Optimising Supermarket Layout Design based on Market Basket Analysis

Project by Yoanna A Kurnianingsih

Background

- The layout of a supermarket is a critical factor in influencing consumer behaviour and store profitability. Efficiently arranged products can significantly enhance the shopping experience, leading to increased customer satisfaction and sales.
- This optimisation of supermarket layout not only aims to improve the shopping experience but also seeks to boost the supermarket's operational efficiency and profitability, making it a vital study in the realm of retail management and consumer behaviour analysis

Project Objective

- My project aims to optimise supermarket layout design by employing Market Basket Analysis (MBA) to determine the confidence values between item pairs and utilise Integer Programming to strategically position items for enhanced shopping efficiency and increased sales

Data Collection

Use a groceries dataset taken from Kaggle (Groceries Market Basket Dataset (kaggle.com))

- 153 transactions
- 19 unique items
- Perform Market Basket Analysis by using Apriori Algorithm to obtain the confidence value for each pair of items

Visit Cold Storage @ NUS Kent Vale to sketch the supermarket layout

Market Basket Analysis

- Market Basket Analysis or MBA is a data mining technique used to uncover purchase patterns, that is, to identify which items are usually purchased together.
- MBA uses association rules to identify and describe customers’ purchase behaviour and discover interesting relationships among items.

Association Rules

1 Support	The support for a set of items is the proportions of all transactions that contain the set.
2 Confidence	Is the support for the co-occurrence of all items in a rule, conditional on the support for the left-hand set alone.
3 Lift	Is the ratio of the observed support to the expected if X and Y are independent.

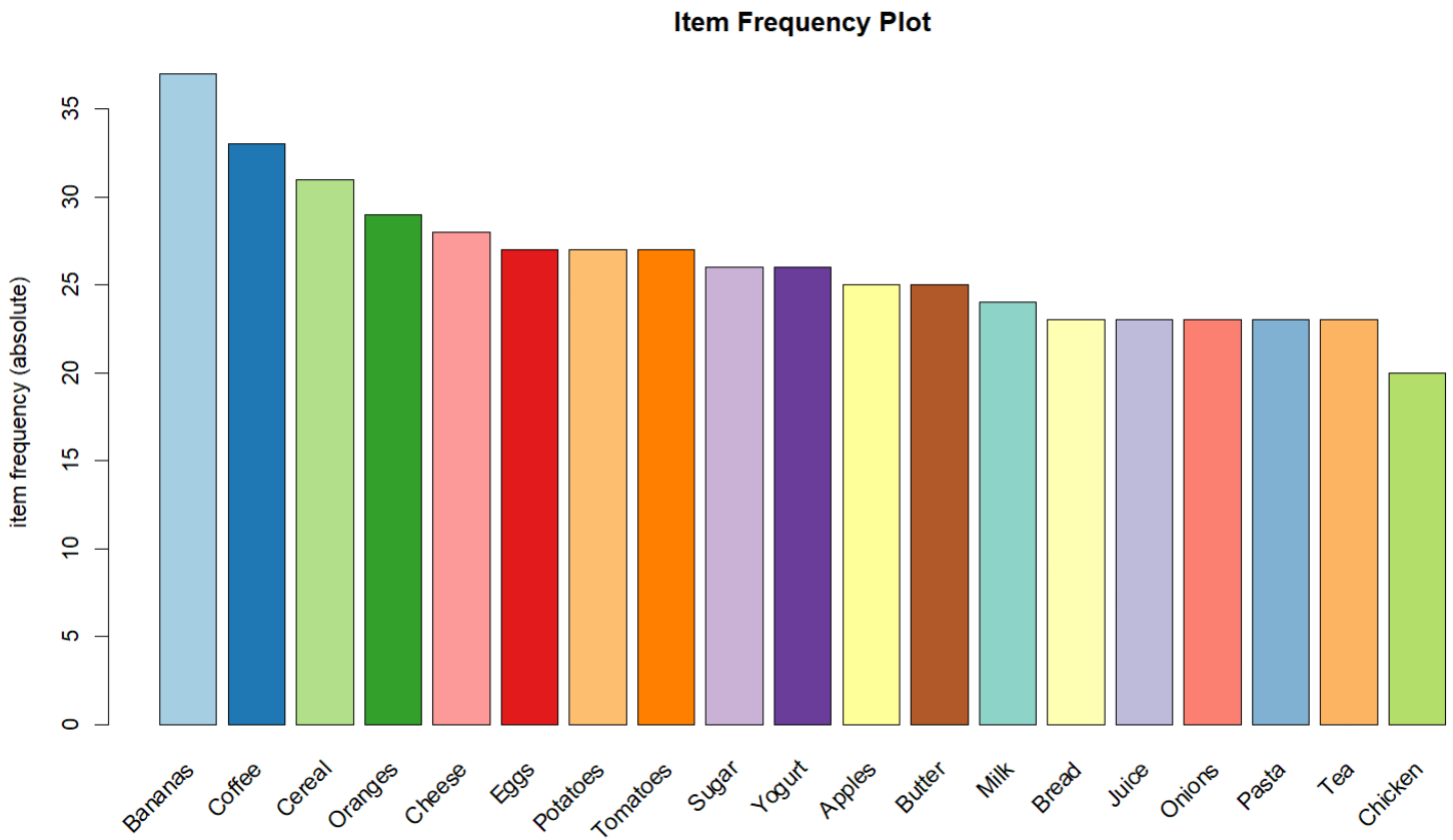
Rule $X \rightarrow Y$

$$Support = \frac{Frequency(X, Y)}{N\ transactions}$$

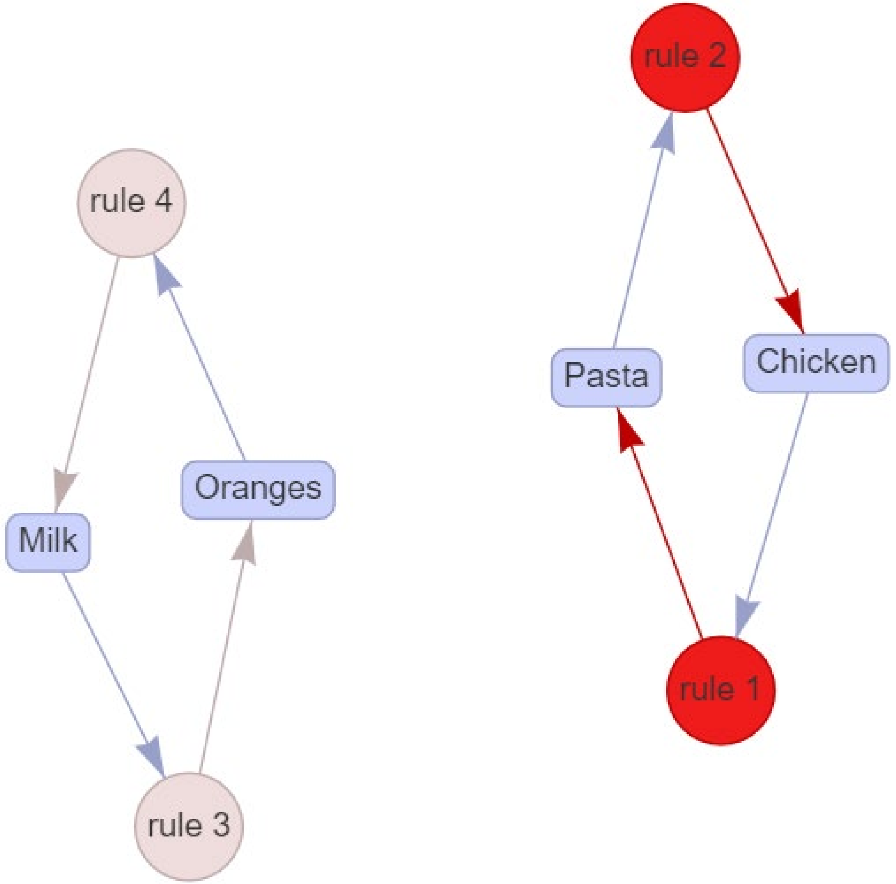
$$\boxed{Confidence = \frac{Support(X, Y)}{Support\ (X)} = \frac{Frequency(X, Y)}{Frequency\ (X)}}$$

