Assumptions and the Mathematical Model for Surplus Unit Allocation Problem – by Baris Tezcan

Executive Summary: In this project, we consider how many surplus units we need to produce for each product to prepare against future demand uncertainty. We leverage our forecasts and product substitutability to decide on the surplus unit amounts we need to manufacture for each product to **i) maximize our profit and ii) minimize the substituted purchases.** We acknowledge that while the substitutability of our products is important, we need to provide what the customers are looking for as much as possible to maintain our brand image and customer relations and to reduce customer churn.

High-Level Summary for Analytics-Focused People: I approach the problem as a two-stage stochastic program, where the first stage decision is to decide on how much surplus to produce for each product, and the second stage decision is the assignment (deciding which product to sell after observing the demand) of products given realized demands (a scenario) of the products. I have used sampling from the given distributions to create multiple scenarios. This approach is known as sample average approximation.

Business assumptions are first maximizing expected profit across all scenarios and then minimizing the maximum substitution amount over the groups to preserve customer satisfaction (and hopefully reduce churn in the next buying season). I also assumed if a product is produced, it should be first used for its demand before substitution. While we can use substitutes, I believe it would hurt the brand's reputation to lack certain products and only have a higher margin version available over a few seasons, so we want to minimize that while maintaining the profits. For this, I have used hierarchical multi-objective optimization. These assumptions are made as quantifying the monetary value of lost sales or substituted products is hard (sometimes impossible) to gather. Still, we can taper the effects by using a second objective rather than assigning speculative weights and lost sales cost amounts (especially for 500 products). If numerical monetary estimates were available for lost sales and the 'hidden' cost of substitution, they could be incorporated into my model with ease.

Details

Sampling: As part of the sample average approximation, I use numpy.stats library to sample and generate the demand as instructed by the project document. The nice thing about sample average approximation is it gives runtime–accuracy trade-off while being flexible with the type of distributions. If the distribution changes next period, my approach is modular enough to handle that. Also, this approach allows me to use linear programming without getting into constrained nonlinear programming. My test runs showed that scenarios around 300 are sufficient as the objective function wasn't drastically changing with additional scenarios. Also, the distributions didn't seem to have huge tails so a risk-neutral approach is selected, in other words I only focus on the average for simplicity and do not consider concepts such as CVaR or VaR as I have no input regarding organizations risk perception and behavior and the distributions seem well-behaved.

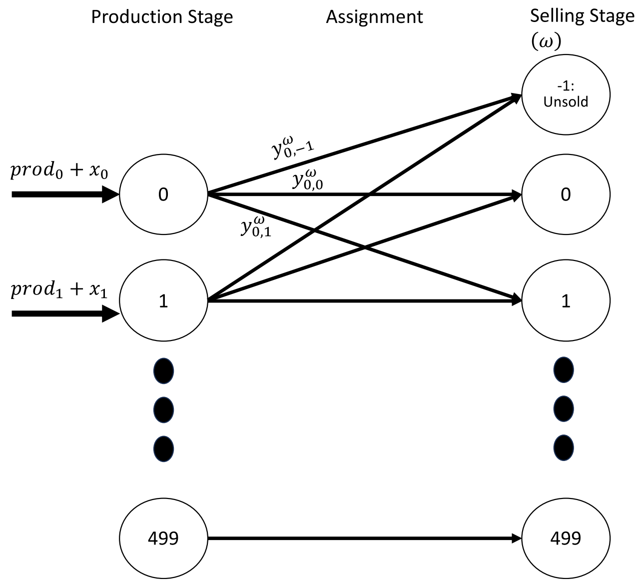

Figure 1:Produce and Sell Network for a Given Scenario

Network and Variables: We can model all the given requirements and my additional assumptions using a linear program. Consider the network in Figure 1, on the left-hand side of the bipartite network we have a production node for each product. The incoming flow to these nodes are the initial production level given as data ($prod_i$) and the surplus amount we want to decide for product i, x_i. We connect each production node to a selling node based on the substitutability group they belong to and also an unsold node, indexed by -1. This node helps track of unsold products and cost calculations. The variable $y^{\omega}_{i,j}$ denoted the amount of product i sold for product j in scenario ω.

