# Incorporating Transferability in Assortment

October 16th, 2025

# Agenda

- Status quo and its limitations
- Prerequisites
  - Migration to GurobiPy
  - Adding transference to the model
- Measuring Transference
  - Causal estimation using natural experiments
    - Solution Overview
    - Data requirements
    - Measuring second order effects
    - Tentative timeline
    - FAQs

# Prerequisites

# Incorporating Transference| Modeling Implications

- Recall the previous model's formulation:

$$\max \sum_i p_i x_i - \sum_{i \in \text{current set}} (1 - \tau_i) \cdot p_i^{\text{current}} \cdot \mathbb{1}[x_i = 0]$$

- This is a linear problem that can be solved using Gurobi or any open-source solver.
- Incorporating transference introduces the term $x_i x_j$ which makes the problem quadratic.

$$\max \quad \underbrace{\sum_i p_i x_i}_{\textbf{(1) Own productivity}} + \underbrace{\sum_i \sum_{j \in \mathcal{N}(i)} T_{i \to j}(1 - x_i) x_j p_j}_{\textbf{(2) Transferability (directional, exogenous)}}$$

- This takes the run time from a few min to several hours.
- And requires migration to GurobiPy

# Estimating Transference Using Natural Experiments



CVSHealth.

# Natural Experiment| Solution In A Nutshell

- Treat Temporary stockouts (OOS) of i act as quasi-exogenous removals from the local choice set.
- Conditioning on customers whose last CDT purchase was i, and restricting to visits during i’s OOS episodes,
- We observe their next choice among available j’s and the outside option (walk).
- With adequate controls (price/promo of j, seasonality, store×week fixed effects), we interpret the observed reallocation as the causal transference pattern for i.

# Natural Experiment| Problem Setup Simplified

- Customer A walks in at week t, and purchases SKU 1 from CDT* *F*.
- At week t+n, the customer makes a subsequent purchase when i is unavailable:
  - ❑ If SKU B from CDT* *F* exists in the customer's basket:
    - The two products A and B are considered substitutable.
  - ❑ If the basket does NOT contain any item from CDT* *F*:
    - 🚶The customer is assumed to have walked.
- For each product, we only keep the top *n* alternatives.
  - ❖ Determining n can be
    - Data Driven
    - Rule Based
    - Hybrid

# Natural Experiment| Features and Setup

- Unit of observation: customer – SKU – week
- Features:
  - ❑ To and from SKU price, promo depth, size, brand, SKU2VEC/description embeddings).
  - ❑ Customer taste: represented by a time-decayed profile derived from their purchase history in the CDT;
  - ❑ Store context such as cluster and operational fixed effects.
  - ❑ Etc.
- Starting Universe: CDT – ATTR1 – ATTR1 + ATTR2

# Natural Experiment| Built On Our Existing Models

- This approach leverages all our developed models:
  - ✓ need states
  - ✓ store clustering
  - ✓ SKU2VEC embeddings
  - ✓ product description embeddings

# Natural Experiment| Blueprint

## Notation and sample

CDT $S$; stores $s$; weeks $t$; users $u$. Availability indicator $A_{kst} \in \{0, 1\}$. Define the visit set for focal $i$:

$$\mathcal{V}_i = \{(u, s, t) : \text{ last CDT purchase} = i, \ A_{ist} = 0, \text{ user visits CDT at } (s, t)\}.$$

Local choice set $C_{st} = \{j \in S : \ A_{jst} = 1\} \cup \{0\}$ (outside option 0).

## Model (multinomial, with taste & context)

For each $(u, s, t) \in \mathcal{V}_i$ and $j \in C_{st}$,

$$U_{ujst} = \alpha_j + \beta^\top X_{ujst} + \gamma^\top Z_{uij} + \eta_{st} + \varepsilon_{ujst}, \tag{1}$$

$$P_{ujst} = \Pr(\text{choose } j \mid C_{st}) = \frac{\exp(U_{ujst})}{\sum_{\ell \in C_{st}} \exp(U_{u\ell st})}. \tag{2}$$

where

$X_{ujst}$: alternative-specific covariates (price, promo, size/brand, embeddings, store attributes).
$Z_{uij}$: pairwise similarity between $i$ and $j$ (embedding distances, format match, brand lineage).
$\eta_{st}$: store×week fixed effects.
Options: (a) Mixed logit with random coefficients; (b) segment mixture on taste.