$$Lift = \frac{Support(X, Y)}{Support(X) * Support(Y)}$$

Most Frequent Items Purchased



Strongest Association

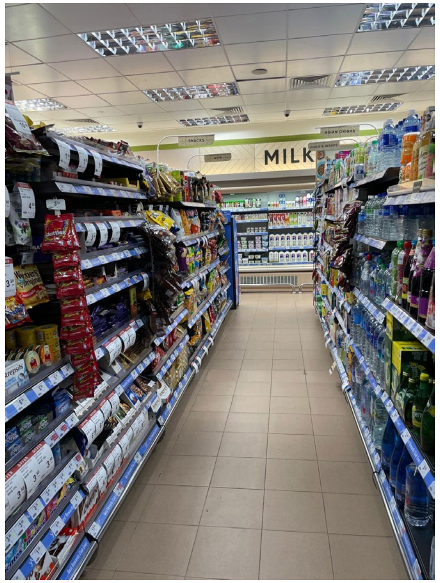


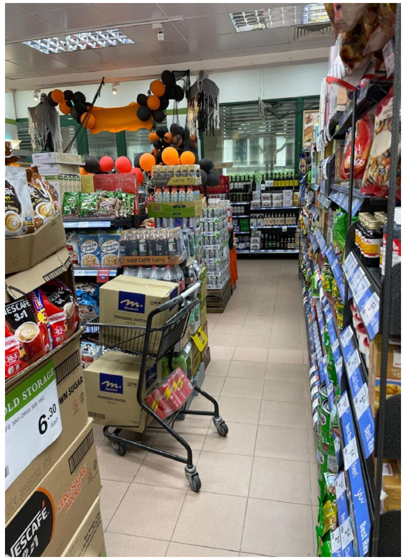
Confidence Matrix

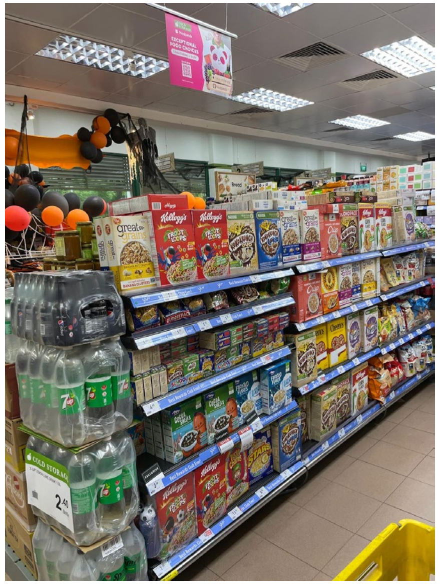
Filter

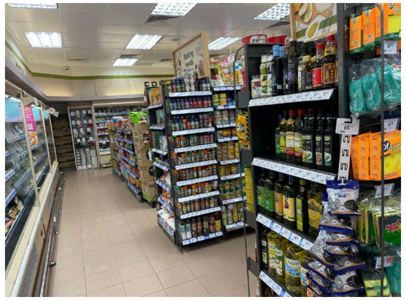
	Chicken	Apples	Tea	Sugar	Pasta	Eggs	Onions	Juice	Yogurt	Milk	Tomatoes	Bread	Butter	Oranges	Cheese	Cereal	Potatoes	Coffee	Bananas
Chicken	0.00000000	0.25000000	0.10000000	0.15000000	0.4000000	0.20000000	0.2500000	0.2500000	0.1500000	0.1500000	0.35000000	0.30000000	0.35000000	0.30000000	0.1500000	0.35000000	0.4000000	0.1500000	0.4000000
Apples	0.20000000	0.00000000	0.20000000	0.36000000	0.2400000	0.16000000	0.1600000	0.0800000	0.3200000	0.2800000	0.24000000	0.28000000	0.16000000	0.16000000	0.2400000	0.12000000	0.2800000	0.2000000	0.0800000
Tea	0.08695652	0.21739130	0.00000000	0.21739130	0.2608696	0.17391304	0.2608696	0.3478261	0.2608696	0.2173913	0.08695652	0.21739130	0.34782609	0.21739130	0.3043478	0.26086957	0.2608696	0.3478261	0.2173913
Sugar	0.11538462	0.34615385	0.19230769	0.00000000	0.2307692	0.15384615	0.2307692	0.1538462	0.2692308	0.1923077	0.34615385	0.19230769	0.07692308	0.07692308	0.2307692	0.07692308	0.2307692	0.3076923	0.2692308
Pasta	0.34782609	0.26086957	0.26086957	0.26086957	0.0000000	0.26086957	0.2173913	0.1739130	0.1739130	0.2173913	0.17391304	0.26086957	0.13043478	0.13043478	0.1739130	0.26086957	0.2608696	0.3913043	0.3478261
Eggs	0.14814815	0.14814815	0.14814815	0.14814815	0.2222222	0.00000000	0.1851852	0.2592593	0.1851852	0.1481481	0.29629630	0.07407407	0.29629630	0.11111111	0.1851852	0.25925926	0.2592593	0.2222222	0.2592593
Onions	0.21739130	0.17391304	0.26086957	0.26086957	0.2173913	0.21739130	0.0000000	0.1739130	0.1304348	0.3043478	0.21739130	0.26086957	0.13043478	0.17391304	0.3913043	0.43478261	0.2608696	0.3043478	0.2608696
Juice	0.21739130	0.08695652	0.34782609	0.17391304	0.1739130	0.30434783	0.1739130	0.0000000	0.2608696	0.1739130	0.21739130	0.21739130	0.34782609	0.17391304	0.2173913	0.21739130	0.3478261	0.3913043	0.3478261
Yogurt	0.11538462	0.30769231	0.23076923	0.26923077	0.1538462	0.19230769	0.1153846	0.2307692	0.0000000	0.1923077	0.19230769	0.23076923	0.19230769	0.30769231	0.2692308	0.15384615	0.1923077	0.3461538	0.2692308
Milk	0.12500000	0.29166667	0.20833333	0.20833333	0.2083333	0.16666667	0.2916667	0.1666667	0.2083333	0.0000000	0.29166667	0.25000000	0.29166667	0.50000000	0.2500000	0.20833333	0.2500000	0.2500000	0.2500000
Tomatoes	0.25925926	0.22222222	0.07407407	0.33333333	0.1481481	0.29629630	0.1851852	0.1851852	0.1851852	0.2592593	0.00000000	0.18518519	0.25925926	0.18518519	0.2222222	0.25925926	0.3333333	0.1851852	0.2962963
Bread	0.26086957	0.30434783	0.21739130	0.21739130	0.2608696	0.08695652	0.2608696	0.2173913	0.2608696	0.2608696	0.21739130	0.00000000	0.34782609	0.39130435	0.2608696	0.26086957	0.3913043	0.4782609	0.3043478
Butter	0.28000000	0.16000000	0.32000000	0.08000000	0.1200000	0.32000000	0.1200000	0.3200000	0.2000000	0.2800000	0.28000000	0.32000000	0.00000000	0.28000000	0.2400000	0.28000000	0.3600000	0.2000000	0.3600000
Oranges	0.20689655	0.13793103	0.17241379	0.06896552	0.1034483	0.10344828	0.1379310	0.1379310	0.2758621	0.4137931	0.17241379	0.31034483	0.24137931	0.00000000	0.3103448	0.17241379	0.3448276	0.2413793	0.2758621
Cheese	0.10714286	0.21428571	0.25000000	0.21428571	0.1428571	0.17857143	0.3214286	0.1785714	0.2500000	0.2142857	0.21428571	0.21428571	0.21428571	0.32142857	0.0000000	0.42857143	0.1785714	0.2500000	0.2857143
Cereal	0.22580645	0.09677419	0.19354839	0.06451613	0.1935484	0.22580645	0.3225806	0.1612903	0.1290323	0.1612903	0.22580645	0.19354839	0.22580645	0.16129032	0.3870968	0.00000000	0.2258065	0.2580645	0.2903226
Potatoes	0.29629630	0.25925926	0.22222222	0.22222222	0.2222222	0.25925926	0.2222222	0.2962963	0.1851852	0.2222222	0.33333333	0.33333333	0.33333333	0.37037037	0.1851852	0.25925926	0.0000000	0.2962963	0.3703704
Coffee	0.09090909	0.15151515	0.24242424	0.24242424	0.2727273	0.18181818	0.2121212	0.2727273	0.2727273	0.1818182	0.15151515	0.33333333	0.15151515	0.21212121	0.2121212	0.24242424	0.2424242	0.0000000	0.1818182
Bananas	0.21621622	0.05405405	0.13513514	0.18918919	0.2162162	0.18918919	0.1621622	0.2162162	0.1891892	0.1621622	0.21621622	0.18918919	0.24324324	0.21621622	0.2162162	0.24324324	0.2702703	0.1621622	0.0000000

