<u>CUSTOMER LIFETIME VALUE STRATEGY FOR PYTHON LOGISTICS</u>

SHREYA MURALEEDHARAN SWAPNA

ANS 1) RFM Model-Based Scoring Criteria

Recency measures how recently a chain placed a delivery order. Chains with recent activity—especially for perishables—indicate higher urgency. Scoring is assigned as follows:

- **Score 5** → Delivery within 0–7 days
- **Score 4** → Delivery within 8–14 days
- **Score 3** → Delivery within 15–21 days
- **Score 2** → Delivery within 22–30 days
- **Score 1** → Delivery more than 30 days ago

These ordering behaviours place supermarkets with more active, and time-sensitive demand at a higher priority.

Frequency reflects the number of orders placed in the past three months divided by the number of stores in the chain, matters of scale to ensure fairness amongst chains of different sizes. Scoring is established as follows:

- **Score 5** → 1.5 or more orders per store
- **Score 4** → between 1.2 and 1.49 orders per store
- **Score 3** → between 0.8 and 1.19 orders per store
- **Score 2** → between 0.5 and 0.79 orders per store
- **Score 1** → less than 0.5 orders per store

This reflects the level of regularity of dependence, by each chain, on Python Logistics.

Monetary reflects the total logistics revenue generated by each chain, but takes into account different weighted revenue depending on how lucrative the order is (ie. tendered refrigerated perishables typically yield higher margins than dry good delivery). Scoring is established as follows:

- **Score 5** → $100,000 or more
- **Score 4** → $75,000 to $99,999
- **Score 3** → $50,000 to $74,999
- **Score 2** → $25,000 to $49,999
- **Score 1** → Less than $25,000

This ensures that chains generating higher and more profitable revenue are prioritized.

This tailored RFM model enables Python Logistics to identify and rank its most valuable supermarket chains based on timeliness, frequency, and profitability—adjusted for business context. Hence, a data-driven incentive to assist in truck assignment, properly targeted marketing, and develop longer-term relationships, while improving the overall use of their limited logistics resources.

Metric	Criteria	Score
Recency	Delivery within 0–7 days	5
	Delivery within 8–14 days	4
	Delivery within 15–21 days	3
	Delivery within 22–30 days	2
	Delivery more than 30 days ago	1
Frequency	1.5 or more orders per store	5
	Between 1.2 and 1.49 orders per store	4
	Between 0.8 and 1.19 orders per store	3
	Between 0.5 and 0.79 orders per store	2
	Less than 0.5 orders per store	1
Monetary	Revenue of $100,000 or more	5
	Revenue between $75,000 and $99,999	4
	Revenue between $50,000 and $74,999	3
	Revenue between $25,000 and $49,999	2
	Revenue less than $25,000	1

Table 1: RFM Scoring Criteria for Supermarket Customers

ANS 2): Applying Choice Models for Granular Targeting

Building on the RFM segmentation, Python Logistics can use choice models—such as logistic regression or multinomial logit—to predict the likelihood of future engagement from supermarket chains. The models contain recency of perishable orders, frequency adjusted for stores, and revenue from high-margin deliveries as variables to predict the probability of a customer's continued profitable engagement with Python.

By reviewing historic behaviours and characteristics, each customer can be assessed with a choice score representing the likelihood of future engagement within a value score. Customers can then be ranked and segmented into targeted groups based on their score, offering a more nuanced view than RFM tiers alone.

This enables strategic targeting: top-scoring clients can receive priority truck allocation and personalized marketing, while mid-tier clients may benefit from service incentives, and low-scoring clients can be engaged minimally. Trade-offs of time and resources will be most productive when they focus on customers with a greater likelihood of returns based on RFM and choice scores.

In summary, choice models transform RFM scoring into a probability-driven targeting tool, enabling Python Logistics to align operational and marketing resources with the customers most likely to drive long-term value.

Customer Tier	Probability Range	Targeting Strategy
Tier 1	0.90 and above	Priority truck allocation, personalized service
Tier 2	0.70 – 0.89	Flexible delivery slots, targeted offers
Tier 3	0.40 – 0.69	Standard service and scheduling
Tier 4	Below 0.40	Minimal engagement, reserve resources

Table 2: Choice Model-Based Decile Segmentation

ANS 3) Customer Evolution and Lifetime Valuation Using Transition Matrix

Python Logistics can use a transition matrix to model how customers evolve across behaviour-based segments to evaluate the long-term value of supermarket chains. They can then apply discount rates to make reasonable estimates of the present value of future profits, which is essential for computing Customer Lifetime Value (CLV).

A transition matrix quantifies the likelihood of a customer moving from one segment to another over time, such as from "Gold" to "Silver" or from "Active" to "Inactive." These segments are defined using RFM logic: recent perishable orders (Recency), frequency normalized by store count (Frequency), and revenue from high-margin deliveries (Monetary). For example, a customer may have a 50% probability of remaining Gold, a 30% probability of moving to Silver, and a 20% probability of falling into an even lower segment next quarter.Python Logistics can estimate these probabilities using historical quarterly data. Over multiple periods, the transition matrix makes it possible to predict each customer's likely future state, thus making the evolution of the customer measurable, by segment. The final "Lost" or "Inactive" segment is treated as an absorbing state—customers here no longer contribute revenue.

Once expected revenues are estimated for each future state, discounting is applied to adjust for the time value of money and risk. For instance, profits from year two may be multiplied by a discount factor like 0.83 (at 10%) to get the present value. This process allows for near-term revenue to be weighted more than that associated with uncertain future income.

Python Logistics can calculate risk-adjusted CLV by combining segment transitions and discounted revenue. This supports smarter decisions—like retaining high-value clients, improving service to grow marginal ones, or de-prioritizing low-CLV accounts—ensuring resources are aligned with evolving customer potential.

From \ To	Gold	Silver	Inactive
Gold	50%	30%	20%
Silver	25%	50%	25%
Inactive	0%	10%	90%

Table 3: Sample Transition Matrix for Supermarket Segments

References:

• Open AI. (2025). *ChatGPT* (GPT-4o) [Large Language Model]. https://chatgpt.com

• Lilien, G. L., Rangaswamy, A., & De Bruyn, A. (2017). *Principles of marketing engineering and analytics* (3rd ed.). DecisionPro, Inc.

• Iacobucci, D., & Churchill, G. A. (2018). *Marketing research: Methodological foundations* (12th ed.). CreateSpace Independent Publishing Platform.

• Lilien, G. L., Kotler, P., & Moorthy, K. S. (2007). *Marketing models*. Trafford Publishing.