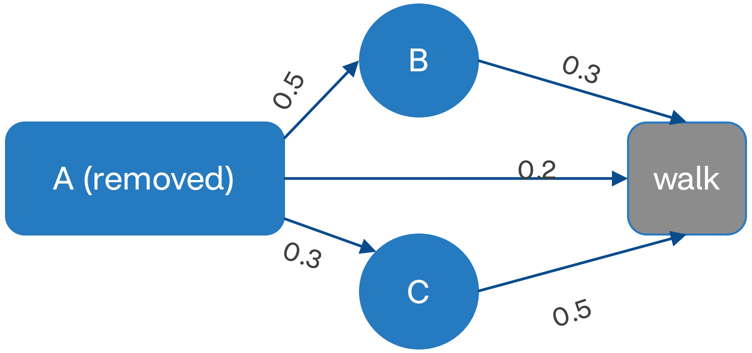
# Natural Experiment| Going Beyond First-Order Transference

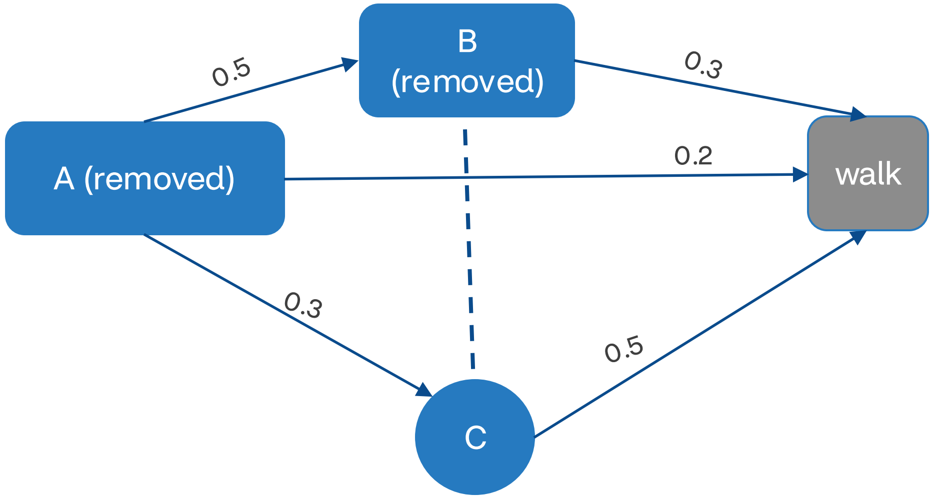
- The transference model measures transfer from A to B, assuming everything else remains the same.
- In reality, multiple SKUs are removed during resets.
- This can underestimate retained sales and distort productivity when substitution chains are deep.
- Incorporating the interaction of n SKUs takes the problem from a quadratic format to an n-degree polynomial.

