MGSC - 670
Revenue Management

Group Assignment -1: Markdown Strategy

May 10, 2022

Group

Aishwarya Choukekar (260985917)

Chelsea Hon (261010089)

Jingyuan Wang (260609651)

Rebecca Mukena Yumba (261003552)

1. Introduction

This report details our approach to generate a markdown pricing strategy for the retailer game described on randhawa.us. The retailer game has 2 main goals:

1. Develop a generic markdown pricing strategy for a retailer.
2. Maximize the total revenue when selling some inventory over 15 weeks

We used a sample output of the game with 3 different strategies on 15 items: 10% discount in the 7th week, 20% discount on the 8th week and 40% discount on the 10th week. We came up with 2 approaches: one simplistic and another more quantitative.

2. Initial Estimation Approach

In this section we discuss a heuristic method to estimate the optimal strategy based on sales data with the objective to have 0 remaining inventory in-and-only-in the last week and the highest possible revenue.

We started by plotting all the strategies for the 15 runs from the sales-data.xlsx file. These graphs showed which strategies work better to reduce the remaining inventory in the last week (see graphs in the appendix section Plots of the Markdown Strategy).

We can gather these insights from the graphs:

- Strategy 1 Early Conservative Prevention: Discounting by 10% on the 7th week consistently leads to a higher level of inventory remaining. It has the lowest revenue mean and highest variance in 5 runs compared to other strategies. Potential to maximize revenue is observed.
- Strategy 2 The Middle Ground: The 20% discount on the 8th week seems like a good middle ground between the 2 previous strategies, however we do not obtain the ideal scenario (0 remaining inventory in-and-only-in the last week) and we sometimes even get 0 inventory before final week.
- Strategy 3 Delayed Aggressive Correction: Activating the 40% discount on the 10th week almost consistently leads to lower remaining inventory on the 15th week (and sometimes before the last week). It has the highest revenue mean and lowest variance in 5 runs, yet less likely obtaining the maximum revenue.

Strategy 2 and 3 seem a little more aggressive and can diminish the likelihood to have remaining inventory at the end of the sale cycle but they do not guarantee the highest revenue as sometimes the sale < demand in the final weeks.

In order to visualize the change in sales over the weeks after the markdown, we also plotted the Δsales = (sales this week - sales previous week) / sales previous week (see graphs in the appendix section Plots of the Sales Difference After Markdown). These graphs show the following insights:

- The 10% and 20% discount on the 7th and 8th week respectively lead to an average of 0% change in the sales from week to week whereas the 40% discount on the 10th week lead to a decreasing demand.

All these observations can lead us to believe that the ideal strategy would consider an early discount (earlier than the 7th week) and avoid the more aggressive discount towards the end of the sales period. An example would be:

- 10% discount on the 4-6th week 20% discount on the 9-11th week.

In order to validate our intuition we also devised a more quantitative approach to determine the best strategy.

3. Our Mathematical Solution

With previous learning and understanding of the problem, we move forward to establish an optimization model that populates strategy based on the original price level, discount level, and demand function with respect to the given price. For data input, we decided to use the historical sales data set named 'Sales-Data.xlsx', available on the website.

There are some static information/assumptions given as follows:

I. Inventory capacity: 2,000 units
II. Selling period: 15 weeks
III. Discount rates: 10%, 20%, 40%
IV. Unit sale price at initial: $60
V. The first week's sales are given and generated by random
VI. Demand lift with respect to price is homogenous across items

For our approach, we divided this into 2 tasks: first, model the demand function via regressions using historical sales data; second, establish a revenue optimization model with demand estimation determined previously.