Sketching the Supermarket Layout



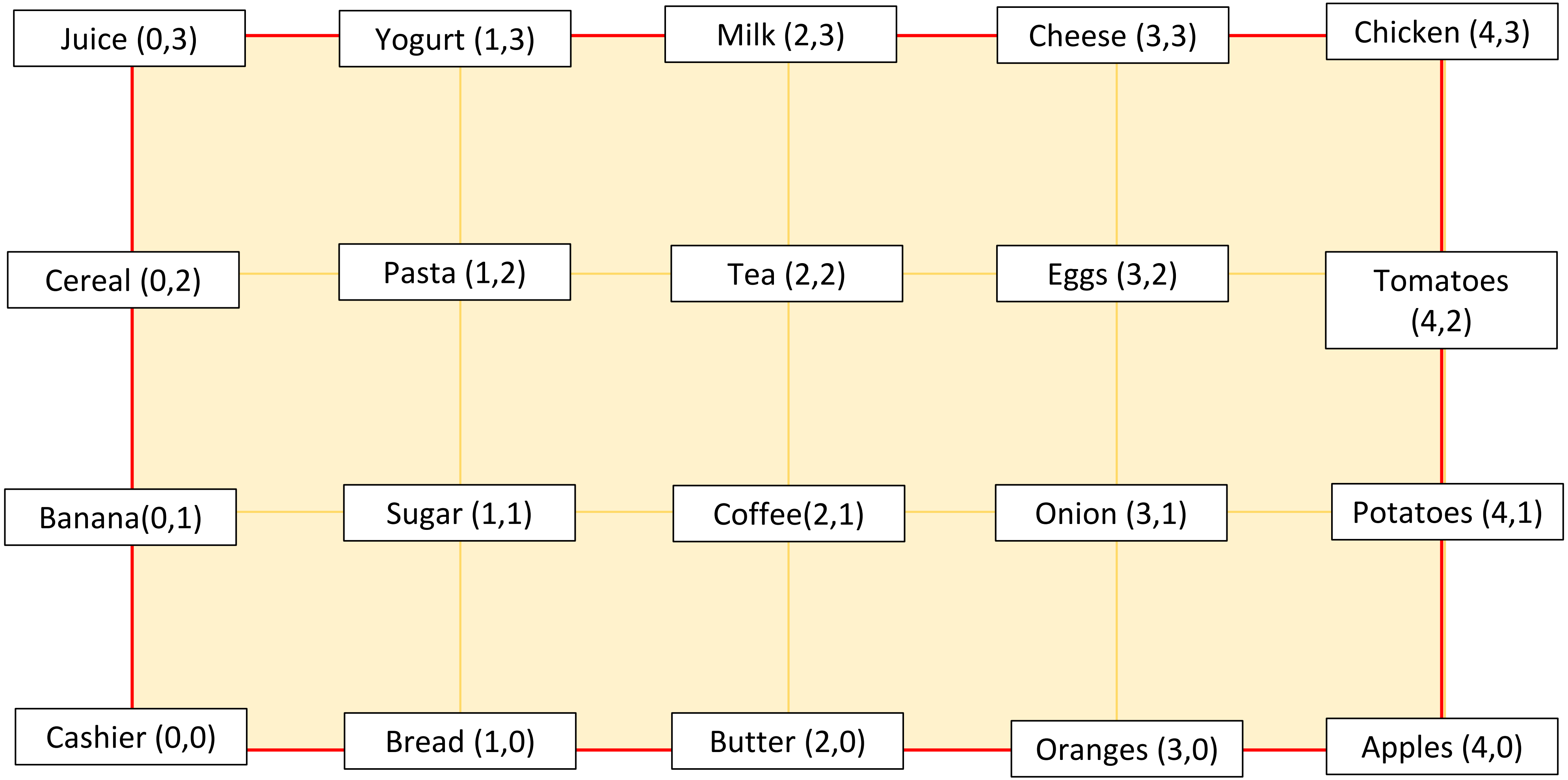




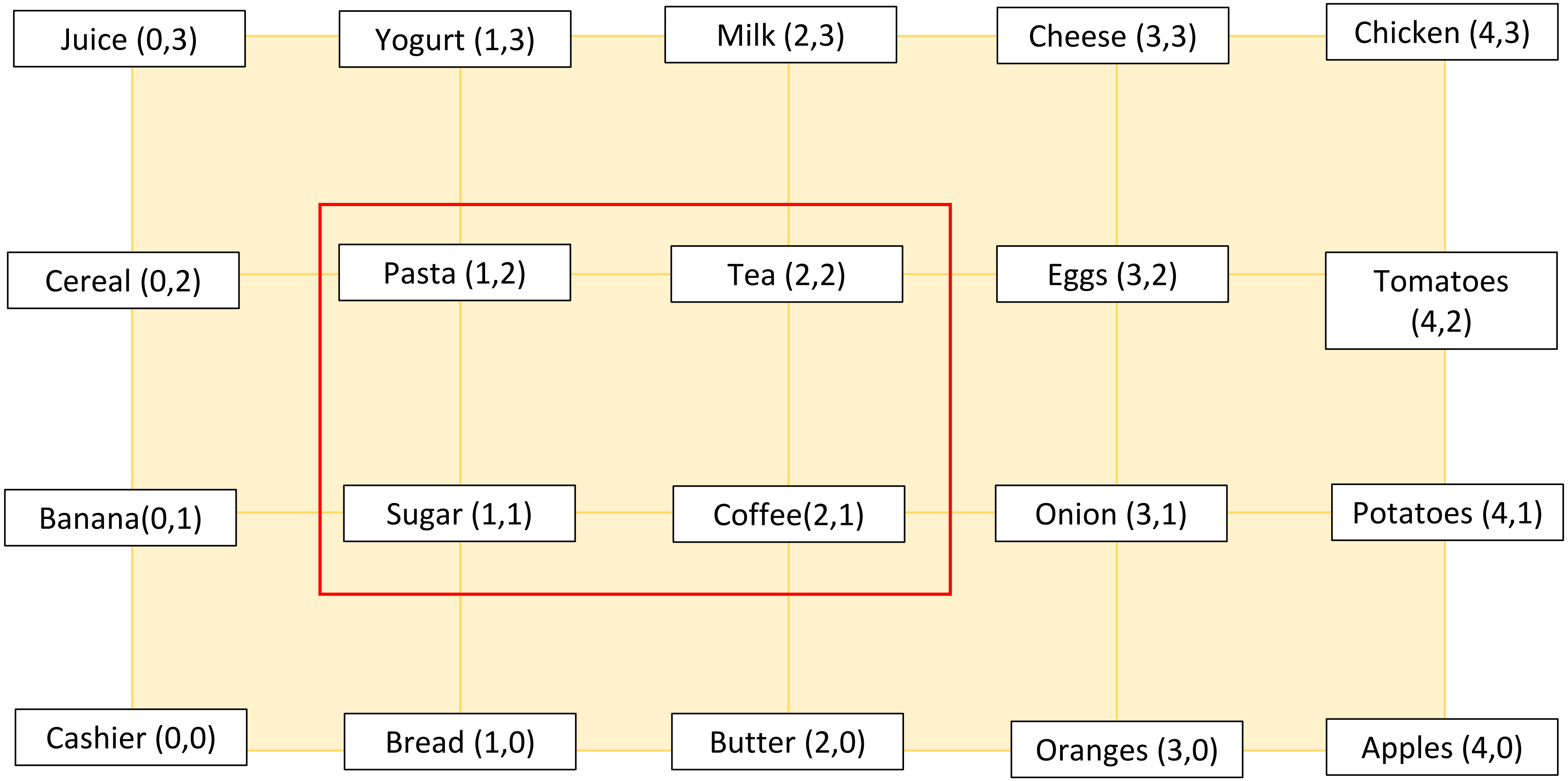


First Round

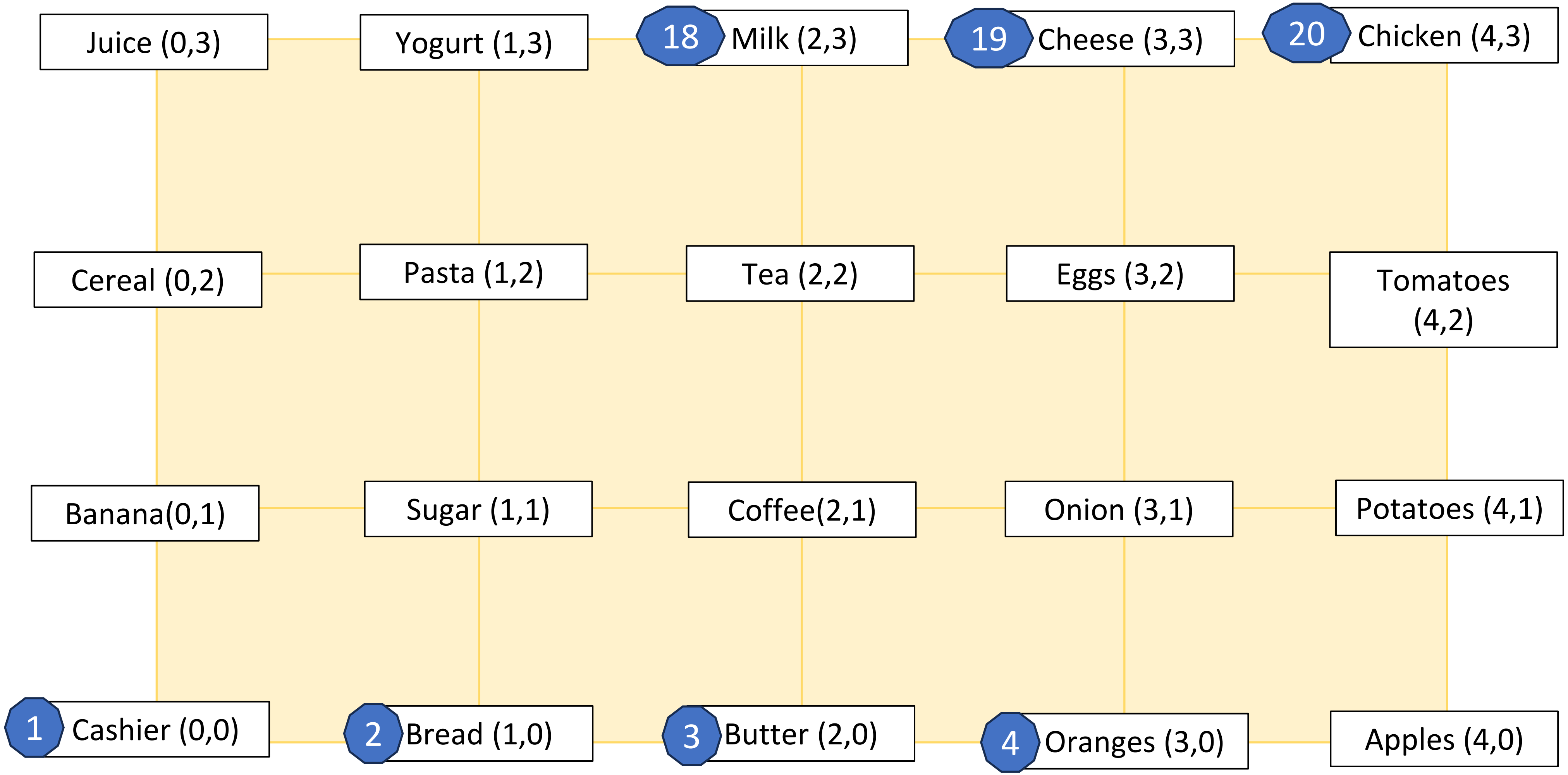
Second Round



Third Round


Juice (0,3)
Yogurt (1,3)
Milk (2,3)
Cheese (3,3)
