1) Customer CLV Tiers — BG/NBD + Gamma–Gamma (margin-aware, H=180d, W=365d)

Goal & decision

Forecast each customer's **6-month gross margin** and map to tiers {**VIP**, **High**, **Med**, **Low**} for eligibility/budgeting and guardrails (e.g., exclude VIP from deep blanket promos).

Feature sources

- **orders (Mongo)**: `amount` , `cogs` , `tax` , `shipping` , `discount` , `returned` , `ts` → `order_margin_cents`
- **customers (Mongo)**: `first_purchase_ts` , `last_purchase_ts` (derive T and t_x)
- *(Optional)* currency normalization (store scope)

Compact formula

BG/NBD — expected future transactions on horizon H:

$$\mathbb{E}[N(H) \mid x, t_x, T] = \frac{a+b+x-1}{a-1} \cdot \frac{1 - \left(\frac{\alpha+T}{\alpha+T+H}\right)^{r+x} {}_2F_1\!\left(r+x,\ b+x;\ a+b+x-1;\ \frac{H}{\alpha+T+H}\right)}{1 + \mathbf{1}_{\{x>0\}} \cdot \frac{a}{b+x-1} \left(\frac{\alpha+T}{\alpha+t_x}\right)^{r+x}}$$

Gamma–Gamma — expected per-transaction margin (Fader–Hardie):

$$\mathbb{E}[M \mid x, \bar{m}] = \frac{(q+x)\,(\gamma + x\,\bar{m})}{(q-1)\,(p+x)} \qquad (q>1)$$

CLV assembly (no PV discount on 6–12m horizons):

$$\mathrm{CLV}_H = \mathbb{E}[N(H)] \cdot \mathbb{E}[M]$$

Full expansion (worked on one customer)

Take customer C3 from metrics:

$$\begin{aligned} \textbf{Inputs:}\quad & x=5,\ T=323,\ t_x=307,\ \bar{m}=11{,}591 \text{ cents} \\ \textbf{Hyper-params (toy):}\quad & r=0.8,\ \alpha=30,\ a=1.5,\ b=3.0,\ p=6,\ q=4.5,\ \gamma=2000,\ H=180 \end{aligned}$$

Step A — Gamma–Gamma monetary (correct)

$$\begin{aligned} \mathbb{E}[M] &= \frac{(q+x)\,(\gamma + x\,\bar{m})}{(q-1)\,(p+x)} \\ &= \frac{(4.5+5)\,\big(2000 + 5 \cdot 11{,}591\big)}{(4.5-1)\,(6+5)} \\ &= \frac{9.5 \cdot 59{,}955}{3.5 \cdot 11} = \frac{569{,}572.5}{38.5} \ \approx\ \mathbf{14{,}794.09} \text{ cents} \end{aligned}$$

Step B — BG/NBD incidence (Excel short series)

Let

$$z = \frac{H}{\alpha + T + H} = \frac{180}{30 + 323 + 180} = \frac{180}{533} = \mathbf{0.3377110694}, \quad A = r + x = 5.8,\ B = b + x = 8,\ C = a + b + x - 1 = 8.5.$$

Approximate the Gauss hypergeometric (good to $\sim 10^{-3}$ with $K \approx 10$ terms):

$${}_2F_1(A, B; C; z) \approx \sum_{k=0}^{K} \frac{(A)_k (B)_k}{(C)_k} \frac{z^k}{k!}, \qquad (U)_0 = 1,\ (U)_k = \prod_{j=0}^{k-1} (U + j)$$

Assemble numerator/denominator per BG/NBD to obtain the **exact** expectation (to 10 d.p. here):

$$\mathbb{E}[N(180)] = \mathbf{1.8268131761}.$$

Step C — CLV (exact values)

$$\mathrm{CLV}_{180} = \mathbb{E}[N(180)] \cdot \mathbb{E}[M] = 1.8268131761 \times 14{,}794.090909\ldots = \mathbf{27{,}026.04}\ \text{cents}\ \ (\$270.2604)$$

> **Rounding policy:** if you prefer rounded inputs, keep $E_N = 1.83$ and report $\mathrm{CLV}_{180} \approx 27{,}073$ c; otherwise use the exact route above and round *once* at the end (recommended).