# Natural Experiment| Going Beyond First-Order Transference

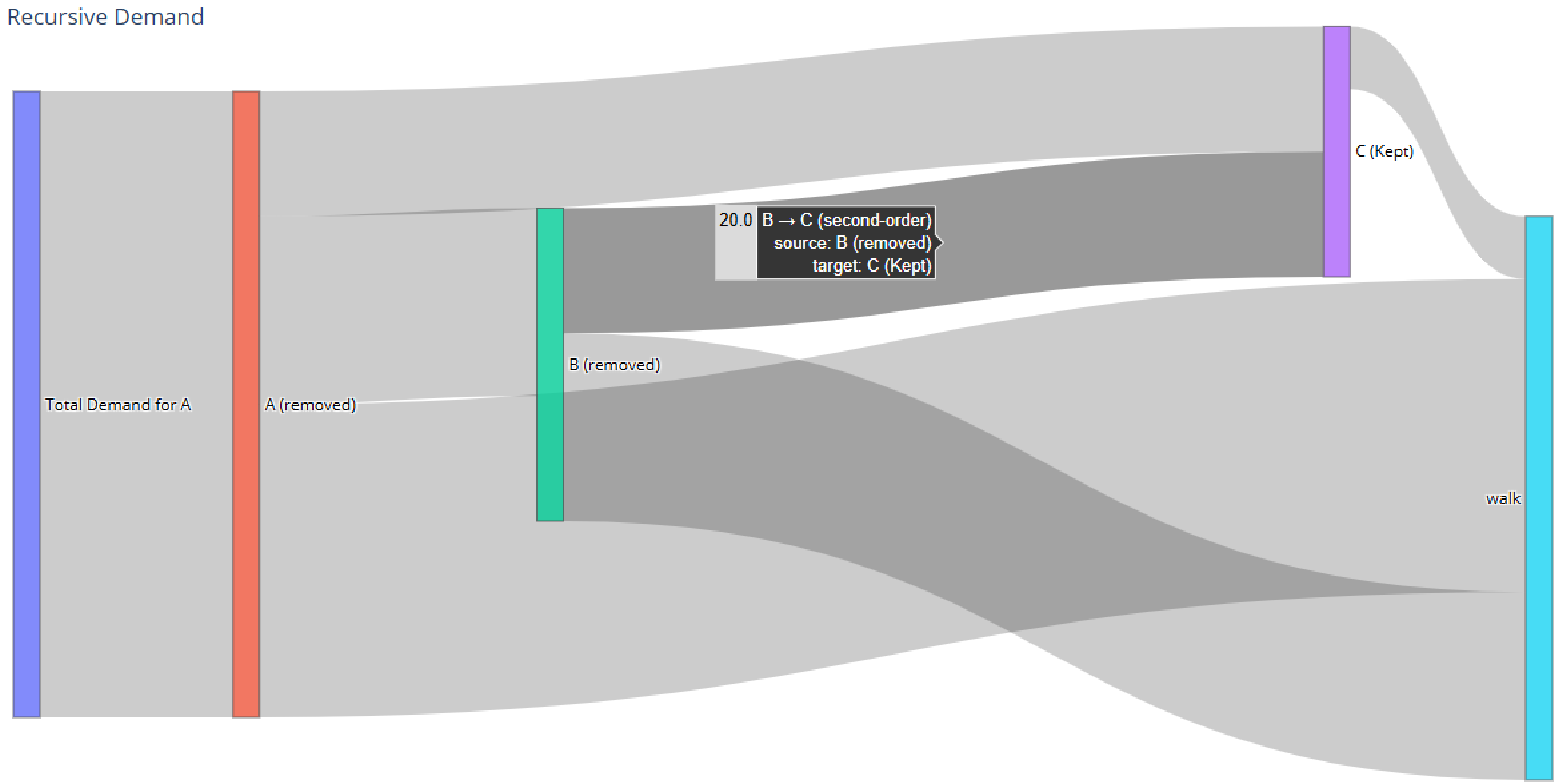
- Removing multiple SKUs takes some of the demand from the first alternative to the second priority alternatives.
  ➔ Some customers would have chosen B, but now that B is not available, a fraction would choose C



# Natural Experiment| Going Beyond First-Order Transference



CVSHealth.

# Natural Experiment| Capturing Retained Demand With Recursive Flows

- We use Markov chains to simulate how lost demand continues to reroute until it reaches a product that remains or walks away.
- This creates and “absorbing” network where kept SKUs capture cascading demand.
- The resulting correction tells us how much demand truly stays within the assortment.
- The correction quantifies how complete the absorption really is.

# Natural Experiment| Capturing Second Order Effect: Approach

**Workflow:**

1- Optimize assortment with the current model.

