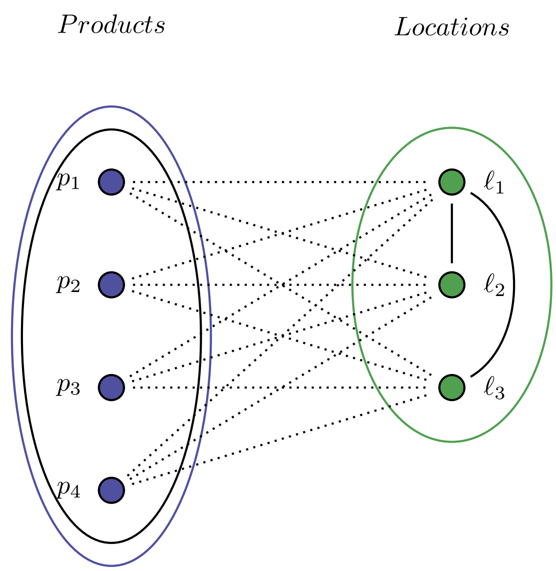

Figure 8: The bipartite graph $G_x = (P(C_x) \cup L(C_x), E)$ with graph U_x inserted.

As stated in Definition 2, the locations allocated to each subgroup should form a connected subgraph in graph B_x. The same applies for the locations allocated to each group regarding graph U_x. To illustrate that a feasible solution should satisfy these requirements, consider a feasible assignment that is depicted in Figure 9 regarding subgroups and Figure 10 regarding groups.

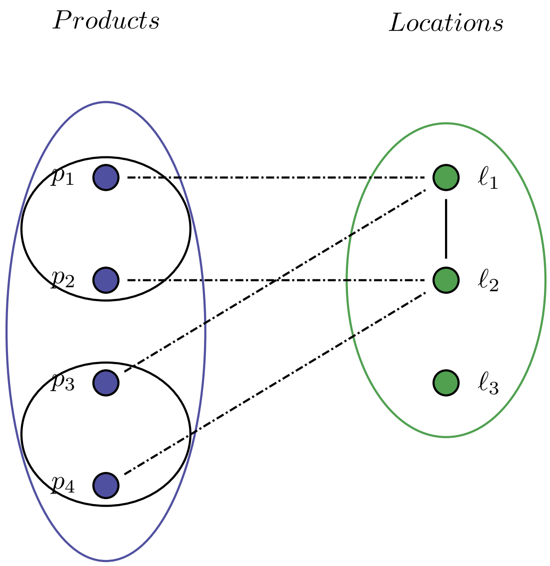

Figure 9: Feasible solution regarding subgroups since the locations allocated to subgroup S_1 form a connected subgraph in B_x by the edge between ℓ_1 and ℓ_2. The same applies for the locations allocated to subgroup S_2.

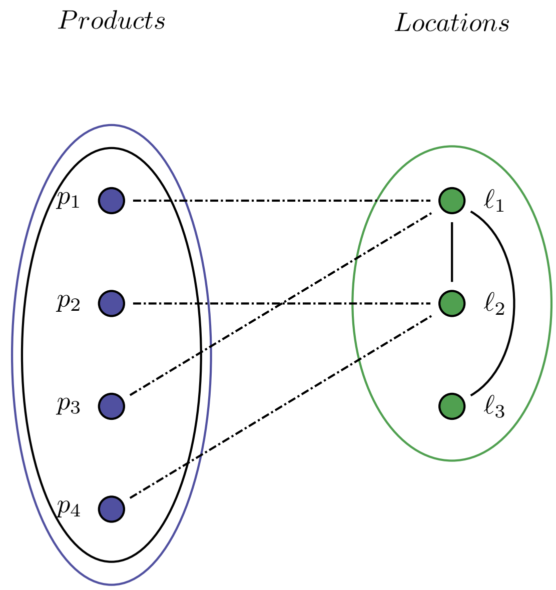

Figure 10: Feasible solution regarding groups since the locations allocated to the one existing group R_1 form a connected subgraph in U_x by the edge between ℓ_1 and ℓ_2.

We illustrate the assignment in two figures because the assignment illustrated in both figures should be feasible regarding subgroups and groups. As explained in the figures, this assignment is feasible because the locations allocated to each subgroup form a connected subgraph in graph B_x and the locations allocated to each group form a connected subgraph in graph U_x.

To illustrate an infeasible assignment, consider the assignment which is illustrated in Figure 11 *regarding subgroups and in Figure* 12 *regarding groups.*

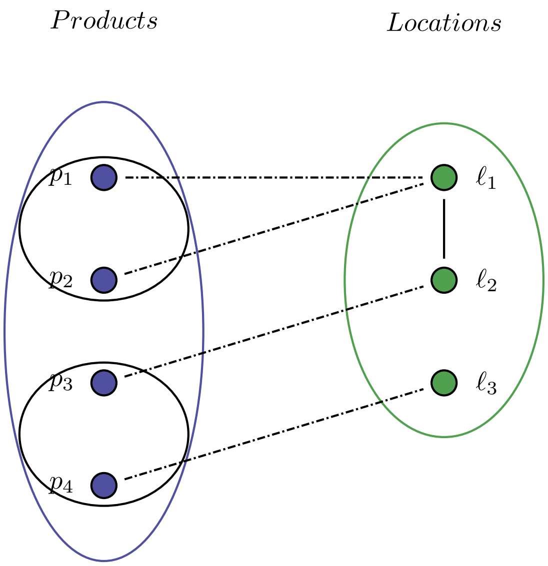

Figure 11: Infeasible solution regarding subgroups since the locations allocated to subgroup S_2 do not form a connected subgraph in graph B_x.

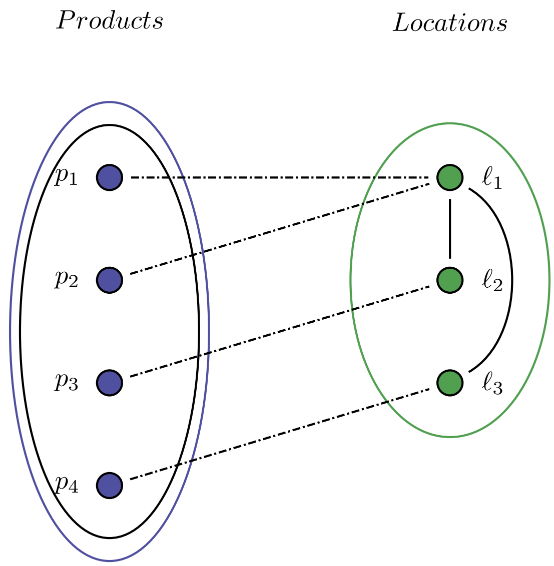

Figure 12: Feasible solution regarding groups since the locations allocated to group R_1 form a connected subgraph in U_x.

As explained in Figure 11*, the assignment illustrated in Figure* 11 *and Figure* 12 *is infeasible because the locations allocated to subgroup S_2 do not form a connected subgroup in B_x since location ℓ_2 and ℓ_3 are not connected in B_x.*

Now all the feasibility requirements are formulated, the next section will describe the definition of the categorized assignment problem with the additional feasibility requirements.

3.2.3 Definition of the categorized assignment problem with feasibility requirements

Recall that the goal is to find a feasible revenue maximizing assignment of products $P(C_x)$ to locations $L(C_x)$ for every category $C_x \in C$. These revenue maximizing assignments for the categories $\{C_1, \ldots, C_{n_c}\}$ together will form the revenue maximizing assignment of the store. An illustrated example of this is given in Figure 2. In the previous section we described the requirements for a feasible solution to the categorized assignment problem. Therefore we now can formally define the categorized assignment problem, which we will refer to as the CA-problem:

Definition 3. ***The categorized assignment problem***
Given a set of products $P(C_x)$ with corresponding subgroups $S(C_x)$ and groups $R(C_x)$, a set of locations $L(C_x)$, the expected weekly revenues $Y(p_i, \ell_j)$ $\forall p_i \in P(C_x), \ell_j \in L(C_x)$, the location capacities $A(\ell_j)$ and the graphs B_x and U_x, find the revenue maximizing assignment of products to locations such that the assignment satisfies the feasibility requirements of Definition 1 *and* 2*.*

4 Estimating the expected weekly revenues and an Integer Linear Program formulation to solve the CA-problem.

This section explains how to compute the revenue maximizing assignment for the categorized assignment problem formulated in Definition 3. Section 4.1 describes the way the expected weekly revenues are predicted, which are needed as input for the categorized assignment problem. Section 4.2 outlines the algorithm which is used to find the revenue maximizing assignment to the CA-problem.

4.1 Estimate the expected weekly revenue for the categorized assignment problem

This section explains how the expected weekly sales $X(p_i, \ell_j)$ $\forall p_i \in P(C_x), \ell_j \in L(C_x)$ are predicted. These are necessary to predict the expected weekly revenues $Y(p_i, \ell_j)$, which are simply predicted by multiplying the expected weekly sales $X(p_i, \ell_j)$ by its gross profit $g(p_i)$.
To predict the expected weekly sales, we use the historical sales of products on their historical locations as depicted in Table 1. However, the historical locations of almost every product is a subset of the locations it can be assigned to, $L(C_x)$. Estimating the expected weekly sales of a product on a historical location sounds reasonable, but estimating the expected weekly sales of a product on a location it never stood before, requires another approach.

Before we will explain how we predict the expected weekly sales $X(p_i, \ell_j)$ $\forall p_i \in P(C_x), \ell_j \in L(C_x)$, we want to address that we are aware of the fact that the expected weekly sales will be estimated by simple methods and that these methods are imperfect. This is caused by the quality and characteristics of the available historical weekly sales data, depicted in Table 1. That is, it is difficult to fit a distribution to the historical weekly sales of a product because many products have not been sold for weeks and when they were sold, the quantity varied a lot. Furthermore, given the available sales data, the only features we were able to use, are the location and the product itself. The quality and characteristics of the data will be further discussed in Section 7.

Now we shall explain the different approaches to predict $X(p_i, \ell_j)$, depending on whether ℓ_j is a historical or non-historical location of product p_i.
Section 4.1.1 will provide a method to predict $X(p_i, \ell_j)$ on historical locations of each product p_i and Section 4.1.2 will explain the method whereby $X(p_i, \ell_j)$ is predicted on non-historical locations of product p_i.

4.1.1 Estimate the expected weekly sales of products on historical locations

We are interested in estimating the expected weekly sales $X(p_i, \ell_j)$ for all products $p_i \in P(C_x)$ on its historical locations $\ell_j \in L_i(C_x)$ using the historical weekly sales data. This section will propose two simple methods that will predict these expected weekly sales of products on its historical locations. The first method is taking the average of all historical weekly sales of a product p_i on its historical location ℓ_j. The second method is to use a Poisson regression with the product p_i and historical location ℓ_j as predictors.
After the explanation of both methods, we will validate and argue which method is the most appropriate to use and conclude with the formula we will use to predict the expected weekly sales of products on its historical locations.

In the following two paragraphs the proposed methods will be explained.

Taking the average

Taking the average of the historical weekly sales of a product in order to predict the expected weekly sales is intuitive and applicable. Consider the following example to illustrate this:

Example 3. *Assume that the following sales data is available of product p_2:*

Product	Week	Weekly sales	Location	Subgroup	Group	Category
p_2	w_1	3	ℓ_3	S_1	R_3	C_1
p_2	w_2	0	ℓ_3	S_1	R_3	C_1
p_2	w_3	1	ℓ_3	S_1	R_3	C_1
p_2	w_4	0	ℓ_5	S_1	R_3	C_1
p_2	w_5	3	ℓ_5	S_1	R_3	C_1
p_2	w_6	4	ℓ_5	S_1	R_3	C_1

Table 3: Historical sales of a certain product.

The prediction of the expected weekly sales $X(p_2, \ell_3)$ and $X(p_2, \ell_5)$ is executed as follows:

$$\hat{X}(p_2, \ell_3) = \frac{1}{3}\sum_{t=1}^{3} Q_t(p_2, \ell_3) = 1\frac{1}{3}$$
$$\hat{X}(p_2, \ell_5) = \frac{1}{3}\sum_{t=4}^{6} Q_t(p_2, \ell_5) = 2\frac{1}{3}.$$

So given these historical sales, it is expected that the weekly sales of product p_2 on location ℓ_3 and ℓ_5 will amount to $1\frac{1}{3}$ and $2\frac{1}{3}$ respectively.

Recall that the weeks a product p_i was positioned on its historical location ℓ_j, is denoted by $W(p_i, \ell_j)$.
In general, using the average of historical sales to estimate the expected weekly sales of a product $p_i \in P(C_x)$ on a historical location $\ell_j \in L_i(C_x)$, is done in the following way:

$$\hat{X}(p_i, \ell_j) = \frac{\sum_{w_t \in W(p_i,\ell_j)} Q_t(p_i, \ell_j)}{|W(p_i, \ell_j)|}. \tag{1}$$

Poisson regression

The second method which can be used to predict the expected weekly sales of a product on its historical location is a Poisson regression. A regression is utilized to reflect the relationship between a response variable and a set of predictors. In our case the response variable is the weekly sales of a product on its historical location and the predictors are the locations and the products. To use a Poisson regression to predict the expected weekly sales $X(p_i, \ell_j)$, we have to assume that the weekly sales of a product on its historical location follow a poisson distribution [8].
Since the weekly sales of a product on its historical location can be described as a count of events in the time interval of a week, the weekly sales might follow a Poisson distribution [9]. Therefore it is a reasonable attempt to predict the expected weekly sales by a poisson regression.
However, it appears that the available historical weekly sales of a product does not fit the Poisson distribution very well due to the poor quality of the data. Nevertheless, we want to compare the approach of taking the average with another method and therefore we do perform a poisson regression and analyze the performances of both methods.

Where an ordinary least squares (OLS) regression aims to model the expected value of the response variable on itself, a Poisson regression aims to model the natural log of the expected value of the response variable. To understand what a Poisson regression precisely does to predict the expected weekly sales, we will give two examples. In Example 4 only the location is used as a predictor and in Example 5 both the location and the product are a predictor.

Example 4. *Consider the weekly sales of Table* 3 *again. These are illustrated in Figure* 13.

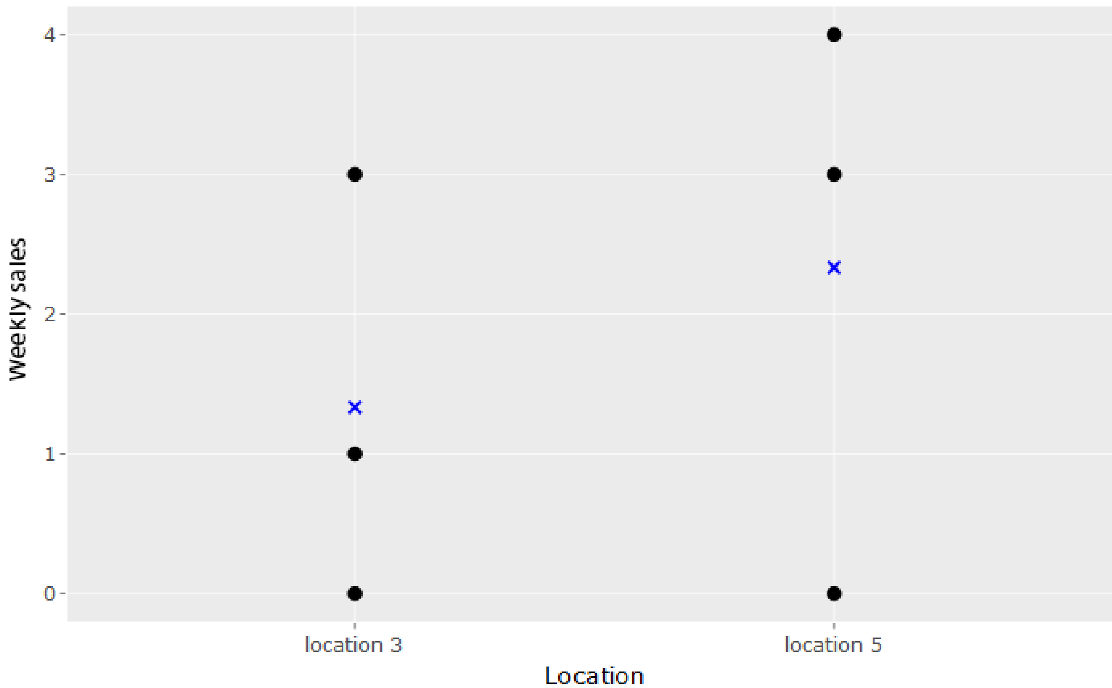

Figure 13: Visualization of weekly sales of a product on two locations illustrated as dots, together with the expected weekly sales per location illustrated as crosses.

