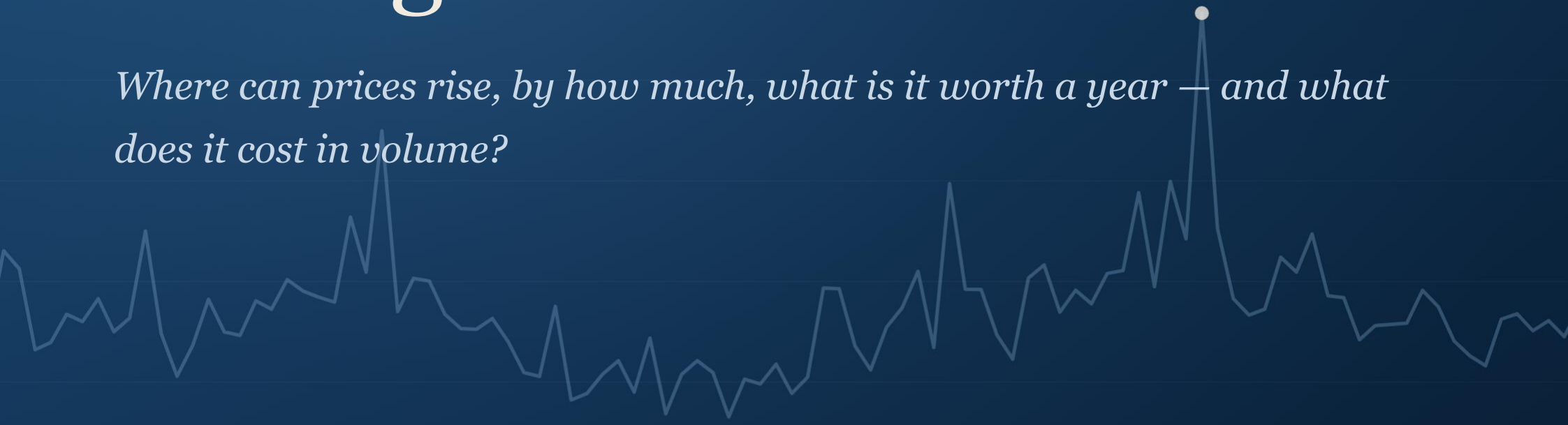
Pricing Power Audit

Where can prices rise, by how much, what is it worth a year — and what does it cost in volume?



Kevin Larretche kevin.larretche@gmail.com

AUGUST 2026 · 387 STORES · 899 PRODUCTS · COSTS ASSUMED

THE SETTING

A real grocery chain, two years of receipts

Everything in this document is measured on one public, real dataset — nothing is simulated, nothing is illustrative.

THE COMPANY A US grocery retailer — 387 stores across two regions. Its checkout data, week by week for two full years, was published by dunnhumby as a public research file ("Carbo-Loading").

THE SHELF Four dry-grocery categories — pasta, pasta sauce, syrups, pancake mixes: 899 products, $8.97M of sales, 6.1 million units.

THE AIM Answer the cover's four questions, product by product — and demonstrate the full audit end-to-end on real data, so the method can be judged before it runs on yours.

THE CATCH The file is deliberately imperfect — no unit costs, no list prices. That is the point: the method has to survive the file a real business actually has.

THE ANSWER

Raise 39 prices — about $59k a year on the table, measured, tested, and staged so a wrong move reverses in two weeks

A complete demonstration on real retail data — the public dunnhumby Carbo-Loading file. A realistic volume year moves the prize between $56k and $63k; in the currency that scales, +1.3 points of gross margin on the category book. Unit costs are assumed and labelled; every number can be exported and checked.

AUGUST 2026 · 24 MONTHS · 387 STORES · 899 PRODUCTS · $8.97M REVENUE

Kevin Larretche · kevin.larretche@gmail.com

PRICING POWER AUDIT

What two years of checkout data can prove

A review of price response across 387 stores, and what it implies for the next price list.

CLIENT
Demonstration — public file

SCOPE
387 stores · 899 SKUs

WINDOW
104 weeks · 2 years

REVENUE
$8.97M · 6.1M units

HIDDEN DISCOUNTS
26.5% of cells · median 21%

PREPARED BY
Kevin Larretche

THE RULES Everything measured is reproducible from the data and code that ship with this deck. What is assumed is labelled. What cannot be measured is named as a limit — never guessed.