Result & steering

- Compute CLV_{180} for all customers; cut tiers by **store quantiles** (e.g., VIP ≥ P90).
- Use tiers for **promo eligibility**, **loyalty budgets**, and **VIP guardrails** (e.g., exclude VIP from deep blanket promos unless uplift is positive).

▼ **metrics: CLV (with feature_sources)** — click to expand

```
metrics:
  clv_example:
    horizon_days: 180
    window_days: 365
    bgnbd_params: { r: 0.8, alpha: 30.0, a: 1.5, b: 3.0 }
    gammagamma_params: { p: 6.0, q: 4.5, gamma_cents: 2000 }
    customer: { id: "C3", T_days: 323, t_last_days: 307, x_repeat: 5, avg_margin_cents: 11591 }
    derived:
      z: 0.3377110694
      E_M_cents: 14794.09
      E_N_180: 1.826813
      CLV_180_cents: 27026.04

feature_sources:
  - mongo.orders
  - mongo.customers

</details>
```

2) Two-Stage Recommender (PDP/PLP) — ANN retrieval → margin-aware ranker

Goal & decision

On PDP/PLP, rank candidates to **maximize expected net gross margin per impression** subject to **stock/exposure/diversity** constraints.

Feature sources

- **product_features**: embedding, popularity, margin_pct, return_rate
- **products**: `price_cents`, `margin_cents`, `stock_units`, `category_id`, `seller_id`, `age_days`
- **interactions**: user–item events (two-tower + ranker training)

Compact formula

For item i with purchase prob p_i, margin m_i, stock-cover penalty LSP_i, return rate r_i (conditional on purchase):

$$S_i = p_i\, m_i \;-\; \alpha\, \mathrm{LSP}_i \;-\; \beta\, p_i\, r_i\, m_i$$

$$\mathrm{LSP}_i = \max\left(0,\; \frac{s_0 - \mathrm{stockCoverDays}_i}{s_0}\right)\, m_0$$

> If you track return handling cost c_{return}, use
> $S_i = p_i\, m_i - \alpha\, \mathrm{LSP}_i - \beta\, p_i\, r_i\, (m_i + c_{\mathrm{return}})$.

Choose top-N by S_i and enforce caps: **OOS filtered;** $\le$ per-category; $\le$ per-seller.

Full expansion (worked subset)

Use $s_0 = 7\ \mathrm{days},\ m_0 = 1000\ \mathrm{cents},\ \alpha = 0.5,\ \beta = 1.0.$

P6 ($p = 0.070,\ m = 1100,\ \mathrm{cover} = 12,\ r = 0.04$)

$$\begin{aligned}\mathrm{LSP}_6 &= \max\left(0, \tfrac{7-12}{7}\right) 1000 = 0\\ \text{return penalty} &= p\, r\, m = 0.07 \cdot 0.04 \cdot 1100 = 3.08\\ S_6 &= 0.07 \cdot 1100 \;-\; 0.5 \cdot 0 \;-\; 3.08 \;=\; \mathbf{73.92}\ \mathrm{cents}\end{aligned}$$

P1 ($p = 0.060,\ m = 1200,\ \mathrm{cover} = 10,\ r = 0.02$)

$$\begin{aligned}\mathrm{LSP}_1 &= 0\\ \text{return penalty} &= 0.06 \cdot 0.02 \cdot 1200 = 1.44\\ S_1 &= 0.06 \cdot 1200 \;-\; 0 \;-\; 1.44 \;=\; \mathbf{70.56}\ \mathrm{cents}\end{aligned}$$