We predict the expected weekly sales $X(p_2, \ell_j)$ with the location $\ell_j \in L_2(C_x)$ as predictor. The product p_i is not a predictor yet, because in this example we only possess over the sales data of product p_2.
Since the Poisson regression models the natural log of the expected weekly sales, the model will be of the form:

$$ln(\hat{X}(p_2, \ell_j)) = \beta_0 + \beta_j + \mu_{2,j}$$
$$\iff \hat{X}(p_2, \ell_j) = \mathrm{e}^{\beta_0 + \beta_j + \mu_{2,j}}.$$

The Poisson regression reveals the relationship between the expected weekly sales and the locations of this product p_2. To notice the effect a location has on the expected weekly sales, some location has to be selected as a benchmark such that the other locations can be compared to this benchmark location. In our example location ℓ_3 is the benchmark location, so the expected weekly sales of product p_2 on location ℓ_3 is given by $\hat{X}(p_2, \ell_3) = \mathrm{e}^{\beta_0}$. The effect in weekly sales of changing location from ℓ_3 to ℓ_5 is given by $\mathrm{e}^{\beta_5 + \mu_{2,5}}$ and hence $\hat{X}(p_2, \ell_5) = \mathrm{e}^{\beta_0 + \beta_5 + \mu_{2,5}}$. These expected weekly sales are denoted as crosses in Figure 13. We will explain how the coefficients such as β_0, β_5 and $\mu_{2,5}$ are estimated if we add the product as a predictor.

So with one product, the Poisson regression with the location as predictor is given by:

$$\hat{X}(p_i, \ell_j) = \mathrm{e}^{\beta_0 + \beta_j + \mu_{i,j}}.$$

For every location ℓ_j that is not equal to the benchmark location, there is a coefficient β_j and $\mu_{i,j}$ that will correct for a change in location from the benchmark location to ℓ_j.

However, we possess over the historical weekly sales data of all products $P(C_x)$. Therefore not only the location is a predictor, but the product as well. Besides a benchmark location, we also need a benchmark product, which results in a benchmark (product, location) combination. From this benchmark a correction will be made for a change in product and location.

Hereby the Poisson regression with the product and location as categorical predictor will be of the form:

$$\begin{aligned} ln(\hat{X}(p_i, \ell_j)) =& \gamma_0 + \alpha_i + \beta_j + \mu_{i,j} \\ \iff \hat{X}(p_i, \ell_j) =& \mathrm{e}^{\gamma_0 + \alpha_i + \beta_j + \mu_{i,j}}. \end{aligned} \tag{2}$$

For the benchmark product p_i and benchmark location ℓ_j, the expected weekly sales are given by $\hat{X}(p_i, \ell_j) = \mathrm{e}^{\gamma_0}$. With Equation 2 a change in product and change in location are considered dependent from each other. This also is the case since the sale of a product p_i occurred when it was positioned on its historical location ℓ_j. Therefore we make a correction for a change in product p_i with coefficient α_i, for a change in location ℓ_j with coefficient β_j and the interaction of those changed product and location is given by coefficient $\mu_{i,j}$. For each product $p_i \in P(C_x)$ and location $\ell_j \in L_i(C_x)$, the coefficients α_i, β_j and $\mu_{i,j}$ are determined by the method of maximum log-likelihood [8].

Within the program R, we predicted the expected weekly revenues $X(p_i, \ell_j)$ via Equation 1 and Equation 2. It turns out that both approaches resulted in the exact same predictions. It appears that by including the interaction coefficient $\mu_{i,j}$ in Equation 2, the Poisson regression predicts the same expected weekly sales as the expected weekly sales predicted by taking the average. A proof of this is included in Appendix *A*.

Since we want to compare the approach of taking the average with another approach, we will perform a Poisson regression with the product and location as predictor under the assumption that the product and location are independent from each other. So the interaction coefficient is not included. Then the Poisson regression to predict the expected weekly sales is given by:

$$\begin{aligned} ln(\hat{X}(p_i, \ell_j)) =& \gamma_0 + \alpha_i + \beta_j \\ \iff \hat{X}(p_i, \ell_j) =& \mathrm{e}^{\gamma_0 + \alpha_i + \beta_j}. \end{aligned} \tag{3}$$

To demonstrate how the Poisson regression of Equation 3 works, consider Example 5:

Example 5. *Consider the weekly sales of product* p_2 *and* p_4*, illustrated within Figure* 14 *as black and grey dots respectively.*

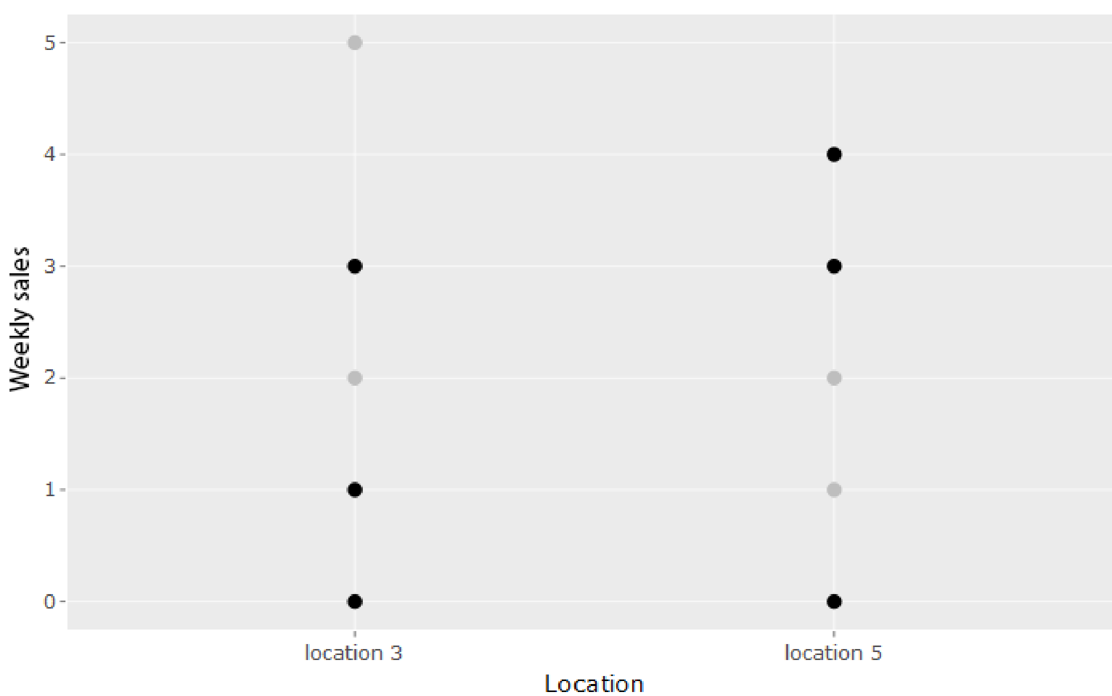

Figure 14: Visualization of weekly sales of two products on two locations.

In this example product p_2 and location ℓ_3 are chosen as the benchmark product and location. Hence, $\hat{X}(p_2, \ell_3) = e^{\gamma_0}$ is the expected weekly sales of product p_2 on its historical location ℓ_3. The expected weekly sales $X(p_2, \ell_5)$, $X(p_4, \ell_3)$ and $X(p_4, \ell_5)$ are given by $\hat{X}(p_2, \ell_5) = e^{\gamma_0+\beta_5}$, $\hat{X}(p_4, \ell_3) = e^{\gamma_0+\alpha_4}$ and $\hat{X}(p_4, \ell_5) = e^{\gamma_0+\alpha_4+\beta_5}$.

So using the Poisson regression, Equation 3 is used to predict the expected weekly sales of products on its historical locations.

Observe that a correction β_j that is made for a change in location ℓ_j from the benchmark location, is based on the sales of the products that were positioned on this location ℓ_j in the past. So the correction β_j is actually induced by the products that were sold on location ℓ_j, and not only by the characteristics of the location itself. This will be further discussed in Section 7.

Computational validation

We proposed two methods to predict the expected weekly sales of products on historical locations. In order to determine which method we shall use, a validation on the predictions via Equation 1 and Equation 3 will be executed. The program R is used for this.

We are aware of the fact that given the quality of the data, we can not be very certain about the conclusions we can draw from the validations. However, it does give an indication about the performance of both methods.

For the validation we split the data set D, illustrated in Table 1, into two mutually exclusive subsets, named the training set $D_{training}$ and the test set D_{test}. We will validate the methods for all products P to obtain a validation that is applicable for the whole store.
The training set is used to train the model, i.e. to predict the expected weekly sales $X(p_i, \ell_j)\ \forall\ p_i \in P, \ell_j \in L_i(C_x)$. These expected weekly sales can be compared to the actual weekly sales the corresponding product and location included in the test set. In order to validate each expected weekly sales $X(p_i, \ell_j)$, every (p_i, ℓ_j) combination needs to be present in the test set as well as in the training set. Therefore we include 20 % of the weekly sales of each product on its historical location (p_i, ℓ_j) in the test set, rounded up to an integer. The other 80 % is included in the training set. The training set is therefore defined by $D_{training} = D \setminus D_{test}$. The weeks of historical sales of product p_i on historical location ℓ_j that are included in D_{test} are denoted by $W^{(test)}(p_i, \ell_j)$, where $W^{(test)}(p_i, \ell_j) \subseteq W(p_i, \ell_j)$.
In the validation, each prediction $X(p_i, \ell_j)$ will be compared to every occurring $Q_t(p_i, \ell_j)$ in D_{test}. During the validation a metric has to be chosen to compare the predictions with the real weekly sales. For this validation we will use the mean absolute error (MAE) to determine the performance of both methods. The performance of a model will then be given by:

$$MAE = \frac{\sum_{p_i \in P} \sum_{\ell_j \in L_i(C_x)} \sum_{w_t \in W^{(test)}(p_i, \ell_j)} \left| \hat{X}(p_i, \ell_j) - Q_t(p_i, \ell_j) \right|}{\sum_{p_i \in P} \sum_{\ell_j \in L_i(C_x)} |W^{(test)}(p_i, \ell_j)|}. \tag{4}$$

Recall that $Q_t(p_i, \ell_j)$ is the number of sales of product p_i on its historical location ℓ_j in week w_t.
Thus, the MAE is the average deviation between the predicted expected weekly sales and the historical weekly sales over all observations of products on its historical locations that are included in D_{test}.
If $X(p_i, \ell_j)$ is predicted by the Poisson regression, we denote the MAE by $MAE_{Poisson}$ and if the average is used to predict $X(p_i, \ell_j)$, the MAE is denoted by $MAE_{average}$.

To compare the performance of the methods as objective as possible, we need to validate the predictions of the methods on different test sets, i.e. validate whether the predictions of the methods fit different samples of reality.
Therefore we will compute twenty different samples of D from which D_{test} and thereby also D_{train} is obtained. In each of those twenty runs, we compute the $MAE_{Poisson}$ and $MAE_{average}$. In this way, each method is trained and tested on a different part of the data set D. The results of the MAE for both methods

are depicted in Table 4 together with the mean of the historical sales of the test set D_{test} of each run.

run	$MAE_{Poisson}$	$MAE_{average}$	Mean of historical sales
1	1.1576	1.1762	1.0302
2	1.0995	1.1102	1.0532
3	1.2157	1.2329	1.1671
4	1.0877	1.0859	1.0818
5	1.1413	1.1495	1.0741
6	1.0602	1.0786	1.1732
7	1.1471	1.1777	1.1980
8	1.1317	1.1389	1.0238
9	1.1586	1.1841	1.2608
10	1.1088	1.1198	1.1745
11	1.1140	1.1232	1.1149
12	1.1098	1.1142	1.1628
13	1.1028	1.1314	1.1538
14	1.0771	1.0901	1.1276
15	1.1470	1.1688	1.2687
16	1.1239	1.1411	1.1112
17	1.1391	1.1570	1.1239
18	1.0961	1.0948	1.1366
19	1.0332	1.0575	1.0887
20	1.1238	1.1395	1.1642

Table 4: $MAE_{Poisson}$ and $MAE_{average}$ for 20 runs.

In order to interpret each MAE, the last column is added which indicates the mean of the historical sales of the corresponding D_{test}. In each run, the MAE represents the mean error the predictions deviate from the mean sales. It can be noticed that in most of the runs the deviation is almost equal to the mean. This confirms the remark at the beginning of this section that the methods will be imperfect due to the quality of the data that is available.

Moreover, it can be observed that the differences between $MAE_{Poisson}$ and $MAE_{average}$ are negligible. We already argued that the available sales of the historical weekly sales of a product on its historical location can not be fitted by a Poisson distribution very well. Due to this and the fact that the difference in performance between the Poisson regression and taking the average is negligible, we do not predict the weekly sales on historical locations by a Poisson regression as in Equation 3.
For taking the average, the only requirement is that the average of weekly sales of a product exist. This is the case since the weekly sales are non-negative and bounded.
Therefore we will use the average to predict the expected weekly sales of products on its historical locations.

In the next section we will provide the method by which the expected weekly sales of products on its non-historical locations are estimated. In this method we pretend like the products of a group are identical and therefore use the historical weekly sales of the group $R(p_i)$ to predict the expected weekly sales of a product p_i on the non-historical locations of the group $R(p_i)$. Section 4.1.2 explains this in detail, but to compare the expected weekly sales of a product objectively over all locations $L(C_x)$, we will also use the historical weekly sales of a group $R(p_i)$ to predict the expected weekly sales of the historical locations of the group $R(p_i)$.
Thus, instead of Equation 1, we will use the average of the weekly sales of group $R(p_i)$ to predict $X(p_i, \ell_j)$ for locations $\ell_j \in \bigcup_{p_f \in R(p_i)} L_f(C_x)$:

$$\hat{X}(p_i, \ell_j) = \frac{\sum_{p_f \in R(p_i)} \sum_{w_t \in W(p_i, \ell_j)} Q_t(p_f, \ell_j)}{\sum_{p_f \in R(p_i)} |W(p_f, \ell_j)|} \qquad \forall \ell_j \in \bigcup_{p_f \in R(p_i)} L_f(C_x). \tag{5}$$

4.1.2 Estimate expected weekly sales of products on non-historical locations

So far, we only discussed how to predict the expected weekly sales of products on their historical locations. As mentioned at the beginning of Section 4.1, we are aware of the fact that the methods we use to predict the expected weekly sales are imperfect. This applies to the predictions on historical locations of products, but certainly also to the predictions on non-historical locations of products.