3.1 Demand estimation

3.1.1 Data Exploration

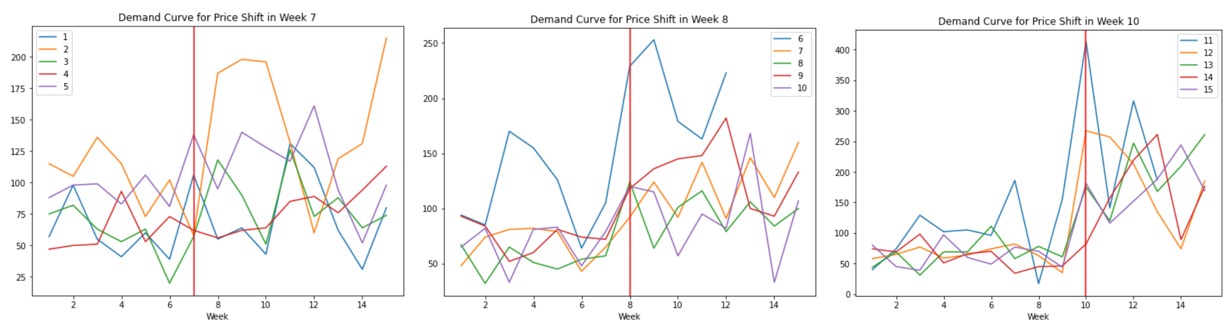
'Sales-Data.xlsx' has 5 attributes initially. It includes sales units of 15 unique products with a given price level at a certain week. All of them have an initial price of $60 corresponding to information (II)(III) above, and each of them contains only 1 price markdown throughout the selling weeks. Generally speaking, there are 3 strategies displayed: markdown at week 7 with 10% off, week 8 with 20% off, and week 10 with 40% off, which we named ‘Scenario’ in the modified version of the historical data file.

3.1.2 Modification and Encoding

Our goal is to estimate demand, however, the data that we have is based on sales. Recall sales = min(Demand, Inventory). To make sure that input represents demand and is not bounded by inventory, we flagged rows that encounter out-of-stock situations, that is where unit sales are significantly below demand level and inventory at the end of the period. These data points are flagged under the attribute ‘Inventory Flag’ and will be excluded from the demand estimation process.

To extract as much information as possible from the historical data set, we made some modifications based on the given historical data set. Aside from the ‘Scenario’ attribute we defined above, we had another variable called ‘Price Change Flag’ to signal any price level change for a given item.

Before we move forward into estimation, we need to encode the time variable ‘Week’ as well. To capture seasonality, we plot the unit sales for items under the 3 scenarios we mentioned above.



We then conclude the following:

- The number of units sold is not necessarily linear along with the timeline, the constant variance assumption might break
- There is a demand lift whenever there's a price drop
- Given the same price level ($60), demand values fell under a similar range
- After Demand lift, the variance of demand distribution tends to increase as well

This also showed that our data set did not present significant evidence of any underlying seasonality. Thus, we did not need to consider or transform time variables, simple label encoding would be enough in this case.

3.1.2 Modeling Historical Data for Demand Estimation

With the full set of data ready for model input, we can consider statistical regression models for demand estimation. However, given the number of data points we have, models with complicated structures might overfit. Models we considered include linear regression, sklearn ensemble methods, LGBM, XGBoost and lastly Neural Networks (sklearn MLP, Tensorflow).

For performance measurements, we have two criteria. MSE (Mean squared error) to capture precision and R-squared score to capture overfitting. Out of all models we fitted, the best model we found is LGBM with default hyperparameters with an R-squared score of 0.392, MSE of 2,636; followed by linear regression with an R-squared score of 0.371 and MSE of 2726.

Given that current optimization solvers available would have trouble taking in decision variables as an input, the optimal solution for that is to consider all possible values and build a lookup table for revenue maximization. Note we have limited time to solve this problem, which means we would not go into depth for every scenario of price change. Instead, we will assume that demand variance will be somewhat similar before and after the price change. Using this terminology, we could establish a look-up table where we have 15 weeks of demand estimation for each of the 4 levels of prices (original price $60 and 3 discount levels mentioned in (III)) given there's no price change, in order to estimate the level of demand at a designated level of price. This table was fed into the optimal demand model we found in the previous step, the demand model output would then be concatenated back to the input table as its corresponding demand level.