P5 ($p = 0.045,\ m = 700,\ \mathrm{cover} = 6,\ r = 0.01$)

$$\begin{aligned}\mathrm{LSP}_5 &= \max\left(0, \tfrac{7-6}{7}\right) 1000 \;=\; 142.857\\ \text{return penalty} &= 0.045 \cdot 0.01 \cdot 700 = 0.315\\ S_5 &= 0.045 \cdot 700 \;-\; 0.5 \cdot 142.857 \;-\; 0.315\\ &= 31.5 \;-\; 71.4285 \;-\; 0.315 \;=\; -\mathbf{40.24}\ \mathrm{cents}\end{aligned}$$

(Compute similarly for remaining items; then sort by S_i and apply caps.)

Result & steering

- Render top-K, then apply post-rank caps (**seller/category diversity**, **OOS**, **low-stock demotions**).
- Monitor **conversion** & **net GM/order**; if stockouts rise, increase α or raise s_0.
- If returns incur costs, include c_{return} in the penalty term.

▼ **metrics: Recommender** (click to expand)


```yaml
metrics:
  recs_example:
    penalties: { alpha: 0.5, beta: 1.0, stock_cover_days_min: 7, m0_cents: 1000 }
    items:
      - { id: P1, cat: A, seller: S1, margin_cents: 1200, stock_cover_days: 10, return_risk: 0.02, p_purchase: 0.060 }
      - { id: P2, cat: A, seller: S2, margin_cents: 800,  stock_cover_days: 6,  return_risk: 0.03, p_purchase: 0.055 }
      - { id: P3, cat: B, seller: S1, margin_cents: 1500, stock_cover_days: 5,  return_risk: 0.02, p_purchase: 0.050 }
      - { id: P4, cat: B, seller: S2, margin_cents: 900,  stock_cover_days: 8,  return_risk: 0.02, p_purchase: 0.040 }
      - { id: P5, cat: C, seller: S3, margin_cents: 700,  stock_cover_days: 6,  return_risk: 0.01, p_purchase: 0.045 }
      - { id: P6, cat: C, seller: S1, margin_cents: 1100, stock_cover_days: 12, return_risk: 0.04, p_purchase: 0.070 }
      - { id: P7, cat: A, seller: S3, margin_cents: 500,  stock_cover_days: 5,  return_risk: 0.01, p_purchase: 0.028 }
      - { id: P8, cat: B, seller: S2, margin_cents: 2000, stock_cover_days: 9,  return_risk: 0.03, p_purchase: 0.035 }

</details>
```


3) Basket / Session-Aware Re-rank — Cart completion with guardrail

Goal & decision

Increase **attach-rate** and **net GM/order** without harming base checkout conversion. Show **N** complements only if the **guardrail** is satisfied.

Feature sources

- **carts**: `items[].product_id` , `qty` , `subtotal_cents`
- **product_graph**: `co_buy_neighbors` , `compatibility_flags`
- **products**: `margin_cents` , `stock_units` , `category_id` , shipping/size flags

Compact formula

Candidate uplift (eligible only; exclude shipping-incompatible or substitutes):

$$\Delta\mathrm{GM}_i \;=\; p_{\mathrm{attach},i} \cdot \mathrm{margin}_i$$

Guardrail across the shown set S:

$$\sum_{i \in S} \Delta\mathrm{GM}_i \;-\; \lambda\,\Delta p_{\mathrm{base}} \;>\; 0$$

with

$$\Delta p_{\mathrm{base}} \;=\; \sum_{i \in S} \mathrm{distractionRisk}_i, \qquad \lambda \;=\; p_{\mathrm{base}} \cdot \mathrm{baseOrderGM}.$$

Full expansion (worked)

From metrics: base order GM $= 10{,}000$ cents, $p_{\mathrm{base}} = 0.10 \Rightarrow \lambda = 1000$ cents.
Eligible $\Delta\mathrm{GM}$:

- A1: $0.09 \times 1200 = 108$

- B1: $0.06 \times 2000 = 120$
- W1: $0.05 \times 1000 = 50$
- A2: $0.07 \times 800 = 56$
  (S1 excluded as substitute; A3 excluded as shipping-incompatible.)