Section 3.2.2 describes how the graphs B_x and U_x are formed for each category, which are representations of the connectivity between the locations $L(C_x)$ regarding subgroups and groups respectively. If we would like to predict the expected weekly sales of a product p_i on a location ℓ_k it never stood before, the product might have been positioned on locations that are in close proximity, i.e. locations that are adjacent to ℓ_k in U_x. However, most products have stood on one or two locations in available dataset. As mentioned at the end of Section 4.1.1, it is more interesting to look at the products of its group $R(p_i)$, i.e. products that are similar to p_i, since it is reasonable that these products together have stood on more locations than the product p_i itself.
Therefore the expected weekly sales of product p_i on the historical locations of its group $R(p_i)$, are predicted by the average sales of its group on this location. We do not take the average of the weekly sales of the product itself anymore because we want to compare the expected weekly revenues of a product, predicted from the expected weekly sales, objectively over all locations. This can only be done if we use the historical sales of a group for all locations. Otherwise, the best performing products of a group will be assigned to one of its historical locations and the worst performing products of a group will be assigned to non-historical locations.

To predict the expected weekly sales on non-historical locations, we will use a weighted average, based on the idea of the kriging method, which uses statistical interpolation on spatial data [10]. To predict the sales of a non-historical location of group $R(p_i)$, we will use the expected weekly sales of group $R(p_i)$ on locations that are in close proximity to the non-historical location of group $R(p_i)$.
Notice that hereafter we will define some new notation in order to explain how this results in a weighted average.

We want to estimate $X(p_i, \ell_k)$ for non-historical location ℓ_k of product group $R(p_i)$ by using the predictions $\hat{X}(p_f, \ell_j)$ of products $p_f \in R(p_i)$ on adjacent locations $\ell_j \in \{L(C_x) | (\ell_k, \ell_j) \in E(U_x)\}$.
This is done in the following way:

$$\hat{X}(p_i, \ell_k) = \begin{cases} \sum_{\ell_j \in L(C_x)} \overline{d_{k,j}^{(i)}} \hat{X}(p_i, \ell_j), & \text{if } \sum_{\ell_j \in L(C_x)} \overline{d_{k,j}^{(i)}} > 0 \\ \frac{\sum_{\ell_j \in L(C_x)} \sum_{p_f \in R(p_i)} \sum_{t \in W(p_i, \ell_j)} Q_t(p_f, \ell_j)}{\sum_{\ell_j \in L(C_x)} \sum_{p_f \in R(p_i)} |W(p_f, \ell_j)|}, & \text{if } \sum_{\ell_j \in L(C_x)} \overline{d_{k,j}^{(i)}} = 0 \end{cases} \tag{6}$$

The clarification of the different cases will be explained after the explanation of the weights $\overline{d_{k,j}^{(i)}}$.
These weights $\overline{d_{k,j}^{(i)}}$ denote the normalized closeness between location ℓ_j and ℓ_k regarding group $R(p_i)$ and therefore $\sum_{\ell_j \in L(C_x)} \overline{d_{k,j}^{(i)}} = 1$.
The weights $\overline{d_{k,j}^{(i)}}$ are obtained by:

$$\overline{d_{k,j}^{(i)}} = \begin{cases} \frac{d_{k,j}^{(i)}}{\sum_{\ell_q \in L(C_x)} d_{k,q}^{(i)}}, & \text{if } \sum_{\ell_q \in L(C_x)} d_{k,q}^{(i)} > 0 \\ 0, & \text{if } \sum_{\ell_q \in L(C_x)} d_{k,q}^{(i)} = 0 \end{cases}$$

The weight $d_{k,j}^{(i)}$ indicate the closeness between the locations ℓ_k and ℓ_j regarding group $R(p_i)$, by a value between zero and one. The weights are determined according to the scenario's that induce the edges in U_x, described in Section 3.2.2. A weight of zero indicates that the location ℓ_j is not comparable to ℓ_k regarding group $R(p_i)$ and a weight of one would indicate that location ℓ_j is perfectly comparable to location ℓ_k

regarding group $R(p_i)$:

$$d_{k,j}^{(i)} = \begin{cases} 1, & \text{If } \ell_j = \ell_k \text{ and group } R(p_i) \text{ was positioned on location } \ell_j \\ 0.8, & \begin{array}{l}\text{If location } \ell_j \text{ is adjacent to location } \ell_k \text{ according to scenario 1,} \\ \text{and group } R(p_i) \text{ was positioned on location } \ell_j\end{array} \\ 0.6, & \begin{array}{l}\text{If location } \ell_j \text{ is adjacent to location } \ell_k \text{ according to scenario 2,} \\ \text{and group } R(p_i) \text{ was positioned on location } \ell_j\end{array} \\ 0.4, & \begin{array}{l}\text{If location } \ell_j \text{ is adjacent to location } \ell_k \text{ according to scenario 3,} \\ \text{and group } R(p_i) \text{ was positioned on location } \ell_j\end{array} \\ 0.2, & \begin{array}{l}\text{If location } \ell_j \text{ is adjacent to location } \ell_k \text{ according to scenario 4,} \\ \text{and group } R(p_i) \text{ was positioned on location } \ell_j\end{array} \\ 0, & \begin{array}{l}\text{If there is none of the four scenario's applicable to locations } \ell_j \text{ and } \ell_k, \\ \text{or group } R(p_i) \text{ was not positioned on location } \ell_j.\end{array} \end{cases}$$

Note that $d_{k,j}^{(i)} = 1$ is only added for completeness because this weight will never be used. If group $R(p_i)$ was positioned on location $\ell_j = \ell_k$, then $X(p_i, \ell_k)$ is predicted by Equation 5.
The weights $d_{k,j}^{(i)}$ are based on the expertise of a data expert in the field of retail because we could not compute reliable correlations between locations due to the quality of the data. This will also be discussed in Section 7.

Further observe that the weights $d_{k,j}^{(i)}$ depend on the group $R(p_i)$ because not every group is positioned on every location of $L(C_x)$. To illustrate this, consider the case that location ℓ_j might be adjacent to ℓ_k in graph U_x, but if the group of the product p_i which we try to predict, was never positioned on location ℓ_j, this location can not be used for the expected weekly sales of product p_i on non-historical location ℓ_k. For a product of a different group, this group might be positioned on location ℓ_j and this location can be used to estimate the expected weekly sales. Therefore a weight $d_{k,j}^{(i)}$ depends on the group $R(p_i)$.

At last, consider the case in Equation 6 where the group $R(p_i)$ was not positioned on any of the adjacent locations of ℓ_k in the past. Then there is no location to use for the estimation of the expected weekly sales $X(p_i, \ell_k)$ and therefore $\sum_{\ell_j \in L(C_x)} d_{k,j}^{(i)} = 0$ and thus $\sum_{\ell_j \in L(C_x)} \overline{d_{k,j}^{(i)}} = 0$. In this case, the expected weekly sales $X(p_i, \ell_k)$ will be predicted by the average sales of the group $R(p_i)$ over all locations of $L(C_x)$.

So far we have a method to predict the expected weekly sales $X(p_i, \ell_j)$ for the historical locations and non-historical locations of its group $R(p_i)$. As input for computing the revenue maximizing assignment of category C_x, we need the expected weekly revenues $Y(p_i, \ell_j)$ for all products $P(C_x)$ on locations $L(C_x)$. The expected weekly revenues $Y(p_i, \ell_j)$ are predicted by multiplying each expected weekly sales $X(p_i, \ell_j)$ by its gross profit $g(p_i)$:

$$\hat{Y}(p_i, \ell_j) = g(p_i) \cdot \hat{X}(p_i, \ell_j) \qquad \forall p_i \in P(C_x), \ell_j \in L(C_x). \tag{7}$$

Summarizing, the expected weekly sales of a product p_i on the historical locations of its group $R(p_i)$ are predicted by the average sales of the group $R(p_i)$ on this location:

$$\hat{X}(p_i, \ell_j) = \frac{\sum_{p_f \in R(p_i)} \sum_{w_t \in W(p_i, \ell_j)} Q_t(p_f, \ell_j)}{\sum_{p_f \in R(p_i)} |W(p_f, \ell_j)|} \qquad \forall \ell_j \in \bigcup_{p_f \in R(p_i)} L_f(C_x).$$

To predict the expected weekly sales of the product p_i on the non-historical locations of its group $R(p_i)$, we use a weighted average on the expected weekly sales of the products of its group $R(p_i)$ on locations that are adjacent to these non-historical locations in U_x. If it occurs that the group $R(p_i)$ was not positioned on locations that are adjacent to a non-historical location of group $R(p_i)$, the expected weekly sales of product p_i on this location will be the average sales of the group $R(p_i)$ over all locations $L(C_x)$:

$$\hat{X}(p_i, \ell_k) = \begin{cases} \sum_{\ell_j \in L(C_x)} \overline{d_{k,j}^{(i)}} \hat{X}(p_i, \ell_j), & \text{if } \sum_{\ell_j \in L(C_x)} \overline{d_{k,j}^{(i)}} > 0 \\ \frac{\sum_{\ell_j \in L(C_x)} \sum_{p_f \in R(p_i)} \sum_{t \in W(p_i, \ell_j)} Q_t(p_f, \ell_j)}{\sum_{\ell_j \in L(C_x)} \sum_{p_f \in R(p_i)} |W(p_f, \ell_j)|}, & \text{if } \sum_{\ell_j \in L(C_x)} \overline{d_{k,j}^{(i)}} = 0 \end{cases}$$

With the expected weekly sales $X(p_i, \ell_j)$ and gross profits $g(p_i)$, we can predict the expected weekly revenues $Y(p_i, \ell_j)$:

$$\hat{Y}(p_i, \ell_j) = g(p_i) \cdot \hat{X}(p_i, \ell_j) \qquad \forall p_i \in P(C_x), \ell_j \in L(C_x).$$

4.2 Compute the revenue maximizing assignment to the categorized assignment problem

This section will explain how to obtain the revenue maximizing assignment for the CA-problem, defined in Definition 3. It appears that the CA-problem is NP-hard, which will be proved in Section 4.2.1. After that, Section 4.2.2 will provide the Integer Linear Program whereby the revenue maximizing assignment for the CA-problem is computed.

4.2.1 NP-hardness

To show that the CA-problem is NP-hard, we will prove that a simplified version of the CA-problem is NP-hard. If this holds, then the CA-problem itself is also NP-hard.
We will derive to this simplified version in two steps. First assume that the requirement of Definition 2 is only considered for groups, that is that the locations allocated to every group should form a connected subgraph in U_x. So focused on one group, the aim is to assign products of this group to locations such that these locations form a connected subgraph in the corresponding graph U_x and the number of assigned products to a vertex does not exceed its capacity. This can be defined mathematically as the following problem:

Capacitated connected subgraph problem (CCS)
Given a graph $U_x = (V, E)$ with vertex capacities $A(v)$, vertex weights $w(v)$ and a positive integer k, find a connected subgraph (V', E') with vertex loads $r(v)$ such that $\sum_{v \in V'} r(v) = k$, $r(v) \leq A(v)\ \forall v \in V'$ and $\sum_{v \in V'} r(v) \cdot w(v)$ is maximized.

One may think of k as the number of products of some group that have to be assigned. The vertices V of U_x correspond to the locations $L(C_x)$. The vertex load $r(v)$ is the number of products which is assigned to vertex v and trivially, this number of products may not exceed its capacity $A(v)$. The vertex weights $w(v)$ correspond to expected weekly revenue of products to locations. Here the expected weekly revenue of every product on a location v has the same weight $w(v)$, so this also is a simplification of the CA-problem.
Now we will simplify the CCS-problem by setting all vertex capacities equal to one. Actually we neglect the capacity constraint by allowing a vertex to be allocated to only one product and therefore the vertex loads $r(v)$ become unnecessary. With this simplification we will study a well known problem in the literature:

Maximum weight connected subgraph (MWCS)
Given a graph $G = (V, E)$ with vertex weights $w(v)$ and a positive integer k, find a connected subgraph $H = (V', E')$ with $|V'| \leq k$ such that $\sum_{v \in V'} w(v)$ is maximized.

Below we will provide a proof to show that $MWCS$ is a NP-hard problem even if $w(v) \in \{0, 1\}\ \forall v \in V$ [11]. We do this by using a special case of the well known Steiner tree problem, which is a NP-hard problem [12]:

Steiner Tree (ST)
Given a graph $\overline{G} = (\overline{V}, \overline{E})$, $R \subseteq \overline{V}$, find a tree $T = (\overline{V'}, \overline{E'})$ in $\overline{G}$ that spans R with a minimum number of vertices $|\overline{V'}|$.

We will prove that $MWCS$ with binary vertex weights is NP-hard by showing that the decision version of $MWCS$ is NP-complete.

Theorem 1. *The decision version of the $MWCS$ problem is NP-complete, even if the vertex weights are restricted to be binary: $w(v) \in \{0, 1\}\ \forall v \in V$.*

Proof.
The decision version of the $MWCS$ with binary vertex weights is formulated as follows:
Given a graph $G = (V, E)$ with vertex weights $w(v) \in \{0, 1\}$, positive integers k and z , is there a connected subgraph $H = (V', E')$ with $|V'| \leq k$ and $w(H) \geq z$?

First, it is not hard to see that the decision version of the $MWCS$ is in NP. Given a subgraph $H = (V', E')$ of an instance of $MWCS$, it can be decided in polynomial time whether this subgraph H satisfies the conditions $|V'| \leq k$, $w(H) \geq z$ and whether the subgraph H is connected [13].

The decision version of the ST problem is as follows:
Given a graph $\overline{G} = (\overline{V}, \overline{E})$, $R \subseteq \overline{V}$, an integer $l \geq |R|$, is there a tree $T = (\overline{V'}, \overline{E'})$ in $\overline{G}$ that spans R with $|\overline{V'}| \leq l$?

To reduce this problem to the decision version of the $MWCS$ problem, we introduce the polynomial-time computable mapping $f : I_{ST} \rightarrow I_{MWCS}$ that maps every instance of the Steiner tree to an instance of the $MWCS$. If we can prove that this mapping maps every yes-instance of I_{ST} to a yes-instance of I_{MWCS} and idem dito for every no-instance, then we have found a reduction from ST to $MWCS$.
Let $\overline{G} = (\overline{V}, \overline{E})$, $R \subseteq \overline{V}$, an integer $l \geq |R|$ be an instance of I_{ST}. The mapping f is then defined as follows:

1. Create a node-weighted graph $G = (V, E)$, with $V = \overline{V}$, $E = \overline{E}$ and vertex weights:

$$w(v) = \begin{cases} 1, & \forall v \in R \\ 0, & \forall v \in V \setminus R \end{cases}$$

2. $z = |R|$, $k = l$.

Consider a yes-instance of I_{ST}: A Steiner tree T of q vertices ($q \leq l$) and $q - 1$ edges (since T is a tree) that spans all vertices of R. Now consider this tree T as the subgraph H, that is $H = T$.
Then in $G = (V, E)$ this tree H contains less than or equal to k vertices and $w(H) \geq z$ since all vertices with $w(v) = 1$ are contained in T by construction. Since a tree is a connected subgraph, H is a connected subgraph. So every yes-instance of I_{ST} will be mapped to a yes-instance of I_{MWCS}.

Consider a no-instance of I_{ST}: There is no tree $T = (\overline{V'}, \overline{E'})$ in $\overline{G}$ with $|\overline{V'}| \leq l = k$ that spans all R vertices. This means that all Steiner trees in $\overline{G}$ contain $|\overline{V'}| > k$ vertices. So in G, any connected subgraph H with a weight $w(H) \geq z$ contains more than k vertices.
We show this by contradiction. Assume that there is a connected subgraph $H = (V', E')$ in G with $|V'| \leq k = l$ and $w(H) \geq z = |R|$, then in $\overline{G}$ there should be a tree $T = (\overline{V'}, \overline{E'})$ that spans R with $|\overline{V'}| \leq l$ because of the following:

1. If the connected subgraph H described above already is a tree, $T = H$ would have been a tree in $\overline{G}$ that spans R and contains $|T(\overline{V'})| \leq l$ vertices.
2. If the connected subgraph H described above is not a tree and thus contains one or more cycles, then we can remove edges from H under the condition that H remains connected until eventually a tree T is obtained. By this, T would be a tree in $\overline{G}$ that still spans R and still contains $|T'(\overline{V'})| \leq l$ vertices.