THE TRACK RECORD Blind-verified on a second public file (dunnhumby "Breakfast at the Frat"): three category estimates lodged in writing before the engine ran — all matched, two to the second decimal.

THE PROMISE If you finish this document and still have to interpret the analysis yourself, we failed. Every product leaves with a decision.

What to do, product by product

Action	Products	Detail
Raise now — phase 1, weeks 1–2	5	Thin Spaghetti +10%; Ragu Traditional, AJ Original Syrup, AJ Buttermilk Mix, Prego Regular +5%
Raise in staged steps	34	The rest of the 39 — pre-cleared at assumed costs, re-cleared on your cost file before moving
Hold	860	No move without evidence; too little price variation is said openly
Watch the basket	—	Substitution between the big sauces is measured; the weekly monitor holds every complement

GROSS MARGIN

+$59k / yr

at assumed costs, replayed on last year's volumes net of the measured response — 28.7% → 30.0% on the same revenue base

VOLUME

inside the line

every move sits within its affordable loss — e.g. Ragu: expected −2.5% vs affordable −15.4%; all 39 pairs in the workbook

THE NUMBER THAT SCALES

+1.3 points

of gross margin on four categories — on a $50M category book, the same 1.3 points is about $650k a year

Five steps from raw file to a reversible price list

PART 1 The money sits in four slow, stable categories — pasta sauce alone is 43% of $8.97M, no hero SKU carries the book, so pricing is a portfolio decision.

PART 2 The raw data hides promotions: 26.5% of the file sells quietly below shelf, and the file's own flags catch roughly one in nine. Promo exposure is measured explicitly before any elasticity is trusted.

PART 3 Measured store by store, week by week, demand answers price weakly — and when a price moves, part of the volume steps sideways to the next brand, not out the door.

PART 4 39 raises are pre-cleared against what each margin can afford to lose — about $59k a year of gross margin, +1.3 points at assumed costs.

PART 5 The rollout is reversible by design: staged steps, a weekly tripwire, and a 192-pair store test for everything history cannot answer.

PART 1 OF 5

The business in the file

1

Before touching a price: where the money is, how steady it is, and why no single product can carry the answer.

Pasta sauce is the business: 43% of $8.97M

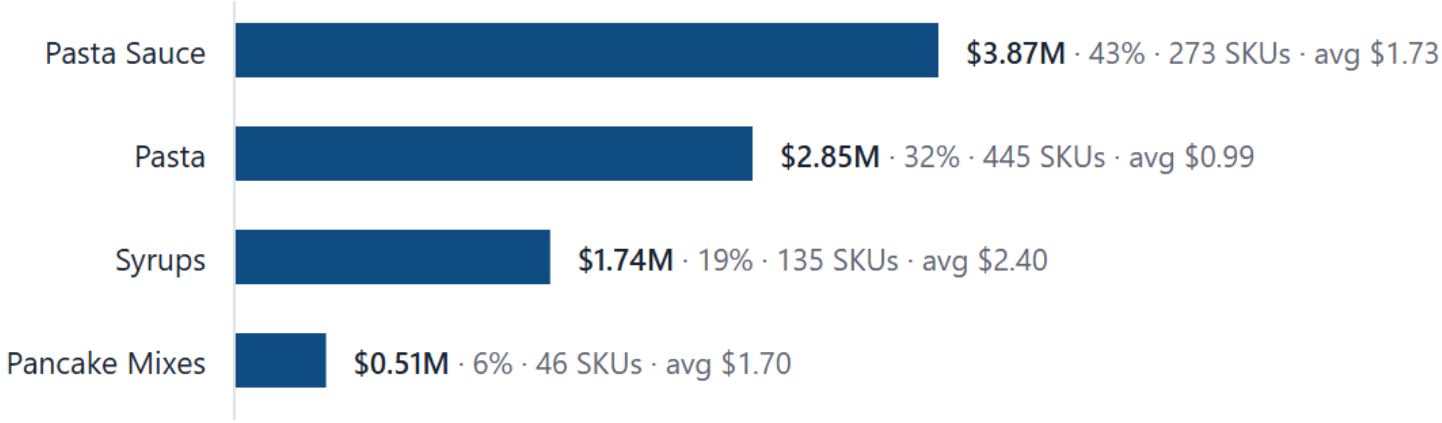


Exhibit 1 — revenue by category — 104 weeks, all 387 stores; share, SKU count and average paid price per category.