Pick **top-3** under category cap (≤ 1 per category among shown): **B1, A1, W1**.

$$\sum \Delta\mathrm{GM} \;=\; 120 + 108 + 50 \;=\; \mathbf{278}, \qquad \Delta p_{\mathrm{base}} \;=\; 0.008 + 0.005 + 0.003 \;=\; \mathbf{0.016}$$

Guardrail:

$$278 \;-\; 1000 \times 0.016 \;=\; 278 - 16 \;=\; \mathbf{262} \;>\; \mathbf{0} \;\Rightarrow\; \mathbf{SHOW}.$$

Result & steering

- Render the three suggestions. If the guardrail fails, reduce N or tighten candidate filters (e.g., raise the margin floor).

▼ **metrics: Basket (with feature_sources)** — click to expand


```yaml
metrics:
  basket_example:
    base:
      base_order_gm_cents: 10000
      base_conversion: 0.10
      lambda_cents: 1000
    candidates:
      - { id: A1, category: ACC,
          margin_cents: 1200, p_attach: 0.09,
          cos_to_hero: 0.30, ship_ok: true, distraction_risk: 0.005 }
      - { id: A2, category: ACC,
          margin_cents: 800,  p_attach: 0.07,
          cos_to_hero: 0.25, ship_ok: true, distraction_risk: 0.004 }
      - { id: S1, category: SUB,
          margin_cents: 3000, p_attach: 0.06,
          cos_to_hero: 0.85, ship_ok: true, distraction_risk: 0.015 }  # exclude (substitute)
      - { id: A3, category: ACC,
          margin_cents: 1500, p_attach: 0.05,
          cos_to_hero: 0.10, ship_ok: false, distraction_risk: 0.006 } # exclude (shipping)
      - { id: B1, category: BUND,
          margin_cents: 2000, p_attach: 0.06,
          cos_to_hero: 0.40, ship_ok: true, distraction_risk: 0.008 }
      - { id: W1, category: WARR,
          margin_cents: 1000, p_attach: 0.05,
          cos_to_hero: 0.10, ship_ok: true, distraction_risk: 0.003 }

feature_sources:
  - mongo.carts
  - mongo.product_graph
  - mongo.products

</details>
```

4) Promotion Uplift — Multi-treatment, cost-aware; persistent holdouts

Goal & decision

Assign each customer one arm {**NO_OFFER**, **10%off**, **20%off**, **FreeShip**, …} to maximize **incremental GM** under a **budget** and **per-arm caps**, while **suppressing negative uplift**.

Feature sources

- **campaign_features_at_assign**: pre-treatment feature snapshot
- **campaign_budgets**: `budget_cents` , `cap_per_treatment` , `unit_costs`
- **campaign_exposures**: persisted assignments (incl. control)

Compact formula

Per-arm incremental GM (expected):

$$\Delta\mathrm{GM}_t(x) = \tau_t(x) \cdot m_t - \mathrm{cost}_t$$

Budgeted assignment (knapsack; one arm per user):

$$\begin{aligned} \max_{y_{i,t}\in\{0,1\}} \; & \sum_{i,t} \Delta\mathrm{GM}_t(x_i)\, y_{i,t} \\ \text{s.t.} \; & \sum_t y_{i,t} \le 1, \qquad \sum_{i,t} \mathrm{EC}_t(x_i)\, y_{i,t} \le B, \qquad \sum_i y_{i,t} \le C_t \end{aligned}$$

with

$$\mathrm{EC}_t(x) = \tau_t(x) \cdot \mathrm{discountCost}_t \; .$$

Holdouts: deterministic, persisted control (e.g., stable hash on $(\mathrm{customer}, \mathrm{campaign})$); **no reassignment mid-campaign**.