In both cases, this leads to a contradiction since we assumed such a Steiner tree did not exist in $\overline{G}$.
So every no-instance of I_{ST} will indeed be mapped to a no-instance of I_{MWCS}.

Thus, we can map every yes-instance of I_{ST} to a yes-instance of I_{MWCS} and map every no-instance of I_{ST} to a no-instance of I_{MWCS} in polynomial time.
Since ST is a NP-hard problem and we found a polynomial-time computable reduction from the decision version of ST to the decision version of $MWCS$, which is in NP, we can conclude that the decision version of $MWCS$ is NP-complete. ■

We proved that the decision version of $MWCS$ is NP-complete and therefore $MWCS$ is NP-hard to solve. Since the $MWCS$ is a simplified version of the CCS problem, the CCS problem is also a NP-hard problem. As argued earlier, to solve the CA-problem, the CCS problem has to be solved for every group. Thus, with the proof above we can state that the CA-problem is also a NP-hard problem.

Now we know that the CA-problem is NP-hard, it is justified to set op an Integer Linear Program (ILP) to solve the CA-problem. The following section describes the way this ILP is created and ends with the ILP as a whole.

4.2.2 Integer Linear Program to solve the CA-problem

We are going to set up an ILP to solve the CA-problem.
First of all, we will denote the expected weekly revenues $Y(p_i, \ell_j)$ by $w(p_i, \ell_j)$ $\forall p_i \in P(C_x), \ell_j \in L(C_x)$.
Furthermore, we have to indicate whether a product is assigned to a location within a solution. Therefore we define the following binary variable:

$$x(p_i, \ell_j) = \begin{cases} 1, & \text{When product } p_i \text{ is assigned to location } \ell_j \\ 0, & \text{Otherwise.} \end{cases}$$

In the assignment every product should be assigned to exactly one location. Using the variable $x(p_i, \ell_j)$, this requirement can be formulated as the following constraint:

Allocation constraint.

$$\sum_{\ell_j \in L(C_x)} x(p_i, \ell_j) = 1 \ \forall p_i \in P(C_x). \tag{8}$$

Furthermore Section 3.2.2 already describes the feasibility constraints that a solution has to satisfy. As said in Definition 1, the number of assigned products to a location ℓ_j, may not exceed its capacity $A(\ell_j)$. This is translated into the following constraint:

Capacity constraint.

$$\sum_{p_i \in P(C_x)} x(p_i, \ell_j) \leq A(\ell_j) \ \forall \ell_j \in L(C_x). \tag{9}$$

Recall that every product p_i belongs to a subgroup $S(p_i)$ and a group $R(p_i)$. Thus, besides a set of products $P(C_x)$, every category C_x also contains a set of subgroups and groups denoted by $S(C_x)$ and $R(C_x)$ respectively.
In addition to the capacity constraint, it has to be ensured that the products of a subgroup are assigned to locations that form a connected subgraph in B_x. Before stating this in a constraint, we first need a variable that indicates whether a subgroup $S_h \in S(C_x)$ is assigned to a location ℓ_j:

$$y(S_h, \ell_j) = \begin{cases} 1, & \text{When subgroup } S_h \text{ is assigned to location } \ell_j \\ 0, & \text{Otherwise.} \end{cases}$$

Whether a subgroup is assigned to a location ℓ_j, only depends on the fact whether a product of this subgroup is assigned to location ℓ_j. If one or more products of the subgroup are assigned to location ℓ_j, the subgroup is also assigned to location ℓ_j. If none of the products of the subgroup is assigned to location ℓ_j, the subgroup itself is neither assigned to location ℓ_j. To force this, we use the following two constraints:

Coupling constraints.

$$y(S_h, \ell_j) \geq x(p_i, \ell_j) \ \forall S_h \in S(C_x), p_i \in S_h, \ell_j \in L(C_x) \tag{10}$$

$$y(S_h, \ell_j) \leq \sum_{p_i \in S_h} x(p_i, \ell_j) \ \forall S_h \in S(C_x), \ell_j \in L(C_x). \tag{11}$$

Consider the case where $x(p_i, \ell_j) = 0$ for all products $p_i \in S_h$. Then $y(S_h, \ell_j)$ is free to choose regarding the constraint of Equation 10, but the constraint of Equation 11 forces $y(S_h, \ell_j) = 0$ because the value of the sum equals zero.

If there is some product $p_i \in S_h$ for which $x(p_i, \ell_j) = 1$, then $y(S_h, \ell_j)$ is free to choose regarding the constraint of Equation 11 because the value of the sum is one or higher. However, the constraint of Equation 10 enforces $y(S_h, \ell_j) = 1$ because there is at least one product of subgroup S_h that is assigned to location ℓ_j. Thus, the constraints of Equation 10 and 11 force $y(S_h, \ell_j)$ to be the correct binary value.

Now we know whether a subgroup is positioned on a location or not, it is possible to check if the locations a subgroup is assigned to, indeed form a connected subgraph in B_x. As explained in Section 3.2.2, the graph B_x only consists of paths. Therefore there exists one or no path between two vertices ℓ_j and ℓ_k in B_x. The locations located on such a path are defined by the set $P(\ell_j, \ell_k)$. Using each set $P(\ell_j, \ell_k)$, we can formulate the following constraint which checks for every subgroup S_h if the vertices for which $y(S_h, \ell_j) = 1$ form a connected subgraph in B_x:

Connectivity constraint regarding subgroups.

$$y(S_h, \ell_j) + y(S_h, \ell_k) - \sum_{l_r \in P(\ell_j, \ell_k)} y(S_h, l_r) \leq 1 \ \forall S_h \in S(C_x), \forall \ell_j, \ell_k \in \{L(C_x) | (\ell_j, \ell_k) \notin E(B_x)\}. \tag{12}$$

Observe that only for the case where $y(S_h, \ell_j) = 1$ and $y(S_h, \ell_k) = 1$ it has to be determined whether the constraint is satisfied. For all other cases the constraint is automatically satisfied since the left-hand side is one or less.
So the constraint verifies for each pair of vertices $\{(\ell_j, \ell_k) | (\ell_j, \ell_k) \notin E(B_x)\}$ for which $y(S_h, \ell_j) = 1$ and $y(S_h, \ell_k) = 1$, if there is at least one vertex $\ell_r \in P(\ell_j, \ell_k)$ for which $y(S_h, l_r) = 1$. If this holds, then the locations a subgroup is assigned to, form a path and thus a connected subgraph in B_x. Note that even if $y(S_h, \ell_j) = 1$ and $y(S_h, \ell_k) = 1$, the constraint does not need to be checked if ℓ_j and ℓ_k are adjacent vertices in B_x because adjacent vertices are always connected.
To illustrate that the constraint of Equation 12 correctly verifies connectivity of the induced subgraph, consider an extension of the graph B_x which is depicted in Figure 5 of Section 3.2.2:

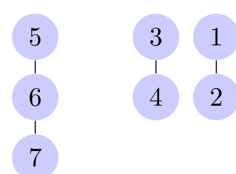

Figure 15: An extended version of the graph $B_x = (V, \widetilde{E})$ of Figure 5.

We added a seventh vertex to create the pair (ℓ_5, ℓ_7) where the vertices are reachable but not adjacent. Consider the case that $y(S_h, l_5) = 1$ and $y(S_h, l_7) = 1$, but $y(S_h, l_6) = 0$ for some $S_h \in S(C_x)$. With $P(l_5, l_7) = \{l_6\}$, the constraint of Equation 12 will not be satisfied since $y(S_h, l_6) = 0$ and therefore the left-hand side of the constraint is two, which is higher than one.
If $y(S_h, l_5) = 1$, $y(S_h, l_7) = 1$ and $y(S_h, l_6) = 1$, the constraint of Equation 12 is satisfied and the locations $\{l_5, l_6, l_7\}$ the subgroup S_h is assigned to, indeed form a connected subgraph in B_x.
Secondly, consider the case that $y(S_h, l_5) = 1$, $y(S_h, l_3) = 1$, then constraint 12 will be violated since $P(l_3, l_5) = \{\emptyset\}$ because ℓ_3 and ℓ_5 are not reachable. Thus, the constraint of Equation 12 is correctly violated for locations that are not reachable in B_x.

Besides the connectivity check for the locations that are allocated to each subgroup, we have to verify whether the locations assigned to each group R_d form a connected subgraph in U_x.
To specify whether a group is assigned to a location, we create the variable $z(R_d, \ell_j)$:

$$z(R_d, \ell_j) = \begin{cases} 1, & \text{When group } R_d \text{ is assigned to location } \ell_j \\ 0, & \text{Otherwise.} \end{cases}$$

Similar to subgroups and products, a group R_d is assigned to a location ℓ_j if one or more subgroups of this group are assigned to location ℓ_j. To provoke this, we need the following two constraints which are similar to the constraints of Equation 10 and 11:

Coupling constraints.

$$z(R_d, \ell_j) \geq y(S_h, \ell_j) \ \forall R_d \in R(C_x), S_h \in R_d, \ell_j \in L(C_x) \tag{13}$$

$$z(R_d, \ell_j) \leq \sum_{S_h \in R_d} y(S_h, \ell_j) \ \forall R_d \in R(C_x), \ell_j \in L(C_x). \tag{14}$$

Now we know whether a group is assigned to a location in U_x, we can focus on the connectivity of all the locations a group R_d is assigned to. In comparison to B_x, in U_x there might exist several paths between vertices ℓ_j and ℓ_k because U_x can contain cycles. Therefore the constraint of Equation 12 should be adjusted in order to check connectivity regarding groups.
To specify the paths corresponding to locations ℓ_j and ℓ_k, we first create the set that consists of the paths between every two vertices in U_x, i.e. the set that contains all paths of U_x: that is, $T = \{t_1, t_2, \ldots, t_{n_t}\}$. Here a path $t_m \in T$ contains the vertices $\ell_j \in L(C_x)$ this path t_m consists of without the source and destination vertex of the path. These paths can be found using a depth-first search [14]. The paths within U_x corresponding to source ℓ_j and destination ℓ_k are denoted by the set $T(\ell_j, \ell_k)$ and is a subset of T: that is, $T(\ell_j, \ell_k) \subseteq T$.
Comparable to the constraint of Equation 12, for every group R_d we want to check if any two locations (ℓ_j, ℓ_k) for which $z(R_d, \ell_j) = 1$ and $z(R_d, \ell_k) = 1$, are connected via vertices that are also assigned to group R_d, i.e. if there is a path $t_m \in T(\ell_j, \ell_k)$ where all vertices of this path are also assigned to R_d. To specify whether this holds for a given path $t_m \in T$ and group R_d, we introduce the following variable:

$$\gamma(t_m, R_d) = \begin{cases} 1, & \text{If } z(R_d, \ell_j) = 1 \text{ for all vertices } \ell_j \in t_m \\ 0, & \text{Otherwise.} \end{cases}$$

To force $\gamma(t_m, R_d)$ to the correct binary value, we need the following two constraints:

Path constraints.

$$\gamma(t_m, R_d) \leq z(R_d, \ell_j) \ \forall R_d \in R(C_x), t_m \in T, \ell_j \in t_m \tag{15}$$

$$\gamma(t_m, R_d) \geq \Big(\sum_{\ell_j \in t_m} [z(R_d, \ell_j) - 1] \Big) + 1 \ \forall R_d \in R(C_x), t_m \in T. \tag{16}$$

If $z(R_d, \ell_j) = 1$ for all vertices $\ell_j \in t_m$, $\gamma(t_m, R_d)$ is free to choose regarding the constraint of Equation 15. However, the right-hand side of the constraint of Equation 16 is equal to one since the summation equals zero. As a result, $\gamma(t_m, R_d)$ is set to one correctly.
If there is a vertex $\ell_j \in t_m$ for which $z(R_d, \ell_j) = 0$, $\gamma(t_m, R_d)$ is free to choose regarding the constraint of Equation 16 since the right-hand side of this constraint is zero or less. However, the constraint of Equation 15 forces $\gamma(t_m, R_d)$ to be zero since $\gamma(t_m, R_d) \leq 0$ for the location ℓ_j that was not assigned to group R_d.

Now it is correctly specified whether all vertices of a path are assigned to a group, we can model a constraint that verifies for any pair of locations (ℓ_j, ℓ_k) for which $z(R_d, \ell_j) = 1$ and $z(R_d, \ell_k) = 1$ whether there is a path $t_m \in T(\ell_j, \ell_k)$ for which $\gamma(t_m, R_d) = 1$ that assures connectivity between ℓ_j and ℓ_k. If there is such a path for any two locations that are assigned to group R_d, the locations the group R_d is assigned to, form a connected subgraph in U_x. This is done by the following constraint:

Connectivity constraint regading groups.

$$\sum_{t_m \in T(\ell_j, \ell_k)} \gamma(t_m, R_d) \geq 1 - \big[1 - z(R_d, \ell_j)\big] - \big[1 - z(R_d, \ell_k)\big] \ \forall R_d \in R(C_x), \ell_j, \ell_k \in \{L(C_x) | (\ell_j, \ell_k) \notin E(U_x)\}. \tag{17}$$

For the case that $z(R_d, \ell_j) = 1$ and $z(R_d, \ell_k) = 1$, the constraint of Equation 17 is only satisfied if there is at least one path $t_m \in T(\ell_j, \ell_k)$ for which all the vertices on this path are assigned to group R_d and thus ℓ_j and ℓ_k are connected via this path.
If $z(R_d, \ell_j) = 0$, $z(R_d, \ell_j) = 0$ or both are zero, the constraint is automatically satisfied since the right-hand side is zero or less.
With the constraint of Equation 17, we have modelled all the feasibility constraints a solution for the CA-problem has to satisfy.

Briefly summarizing, for a given category C_x, the CA-problem wants to assign the corresponding products $P(C_x)$ to the corresponding locations $L(C_x)$ with the objective to maximize the total expected weekly revenue of the assignment $\sum_{p_i \in P(C_x), \ell_j \in L(C_x)} x(p_i, \ell_j) w(p_i, \ell_j)$.
The locations $L(C_x)$ located in the store layout induce the connectivity graphs B_x and U_x regarding subgroups and groups respectively. The locations in the revenue maximizing solution that are assigned to a subgroup S_h should form a connected subgraph in B_x such that the products of subgroup S_h are positioned in close proximity to each other. Comparable, the locations in the revenue maximizing solution that are assigned to a group R_d should form a connected subgraph in U_x such that the products of this group R_d are positioned in close proximity to each other.
Furthermore the number of products assigned to a location may not exceed its capacity and every product is assigned to one location only.