Note that at this level, we can consider using this as demand information that is input into the optimization model, but the time and computation cost would be high and we are very likely to land at an infeasible solution. To further generalize this information, we took the mean of demand level grouping by each price level, which led to the following table:

Price Level	**Predicted Demand Level**	**Demand Lift Factor**
60	73.0	1.00
54	99.0	1.36
48	131.0	1.79
36	198.0	2.71

The demand lift factor we calculated here will be further used in the next part when we establish the optimization model. This could be combined with the assumption (VI), so that our formulation together with this demand lift factor could be further extended and applied to any initial price other than $60 given in (IV).

3.2 Optimization Model

With the goal of maximizing revenue and formulating the markdown strategy, we decided to formulate an optimization model using Gurobi. But first, let's have an overview of all the information we have to build the model.

3.2.1 Given Data and Inputs

Sticking to the rules laid out for games, we can pick up on a few details, which we include as pre-defined variables.

Pre-defined variable	**Value**	**Description**
units	2000	Total number of units in inventory to be sold
disc	[1, 0.9, 0.8, 0.6]	Multiplier for price after applying discount rates applicable i.e. $60 (1), $54 (0.9), $48 (0.8), $36 (0.6)

In addition to the above two values, we would be taking two user inputs to ensure our model efficiency despite the dynamic nature of the market.

Input variable	**Value**	**Description**
price	*Starting price* ($60)	This is the initial undiscounted selling price of one unit.
initial_demand	*Dynamic value*	Sales in week 1. We use the demand for product in week 1 as base for further future predicted demand.

3.2.2 Assumptions and Derived Values

By using a multiplier factor to predict demand based on four different price options, we are taking into consideration the average demand for the product at that particular price and using this average predicted demand throughout. Below are the calculated multiplier values for demand rate.

Variable	Derived Value	Description
demand_rate	[1.0, 1.36, 1.79, 2.71]	Factor multiplier for predicted demand based on decided price, as calculated from the previous section.

3.2.3 Formulation

To get the optimal markdown strategy, we are using non-linear mixed integer programming. Our problem formulation is as below:

Decision Variables:

p_i : optimal price to be selled at for the week i.

$d\_ind_{i,j}$: binary variable indicating which price j will be selected from four options for week i.

$sale_i$: variable to store the predicted demand for week i

Objective:

The objective is to maximize revenue i.e. product of price decided and predicted demand.

$$Maximize \sum_{m=1}^{15} p_i sale_i$$

Subject to:

1. Week-1 constraints: These are some initial values set for price in week 1 and default discount indicator as 1, as we select price = 60.

   $$p_{i=0} = price$$

   $$d\_ind_{i=0,j=0} = 1$$

2. Total number of units in inventory: Limiting total predicted demand within the units available for selling.

   $$\sum_{i=1}^{15} \sum_{j=1}^{4} d\_ind_{i,j} * d_j \leq units$$

3. Only one discount at a time: Only one of the four price choices available would be selected at a given time.

   $$\sum_{j=1}^{4} d\_ind_{i,j} == 1 \ \forall i = 1, \ldots, 15$$

4. Sales constraint: Actual sales would be less than or equal to predicted average sales for particular price.

$$sale_i \leq \sum_{j=1}^{4} d\_ind_{i,j} * d_j \ \forall i = 1, \ldots, 15$$

5. Maintain or reduce price constraint: The price of next week should be less than or equal to next week.

$$\sum_{j=1}^{4} d\_ind_{i+1,j} * disc_j * price \leq \sum_{j=1}^{4} d\_ind_{i,j} * disc_j * price \ \forall i = 1, \ldots, 14$$

6. Price constraint: Based on the above limit and considering the discount indicator, calculating price for following weeks.

$$p_{i+1} == \sum_{j=1}^{4} d\_ind_{i+1,j} * disc_j * p0 \ \forall i = 1, \ldots, 14$$

3.2.4 Implementation

The above formulated maximization problem was then programmed to fit the Gurobi model using python. (Please refer to the Jupyter notebook attached for the code). For running the program, we would first be taking input price (which for the purpose of a given assignment is fixed to 60), followed by another user input asking the sales for the first week. Based on the popularity of the product in week 1, the model thus formulates a strategy for following 14 weeks based on estimated demand.