SO WHAT four categories, one shelf logic each: sauce is branded ($1.73 avg), pasta is a price-point aisle ($0.99 avg), syrup carries the premium ($2.40 avg). Price rules must differ by aisle — and they do, in the price list.

Source: dunnhumby Carbo-Loading, shaped store-week file (3,425,300 rows) — computed by make_exhibits.py, reproducible in one run.

Two steady years: $86k a week, year two +5.8%

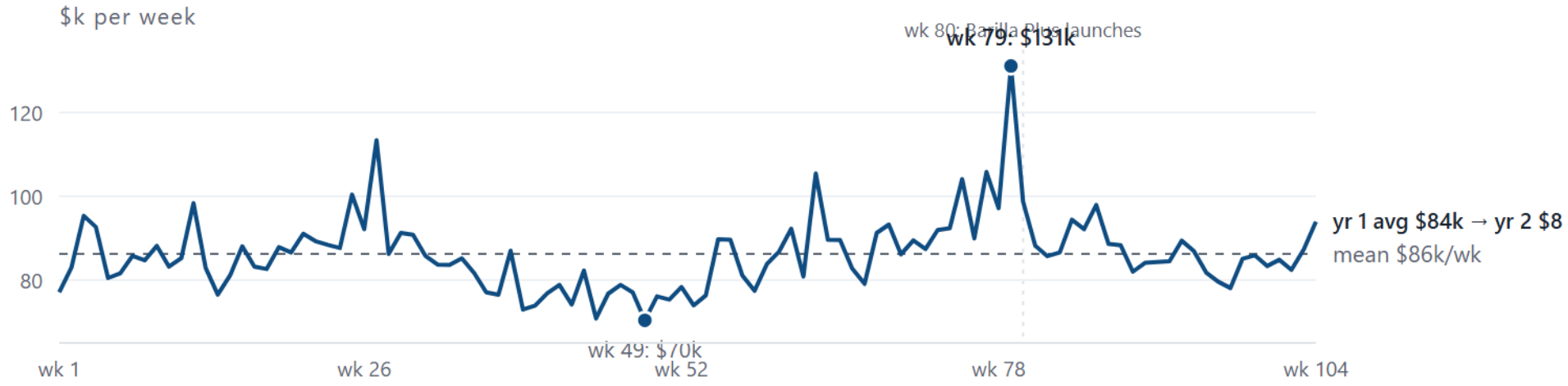


Exhibit 2 — weekly revenue, all four categories, $k per week — dashed grey = the two-year mean ($86k). Year two runs 5.8% above year one; the week-80 marker is the Barilla Plus launch.

SO WHAT stable base — no boom, no collapse: weeks run $70k to $131k around a $86k mean. Price moves land on a calm background and read cleanly against matched controls.

Source: dunnhumby Carbo-Loading, shaped store-week file (3,425,300 rows) — computed by make_exhibits.py, reproducible in one run.

No hero SKU: the top 10 carry just 10.8% of revenue

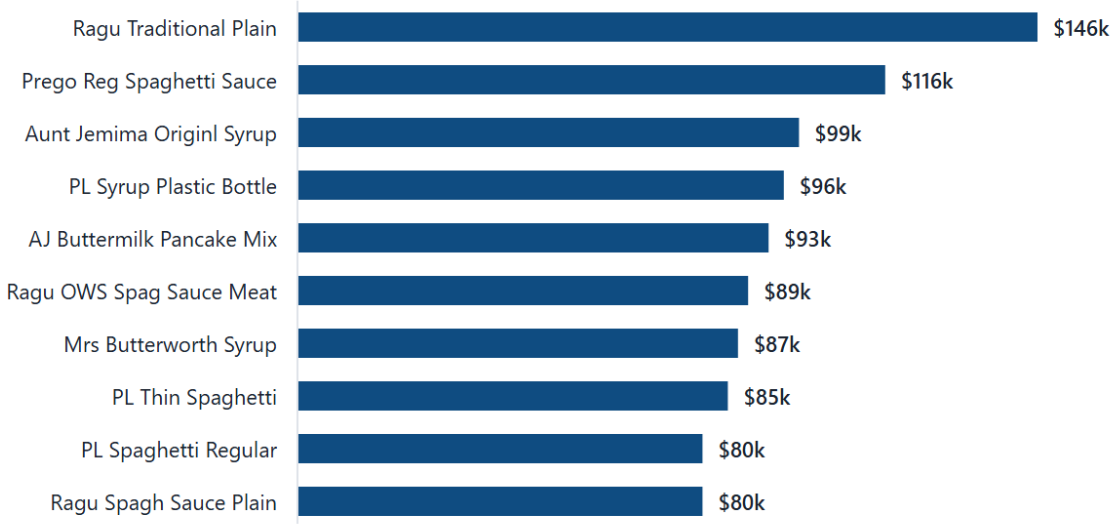


Exhibit 3 — the ten largest products by revenue, $k over 104 weeks.