2- Run post optimization recursive transference Check.

3- If

- gap < 5% ➔ proceed
- gap >5% ➔ apply extended objective

$$\max_{x} \underbrace{\sum_{i} p_i x_i}_{\text{own productivity}} + \underbrace{\sum_{i,j} T_{i\to j}(1-x_i)x_j p_i}_{\text{first-order transfer}} + \underbrace{\sum_{i,j,k} T_{i\to j}T_{j\to k}(1-x_i)(1-x_j)x_k p_k}_{\text{second-order transfer}}.$$

**Benefits:**

1- Incorporates multi-SKU removal effects

2- Leverages the fact that the model by default prioritizes high absorbing SKUs.

3- Accounts for second order effects surgically and only when assortment is likely to change.

4- Keeps solver performance fast and the model explainable

# Natural Experiment| A Small Example

| SKU | Base Productivity | Decision |
|---|---|---|
| A | 100 | Remove |
| B | 80 | Remove |
| C | 90 | Keep |

| From ➔ To | A | B | C | Walk |
|---|---|---|---|---|
| A | - | 0.5 | 0.3 | 0.2 |
| B | 0.2 | - | 0.4 | 0.6 |

What Model Does:

- A is gone ➔ C gets 0.3*90 =27
- B is gone ➔ C gets 0.4*90 =36
- C Stays ➔ 90 stay

Total: 27+36+90=153

What Markov measures

- A is gone **&** B is gone ➔ C gets 0.5*0.4*90 + 0.3*90 = 45
- B is gone **&** A is gone ➔ C gets 0.2*0.3*90 + 0.4*90 = 41.4
- C Stays ➔ 90 stay

Total: 45+41+90=176

**Gap vs the model:** 176/153-1 = 15%

# Natural Experiment| FAQ

| Question | Response |
| --- | --- |
| 1- How does the OOS data look like and how confident are we in it? | The OOS data is at store-SKU-Week level and identifies the number of days a SKU has been OOS during that week. We are working with the inventory team to obtain a quantified assessment of the data quality. |
| 2- Let's say two highly substitutable items are not available in the same store. Wouldn't the model underestimate their transference? | When two items i and j are rarely co-carried in the same store, we still infer $T_{i \to j}$ from other stores where j is available by leveraging feature-driven substitution structure (brand/size proximity, embeddings) and hierarchical shrinkage. For stores where j is not carried at scoring time, we re-normalize $T_{i \to j}$ over the locally available set (see Appendix) |
| 3- CDT and even ATTR1+ATTR2 make a huge universe of alternatives. How do we make the data size reasonable? | For each purchase, we sample only a subset of alternative SKUs as negative observations. The sampling can be weighted using SKU2VEC and/or description embedding similarity to the focal SKU. |

# Natural Experiment| FAQ Cont’d

| Question | Response |
|---|---|
| 4- Transference from A to B is not necessarily equal to transference from B to A. For example, a national brand vs store brand. How do you address that? | The model takes this into account, and the transference matrix is not symmetric. |
| 5- How do you account for customer heterogeneity? | Store fixed effects (for which we will leverage the clustering algorithm features) enables us to control for both customer heterogeneity, and store differences. Alternatively, the modeling can be done at cluster level. |
| 6- Can the model’s inferences really be interpreted as causal? What are the limitations? | Although OOS provides a clean natural experiment, it may understate or overstate long-run behavior under a planogram reset. In OOS episodes, some customers may defer purchase until i returns, whereas permanent resets force immediate re-optimization of preferences. This introduces bias relative to steady-state behavior; we flag this when applying T to long-horizon decisions and complement with robustness checks (event studies, longer OOS spells). |