Chicken (4,3)
Cereal (0,2)
Pasta (1,2)
Tea (2,2)
Eggs (3,2)
Tomatoes (4,2)
Banana(0,1)
Sugar (1,1)
Coffee(2,1)
Onion (3,1)
Potatoes (4,1)
Cashier (0,0)
Bread (1,0)
Butter (2,0)
Oranges (3,0)
Apples (4,0)

Third Round



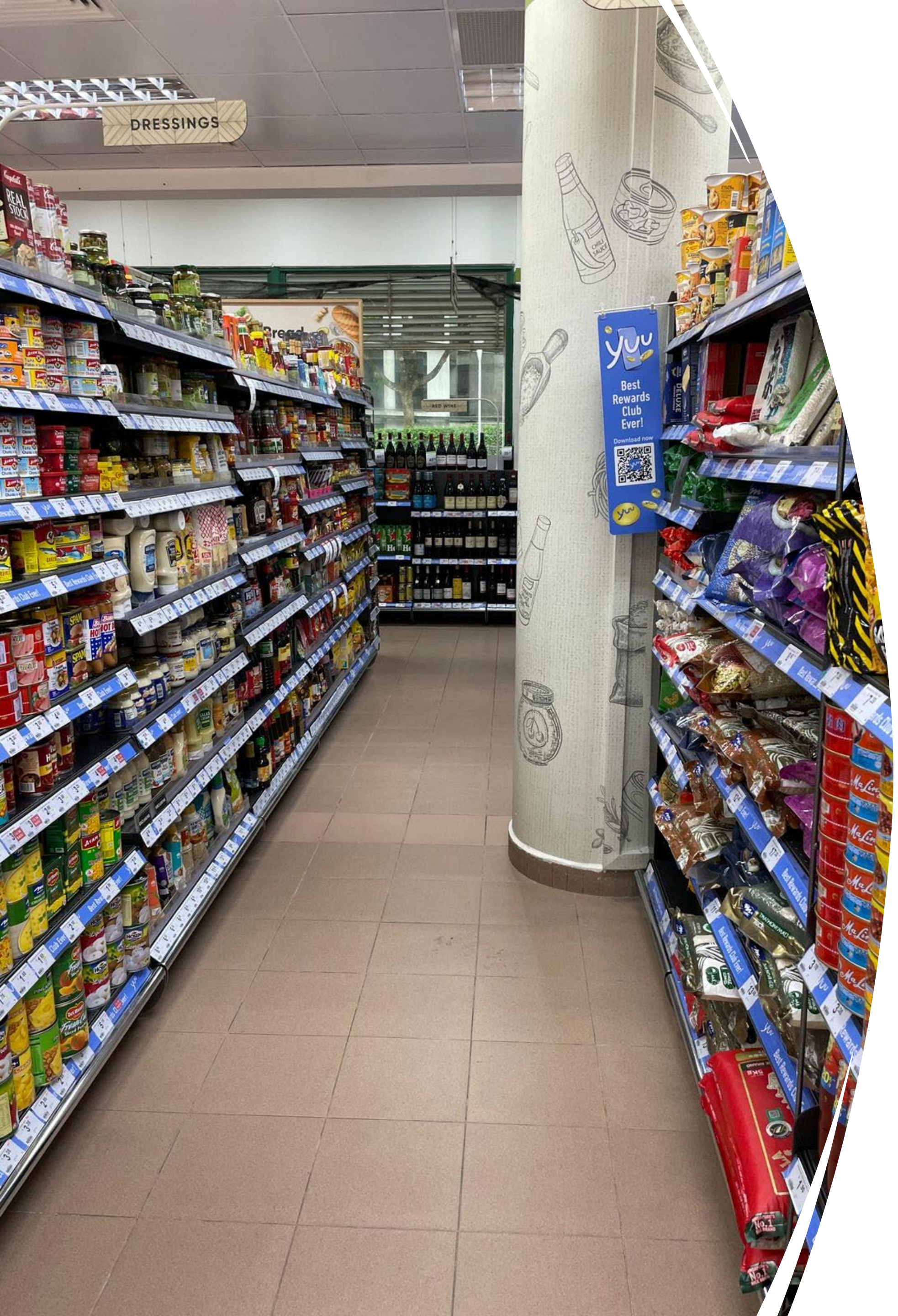
Juice (0,3)
Yogurt (1,3)
18
Milk (2,3)
19
Cheese (3,3)
20
Chicken (4,3)
Cereal (0,2)
Pasta (1,2)
Tea (2,2)
Eggs (3,2)
Tomatoes (4,2)
Banana(0,1)
Sugar (1,1)
Coffee(2,1)
Onion (3,1)
Potatoes (4,1)
1
Cashier (0,0)
2
Bread (1,0)
3
Butter (2,0)
4
Oranges (3,0)
Apples (4,0)