The Integer Linear Program as a whole then looks as follows:

$$\max \sum_{p_i \in P(C_x)} \sum_{\ell_j \in L(C_x)} w(p_i, \ell_j) x(p_i, \ell_j)$$

subject to

$$\sum_{\ell_j \in L(C_x)} x(p_i, \ell_j) = 1 \qquad \forall p_i \in P(C_x) \qquad \textit{(Eq. 8)}$$

$$\sum_{p_i \in P(C_x)} x(p_i, \ell_j) \leq A(\ell_j) \qquad \forall \ell_j \in L(C_x) \qquad \textit{(Eq. 9)}$$

$$y(S_h, \ell_j) \geq x(p_i, \ell_j) \qquad \forall S_h \in S(C_x), p_i \in S_h, \ell_j \in L(C_x) \qquad \textit{(Eq. 10)}$$

$$y(S_h, \ell_j) \leq \sum_{p_i \in S_h} x(p_i, \ell_j) \qquad \forall S_h \in S(C_x), \ell_j \in L(C_x) \qquad \textit{(Eq. 11)}$$

$$y(S_h, \ell_j) + y(S_h, \ell_k) - \sum_{l_r \in P(\ell_j, \ell_k)} y(S_h, l_r) \leq 1 \qquad \begin{array}{l} \forall S_h \in S(C_x), \\ \ell_j, \ell_k \in L(C_x) | (\ell_j, \ell_k) \notin E(B_x) \end{array} \qquad \textit{(Eq. 12)}$$

$$z(R_d, \ell_j) \geq y(S_h, \ell_j) \qquad \forall R_d \in R(C_x), S_h \in R_d, \ell_j \in L(C_x) \qquad \textit{(Eq. 13)}$$

$$z(R_d, \ell_j) \leq \sum_{S_h \in R_d} y(S_h, \ell_j) \qquad \forall R_d \in R(C_x), \ell_j \in L(C_x) \qquad \textit{(Eq. 14)}$$

$$\gamma(t_m, R_d) \leq z(R_d, \ell_j) \qquad \forall R_d \in R(C_x), t_m \in T, \ell_j \in t_m \qquad \textit{(Eq. 15)}$$

$$\gamma(t_m, R_d) \geq \sum_{\ell_j \in t_m} \big[z(R_d, \ell_j) - 1 \big] + 1 \qquad \forall R_d \in R(C_x), t_m \in T \qquad \textit{(Eq. 16)}$$

$$\sum_{t_m \in T(\ell_j, \ell_k)} \gamma(t_m, R_d) \geq 1 - \big[1 - z(R_d, \ell_j) \big] - \big[1 - z(R_d, \ell_k) \big] \qquad \begin{array}{l} \forall R_d \in R(C_x), \\ \ell_j, \ell_k \in \{ L(C_x) | (\ell_j, \ell_k) \notin E(U_x) \} \end{array} \qquad \textit{(Eq. 17)}$$

$$x(p_i, \ell_j), y(S_h, \ell_j), z(R_d, \ell_j), \gamma(t_m, R_d) \in \{0, 1\}.$$

Adding a penalty to the objective function

It appears that historical store layouts not always satisfy the feasibility constraints which verify the connectivity of the locations allocated to a subgroup or group. Thus given the input for a category, it could be that the ILP can not find a feasible solution.
Therefore we want to allow subgroups and groups to violate with the constraints that are forcing the connected subgraph, but minimize these number of violations. To achieve this, we create significantly high penalties M_{sub} and M_{group} that can be subtracted from the total revenue of the solution. The penalties should be chosen significantly high such that it is not attractive to allow a penalty in order to create a revenue increase which is higher then this chosen penalty.
So ideally we want a solution to satisfy the constraint of Equation 12 for each subgroup S_h. If this is not possible, a penalty shall be given for this subgroup S_h and locations $\ell_j, \ell_k \in \{L(C_x)|(\ell_j, \ell_k) \notin E(B_x)\}$, using a binary variable that is defined as follows:

$$PenSub(S_h, \ell_j, \ell_k) = \begin{cases} 1, & \text{If the locations } \ell_j \text{ and } \ell_k \text{, for which } y(S_h, \ell_j) = 1 \text{ and } y(S_h, \ell_k) = 1, \\ & \text{are not connected in } B_x \text{ via locations that are also assigned to subgroup } S_h \\ 0, & \text{Otherwise.} \end{cases}$$

Using this variable we can add a penalty to the objective function of the ILP, which looks as follows:

$$\max \sum_{p_i \in P(C_x), \ell_j \in L(C_x)} w(p_i, \ell_j) x(p_i, \ell_j) - \sum_{S_h \in S(C_x), \ell_j, \ell_k \in \{L(C_x)|(\ell_j, \ell_k) \notin E(B_x)\}} M_{sub} PenSub(S_h, \ell_j, \ell_k).$$

To force $PenSub(S_h, \ell_j, \ell_k)$ to the correct binary value, we will adapt the constraint of Equation 12 as follows:

Connectivity constraint regarding subgroups with a penalty included.

$$y(S_h, \ell_j) + y(S_h, \ell_k) - \sum_{l_r \in P(\ell_j, \ell_k)} y(S_h, l_r) \leq 1 + PenSub(S_h, \ell_j, \ell_k) \quad \begin{array}{l} \forall S_h \in S(C_x), \\ \ell_j, \ell_k \in \{L(C_x)|(\ell_j, \ell_k) \notin E(B_x)\}. \end{array} \tag{18}$$

The binary variable $PenSub(S_h, \ell_j, \ell_k)$ on the right-hand side of the constraint of Equation 18 can be used to force this constraint to be satisfied. Since we want to maximize the objective function, we would like to minimize the number that $PenSub(S_h, \ell_j, \ell_k) = 1$ occurs and ideally there does not occur a penalty regarding subgroups at all.
Therefore, the case that $PenSub(S_h, \ell_j, \ell_k) = 1$ will only occur if there is a subgroup $S_h \in S(C_x)$ for which it is impossible to assign to locations that form a connected subgraph in B_x.

Besides subgroups we want the locations assigned to a group $R_d \in R(C_x)$ to form a connected subgraph in U_x. Comparable to the situation for subgroups, it might occur that there is no solution for which the constraint of Equation 17 is always satisfied. Therefore we will introduce a binary variable like is done for the subgroups in order to allow a solution which violates the constraint of Equation 17 against a significant high penalty M_{group}.
To achieve this, we define the binary variable that indicates whether a group R_d with locations $\ell_j, \ell_k \in \{L(C_x)|(\ell_j, \ell_k) \notin E(U_x)\}$ violates the constraint of Equation 17:

$$PenGroup(R_d, \ell_j, \ell_k) = \begin{cases} 1, & \text{If the locations } \ell_j \text{ and } \ell_k \text{, for which } z(R_d, \ell_j) = 1 \text{ and } z(R_d, \ell_k) = 1, \\ & \text{are not connected in } U_x \text{ via locations that are also assigned to group } R_d \\ 0, & \text{Otherwise.} \end{cases}$$

The constraint of Equation 17 will be adjusted similarly as is done for the constraint of Equation 12 regarding subgroups to determine whether each $PenGroup(R_d, \ell_j, \ell_k)$ is set to one or zero.

Connectivity constraint regarding groups with a penalty included.

$$\sum_{t_m \in T(\ell_j, \ell_k)} \gamma(t_m, R_d) \geq 1 - \big[1 - z(R_d, \ell_j)\big] - \big[1 - z(R_d, \ell_k)\big] - PenGroup(R_d, \ell_j, \ell_k) \quad \begin{array}{l} \forall R_d \in R(C_x), \\ \ell_j, \ell_k \in \{L(C_x)|(\ell_j, \ell_k) \notin E(U_x)\}. \end{array} \tag{19}$$

Similarly to the case regarding subgroups, $PenGroup(R_d, \ell_j, \ell_k)$ is only set to one if it is not possible to assign locations to a group R_d such that these locations form a connected subgraph in U_x. If $PenGroup(R_d, \ell_j, \ell_k) = 1$, a penalty M_{group} will be subtracted from the objective function. We want to minimze the number of penalties regarding subgroups and groups since we want to maximize the objective function.
Including these penalties regarding subgroups and groups, the final version of the ILP looks as follows:

$$\max \Big[\sum_{p_i \in P(C_x), \ell_j \in L(C_x)} w(p_i, \ell_j) x(p_i, \ell_j) - \sum_{S_h \in S(C_x), \ell_j, \ell_k \in \{L(C_x) | (\ell_j, \ell_k) \notin E(B_x)\}} M_{sub} PenSub(S_h, \ell_j, \ell_k)$$
$$- \sum_{R_d \in R(C_x), \ell_j, \ell_k \in \{L(C_x) | (\ell_j, \ell_k) \notin E(U_x)\}} M_{group} PenGroup(R_d, \ell_j, \ell_k) \Big]$$

subject to

$$\sum_{\ell_j \in L(C_x)} x(p_i, \ell_j) = 1 \qquad \forall p_i \in P(C_x) \qquad \textit{(Eq. 8)}$$

$$\sum_{p_i \in P(C_x)} x(p_i, \ell_j) \leq A(\ell_j) \qquad \forall \ell_j \in L(C_x) \qquad \textit{(Eq. 9)}$$

$$y(S_h, \ell_j) \geq x(p_i, \ell_j) \qquad \forall S_h \in S(C_x), p_i \in S_h, \ell_j \in L(C_x) \qquad \textit{(Eq. 10)}$$

$$y(S_h, \ell_j) \leq \sum_{p_i \in S_h} x(p_i, \ell_j) \qquad \forall S_h \in S(C_x), \ell_j \in L(C_x) \qquad \textit{(Eq. 11)}$$

$$y(S_h, \ell_j) + y(S_h, \ell_k) - \sum_{l_r \in P(\ell_j, \ell_k)} y(S_h, l_r) \leq 1 + PenSub(S_h, \ell_j, \ell_k) \qquad \forall S_h \in S(C_x), \ell_j, \ell_k \in \{L(C_x) | (\ell_j, \ell_k) \notin E(B_x)\} \qquad \textit{(Eq. 18)}$$

$$z(R_d, \ell_j) \geq y(S_h, \ell_j) \qquad \forall R_d \in R(C_x), S_h \in R_d, \ell_j \in L(C_x) \qquad \textit{(Eq. 13)}$$

$$z(R_d, \ell_j) \leq \sum_{S_h \in R_d} y(S_h, \ell_j) \qquad \forall R_d \in R(C_x), \ell_j \in L(C_x) \qquad \textit{(Eq. 14)}$$

$$\gamma(t_m, R_d) \leq z(R_d, \ell_j) \qquad \forall R_d \in R(C_x), t_m \in T, \ell_j \in t_m \qquad \textit{(Eq. 15)}$$

$$\gamma(t_m, R_d) \geq \sum_{\ell_j \in t_m} \big[z(R_d, \ell_j) - 1 \big] + 1 \qquad \forall R_d \in R(C_x), t_m \in T \qquad \textit{(Eq. 16)}$$

$$\sum_{t_m \in T(\ell_j, \ell_k)} \gamma(t_m, R_d) \geq 1 - \big[1 - z(R_d, \ell_j)\big] - \big[1 - z(R_d, \ell_k)\big] - PenGroup(R_d, \ell_j, \ell_k) \qquad \forall R_d \in R(C_x), \ell_j, \ell_k \in \{L(C_x) | (\ell_j, \ell_k) \notin E(U_x)\} \qquad \textit{(Eq. 19)}$$

$$x(p_i, \ell_j), y(S_h, \ell_j), z(R_d, \ell_j), \gamma(t_m, R_d), PenSub(S_h, \ell_j, \ell_k), PenGroup(R_d, \ell_j, \ell_k) \in \{0, 1\}.$$

With this ILP, we constructed a model to obtain a revenue maximizing assignment for each category C_x. By combining the assignments of each category, the revenue maximizing assignment of the store is obtained. In Section 5 we will compare the revenues between the revenue maximizing assignment and the historical assignments for each category.

5 Computational results of the categorized assignment problem

This section provides the revenue maximizing solution of each category C_x, obtained by using the ILP which is outlined in Section 4.2.2. These expected weekly revenues will be compared to the historical average weekly revenue of each category in order to determine whether the approach to assign products to locations according to this research might be profitable for the store. Furthermore the feasibility of the revenue maximizing assignment of one category will be explained in detail to obtain a deeper understanding of the feasibility constraints of the ILP.

The ILP described in Section 4.2.2 was implemented in the program AIMMS. For computing the revenue maximizing assignments we use a windows machine with an intel Core i5-7200U @2.5 GHz CPU and a CPLEX 12.8.0 solver within AIMMS.

To obtain some insight into the size of the input for each category, we will provide an overview of the number of locations, products, subgroups, groups and paths. This is illustrated in Table 5:

	Number of locations: $\|L(C_x)\|$	Number of products: $\|P(C_x)\|$	Number of subgroups: $\|S(C_x)\|$	Number of groups: $\|R(C_x)\|$	Number of paths in U_x: $\|T(C_x)\|$
Category C_1	18	359	58	14	349
Category C_2	19	597	72	10	956
Category C_3	8	192	10	4	157
Category C_4	11	126	11	5	300
Category C_5	7	183	23	6	18
Category C_6	8	192	34	8	11

Table 5: Overview of number of locations, products, subgroups, groups and paths per category.

Besides these sets, the input for the ILP further contains the connectivity graphs B_x and U_x, the location capacities $A(\ell_j)$ for all $\ell_j \in L(C_x)$ and the expected weekly revenues $w(p_i, \ell_j)$ for all $p_i \in P(C_x), \ell_j \in L(C_x)$. The graphs can be directly obtained from the physical positions of the locations $L(C_x)$, which is described in Section 3.2.2. This section also clarifies how the location capacities are obtained. The expected weekly revenues $w(p_i, \ell_j)$ equal $Y(p_i, \ell_j)$ $\forall p_i \in P(C_x), \ell_j \in L(C_x)$, as already mentioned at the beginning of Section 4.2.2.
As last we set the value for the penalties $M_{sub} = M_{group} = \sum_{p_i \in P(C_x)} \sum_{\ell_j \in L(C_x)} \max(w(p_i, \ell_j))$. In this way, it is never attractive to allow a penalty in order to create a revenue increase.

Given these sets as input for each category, the corresponding revenue maximizing assignment can be computed. These revenues are depicted in Table 6 together with the violations if they are applicable. Furthermore the historical average weekly revenue for each category is depicted in order to compare with the expected weekly revenue of the revenue maximizing assignment of the corresponding category. Note that the expected weekly revenue is the revenue without the subtraction of the penalties, so it only consists of $\sum_{p_i \in P(C_x), \ell_j \in L(C_x)} w(p_i, \ell_j) x(p_i, \ell_j)$ for each category C_x.

	Expected weekly revenue of the revenue maximizing assignment	Number of violations regarding subgroups	Number of violations regarding groups	Historical average weekly revenue
Category C_1	€1290,31	0	3 violations regarding group R_6	€845,58
Category C_2	€1681,17	0	0	€615,20
Category C_3	€553,56	1 violation regarding subgroup S_2	0	€398,20
Category C_4	€587,03	0	0	€436,78
Category C_5	€175,79	0	0	€133,94
Category C_6	€1207,64	0	1 violation regarding group R_3	€884,26

Table 6: Weekly revenue for all categories.

In Table 6 we can see that the expected weekly revenue of the revenue maximizing assignment of every category is higher than the historical average weekly revenue of each corresponding category. In Section 5.2

we will visualize this comparison and provide the expected increase in revenue for each category.
Furthermore Table 6 shows the violations that have occurred in the revenue maximizing solution of each category. It appears that for categories C_1, C_3 and C_6 there does not exist a feasible solution to the CA-problem where the locations allocated to each subgroup or each group form a connected subgraph in B_x or U_x respectively. Therefore the revenue maximizing assignments of these categories include one or several violations.

To obtain a better understanding of the violation occurrences within a revenue maximizing assignment and the feasibility of such an assignment in general, Section 5.1 visualizes the solution for category C_3 in detail.

5.1 The revenue maximizing solution of Category C_3

This subsection provides the solution of category C_3 in detail to acquire a better understanding of the feasibility of a solution together with the penalties that may occur.
Category C_3 is taken as example since this category has a relative small number of locations that can be easily illustrated and the solution has one violation regarding subgroups to illustrate.

The input for category C_3 are the sets $L(C_3), P(C_3), S(C_3), R(C_3), T(C_3)$ for which the size is shown in Table 5. The connectivity graphs B_3 and U_3 are as follows:

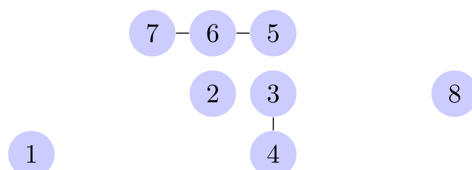

Figure 16: The graph $B_3 = (L(C_3), \widetilde{E})$.