# Natural Experiment| FAQ Cont'd

| Question | Response |
|---|---|
| 7- A customer's subsequent purchases do not necessarily prove substitutability. How does this approach address that? | We assume by looking at a large set of transaction data over a year, repeating switching patterns across a large set of customers reveal substitutability. The core argument is that this approach does not have a systematic bias, and while it might over or underestimate transference in some instances, its error is random. |
| 8- CDT is not truly a transfer garden. How do you define a transfer garden that is applicable to all categories? | The short answer is we don't. CDT is not the transfer garden; it is a starting point for the model to determine what the best transfer garden for each SKU is. Once we have estimated transference from a SKU to all other SKUs, we keep a maximum of n substitutes for each SKU. These 5 may be in the same need state, or share the same ATTR1 with the focal product, or they may not. We have a comprehensive EDA on that. |
| 9- What about new SKUs? | The model is able to estimate transference 'to' new SKUs by leveraging feature-driven substitution structure (brand/size proximity, embeddings). There is no transference 'from' new SKUs. |



CVSHealth.

# Natural Experiment| FAQ Cont'd

| Question | Response |
|---|---|
| 10- Have you tried starting with a granularity level other than CDT? | Yes. We have tried several alternative granularity levels including need state, Attr1, and Attr1+Attr2. We judge the different candidates based on code efficiency, run time, and sanity checks on the results. Our initial results on test categories suggest Attr1+Attr2 might be the most suitable granularity.. |
| 11- What about someone who buys for their family? Wouldn't your model wrongly suggest that a women shampoo is a substitute for a men shampoo? | This approach relies on the frequency of occurrence. Yes, there are instances in the data when the shopping behavior of the customer's might result in swaps that do not belong to the same transfer garden, but they are not frequent enough to show up in our top list. |
| 12- How do you determine if the model's results make sense? | Determining the accuracy of this approach relies on a set of predetermined sanity checks as well as human review. We do not expect 100% perfect results, but we expect model's errors to be small in scale, and detectable. |

# Appendix

# Deploying to Local Assortment

# Natural Experiment| Deployment to Local Assortment

Let $A_s \subseteq S$ be the set of SKUs carried in store $s$. For a focal $i$ with national row $T_{i\to\cdot}$, define

$$S = \sum_{k\in A_s} T_{i\to k}, \quad U = \sum_{k\notin A_s} T_{i\to k}, \quad W = T_{i\to 0}, \qquad S + U + W = 1.$$

We map the national row to store $s$ by choosing a policy for the missing mass $U$:

$$\textbf{R1 (Pure reallocation):} \quad W' = W, \quad T^{(s)}_{i\to j} = \begin{cases} \frac{T_{i\to j}}{S}(1 - W') & j \in A_s, \\ 0 & j \notin A_s, \end{cases} \tag{3}$$

$$\textbf{R2 (Spill to walk):} \quad W' = W + U, \quad T^{(s)}_{i\to j} = \begin{cases} T_{i\to j} & j \in A_s, \\ 0 & j \notin A_s, \end{cases} \tag{4}$$

so that in both cases $\sum_{j\in A_s} T^{(s)}_{i\to j} + T^{(s)}_{i\to 0} = 1$ with $T^{(s)}_{i\to 0} = W'$. A hybrid policy with parameter $\alpha \in [0, 1]$ sets $W' = W + \alpha U$ and $T^{(s)}_{i\to j} = \frac{T_{i\to j}}{S}(1 - W')$ for $j \in A_s$. If $S = 0$, set $T^{(s)}_{i\to 0} = 1$ and all other probabilities to 0.

# Natural Experiment| Deployment to Local Assortment

### Tiny Example (Numerical)

National transference for focal $i$:

$$T_{i\to j_1} = 0.30, \quad T_{i\to j_2} = 0.20, \quad T_{i\to j_3} = 0.15, \quad T_{i\to j_4} = 0.10, \quad T_{i\to 0} = 0.25.$$

Store $s$ carries $A_s = \{j_1, j_4\}$. Then

$$S = \sum_{k\in A_s} T_{i\to k} = 0.30 + 0.10 = 0.40, \quad U = \sum_{k\notin A_s} T_{i\to k} = 0.20 + 0.15 = 0.35, \quad W = T_{i\to 0} = 0.25.$$