Distance Matrix

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17	V18	V19	V20
1	0	1	2	3	4	1	2	3	4	5	2	3	4	5	6	3	4	5	6	7
2	1	0	1	2	3	2	1	2	3	4	3	2	3	4	5	4	3	4	5	6
3	2	1	0	1	2	3	2	1	2	3	4	3	2	3	4	5	4	3	4	5
4	3	2	1	0	1	4	3	2	1	2	5	4	3	2	3	6	5	4	3	4
5	4	3	2	1	0	5	4	3	2	1	6	5	4	3	2	7	6	5	4	3
6	1	2	3	4	5	0	1	2	3	4	1	2	3	4	5	2	3	4	5	6
7	2	1	2	3	4	1	0	1	2	3	2	1	2	3	4	3	2	3	4	5
8	3	2	1	2	3	2	1	0	1	2	3	2	1	2	3	4	3	2	3	4
9	4	3	2	1	2	3	2	1	0	1	4	3	2	1	2	5	4	3	2	3
10	5	4	3	2	1	4	3	2	1	0	5	4	3	2	1	6	5	4	3	2
11	2	3	4	5	6	1	2	3	4	5	0	1	2	3	4	1	2	3	4	5
12	3	2	3	4	5	2	1	2	3	4	1	0	1	2	3	2	1	2	3	4
13	4	3	2	3	4	3	2	1	2	3	2	1	0	1	2	3	2	1	2	3
14	5	4	3	2	3	4	3	2	1	2	3	2	1	0	1	4	3	2	1	2
15	6	5	4	3	2	5	4	3	2	1	4	3	2	1	0	5	4	3	2	1
16	3	4	5	6	7	2	3	4	5	6	1	2	3	4	5	0	1	2	3	4
17	4	3	4	5	6	3	2	3	4	5	2	1	2	3	4	1	0	1	2	3
18	5	4	3	4	5	4	3	2	3	4	3	2	1	2	3	2	1	0	1	2
19	6	5	4	3	4	5	4	3	2	3	4	3	2	1	2	3	2	1	0	1
20	7	6	5	4	3	6	5	4	3	2	5	4	3	2	1	4	3	2	1	0

Objectives

- Maximising the Total Distance between products that are strongly associated
  - Unplanned spending at retail stores has been related to within-trips and store travel distances (Hui et al., 2013)
- Minimising the Total Distance between products that are strongly associated
  - In other words, maximising the total distance between products that are weakly associated



Layout Optimisation

- The decision variables are:

$$x_{ik} = \begin{cases} 1 & \text{if product } i \text{ is located at position } k, \\ 0 & \text{else.} \end{cases}$$

$$x_{jl} = \begin{cases} 1 & \text{if product } j \text{ is located at position } l, \\ 0 & \text{else.} \end{cases}$$

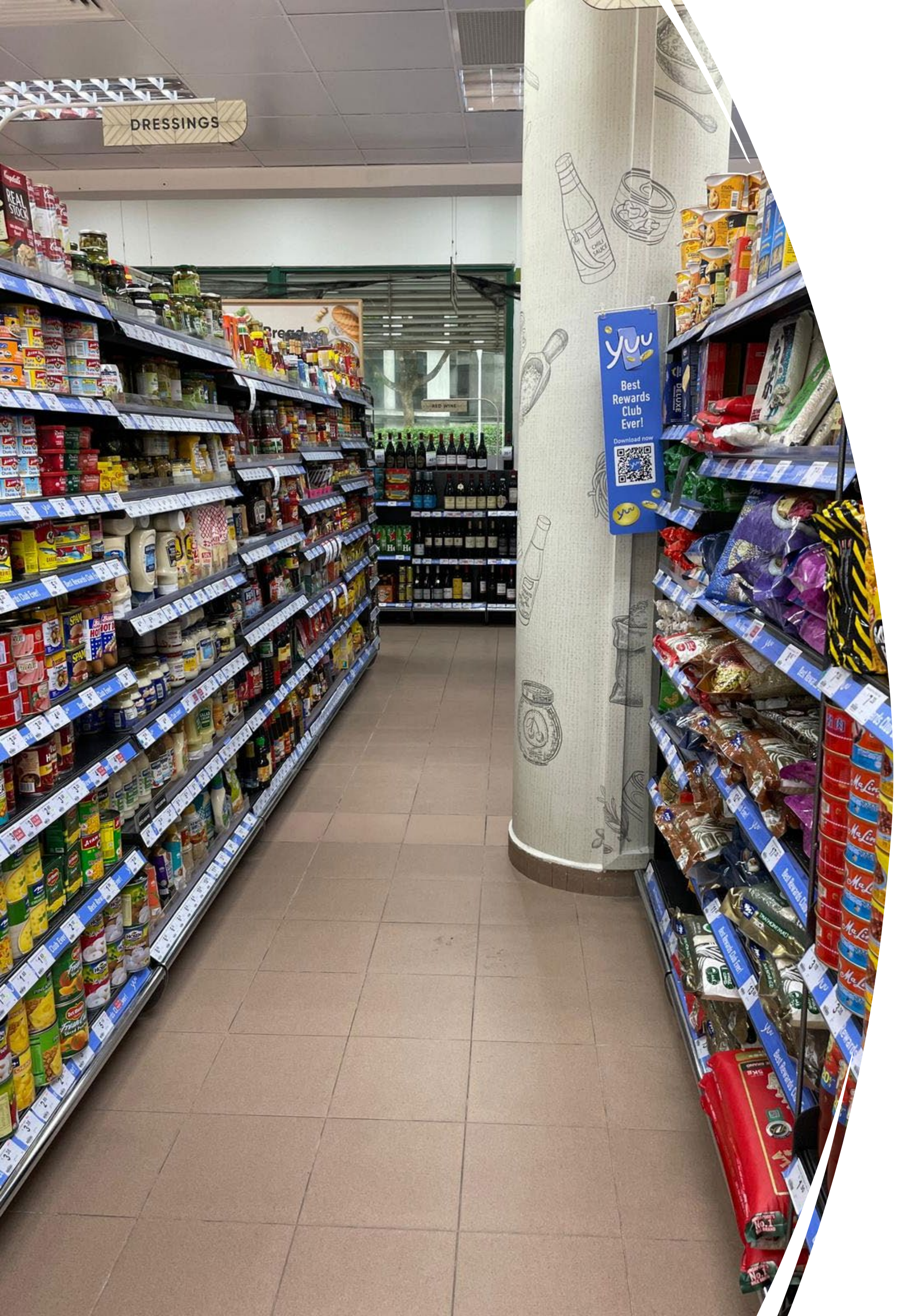
- The objective function can be formulated as:

$$\sum_{ijlk} \boxed{x_{il} x_{jk}} d_{lk} c_{ij}$$

Solution?

introducing variables y_{ijkl}

Where each pair of produced is weighted by their confidence value.