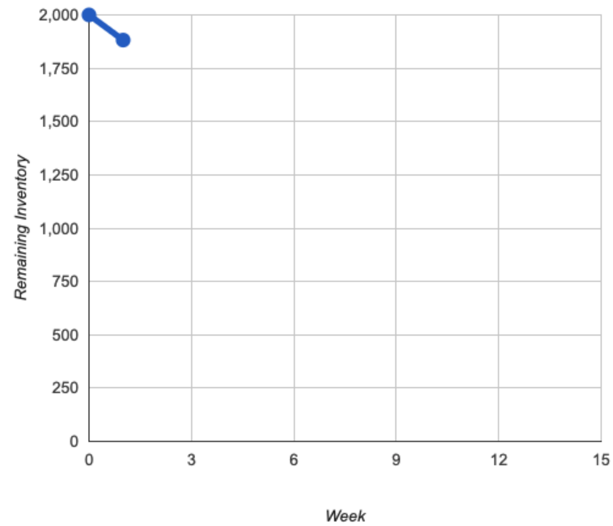
4. Results Illustration

In this section, we are going to implement the markdown strategy for two scenarios.

4.1 Initial Sales = 118

In the first scenario, we take the initial price $60 as fixed by the question and initial sales 118 as randomized.

Week	Price	Sales	Remaining Inventory
1	60	118	1882



Input accordingly in Gurobi notebook and obtain the price per week to maximize revenue.

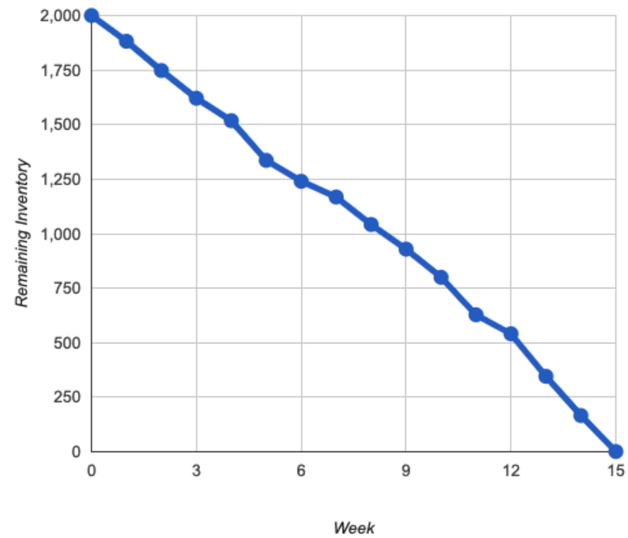
```
In [*]: markdown_strategy()
        Input initial price: 60
        Input initial sales: 118
```

The sample decision variables of price per week implies a markdown strategy of 10% at week 11.

```
    Variable            X
-------------------------
     price_1           60
     price_2           60
     price_3           60
     price_4           60
     price_5           60
     price_6           60
     price_7           60
     price_8           60
     price_9           60
    price_10           60
    price_11           54
    price_12           54
    price_13           54
    price_14           54
    price_15           54
```

Apply the markdown strategy in the retailer game and review the difference with respective to perfect foresight strategy. Highest possible revenue with perfect foresight strategy is achieved.

Week	Price	Sales	Remaining Inventory
1	60	118	1882
2	60	134	1748
3	60	127	1621
4	60	103	1518
5	60	182	1336
6	60	96	1240
7	60	72	1168
8	60	126	1042
9	60	113	929
10	60	129	800
11	54	172	628
12	54	87	541
13	54	195	346
14	54	180	166
15	54	165	1