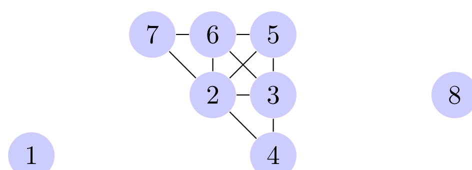

Figure 17: The graph $U_3 = (L(C_3), E)$.

The location capacities $A(\ell_j)$ for all $\ell_j \in L(C_3)$ are as follows:

$$\begin{bmatrix} A(l_1) \\ A(l_2) \\ A(l_3) \\ A(l_4) \\ A(l_5) \\ A(l_6) \\ A(l_7) \\ A(l_8) \end{bmatrix} = \begin{bmatrix} 1 \\ 35 \\ 22 \\ 19 \\ 51 \\ 33 \\ 34 \\ 12 \end{bmatrix}$$

The expected weekly revenues $w(p_i, \ell_j)$ are included in Appendix B.
With these inputs, the ILP computed the revenue maximizing solution from which the revenue is depicted in Table 6. The assignment of subgroups and groups to locations is included in Appendix B, illustrated as

the ouput of AIMMS. To visualize the assignment of subgroups to locations and to illustrate the violation regarding subgroup $S_2 \in S(C_3)$, we will present the locations allocated to every subgroup in Figure 18.

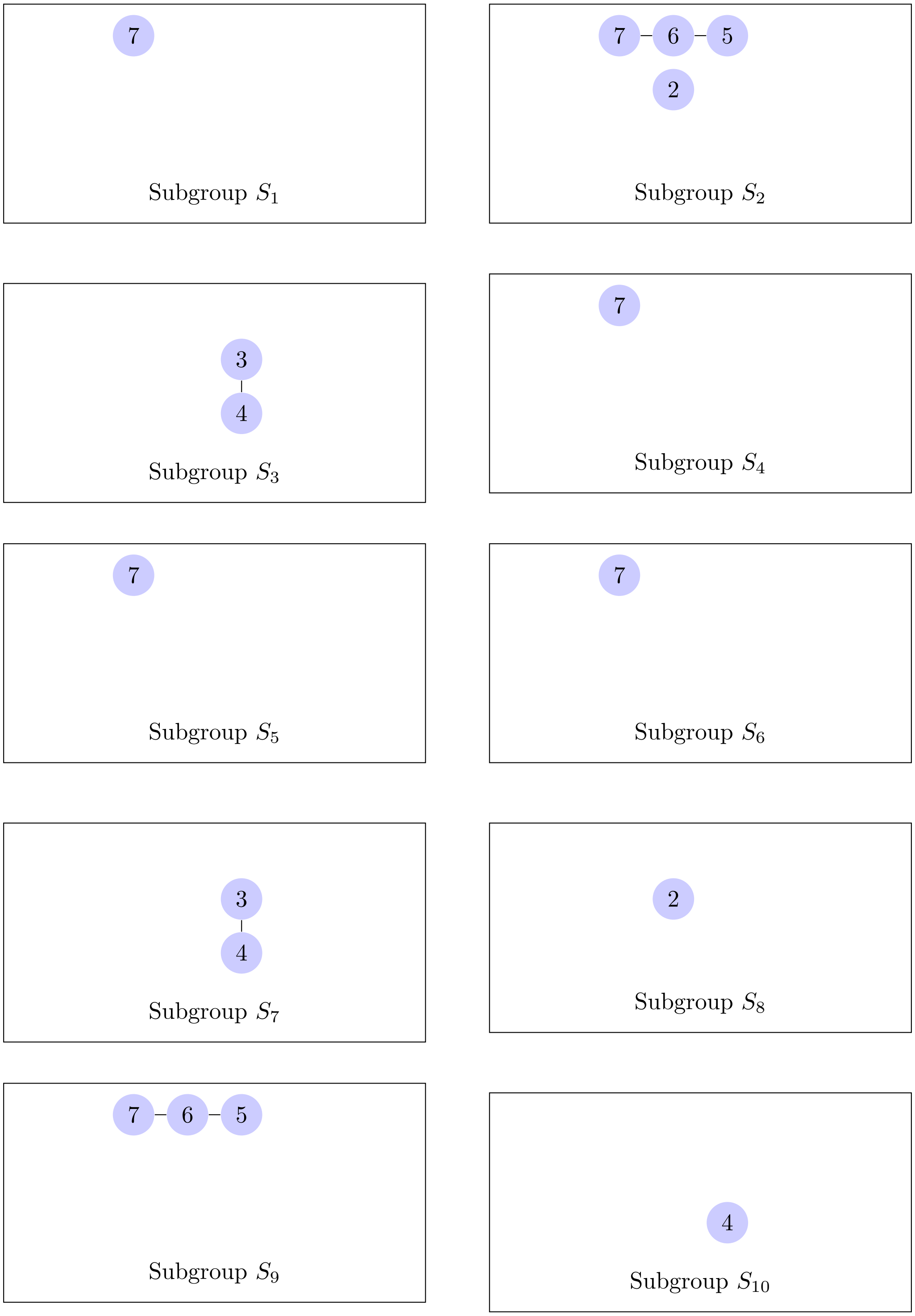

Figure 18: The locations allocated to every subgroup of Category C_3, illustrated as subgraph of B_3.

Every subfigure within Figure 18 represents the locations a subgroup is assigned to. It can be observed that only the locations assigned to subgroup S_2 do not form a connected subgraph in B_3 and induce the violation within the revenue maximizing assignment of category C_3. This violation occurs because it appears that there is no feasible solution for category C_3 for which the locations allocated to every subgroup and group, form a connected subgraph in graph B_3 and graph U_3 respectively. Therefore the ILP determined that it would minimize the number of violations regarding subgroups and groups if subgroup S_2 is allocated to location ℓ_2. This is the only location that is isolated from the other allocated locations $\{\ell_5, \ell_6, \ell_7\}$ of subgroup S_2, which are connected in B_3.

We can also illustrate the locations allocated to every group and confirm that they all form a connected subgraph in U_3. This is done in Figure 19.

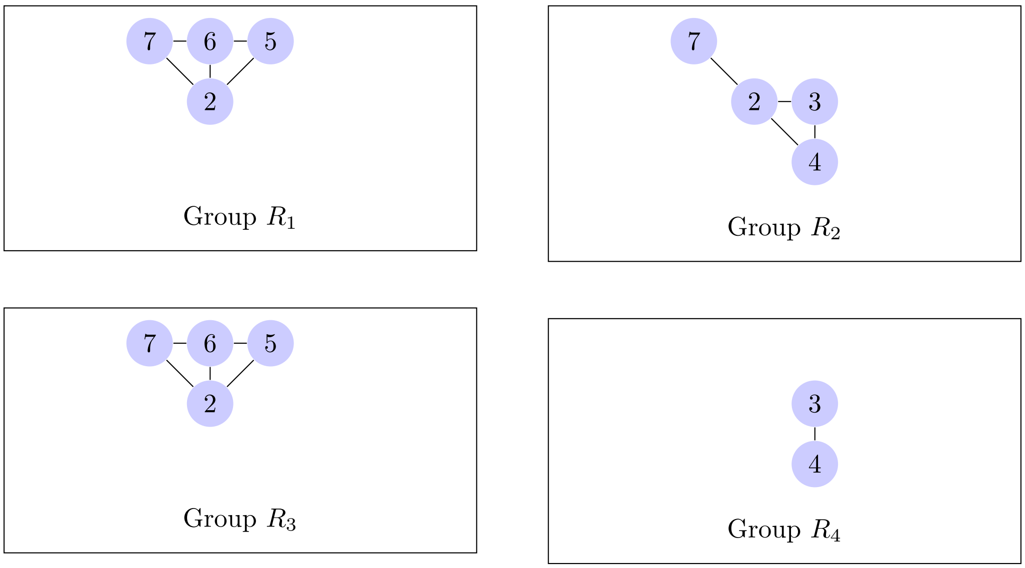

Figure 19: The locations allocated to every group of category C_3, illustrated as subgraph of U_3.

As can be seen, the locations assigned to every group truly form a connected subgraph in U_3.
Thus, Figure 18 and 19 illustrate the feasibility of the revenue maximizing assignment regarding the connectivity of the locations assigned to each subgroup or group.
Furthermore the number of products assigned to each location does not exceed its capacity. This is visualized in Table 7.

	Number of assigned products	Location capacity
ℓ_1	0	1
ℓ_2	34	35
ℓ_3	22	22
ℓ_4	18	19
ℓ_5	51	51
ℓ_6	33	33
ℓ_7	34	34
ℓ_8	0	12

Table 7: Number of products assigned to each location together with its capacity.

Observe that location ℓ_1 and ℓ_8 are not allocated to any product, which is also illustrated in Figure 18 and Figure 19. As can be seen in graph B_3 and U_3 of Figure 16 and Figure 17 respectively, location ℓ_1 and ℓ_8 are isolated. Location ℓ_1 only has a capacity of one so it is not attractive to allocate a product to this location since the other products of its group will be allocated to locations that are not connected to location ℓ_1 in graph B_3 and U_3. This would cause many violations within the assignment of category C_3.

On the other hand, location ℓ_8 has a capacity of twelve and we can observe that the expected revenues on location ℓ_8, depicted in Table 9, are attractive to choose. However, since location ℓ_8 is isolated from the other locations in graph B_3 and U_3, it would induce many violations to allocate products to location ℓ_8. For that reason there are no products assigned to location ℓ_1 and ℓ_8 in the revenue maximizing assignment of category C_3, where only one violation is included.

5.2 Comparison between historical average weekly revenue and expected weekly revenue of the optimal solution

To obtain a better representation of the difference between the expected weekly revenue of the revenue maximizing solution created by the ILP and the historical average weekly revenue of each category, consider the visualization of Table 6 in Figure 20.

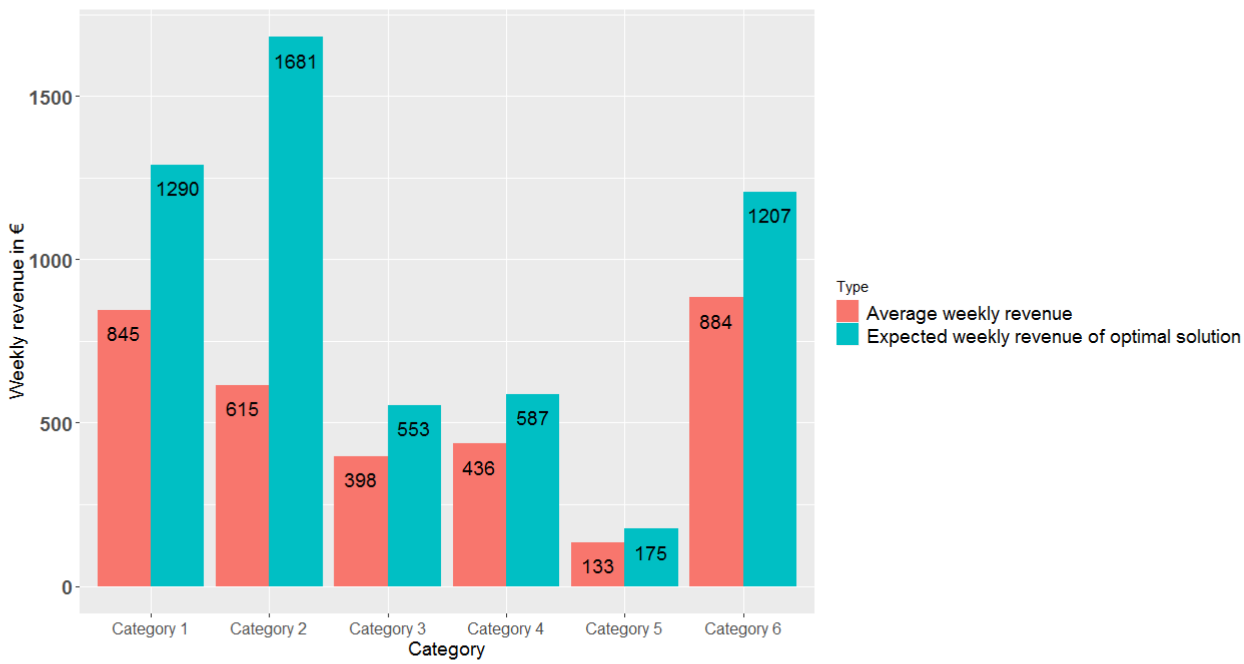

Figure 20: Comparison between the historical average weekly revenue and the expected weekly revenue of the optimal solution for each category.

For every category there is an increase in revenue between the average weekly revenue and the expected weekly revenue of the revenue maximizing solution. These expected increases are given in the following table:

	Expected increase in revenue in %
Category C_1	52 %
Category C_2	173 %
Category C_3	39 %
Category C_4	34 %
Category C_5	31 %
Category C_6	37 %

Table 8: Expected increase in revenue per category.

We can observe that category C_2 has the highest expected increase in revenue and category C_5 the lowest expected increase in revenue. Section 6 provides possible reasons for this.

Finally, we are interested in the total revenue of the whole store. Therefore we illustrate the revenues of Figure 20 together so we can see the difference of the average historical revenue and the expected revenue of the revenue maximizing assignment for the entire store. This is illustrated in Figure 21.

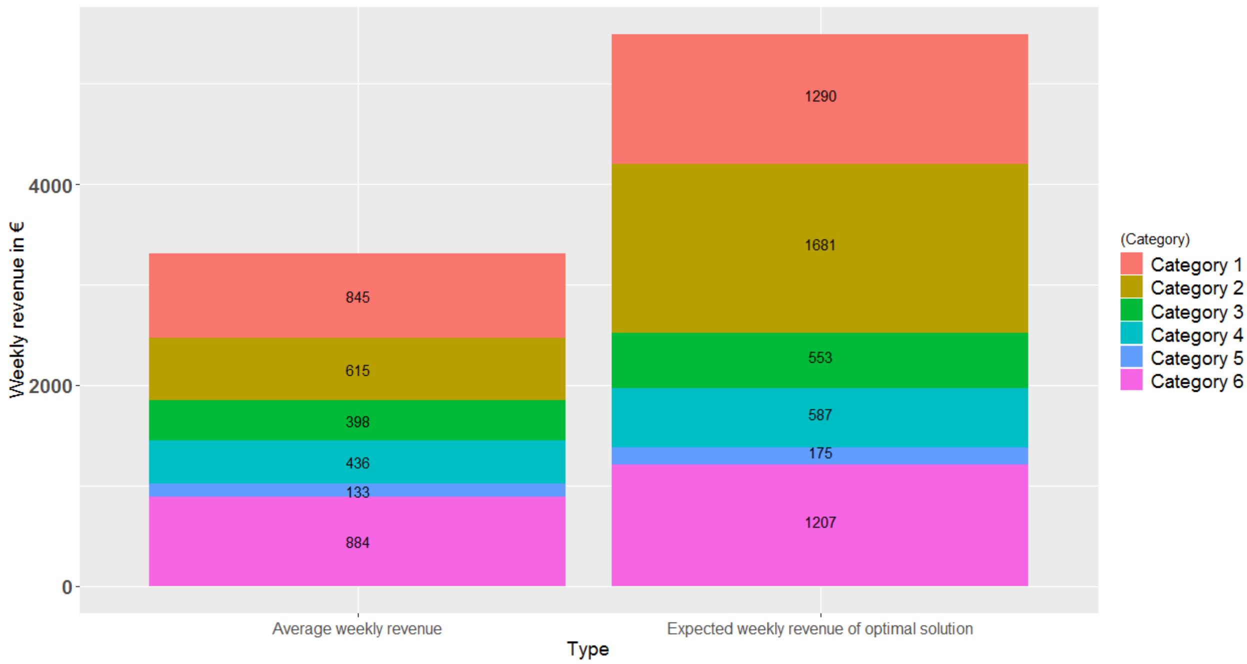

Figure 21: Comparison between the historical average weekly revenue and the expected weekly revenue of the optimal solution over all categories.