SO WHAT

long tail

top 50 products still only carry 34% — pricing power is spread across the range. Hence 39 targeted moves, not one flagship repricing.

All five phase-1 moves sit inside this revenue top 10 — the first raises land where the money already is.

Source: dunnhumby Carbo-Loading, shaped store-week file (3,425,300 rows) — computed by make_exhibits.py, reproducible in one run.

PART 2 OF 5

What the file hides

2

The promotion history is mostly invisible in the raw export — and an elasticity that ignores promo exposure is wrong before it starts.

A quarter of the file is on promotion. Quietly.

QUIET PROMOTIONS

26.5%

of product-store-weeks — one product, one store, one week — sell at least 10% below the shelf price. The typical cut is 21%. The file's own promo flags catch roughly one in nine of them.

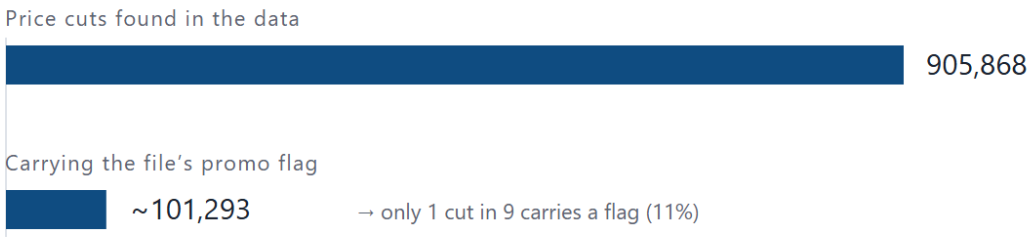


Exhibit 4 — product-store-weeks priced 10%+ below shelf level: cuts found in the data vs cuts carrying the file's own promo flag.

Why this is in the deck. This file carries only the price paid, so promo exposure is controlled week by week through the recorded feature and display flags. A client file with both list and paid price gets the full separation — one elasticity for the shelf price, one for the promo. The 26.5% is why that separation is step one on an engagement.

Source: counted on the store-week panel; shelf price reconstructed per store × product (rebuild_base_price.py).

Promotions move volume three ways — and one of them is a debt

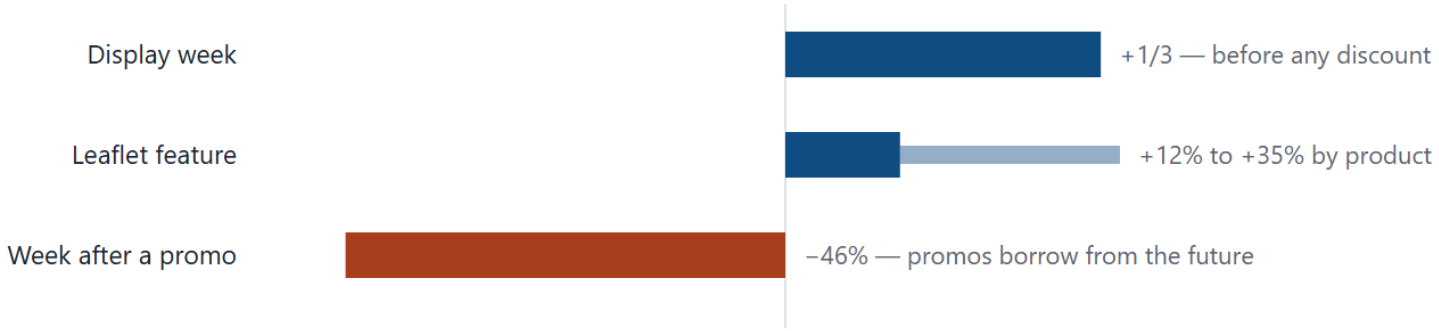


Exhibit 5 — volume effect of merchandising at unchanged price, and the week after a promotion ends. The dip means a discount's true cost includes the empty week after it — most promo calendars never charge for that week.