Maintain Price | 10% | 20% | 40%

Your revenue: $115,146, Perfect foresight strategy: $115,146, Difference: 0.0%

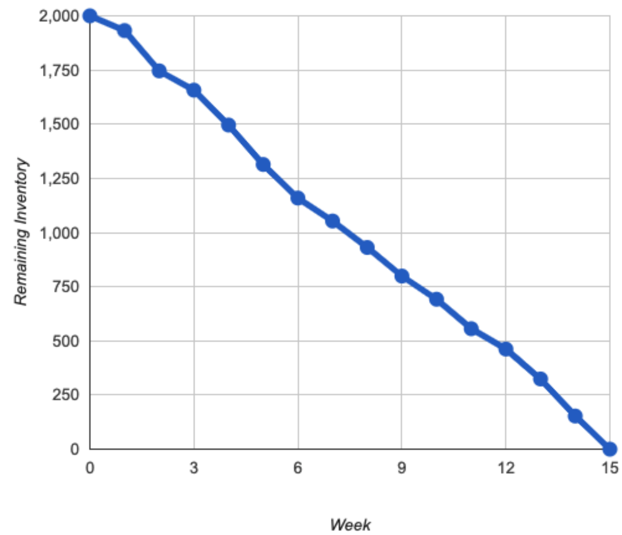
4.2 Initial Sales = 68

Repeat with initial price $60 as fixed by the question but initial sales 68 as randomized.

```
     Variable              X
-------------------------
       price_1            60
       price_2            48
       price_3            48
       price_4            48
       price_5            48
       price_6            48
       price_7            48
       price_8            48
       price_9            48
      price_10            48
      price_11            48
      price_12            48
      price_13            36
      price_14            36
      price_15            36
```

The sample decision variables of price per week implies a markdown strategy of 20% at week 2 and 40% at week 13. Let’s see how it can fulfill demand and achieve the highest possible revenue with perfect foresight strategy.

Week	Price	Sales	Remaining Inventory
1	60	68	1932
2	48	186	1746
3	48	89	1657
4	48	161	1496
5	48	182	1314
6	48	155	1159
7	48	106	1053
8	48	122	931
9	48	132	799
10	48	107	692
11	48	136	556
12	48	94	462
13	36	138	324
14	36	171	153
15	36	153	0



Maintain Price | 10% | 20% | 40%

Your revenue: $91,272, Perfect foresight strategy: $91,296, Difference: 0.0%

5. Strategies and Business Implications

With the revenue optimization model in place, dynamic markdown strategy and initial pricing strategy can be formulated by changing quantity (for various popularity) at fixed price level and changing price level at fixed demand quantity respectively.

5.1 Dynamic Markdown Strategy

Given the list price is fixed at $60, we are able to test on different initial sales scenarios, different popularity and their optimal price p at week w to generalize 6 potential markdown strategies, from more aggressive to more conservative, for the fashion apparel marketing manager in the retailer game, constrained to 2,000 inventory items and non-reversible 10%/20%/40% discounts. (see price per week for all initial sales scenarios in the appendix detailed markdown strategy)

Initial Sales (sale)	Proposed Markdown Strategy	Decision		
		w1	w2	w3+
sale < 52	40% at week 2	Keep	40%	Keep
52 ≤ sale < 75	20% at week 2, followed by 40% from week 3 or after	Keep	20%	40%
75 ≤ sale < 77	20% at week 2	Keep	20%	Keep

$77 \leq$ sale < 98	10% at week 2, followed by 20% from week 3 or after	Keep	10%	20%
$98 \leq$ sale < 131	10% from week 2 or after	Keep	10%	
sale ≥ 131	No markdown is needed	Keep	Keep	Keep

We can consider two extreme scenarios where initial sale is far below or above expectations to understand the pressing need to discount as much as we could to cut loss or no need at all to apply any markdown strategy respectively. The higher the initial sales, or the higher the market reaction in product take-up, the less discount and longer price hold before discounting could gauge demand efficiently and maximize revenue. Basically, the markdown strategy for 15 weeks can be released after week 1 right away for execution.