Full expansion (worked on a few rows)

From metrics: **base margin** $= 3000$ c.
Arm margins after discount: $m_{10} = 2700$ c, $m_{20} = 2400$ c, $m_{\mathrm{ship}} = 2500$ c.
Discount costs per redemption: **10%→1000 c**, **20%→2000 c**, **FreeShip→500 c**
(contact/send cost already included in `discountCost`).

For $U1$ with $\tau_{10} = 0.010,\ \tau_{20} = 0.015,\ \tau_{\mathrm{ship}} = 0.012$:

- $\Delta\mathrm{GM}_{10} = 0.010 \cdot 2700 - 10 = \mathbf{17}$ c
- $\Delta\mathrm{GM}_{20} = 0.015 \cdot 2400 - 30 = \mathbf{6}$ c
- $\Delta\mathrm{GM}_{\mathrm{ship}} = 0.012 \cdot 2500 - 6 = \mathbf{24}$ c

Expected budget spend:

- $\mathrm{EC}_{10} = 0.010 \cdot 1000 = \mathbf{10}$ c
- $\mathrm{EC}_{20} = 0.015 \cdot 2000 = \mathbf{30}$ c
- $\mathrm{EC}_{\mathrm{ship}} = 0.012 \cdot 500 = \mathbf{6}$ c

$\Rightarrow$ Pick **FreeShip** for $U1$ (largest positive $\Delta\mathrm{GM}$ at the lowest expected cost).

Apply across users; sort feasible $(\mathrm{user}, \mathrm{arm})$ pairs by **value-per-cost** and respect caps and total budget B.

Result & steering

- Output treated list (with chosen arms); **NO_OFFER** for the rest.
- If KPI lags: examine **holdouts**, leakage (post-treatment features), or **over-tight caps**; check budget pacing.

▼ **metrics: Uplift (with feature_sources)** — click to expand


```yaml
metrics:
  uplift_example:
    arms:
      - { name: "PCT10",   post_margin_cents: 2700, discountCost_cents: 10 }
      - { name: "PCT20",   post_margin_cents: 2400, discountCost_cents: 30 }
      - { name: "FREESHIP", post_margin_cents: 2500, discountCost_cents: 6 }
    budget:
      total_budget_cents: 2500
      caps: { PCT10: 5, PCT20: 3, FREESHIP: 5 }
    tau_rows:
      - { id: U1,  tau_10: 0.010, tau_20: 0.015, tau_ship: 0.012 }
      - { id: U2,  tau_10: 0.005, tau_20: 0.009,  tau_ship: 0.007 }
      - { id: U3,  tau_10: 0.002, tau_20: 0.005,  tau_ship: 0.003 }
      - { id: U4,  tau_10: 0.008, tau_20: 0.012, tau_ship: 0.010 }
      - { id: U5,  tau_10: 0.006, tau_20: 0.010, tau_ship: 0.008 }
      - { id: U6,  tau_10: 0.001, tau_20: 0.003,  tau_ship: 0.002 }
      - { id: U7,  tau_10: 0.000, tau_20: 0.002,  tau_ship: 0.001 }
      - { id: U8,  tau_10: 0.004, tau_20: 0.006,  tau_ship: 0.005 }
      - { id: U9,  tau_10: 0.009, tau_20: 0.013, tau_ship: 0.011 }
      - { id: U10, tau_10: 0.003, tau_20: 0.005,  tau_ship: 0.004 }
      - { id: U11, tau_10: 0.007, tau_20: 0.010, tau_ship: 0.008 }
      - { id: U12, tau_10: 0.002, tau_20: 0.004,  tau_ship: 0.003 }

feature_sources:
  - mongo.campaign_features_at_assign
  - mongo.campaign_budgets
  - mongo.campaign_exposures

</details>
```


5) Customer Personas — HDBSCAN (primary), K-Means (fallback)

Goal & decision

Discover behavioral **personas** that differ in **engagement**, **value**, and **needs** for **messaging**, **slotting**, and **creative**.