**R2 (Spill missing mass to walk).**

$$W' = W + U = 0.25 + 0.35 = 0.60, \qquad T^{(s)}_{i\to j_1} = 0.30, \; T^{(s)}_{i\to j_4} = 0.10, \; T^{(s)}_{i\to 0} = 0.60.$$

**R1 (Pure reallocation among available).** Scale available items by $\frac{1-W'}{S} = \frac{1-0.25}{0.40} = \frac{0.75}{0.40} = 1.875$ and keep walk unchanged:

$$T^{(s)}_{i\to j_1} = 0.30 \times 1.875 = 0.5625, \quad T^{(s)}_{i\to j_4} = 0.10 \times 1.875 = 0.1875, \quad T^{(s)}_{i\to 0} = 0.25.$$

| **Alternative** | **National $T_{i\to\cdot}$** | **$T^{(s)}$ (R2)** | **$T^{(s)}$ (R1)** |
|---|---|---|---|
| $j_1 \in A_s$ | 0.30 | 0.30 | 0.5625 |
| $j_2 \notin A_s$ | 0.20 | 0 | 0 |
| $j_3 \notin A_s$ | 0.15 | 0 | 0 |
| $j_4 \in A_s$ | 0.10 | 0.10 | 0.1875 |
| walk (0) | 0.25 | 0.60 | 0.25 |

In both policies, the probabilities sum to 1 for the local assortment.

CVSHealth

# First Alternative Approach

# First Approach| Basket to Basket Transference

- The first approach is similar and built on the existing logic
- For each **CDT**, we:
- **Identify Relevant Baskets**: Filters baskets containing products from the specific CDT
- **Temporal Sequence Analysis:** Examines consecutive shopping trips for each customer (aggregated weekly)
- **Substitution Detection:** Identifies two types of substitution patterns:
  - Complete Substitution
    - Customer completely switches from one set of SKUs to another set within the same CDT
    - **Inference**: All SKUs in basket1 can substitute to all SKUs in basket2
  - Partial Substitution
    - Customer maintains some SKUs but switches others within the same CDT
    - **Inference**: Disappeared SKUs transfer to appeared SKUs

# First Approach| Counting Transference

- Transference from SKU i to j is estimated as:

$$\text{substitution count}(i \to j)/\text{total transitions}(i)$$

- The final matrix is sparsified to keep only top *n* alternatives per SKU.
    - Can be data driven
    - Can be predetermined
    - Can be hybrid (data driven with a max)
- Transference values are calibrated using SKU walk rates so they all sum up to 1.
- Walk rate for SKU i is defined as quantity share of single item buyers of SKU i over all SKU i sales.
- The updated objective function becomes:

$$\max \quad \underbrace{\sum_i p_i x_j}_{\textbf{(1) Own productivity}} + \underbrace{\sum_i \sum_{j \in \mathcal{N}(i)} T_{i \to j}(1 - x_i) x_j p_i}_{\textbf{(2) Transferability (directional, exogenous)}}$$



CVSHealth.

# First Approach| Vs The Legacy Model

- The legacy model looks at co-purchases of non-exclusive customers (customers who buy more than 1 product through out the year, transfer rates are determined according to the following example:
  - Customer 1 buys: SKU A (5 units) + SKU J (3 units) + SKU K (2 units)
  - Customer 2 buys: SKU A (3 units) + SKU J (4 units) + SKU L (6 units)
  - Customer 3 buys: SKU A (2 units) + SKU K (8 units) + SKU L (1 unit)
  - Total co-purchases with A are:
    - J: 3+4 = 7
    - K: 2+8 = 10
    - L: 6+1 = 7
    - total (excluding A) = 24
  - Transfer percentages : 7/24, 10/24, 7/24, respectively
- Adjusts for size, trips being >= 5, churn rate, exclusivity, disc SKUs
- Excludes high spend customers