Cost and Profit: We consider a cost of COGS(i) for amounts traveling arcs $(i,-1)\ \forall i \in Products$ to capture the cost of producing and not selling in each scenario. We also gain a profit margin, margin(i) for flows traveling across arcs $(i,j) \forall\ i,j \in Products$. There is a small assumption of selling the product i with it's own margin rather than j's. I can see the other case happening in the business case as well, but it's very easy to change in the model and not critical.

Flow Capacities: We use the capacities given in the input data to bound surplus production variables x_i. I have also added an additional business constraint forcing the lower bound of arc (i,i) for scenario network ω to be $\min\ (prod_i, dem^{\omega}_i)$. This makes sure that the initial production tries to cover the demand as much as it can and the model decides to fulfill the remaining part. If this constraint is not added, we see some very greedy behavior where initial production is fully disregarded and the demand is satisfied with a higher margin substitute. While this might work for a single season, customers will not be happy with this greedy strategy, and there will be consequences.

Auxiliary average substitution variables: For ease of notation and discussion, I have defined a variable holding the value of expected substituted sales for each group, s_{aux_g}. s_{max} variable holds the maximum expected substituted sales over groups.

Linear Program for SUA

Sets:

$$Products: Set\ of\ products, \{0,1,\dots,499\}$$
$$Groups: Set\ of\ groups, \{0,1,\dots,197\}$$

$$I(i): Set\ of\ arcs\ incoming\ to\ node\ i$$
$$O(i): Set\ of\ arcs\ outgoing\ from\ node\ i$$
$$G(g): Set\ of\ nodes\ that\ is\ in\ group\ g$$
$$S: Set\ of\ scenarios$$

Parameters:

$$margin: margin\ for\ product\ i$$
$$cost: COGS\ for\ product\ i$$
$$caps_i: surplus\ production\ upper\ bound\ for\ product\ i$$
$$prod_i: initial\ production\ for\ product\ i$$
$$dem_i^{\omega}: demand\ i\ for\ product\ i\ in\ scenario\ \omega$$
$$\alpha: coefficient\ for\ the\ total\ surplus\ production\ percantage\ compared\ to\ initial\ production$$

$$\max f_1: \sum_{i \in Products} \sum_{j \in O(i) \setminus \{-1\}} \frac{1}{|S|} * \sum_{\omega \in S} margin_i * y_{i,j}^{\omega} - \sum_{i \in Products} \sum_{\omega \in S} cost_i * y_{i,-1}^{\omega}\ (1a)$$
$$\min f_2 : s_{max} (1b)$$
$$s.t.$$
$$x_i \leq caps_i\ \forall i \in Products\ (2)$$
$$\sum_{i \in I(j)} y_{i,j}^{\omega} \leq dem_j^{\omega} \forall \omega \in S, \forall j \in Products\ (3)$$
$$x_i + prod_i = \sum_{j \in O(i)} y_{i,j}^{\omega}\ \forall\ \omega \in S, \forall i \in Products\ (4)$$
$$\sum_{i \in Products} x_i \leq \sum_{i \in Products} prod_i * \alpha\ (5)$$
$$s_{aux_g} = \frac{\sum_{i \in G(g)} \sum_{j \in O(i) \setminus \{i,-1\}} \sum_{\omega \in S} y_{i,j}^{\omega}}{|S|}\ \forall g \in Groups\ (6)$$
$$s_{aux_g} \leq s_{max} \forall g \in Groups\ (7)$$
$$x_i, y_{i,j}^{\omega}, s_{aux_g}, s_{max} \geq 0\ \forall i,j \in Products, \forall \omega \in S\ (8)$$

(1a) is maximizing the profit and (1b) is the minimizing maximum substitution amount over groups. They are hierarchical objectives where maximizing the profit has the priority. (2) limits the surplus production for each product. (3) makes sure that only up to realized demand can be satisfies for each product in a scenario. (4) ensures there is a flow balance between initial production and surplus production and assignment for sales for each product in each scenario. (5) ensures that the total surplus production limit doesn't exceed the alpha fraction of the whole initial production. (6) calculates the substituted amount of sales within each group and (7) bounds each of these with s_{max}. (8) is the domain of variables.

Results: Results on how much surplus to have from each product can be seen by running the script or evaluating the generated csv's/dash_table. These tables contain some additional statistics as well.