It can be observed that the revenue of the store is expected to increase with 66% if the store will assign the products of each category according to the revenue maximizing solution of the corresponding category. Section 6 explains which conclusions can be drawn from these computational results.

6 Conclusion

This section will draw conclusions from the computational results presented in Section 5.

The main conclusion can be drawn from Figure 21. Using the revenue maximizing assignment, the revenue of the store is expected to increase with 66% compared to the average weekly revenue over all weeks W.
A part of this percentage is induced by the uncertainty that is included in the estimated expected weekly revenues $w(p_i, \ell_j)$. But since these expected weekly revenues are all based on the same historical data from which the average weekly revenues for each category are formed, a big part of the 66% is caused by assigning the products of each category to the locations of the corresponding category according to the revenue maximizing assignment.

Furthermore Figure 20 illustrates that Category C_2 induces the highest contribution to the expected weekly revenue increase of the store. Category C_2 also has the highest absolute and relative increase in revenue in comparison to the other categories.
A reason for this might be that Category C_2 has more products to assign in comparison to the other categories, which can be seen in Table 5. Therefore Category C_2 also has more products to assign to the location which contributes to the maximizing revenue of the store and may the absolute revenue be higher as for other categories. Furthermore the graph U_2 has more edges in comparison to other categories, so the locations $L(C_2)$ are positioned in such a way that many of these locations are in close proximity to each other within the store. This can also be seen in the number of paths of T_2, which is significant more as for the other categories. Therefore products of category C_2 are probably more often assigned to the location with the highest expected weekly revenue while the assignment still remains feasible.
The categories C_3 up to C_6 contributes the least to the expected increase in revenue of the store. It can be observed that these categories contain less products to assign and their corresponding locations induce less paths in U_3 up to U_6 respectively.

Even though we realize that there is uncertainty in the expected weekly revenue of each category, regarding the expected weekly revenue increase of 66%, it can be concluded that it is interesting for this store to experiment with the store layout according to the revenue maximizing assignments of each category.

7 Discussion and further research

There are several topics for which discussion and further research is applicable. We will first discuss the topics regarding the estimation of the expected weekly revenues after which we will discuss the topics regarding the revenue maximizing assignment of the CA-problem.

In the previous section we showed that the revenue of the store is expected to increase with 66% if the assignment of product P to locations L is executed according to the revenue maximizing assignment of each category. As stated, it is disputable which part of the 66% is caused by the optimization and which part is induced by the uncertainty of the expected weekly revenues. In the scope of this research, the influence of the uncertainty is not investigated. Thus, we know that there is a large potential to increase the revenue of the store, but the actual increase might be lower than 66 % due to the uncertainty included in the expected weekly revenues.

The uncertainty included in the estimated expected weekly revenues is caused by the quality and features of the available sales data.
Most of the weeks, products were not sold and when they did get sold, the quantity varied a lot. Therefore it is hard to fit a distribution through the available sales data and we decided to estimate the expected weekly revenue by taking the average.
However, if we would have had more available sales data that consisted of several years and where products were sold more often on different locations within the store, the sales data might satisfy the assumptions for a certain distribution such as a Poisson distribution. In that case it might be more reasonable to use a Poisson regression to predict the expected weekly revenues in stead of the simple approach of taking the average.

Furthermore, the only features within the Poisson regression we were able to use were the locations and products. Therefore the influence of a location to the expected weekly sales of the product is mainly based on the weekly sales of the products that were sold on this location. It would be interesting to do more research dedicated to the influence of the location itself in stead of using the products on this location. This could be done by combining the sales data of several stores for which the physical locations within these stores are comparable to each other. In that case there would be more data of locations within a store.

Besides that, the weekly sales of a product on a location might be influenced by other factors like weather, discount or holiday periods. Therefore it would be interesting to include features like this within the Poisson regression and investigate what influence they have on the expected weekly sales of products on locations.
If we would have sales data of several years and more data of external factors, we could even model the expected revenue of a product on a location in a particular week. With that, a revenue maximizing assignment of the store can be computed per week.

For predicting the expected weekly revenue of products on its non-historical locations, we used the approach of taking a weighted average of the expected weekly sales on historical locations that are in close proximity to this non-historical location. This is illustrated in Equation 6. The weights $d_{k,j}^{(i)}$ used in the weighted average are based on the expertise of a data expert in the field of retail. It would be interesting to determine these weights by a more quantitative approach, such as computing the correlations between locations.

As final remark about this topic, be aware of the fact that predicting the expected weekly revenues of products on locations within a store could be a research on its own. The main focus of this research is on the construction of the ILP by which the revenue maximizing assignment of each category is computed. Therefore it was sufficient to predict the expected weekly sales of products on locations by intuitive methods, such that we have the input for the categorized assignment problem.

Regarding the categorized assignment problem, there are a few topics to discuss for which further research is applicable.

The edges in the connectivity graphs B_x and U_x are defined by the scenario's described in Section 3.2.2. These scenario's illustrate when locations are in close proximity to each other and are determined together with a data expert in the field of retail. However, it might be interesting to investigate other ways to define close proximity and thereby the edges of connectivity graphs B_x and U_x. Furthermore it might be interesting to assign weights to the edges in graphs B_x and U_x that can represent distance or the correlations which are discussed earlier in this section.

Within the revenue maximizing assignment of a category, the violations regarding the constraint that products of a subgroup or group are positioned in close proximity to each other, are minimized. This has its consequences regarding the expected revenue of the assignment.
When the uncertainty included in the estimated expected weekly revenues is reduced significantly, it might be interesting to investigate the influence of the constraints regarding this close proximity on the expected revenue of the revenue maximizing solution of each category. This can be done by comparing the revenue maximizing assignment with and without the constraints regarding the close proximity of allocated locations.

This research focuses on which product to assign to which physical location within the store. Here we neglect the quantity of each product that will be assigned to this location, although it might be interesting to add this to the problem.
Furthermore it would be interesting to determine on which height within a location a product contributes the most to the expected revenue of the store. However, this information should then be inserted in the historical sales data, which is missing now.

Finally, the running-time of computing the revenue maximizing assignment via the ILP took several hours for category C_1 and C_2. If a category would consist of significant more locations and paths, it could be that computing the revenue maximizing assignment takes too long. However, the dual version of checking connectivity by paths, is checking connectivity by cuts. If the number of locations and paths increase significant, it might be more running-time efficient to have an ILP formulation that uses cuts in stead of paths to ensure that the locations allocated to each subgroup or group form a connected subgraph in graph B_x or graph U_x respectively. Therefore it would be interesting to set up such an ILP, implement it in AIMMS and investigate whether it reduces the running-time for instances with significant more locations and edges within U_x comparing to the ILP which uses paths to check connectivity.

References

[1] Andrew Lim, Brian Rodrigues, and Xingwen Zhang. Metaheuristics with local search techniques for retail shelf-space optimization. *Management science*, 50(1):117–131, 2004.

[2] Marcel Corstjens and Peter Doyle. A model for optimizing retail space allocations. *Management Science*, 27(7):822–833, 1981.

[3] Marcel Corstjens and Peter Doyle. A dynamic model for strategically allocating retail space. *Journal of the Operational Research Society*, 34(10):943–951, 1983.

[4] Ming-Hsien Yang. An efficient algorithm to allocate shelf space. *European journal of operational research*, 131(1):107–118, 2001.

[5] Ming-Hsien Yang and Wen-Cher Chen. A study on shelf space allocation and management. *International journal of production economics*, 60:309–317, 1999.

[6] Ahmed Ali Ahmed Al Al-Kuwari et al. Solving the generalized assignment problem with location/allocation considerations using heuristic decomposition approaches. Master's thesis, 2015.

[7] Darina Graczová and Peter Jacko. Generalized restless bandits and the knapsack problem for perishable inventories. *Operations Research*, 62(3):696–711, 2014.

[8] Chris Chatfield, Jim Zidek, and Jim Lindsey. *An introduction to generalized linear models*. Chapman and Hall/CRC, 2010.

[9] Sheldon M Ross. *Introduction to probability models*. Academic press, 2014.

[10] Michael L Stein. *Interpolation of spatial data: some theory for kriging*. Springer Science & Business Media, 2012.

[11] Dorit S Hochbaum and Anu Pathria. Node-optimal connected k-subgraphs. *manuscript, UC Berkeley*, 1994.

[12] Bernhard Korte and Jens Vygen. Combinatorial optimization.

[13] Greg Barnes and Walter L Ruzzo. Deterministic algorithms for undirected s-t connectivity using polynomial time and sublinear space. In *Proceedings of the twenty-third annual ACM symposium on Theory of computing*, pages 43–53. ACM, 1991.

[14] Ricardo Simões. Apac: An exact algorithm for retrieving cycles and paths in all kinds of graphs. *Tékhne-Revista de Estudos Politécnicos*, (12):39–55, 2009.

Appendices

A Proof that Poisson regression with interaction is the same as taking the average

Let W be the time set, P the set of products, L the set of locations.
The log-likelihood of the vectors of parameters $\alpha,\ \beta,\ \mu$ is given by:

$$\ell(\alpha, \beta, \mu) = \sum_{w_t \in W,\ p_i \in P,\ l_j \in L} (Q_t(p_i, l_j)(\alpha_i + \beta_j + \mu_{i,j}) - e^{\alpha_i + \beta_j + \mu_{i,j}}).$$

Hence:

$$\frac{\partial}{\partial \mu_{i,j}} \ell(\alpha, \beta, \mu) = \sum_{w_t \in W} (Q_t(p_i, l_j) - e^{\alpha_i + \beta_j + \mu_{i,j}}) := 0$$
$$\iff e^{\alpha_i + \beta_j + \mu_{i,j}} = \frac{1}{|W|} \sum_{w_t \in W} Q_t(p_i, l_j).$$

Now:

$$\frac{\partial}{\partial \beta_j} \ell(\alpha, \beta, \mu) = \sum_{w_t \in W,\ p_i \in P} (Q_t(p_i, l_j) - e^{\alpha_i + \beta_j + \mu_{i,j}}) = \sum_{p_i \in P} (\sum_{w_t \in W} (Q_t(p_i, l_j)) - |W| e^{\alpha_i + \beta_j + \mu_{i,j}}) := 0$$
$$\Longleftarrow e^{\alpha_i + \beta_j + \mu_{i,j}} = \frac{1}{|W|} \sum_{w_t \in W} Q_t(p_i, l_j).$$

The same holds for α_i.

B Input and output for the revenue maximizing assignment of category C_3

This appendix consists of the expected weekly revenues that are used as input for category C_3, illustrated in Table 9. Furthermore, the locations that are allocated to each subgroup and group of Category C_3 are illustrated in Figure 22 and Figure 23 respectively.
In Table 9, the first row corresponds with the first product of $P(C_3)$, the second row with the second product of $P(C_3)$, and so on.

Table 9: The expected weekly revenues $w(p_i, \ell_j)$ in euro of products $p_i \in P(C_3)$ on locations $\ell_j \in L(C_3)$.

Location 1	Location 2	Location 3	Location 4	Location 5	Location 6	Location 7	Location 8
0,28	0,28	0,28	0,28	0,28	0,28	0,28	0,28
0,64	0,64	0,64	0,64	0,64	0,64	0,64	0,64
3,58	3,58	3,63	0,58	3,58	0,58	3,58	3,58
4,79	4,79	4,86	0,77	4,79	0,77	4,79	4,79
1,96	1,96	1,99	0,32	1,96	0,32	1,96	1,96
3,24	3,24	3,29	0,52	3,24	0,52	3,24	3,24
3,87	3,87	3,93	0,62	3,87	0,62	3,87	3,87
3,34	3,34	3,39	0,54	3,34	0,54	3,34	3,34
3,09	3,09	3,13	0,50	3,09	0,50	3,09	3,09
3,16	3,16	3,21	0,51	3,16	0,51	3,16	3,16
3,23	3,23	3,28	0,52	3,23	0,52	3,23	3,23

Table 9 – continued from previous page

Location 1	Location 2	Location 3	Location 4	Location 5	Location 6	Location 7	Location 8
3,01	3,01	3,06	0,49	3,01	0,49	3,01	3,01
2,64	2,64	2,68	0,43	2,64	0,43	2,64	2,64
2,79	2,79	2,83	0,45	2,79	0,45	2,79	2,79
3,83	3,83	3,89	0,62	3,83	0,62	3,83	3,83
2,16	2,16	2,19	0,35	2,16	0,35	2,16	2,16
5,05	5,05	5,12	0,81	5,05	0,81	5,05	5,05
4,51	4,51	4,57	0,73	4,51	0,73	4,51	4,51
3,75	3,75	3,80	0,60	3,75	0,60	3,75	3,75
3,51	3,51	3,56	0,57	3,51	0,57	3,51	3,51
1,48	1,48	1,51	0,24	1,48	0,24	1,48	1,48
5,28	5,28	5,36	0,85	5,28	0,85	5,28	5,28
1,66	1,66	1,69	0,27	1,66	0,27	1,66	1,66
1,25	1,25	1,27	0,20	1,25	0,20	1,25	1,25
6,29	6,29	6,38	1,01	6,29	1,01	6,29	6,29
2,45	2,45	2,48	0,39	2,45	0,39	2,45	2,45
3,54	3,54	3,59	0,57	3,54	0,57	3,54	3,54
2,74	2,74	2,78	0,44	2,74	0,44	2,74	2,74
3,61	3,61	3,66	0,58	3,61	0,58	3,61	3,61
2,65	2,65	2,68	0,43	2,65	0,43	2,65	2,65
8,14	8,14	8,26	1,31	8,14	1,31	8,14	8,14
1,55	1,55	1,57	0,25	1,55	0,25	1,55	1,55
0,80	0,80	0,82	0,13	0,80	0,13	0,80	0,80
1,17	1,17	1,18	0,19	1,17	0,19	1,17	1,17
3,27	3,27	3,31	0,53	3,27	0,53	3,27	3,27
10,72	10,72	10,87	1,73	10,72	1,73	10,72	10,72
2,11	2,11	2,14	0,34	2,11	0,34	2,11	2,11
7,99	7,99	8,11	1,29	7,99	1,29	7,99	7,99
5,11	5,11	5,19	0,82	5,11	0,82	5,11	5,11
7,47	7,47	7,58	1,21	7,47	1,21	7,47	7,47
7,01	7,01	7,12	1,13	7,01	1,13	7,01	7,01
2,48	2,48	2,52	0,40	2,48	0,40	2,48	2,48
3,97	3,97	4,03	0,64	3,97	0,64	3,97	3,97
5,53	5,53	5,61	0,89	5,53	0,89	5,53	5,53
4,88	4,88	4,95	0,79	4,88	0,79	4,88	4,88
4,41	4,41	4,48	0,71	4,41	0,71	4,41	4,41
3,57	3,57	3,62	0,58	3,57	0,58	3,57	3,57
3,50	3,50	3,55	0,56	3,50	0,56	3,50	3,50
4,69	4,69	4,76	0,76	4,69	0,76	4,69	4,69
3,28	3,28	3,33	0,53	3,28	0,53	3,28	3,28
3,48	3,48	3,54	0,56	3,48	0,56	3,48	3,48
6,97	6,97	7,07	1,12	6,97	1,12	6,97	6,97
4,52	4,52	4,59	0,73	4,52	0,73	4,52	4,52
2,44	2,44	2,48	0,39	2,44	0,39	2,44	2,44
4,84	4,84	4,91	0,78	4,84	0,78	4,84	4,84
5,83	5,83	5,92	0,94	5,83	0,94	5,83	5,83
5,78	5,78	5,87	0,93	5,78	0,93	5,78	5,78
4,21	4,21	4,27	0,68	4,21	0,68	4,21	4,21
2,98	2,98	3,03	0,48	2,98	0,48	2,98	2,98
4,80	4,80	4,87	0,77	4,80	0,77	4,80	4,80
1,15	1,15	1,15	1,15	1,15	1,15	1,14	1,36
5,73	5,73	5,73	5,73	5,73	5,73	5,70	6,78
5,39	5,39	5,39	5,39	5,39	5,39	5,36	6,37