# First Approach| Vs The Legacy Model

| The Legacy Model | First Approach |
|---|---|
| Analyzes co-purchase patterns in the same time window, and focuses on con-current multi-SKU purchasing | Looks at consecutive trips, and takes a temporal sequence analysis approach |
| Walk rates are determined as 1 - (transfer rate from non-exclusive customers + churn effects). | Walk rates are determined by the quantity share of customers who only purchase a single SKU within the 'transfer garden' over all SKU sales. |
| Has heuristic methods for size-based adjustments, churn rates, cost components (shrink, write off, BTL, funding), and disc SKUs, cascade effects | Can address the churn point by looking at customers with previous data only and ignoring the first trip of new customers.<br>Does not currently address size adjustments. |
| No consecutive trip analysis | Tracks actual switching patterns and partially captures gradual transference. |
| Complex and difficult to audit, maintain, and explain | Intuitive and easy to maintain, add additional assumptions, and validate |

# First Approach| Advantages and Limitations

| ⭐ Advantages | ⚠ Disadvantages |
|---|---|
| ❑ Is the next iteration of the legacy model<br>❑ Intuitive<br>❑ Able to incorporate existing rules easily<br>❑ Efficient and scalable<br>❑ Takes a data driven approach in determining the transfer garden<br>❑ CDT can be replaced with Attr 1, need state, etc. | ❑ Cannot model the effect of promo and price changes<br>❑ Cannot model cases where substitutes are not in the same store<br>❑ Limited to repeating customers<br>❑ Does not leverage the existing rich set of features we have on SKUs and customers<br>❑ Limited to the existing observations and is not able to infer on censored data<br>❑ Second order effects |

# Second Alternative Approach

# Second Approach| ML Based Transference

- The basket analysis approach is inherently limited. Most importantly it is not able to:
  - ❑ Consider customer specific preferences
  - ❑ Generalize from limited observations
  - ❑ Take into account the effect of promos/price changes/ store specific characteristics
  - ❑ Estimate transference for new SKUs
- The ML based approach addresses these weaknesses by estimating:

$$P[\text{Customer A switching from i to j} \mid \text{SKU i,j features, Customer A's taste}]$$

- By looking at subsequent purchase behavior of the customers.

# Second Approach| Model Features

- At a high level, the model's feature can be classified into two categories:

| Category | Examples |
|---|---|
| Items' Features | • Price<br>• Promo<br>• Dimension<br>• SKU2VEC embedding<br>• Items' similarity based on embeddings<br>• Product maturity<br>• Weight<br>• etc. |
| Customer – Item interaction features | Similarity of the items to the user's taste measured using cosine similarity.<br>Users taste vector: Quantity (or $) weighted average of previous purchases within the same CDT discounted by recency of purchase |

# Second Approach| ML Based Transference

- A classifier with Platt scaling is used to estimate transference likelihood.

| ⭐ Advantages | ⚠️ Disadvantages |
|---|---|
| ❑ Leverages SKU2VEC model's existing features<br>❑ Controls for cross-SKU differences<br>❑ Creates dynamic individual user tastes<br>❑ Is able to incorporate external and user specific features<br>❑ Scalable<br>❑ Allows for starting with a large pool and uses a data driven approach to narrow down the substitutes | ❑ Limited to repeating customers<br>❑ Ignores the impact of out of stock<br>❑ Relies on observed purchases, might miss latent behavior<br>❑ Does not (currently) take customer purchase cycle into account |



CVSHealth.

# Quadratic Formulation

# Incorporating Transference| News Update

- 🥇Migration to GurobiPy is complete 🥇
  - 🏗 + 1,700 lines of code change
  - 🧪 All constraints tested
  - 👟 Optimization run time reduced by more than 60%
  - 🧑‍💻 Undergoing peer review
  - ✨ Expected to be deployed before the end of October
- 🏆Transference Parameter is added to the model🏆
  - 👷 (more on how, in the subsequent pages) 

CVSHealth.