Why this is in the deck. Features run on 3.7% of store-weeks, displays on 1.7% — small enough to miss, big enough to bend any elasticity that ignores them. Both are controlled in every estimate that follows.

Source: engine measurement on the store-week panel; errors clustered by store.

PART 3 OF 5

Measuring price response

3

One product at a time, store by store, week by week — with the checks that would catch us being wrong, run before any client sees a number.

Demand answers price weakly: median −0.72, every estimate above −1.1

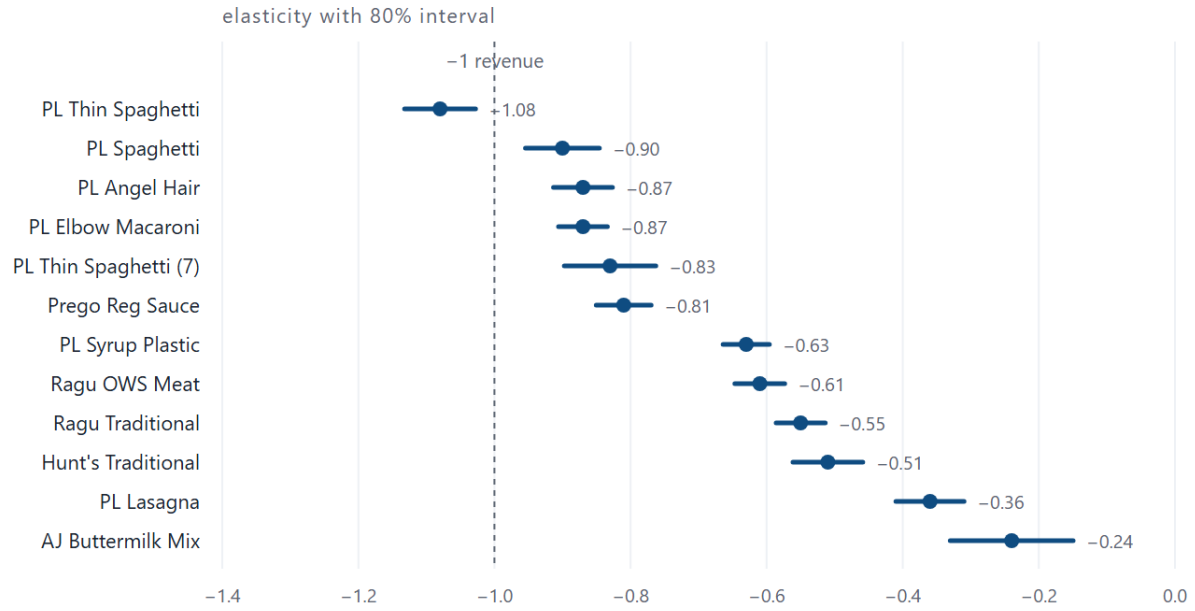


Exhibit 6 — point estimate and 80% interval — the independent fixed-effects check, twelve largest panels (all |t| > 3, up to 33,325 store-weeks). The plan prices on the engine's estimates, which add merchandising controls and read shallower (Ragu: −0.526 vs the check's −0.55). The dashed line at −1 is where a raise stops paying in revenue.

Why this is in the deck. Thin Spaghetti sits at the −1 line, so its +10% is a margin move, priced on its controlled estimate (expected −5.1%). Stress it with the check's steeper −1.08 and the loss is −9.7% — still inside its −11.2% affordable line.

PART 3 · MEASUREMENT

Raise a price, and part of the volume steps sideways — measured, with its limit

SUBSTITUTION — MEASURED

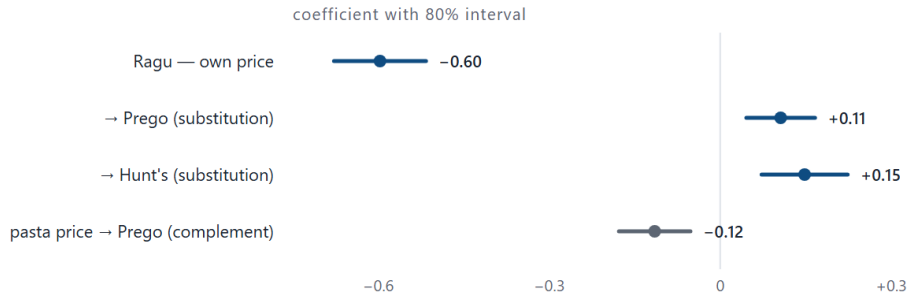


Exhibit 7 — 13,311 store-weeks where all three sauces sold side by side. In units, the two measured rivals alone pick up about a fifth of the leavers; the other 270 sauces are unmeasured, so the shelf keeps more. Barilla Plus launch read: Annex B.