Table 9 – continued from previous page

Location 1	Location 2	Location 3	Location 4	Location 5	Location 6	Location 7	Location 8
6,68	6,68	6,68	6,68	6,68	6,68	6,64	7,90
2,86	2,86	2,86	2,86	2,86	2,86	2,85	3,39
7,66	7,66	7,66	7,66	7,66	7,66	7,61	9,06
4,27	4,27	4,27	4,27	4,27	4,27	4,24	5,05
3,44	3,44	3,44	3,44	3,44	3,44	3,42	4,07
3,91	3,91	3,91	3,91	3,91	3,91	3,89	4,63
3,42	3,42	3,42	3,42	3,42	3,42	3,40	4,05
8,13	8,13	8,13	8,13	8,13	8,13	8,08	9,62
8,12	8,12	8,12	8,12	8,12	8,12	8,07	9,60
5,91	5,91	5,91	5,91	5,91	5,91	5,87	6,98
5,76	5,76	5,76	5,76	5,76	5,76	5,73	6,81
3,84	3,84	3,84	3,84	3,84	3,84	3,82	4,54
3,64	3,64	3,64	3,64	3,64	3,64	3,62	4,30
4,30	4,30	4,30	4,30	4,30	4,30	4,27	5,08
9,32	9,32	9,32	9,32	9,32	9,32	9,26	11,02
3,41	3,41	3,41	3,41	3,41	3,41	3,39	4,03
3,16	3,16	3,16	3,16	3,16	3,16	3,14	3,73
5,59	5,59	5,59	5,59	5,59	5,59	5,55	6,61
12,39	12,39	12,39	12,39	12,39	12,39	12,32	14,66
5,61	5,61	5,61	5,61	5,61	5,61	5,58	6,64
5,75	5,75	5,75	5,75	5,75	5,75	5,71	6,80
3,86	3,86	3,86	3,86	3,86	3,86	3,83	4,56
7,96	7,96	7,96	7,96	7,96	7,96	7,91	9,41
9,26	9,26	9,26	9,26	9,26	9,26	9,20	10,95
3,91	3,91	3,91	3,91	3,91	3,91	3,89	4,63
4,96	4,96	4,96	4,96	4,96	4,96	4,93	5,87
3,95	3,95	3,95	3,95	3,95	3,95	3,93	4,67
0,55	0,55	0,55	0,55	0,55	0,55	0,55	0,55
0,55	0,55	0,55	0,55	0,55	0,55	0,55	0,55
0,55	0,55	0,55	0,55	0,55	0,55	0,55	0,55
0,55	0,55	0,55	0,55	0,55	0,55	0,55	0,55
0,69	0,69	0,69	0,69	0,69	0,69	0,69	0,69
0,69	0,69	0,69	0,69	0,69	0,69	0,69	0,69
0,35	0,35	0,35	0,35	0,35	0,35	0,35	0,35
0,55	0,55	0,55	0,55	0,55	0,55	0,55	0,55
0,55	0,55	0,55	0,55	0,55	0,55	0,55	0,55
0,26	0,26	0,26	0,26	0,26	0,26	0,26	0,26
0,69	0,69	0,69	0,69	0,69	0,69	0,69	0,69
0,18	0,18	0,18	0,18	0,18	0,18	0,18	0,18
1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00
1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00
0,55	0,55	0,55	0,55	0,55	0,55	0,55	0,55
0,55	0,55	0,55	0,55	0,55	0,55	0,55	0,55
0,57	0,57	0,57	0,57	0,57	0,57	0,57	0,57
0,57	0,57	0,57	0,57	0,57	0,57	0,57	0,57
0,56	0,56	0,56	0,56	0,56	0,56	0,56	0,56
0,22	0,22	0,22	0,22	0,22	0,22	0,22	0,22
0,45	0,45	0,45	0,45	0,45	0,45	0,45	0,45
0,24	0,24	0,24	0,24	0,24	0,24	0,24	0,24
0,02	0,02	0,02	0,02	0,02	0,02	0,02	0,02
0,22	0,22	0,22	0,22	0,22	0,22	0,22	0,22
0,48	0,48	0,48	0,48	0,48	0,48	0,48	0,48

Table 9 – continued from previous page

Location 1	Location 2	Location 3	Location 4	Location 5	Location 6	Location 7	Location 8
0,28	0,28	0,28	0,28	0,28	0,28	0,28	0,28
0,62	0,62	0,62	0,62	0,62	0,62	0,62	0,62
0,73	0,73	0,73	0,73	0,73	0,73	0,73	0,73
0,76	0,76	0,76	0,76	0,76	0,76	0,76	0,76
0,04	0,04	0,04	0,04	0,04	0,04	0,04	0,04
0,22	0,22	0,22	0,22	0,22	0,22	0,22	0,22
0,40	0,40	0,40	0,40	0,40	0,40	0,40	0,40
5,45	5,45	5,45	5,45	5,45	5,45	3,33	6,06
7,90	7,90	7,90	7,90	7,90	7,90	4,82	8,78
19,33	19,33	19,33	19,33	19,33	19,33	11,80	21,49
15,46	15,46	15,46	15,46	15,46	15,46	9,44	17,19
12,38	12,38	12,38	12,38	12,38	12,38	7,56	13,76
4,04	4,04	4,04	4,04	4,04	4,04	2,47	4,49
4,15	4,15	4,15	4,15	4,15	4,15	2,53	4,61
7,40	7,40	7,40	7,40	7,40	7,40	4,52	8,23
4,93	4,93	4,93	4,93	4,93	4,93	3,01	5,48
5,40	5,40	5,40	5,40	5,40	5,40	3,29	6,00
7,19	7,19	7,19	7,19	7,19	7,19	4,39	8,00
4,36	4,36	4,36	4,36	4,36	4,36	2,66	4,85
37,38	37,38	37,38	37,38	37,38	37,38	22,83	41,56
43,45	43,45	43,45	43,45	43,45	43,45	26,53	48,30
5,37	5,37	5,37	5,37	5,37	5,37	3,28	5,97
7,80	7,80	7,80	7,80	7,80	7,80	4,76	8,67
27,65	27,65	27,65	27,65	27,65	27,65	16,88	30,73
7,92	7,92	7,92	7,92	7,92	7,92	4,84	8,81
15,99	15,99	15,99	15,99	15,99	15,99	9,76	17,77
7,32	7,32	7,32	7,32	7,32	7,32	4,47	8,14
7,32	7,32	7,32	7,32	7,32	7,32	4,47	8,14
15,99	15,99	15,99	15,99	15,99	15,99	9,76	17,77
5,98	5,98	5,98	5,98	5,98	5,98	3,65	6,64
2,10	2,10	2,10	2,10	2,10	2,10	1,28	2,33
6,35	6,35	6,35	6,35	6,35	6,35	3,88	7,06
7,76	7,76	7,76	7,76	7,76	7,76	4,74	8,63
8,10	8,10	8,10	8,10	8,10	8,10	4,94	9,00
8,05	8,05	8,05	8,05	8,05	8,05	4,91	8,94
9,23	9,23	9,23	9,23	9,23	9,23	5,63	10,26
4,97	4,97	4,97	4,97	4,97	4,97	3,03	5,52
11,85	11,85	11,85	11,85	11,85	11,85	7,24	13,17
28,26	11,04	37,29	11,04	37,29	28,26	28,26	28,26
24,08	9,40	31,76	9,40	31,76	24,08	24,08	24,08
27,50	10,74	36,28	10,74	36,28	27,50	27,50	27,50
27,23	10,63	35,92	10,63	35,92	27,23	27,23	27,23
22,71	8,87	29,97	8,87	29,97	22,71	22,71	22,71
46,33	18,09	61,12	18,09	61,12	46,33	46,33	46,33
47,12	18,40	62,17	18,40	62,17	47,12	47,12	47,12
5,72	2,23	7,54	2,23	7,54	5,72	5,72	5,72
7,63	2,98	10,06	2,98	10,06	7,63	7,63	7,63
5,62	2,19	7,41	2,19	7,41	5,62	5,62	5,62
8,16	3,18	10,76	3,18	10,76	8,16	8,16	8,16
5,75	2,25	7,58	2,25	7,58	5,75	5,75	5,75
7,81	3,05	10,30	3,05	10,30	7,81	7,81	7,81
5,57	2,17	7,34	2,17	7,34	5,57	5,57	5,57

Table 9 – continued from previous page

Location 1	Location 2	Location 3	Location 4	Location 5	Location 6	Location 7	Location 8
7,39	2,89	9,75	2,89	9,75	7,39	7,39	7,39
6,06	2,37	8,00	2,37	8,00	6,06	6,06	6,06
8,42	3,29	11,11	3,29	11,11	8,42	8,42	8,42
5,52	2,15	7,28	2,15	7,28	5,52	5,52	5,52
6,80	2,65	8,97	2,65	8,97	6,80	6,80	6,80
6,03	2,36	7,96	2,36	7,96	6,03	6,03	6,03
6,31	2,47	8,33	2,47	8,33	6,31	6,31	6,31
5,88	2,30	7,76	2,30	7,76	5,88	5,88	5,88
5,38	2,10	7,10	2,10	7,10	5,38	5,38	5,38
5,88	2,30	7,76	2,30	7,76	5,88	5,88	5,88
5,67	2,21	7,48	2,21	7,48	5,67	5,67	5,67
5,90	2,30	7,78	2,30	7,78	5,90	5,90	5,90
5,52	2,15	7,28	2,15	7,28	5,52	5,52	5,52
4,59	1,79	6,05	1,79	6,05	4,59	4,59	4,59
7,19	2,81	9,49	2,81	9,49	7,19	7,19	7,19
6,95	2,71	9,16	2,71	9,16	6,95	6,95	6,95
3,74	1,46	4,93	1,46	4,93	3,74	3,74	3,74
4,85	1,89	6,40	1,89	6,40	4,85	4,85	4,85
15,17	15,17	15,17	15,17	15,17	15,17	15,17	15,17
1,98	1,98	1,98	1,98	1,98	1,98	1,98	1,98
30,59	30,59	30,59	30,59	30,59	30,59	30,59	30,59
7,69	7,69	7,69	7,69	7,69	7,69	7,69	7,69
6,35	6,35	6,35	6,35	6,35	6,35	6,35	6,35
9,53	9,53	9,53	9,53	9,53	9,53	9,53	9,53
0,37	0,37	0,37	0,37	0,37	0,37	0,37	0,37

Figure 22 is the output of AIMMS where is illustrated which locations are allocated to each subgroup. The columns represent the locations and the rows represent the subgroups of category C_3.

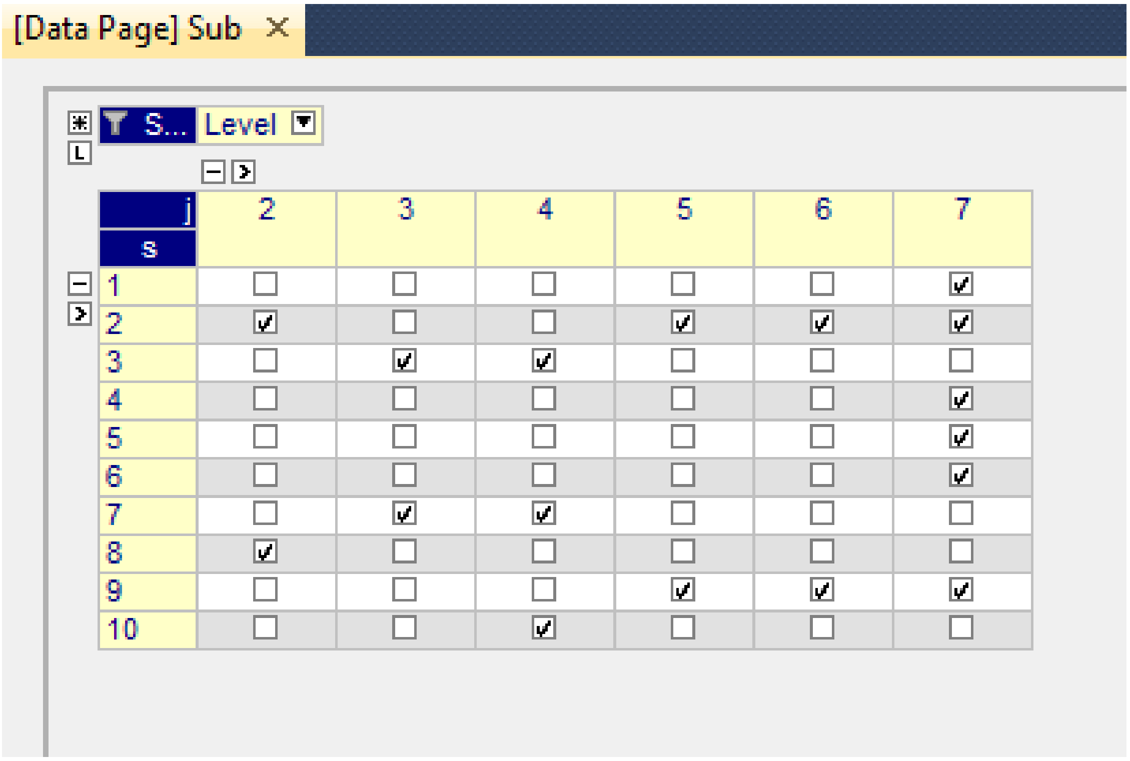

Figure 22: The locations assigned to each subgroup of category C_3.

Figure 23 is the output of AIMMS where is illustrated which locations are allocated to each group. The columns represent the locations and the rows represent the groups of category C_3.

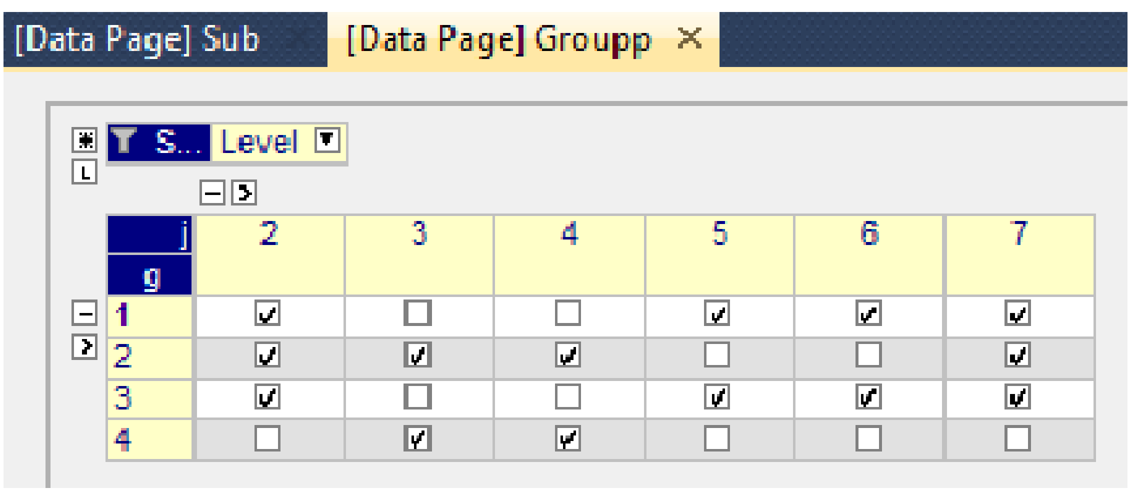

Figure 23: The locations assigned to each group of category C_3.