# Incorporating Transference| Solving a Quadratic Problem

The quadratic formulation creates a Mixed Integer Quadratic Programming (MIQP) problem, which has several computational disadvantages:

- **Solver Complexity:** Quadratic solvers are less mature than linear solvers
- **Solution Time:** Quadratic problems are inherently harder to solve
- **Scaling Issues:** Performance degrades significantly with problem size
- **Memory Usage:** Quadratic terms require more memory to store and process

☢ Our current tests shows that switching from walk rate to transference increases the model run time from 3 minutes to +7 hours.

➢ WHAT CAN WE DO?! ➔ McCormick envelopes to the rescue!

# Incorporating Transference| Solving a Quadratic Problem

- To convert the quadratic problem to a Mixed Integer Linear Programming (MILP) problem, we use McCormick linearization to handle the product term $\left(1 - S_{j,c,g}\right) \times S_{i,c,g}$
- Define $y_{i,j,c,g} \in \{0,1\}$ representing $\left(1 - S_{j,c,g}\right) \times S_{i,c,g}$
- Add the following constraints:

$y_{i,j,c,g} \leq \left(1 - S_{j,c,g}\right)$ If j is selected, y must be 0

$y_{i,j,c,g} \leq S_{i,c,g}$ If i is not selected, y must be 0

$y_{i,j,c,g} \geq \left(1 - S_{j,c,g}\right) + S_{i,c,g} - 1$ Forces y=1 when both conditions met

# Incorporating Transference| Solving a Quadratic Problem

This allows us to replace the quadratic equation:

$$\sum\sum\sum T_{i,j} \times P_{i,c,g} \times \left(1 - S_{j,c,g}\right) \times S_{i,c,g}$$

where

With

$$S_{i,c,g} = \sum_{f \geq 1} x_{i,f,g,c}$$

$$\sum\sum\sum T_{i,j} \times P_{i,c,g} \times y_{i,j,c,g}$$

👑 Making the problem linear once again



CVSHealth.

# Incorporating Transference| Solving a Quadratic Problem

In addition to linearizing the problem, we also take several other measures into account, including:

- ❑ Solving the model with a two-stage approach:
  - ➢ First, solve the productivity maximization part of the model, ignoring transference
  - ➢ Use the solution to the first stage, as the warm start solution to the second model.
- ❑ Tuning Gurobi parameters

| Formulation | Problem Type | Solve Time | Memory Usage | Scalability |
|---|---|---|---|---|
| Original Quadratic | MIQP | ~ 7 hours | High | Poor |
| McCormick Linear | MILP | ~ 12 min | Moderate | Good |
| McCormick + Warm Start | MILP | ~ 5min | Moderate | Excellent |

# Long Term Plan

CVSHealth

# Model Updates Overview| Measuring cross product impacts

- In a slightly simplified manner, the current objective function can be formulated as:

$$\max \sum_i p_i x_i - \sum_{i \in \text{current set}} (1 - \tau_i) \cdot p_i^{\text{current}} \cdot \mathbb{1}[x_i = 0]$$

- The goal is to incorporate the interaction effects of different SKUs

$$\begin{aligned}
\max \quad & \underbrace{\sum_i p_i x_i}_{\textbf{(1) Own productivity}} \\
+ \quad & \underbrace{\sum_i \sum_{j \in \mathcal{N}(i)} T_{i \to j} (1 - x_i) x_j p_j}_{\textbf{(2) Transferability (directional, exogenous)}} \\
& \underbrace{+ \lambda_s \sum_i \sum_{\substack{j \in \mathcal{N}(i) \\ \text{Need}(i) \neq \text{Need}(j)}} c_{ij}\, x_i x_j p_j}_{\textbf{(3) Synergy across need states}} \\
- \quad & \underbrace{\lambda_c \sum_i \sum_{\substack{j \in \mathcal{N}(i) \\ \text{Need}(i) = \text{Need}(j)}} c_{ij}\, x_i x_j p_j}_{\textbf{(4) Cannibalization within need states}}
\end{aligned}$$

Focus for this iteration

CVSHealth®