THE KNOWN-ANSWER TEST

THE SETUP pasta and sauce are complements: cheaper pasta must lift sauce. The sign is known before the study runs.

THE RESULT Prego: −0.12 ± 0.05, the predicted sign. Ragu and Hunt's: too small to separate from zero — reported as exactly that.

THE POINT no wrong signs anywhere. A method that says "too small to see" is one you can trust when it says something is big.

What this file cannot see. Shoppers who leave the store entirely — this data watches one chain. That is exactly what the 192-pair store test exists to catch, before a chain-wide move, not after. Barilla Plus launch read: Annex B.

Source: two-way fixed-effects panel with store-clustered errors (cross_cannibal_forecast.py).

PART 4 OF 5

The recommendations

4

39 pre-cleared raises worth about $59k a year at assumed costs — each one priced, staged, and wired to an automatic revert.

PART 4 · RECOMMENDATIONS

Phase 1: five moves, each cleared with room to spare

Product	Current	Step	New	Expected volume	Affordable	Price point
PL Thin Spaghetti	$0.58	+10.0%	$0.64	−5.1%	−11.2%	crosses 60¢ — watch wk 1
Ragu Traditional Plain	$1.46	+5.0%	$1.53	−2.5%	−15.4%	inside $1.50s
AJ Original Syrup	$3.05	+5.0%	$3.20	−0.9%	−12.7%	inside $3s
AJ Buttermilk Mix	$2.05	+5.0%	$2.15	−1.2%	−15.2%	inside $2s
Prego Reg Spaghetti Sauce	$1.73	+5.0%	$1.82	−3.6%	−15.8%	inside $1.80s

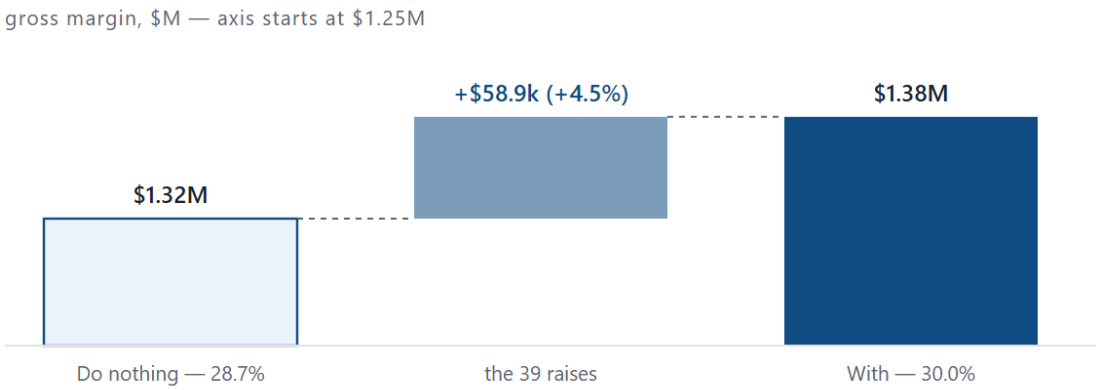
THE OTHER 34 each carries the same pair of numbers in the workbook — every one inside its affordable line at assumed costs, re-cleared on your cost file before it moves

PRICE-IMAGE SCREEN on an engagement, known-value items are screened before phase 1 — a 58-cent pasta is the kind of price customers know by heart, so its tripwire reads weekly from day one

Source: engine run at ASSUMED costs; expected volumes from the controlled estimates; stress-tested at the check estimates (Exhibit 5).

PART 4 · RECOMMENDATIONS

Same year, two futures: $1.32M of margin, or $1.38M



TAKE THE 39 RAISES

+$58,926

(+4.5%) — revenue base $4.61M; GM% 28.7% → 30.0% (+1.3 points); forward valuation $59,730 (worst-case floor $21.7k)

Exhibit 8 — gross margin at ASSUMED costs, next 12 months: do nothing vs the 39 raises on last year's volumes, net of the measured response.