Layout Optimisation

- The decision variables are:

$$x_{ik} = \begin{cases} 1 & \text{if product } i \text{ is located at position } k, \\ 0 & \text{else.} \end{cases}$$

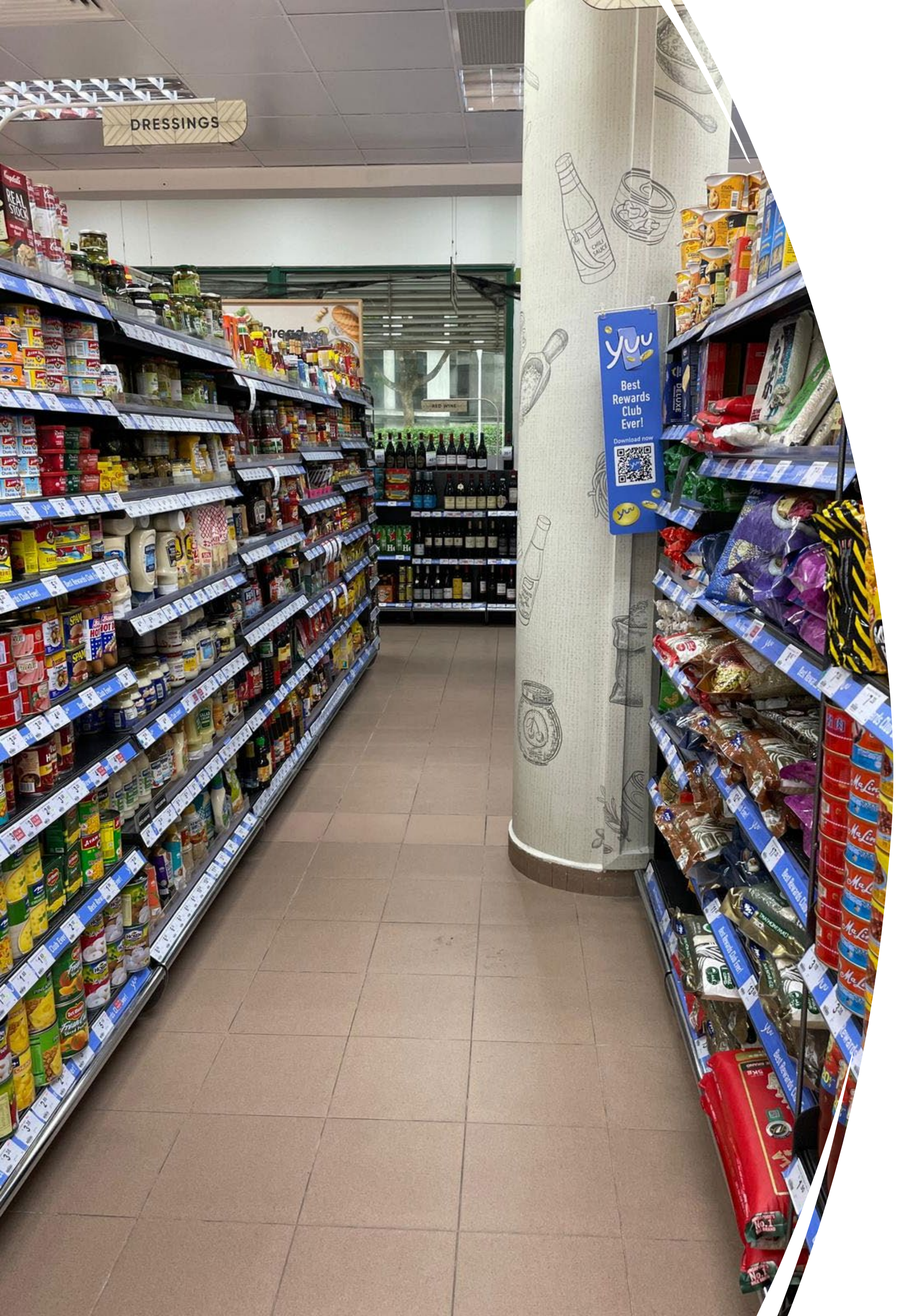
$$x_{jl} = \begin{cases} 1 & \text{if product } j \text{ is located at position } l, \\ 0 & \text{else.} \end{cases}$$

- The objective function can be formulated as:

$$\sum_{ijkl} x_{ik} x_{jl} d_{kl} c_{ij}$$

$$y_{ijkl} \equiv x_{ik} \cdot x_{jl}$$

$$y_{ijlk} = \begin{cases} 1 & \text{if products } i, j \text{ are located at positions } l, k, \text{ respectively,} \\ 0 & \text{else.} \end{cases}$$



Layout Optimisation

- The decision variables are:

$$x_{ik} = \begin{cases} 1 & \text{if product } i \text{ is located at position } k, \\ 0 & \text{else.} \end{cases}$$

$$x_{jl} = \begin{cases} 1 & \text{if product } j \text{ is located at position } l, \\ 0 & \text{else.} \end{cases}$$

- The objective function can be formulated as:

$$\begin{aligned} \text{maximise} \quad & \sum_{ijlk} y_{ijlk} d_k c_{ij} & \\ \text{s.t.} \quad & 2y_{ijlk} \leq x_{ik} + x_{jl}, & \forall i, j, l, k \\ & \sum_{i} x_{ik} = 1, & \forall k \\ & \sum_{k} x_{ik} = 1, & \forall i \\ & y_{ijlk} \in \{0, 1\}, x_{ik} \in \{0, 1\}. & \end{aligned}$$