Feature sources

- **orders** → R, F, M (margin) within window
- **events** → engagement signals (active days, opens/clicks)
- **orders by category** → row-normalized **category share** vector

Compact formula

HDBSCAN uses **mutual-reachability distance** with k-NN core distances:

$$\mathrm{mreach}_k(a,b) = \max\{\mathrm{core}_k(a),\ \mathrm{core}_k(b),\ d(a,b)\}$$

where $\mathrm{core}_k(a)$ is the k-NN core distance and $d(a,b)$ is Euclidean.
Clusters arise from the MST of mreach_k with **stability pruning**.

Silhouette (quality, any clustering):

$$s_i = \frac{b_i - a_i}{\max(a_i, b_i)}, \qquad a_i = \text{mean intra-cluster dist}, \quad b_i = \min_{\text{other clusters}} \text{mean inter-cluster dist}$$

Full expansion (worked)

We compute silhouette for one point (assuming HDBSCAN found two clusters A and B).

Cluster A (C1, C2, C3) — 2-D z-features:
C1 $= (1.00, 1.10)$, C2 $= (0.90, 1.00)$, C3 $= (1.10, 0.90)$

Cluster B (C4, C5, C6):
C4 $= (-1.00, -1.10)$, C5 $= (-0.90, -1.00)$, C6 $= (-1.10, -0.90)$

For C1 — intra-cluster distances and a_1:

$$\begin{aligned}
d(C1, C2) &= \sqrt{(0.10)^2 + (0.10)^2} \;=\; \sqrt{0.02} \;=\; 0.1414 \\
d(C1, C3) &= \sqrt{(0.10)^2 + (0.20)^2} \;=\; \sqrt{0.05} \;=\; 0.2236 \\
a_1 &= \frac{0.1414 + 0.2236}{2} \;=\; 0.1825
\end{aligned}$$

Inter-cluster distances to B and b_1:

$$\begin{aligned}
d(C1, C4) &= \sqrt{(2.0)^2 + (2.2)^2} \;=\; \sqrt{8.84} \;=\; 2.973 \\
d(C1, C5) &= \sqrt{(1.9)^2 + (2.1)^2} \;=\; \sqrt{8.02} \;=\; 2.834 \\
d(C1, C6) &= \sqrt{(2.1)^2 + (2.0)^2} \;=\; \sqrt{8.41} \;=\; 2.900 \\
b_1 &= \frac{2.973 + 2.834 + 2.900}{3} \;=\; 2.9023
\end{aligned}$$

Silhouette for C1:

$$s_1 = \frac{b_1 - a_1}{\max(a_1, b_1)} = \frac{2.9023 - 0.1825}{\max(0.1825,\ 2.9023)} = \mathbf{0.938}$$

Result & steering

- Keep clusters with good **stability** and average **silhouette** (e.g., ≥ 0.35).
- Label personas (e.g., "Premium Loyalists", "Bargain Browsers") for targeted content/offers.
- If HDBSCAN yields < 3 clusters or > 20% noise → fallback to **K-Means** with $k \in \{4, 5, 6\}$, pick via **silhouette/BIC**.

▼ **metrics: Personas (with feature_sources)** — click to expand


```yaml
metrics:
  personas_example:
    hdbscan:
      min_cluster_size: 3
      min_samples: 2
      feature_space: "z-scored [R,F,M,Engagement,DiscountRatio,CategoryShares...]"
    toy_points:
      A:
        - { id: C1, f1: 1.00,  f2: 1.10 }
        - { id: C2, f1: 0.90,  f2: 1.00 }
        - { id: C3, f1: 1.10,  f2: 0.90 }
      B:
        - { id: C4, f1: -1.00, f2: -1.10 }
        - { id: C5, f1: -0.90, f2: -1.00 }
        - { id: C6, f1: -1.10, f2: -0.90 }
    silhouette_example:
      a_i: 0.1825
      b_i: 2.9023
      s_i: 0.938

feature_sources:
  - mongo.orders
  - mongo.events

</details>
```