It is also worth to note that the optimal strategy in most cases consists of multi-step discount which immediately discounts in week 2 to address initial market reaction, followed by further price reduction in subsequent weeks (from week 3 or after) until sale is not catching up to create demand lift at a more granular level. Applying this concept, we can hypothesize a potential enhancement to take real-time performance data as time dimensional variables and re-run Gurobi models to determine the next-best-action more accurately and dynamically. Yet, considering the complexity of web scraping and good enough results we achieved, our team decided to keep this extension in the backlog.

5.2 Initial Pricing Strategy

Apart from popularity or market reaction upon introduction of the brand new fashion apparel, we want to know if the initial list price also plays an important role in determining the subsequent markdown strategy. In the below table, we ran a few experimental trials with the Gurobi model to understand the optimal markdown strategy with changing price level, assuming demand quantity is fixed at 100 (not definable in the retailer game).

Initial Price (p1)	Initial Sales	Proposed Markdown Strategy	Decision		
			w1,2	w3	w4+
40	100	10% at week 3	Keep	10%	Keep
50	100	10% at week 3	Keep	10%	Keep
60	100	10% at week 3	Keep	10%	Keep
70	100	10% at week 3	Keep	10%	Keep
80	100	10% at week 3	Keep	10%	Keep

90	100	10% at week 3	Keep	10%	Keep
100	100	10% at week 3	Keep	10%	Keep
110	100	10% at week 3	Keep	10%	Keep
120	100	10% at week 3	Keep	10%	Keep

It is observed that given discounts at 10%/20%/40% and demand rate with respect to price are fixed, the optimal markdown strategy in terms of percentage discount remains the same across various initial prices. This aligns with our demand estimation approach that demand lift with respect to price is homogeneous across items - not only reflected in subsequent price changes but also the initial pricing. Visually, the same demand curve moves and scales along decreasing price and increasing demand in demand estimation.

Overall, the more important decision and final retailer game quick wins strategy would be the dynamic markdown based on initial demand. In reality, the fashion apparel marketing manager might also want to consider different combinations of markdown options for better efficiency to meet changing demand, allow flexible mark-up and mark-down for sensitivity to adapt to market fluctuation, and capture demand data in a more timely fashion for potential demand rate adjustments in the assumption as well as markdown strategy optimization model re-un.

6. Conclusion

In this report, we attempted to estimate the demand based on historical data to derive demand rate as factor multiplier for predicted demand with respect to price, which is homogeneous across items / games. The revenue optimization tool using Gurobi provides a Mathematical approach to resolve the non-linear problem of maximizing revenue with random demand. It can be concluded that initial sale or popularity (but not list price) is crucial in determining the dynamic markdown strategy in the beginning of the retailer game, assuming discounts and demand lift factors are fixed.

7. Appendix

7.1 Github Code Repository

https://github.com/anarlewang/revenue_markdown_strategy

Plots of the Markdown Strategy: Plot strategy for each item

Plots of the Sales Difference After Markdown: Plot sales difference for each item

7.2 Detailed Markdown Strategy

Assumptions

- Inventory capacity: 2,000 units
- Selling period: 15 weeks
- Discount rates: 10%, 20%, 40%
- Unit sale price at initial: $60

Legend - price and discount rates

No	10%	20%	40%
$60	$54	$48	$36

Scenarios - initial sale and markdown strategy

Initial Sales (sale)	Proposed Markdown Strategy	Price p at week w														
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	40% at week 2															
…																
50																
51																
52	20% at week 2, followed by 40% from week 3 or after															
53																
54																
55																
56																
57																
58																
59																

60																
61																
62																
63																
64																
65																
66																
67																
68																
69																
70																
71																
72																
73																
74																
75	20% at week 2															
76																
77	10% at week 2, followed by 20% from week 3 or after															
78																
79																
80																
81																
82																
83																
84																
85																
87																
88																
89																
90																
91																
92																
93																
94																
95																
96																
97																
98	10% from week 2 or after															
99																
100																
101																
102																

103	
104	
105	
106	
107	
108	
109	
110	
111	
112	
113	
114	
115	
116	
117	
118	
119	
120	
121	
122	
123	
124	
125	
126	
127	
128	
129	
130	
131	No markdown
…	
2000	