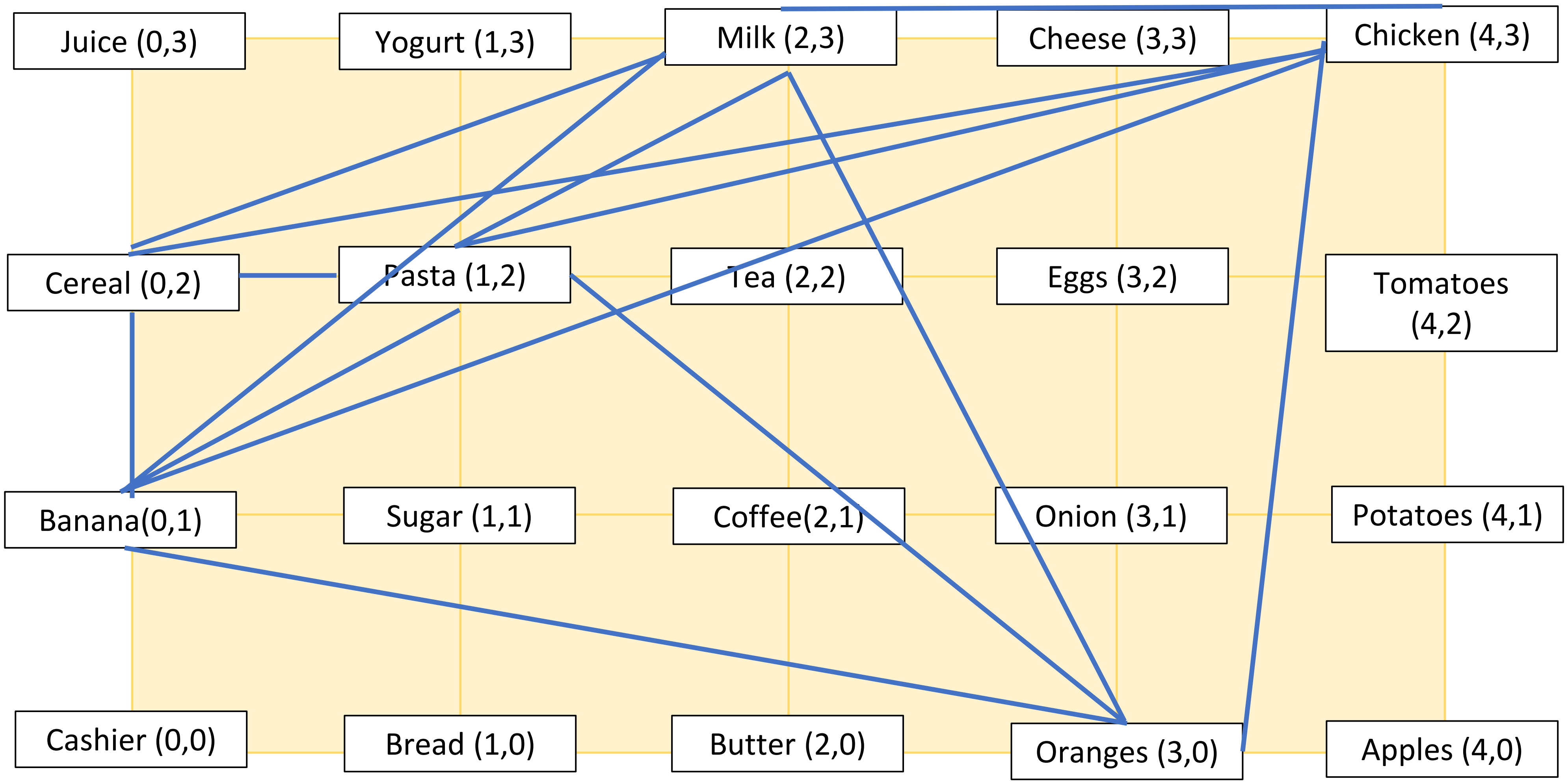
Running the Programme in R

- Time taken for the full dataset and layout was too long
  - 19 items x 19 positions, > 12 hours
  - 10 items x 10 positions, > a few hours
  - 8 items x 8 positions, > an hour

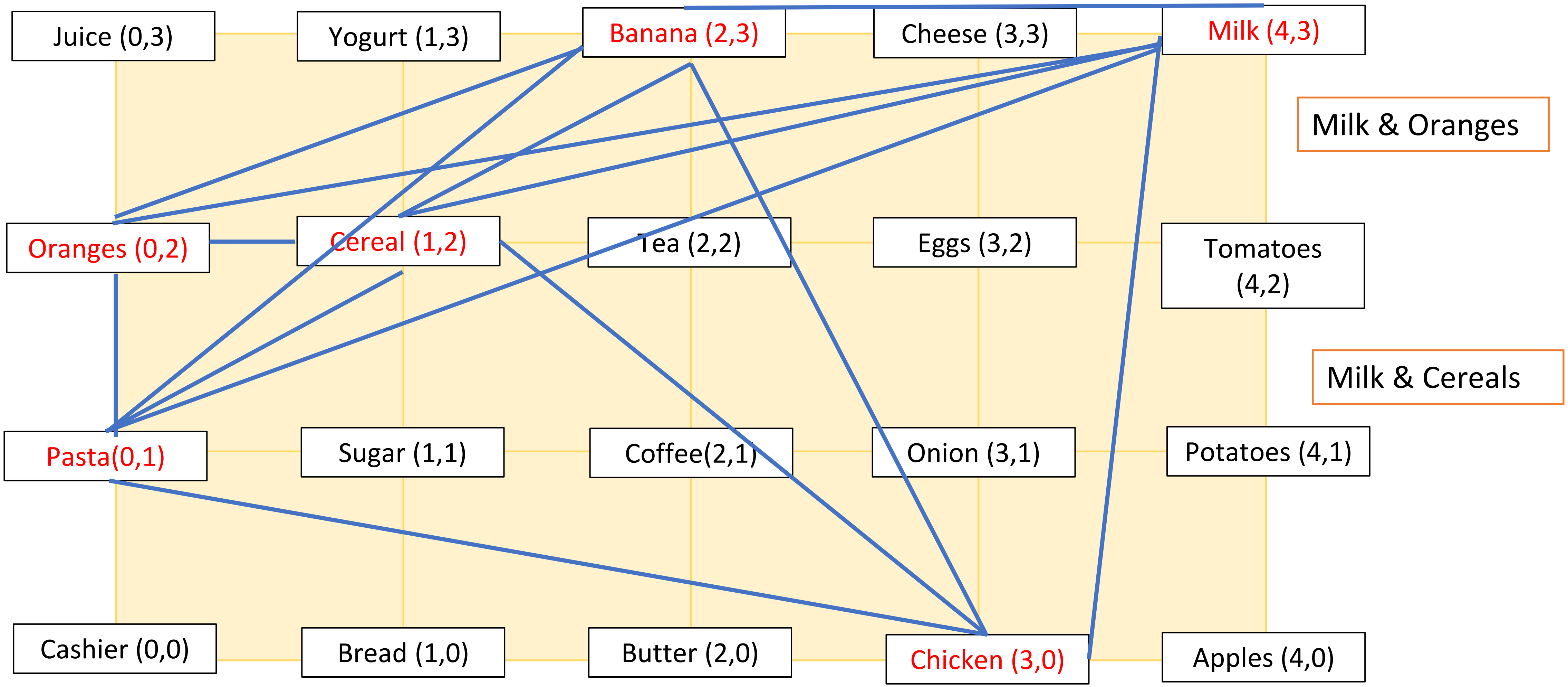
- Decided to use a 6 items x 6 positions subset. What are the items that I think are important?

{Milk, Orange, Pasta, Chicken, Banana, Cereal, Coffee}

Original Layout
Original Distance * Confidence: 13.050



Maximise Total Distance for High Confidence
Objective Value: 27.697



Limitation & Future Work

LIMITATIONS

1. THE TIME TAKEN TO SOLVE THE PROBLEM CAN BE SIGNIFICANT

2. THE CURRENT MODEL MAY NOT SCALE EFFICIENTLY TO VERY LARGE SUPERMARKETS

3. SUPERMARKET INVENTORY AND LAYOUT PREFERENCES MAY CHANGE FREQUENTLY.

4. THE CURRENT MODEL MIGHT BE TAILORED TO A SPECIFIC SUPERMARKET'S LAYOUT.

5. THE MODEL MAY NOT FULLY CAPTURE THE COMPLEXITIES OF CUSTOMER BEHAVIOUR

Future Work:

1. We can try heuristic or parallel computing to reduce the computing time

2. Create a dynamic model that can update the layout periodically based on real-time data.

3. Integrate machine learning to predict changes in customer behaviour and adjust the layout accordingly.

Reference

- Bermudez, J., Apolinario, K., & Abad, A. G. (2016). Layout optimization and promotional strategies design in a retail store based on a market basket analysis. In *14th LACCEI International Multi-Conference for Engineering, Education, and Technology*.
- Hui, S. K., Inman, J. J., Huang, Y., & Suher, J. (2013). The effect of in-store travel distance on unplanned spending: Applications to mobile promotion strategies. *Journal of Marketing*, *77*(2), 1-16.