Why this is in the deck. Valued twice on separate volume bases — last year's actuals and the 12-month forecast — landing 1.4% apart; the replay uses no forecast at all. The worst-case floor assumes every product's forecast misses the bad way at once — the next slide draws the realistic path over three held years. GM% convention: margin uplift over the unchanged revenue base — the raises also add revenue, so this understates.

Source: year-2 revenue and gross margin computed from the shaped file at the ASSUMED costs (cross_cannibal_forecast.py); uplift = engine replay value.

Held for three years, the same 39 moves are worth about $179k

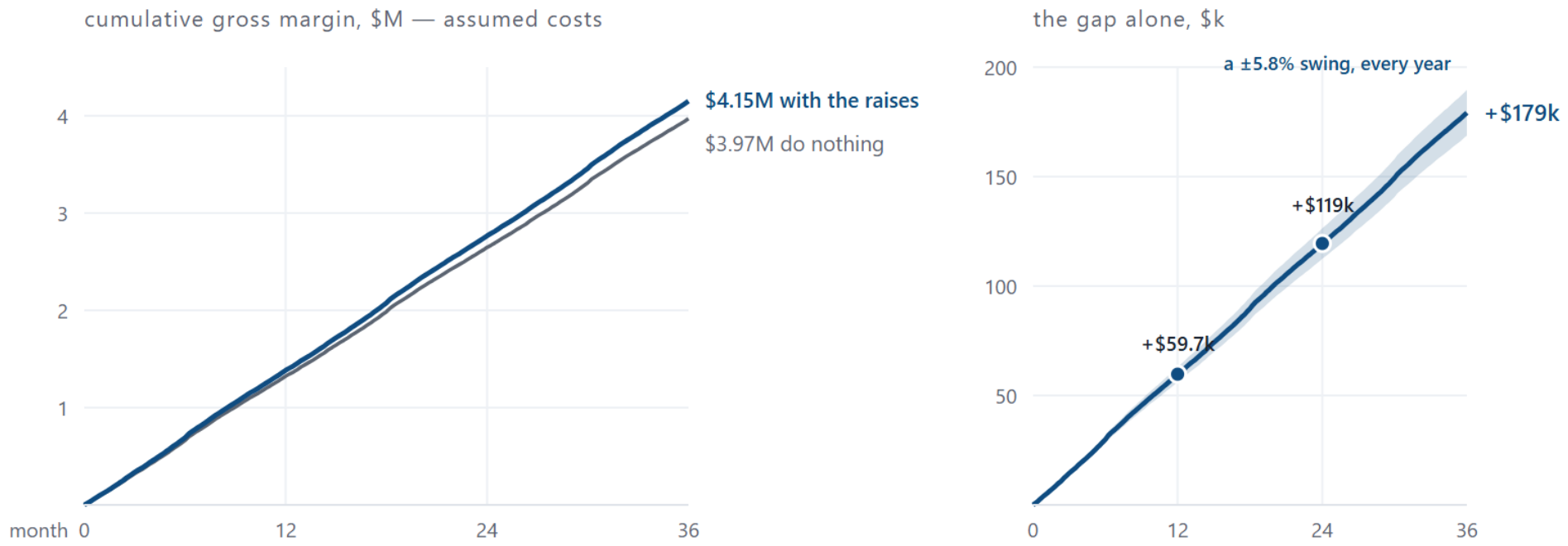


Exhibit 9 — left: cumulative gross margin at ASSUMED costs over 36 months — navy = with the 39 raises, grey = do nothing; year 1 follows year-2's real weekly pattern, years 2–3 hold the book flat (no growth claimed). Right: the gap alone — +$59.7k after year one, +$119k after two, about +$179k after three; shaded = the file's own ±5.8% year-over-year swing applied every single year ($169k–$190k at month 36).

Why the band is this size — and where the worst case went. The band carries the biggest volume swing this file ever showed, both ways, every year. The engine's $21.7k floor still exists — it assumes all 39 volume forecasts miss the bad way at once, so it lives in the workbook, not on the chart. Prices are re-cleared on real costs each year, and the weekly tripwire keeps a wrong estimate a weeks-long problem, not a three-year one.

Source: year-1 path = year-2 weekly revenue at the assumed margin rates; held years = arithmetic (book flat, prices held); central uplift = engine forward valuation; swing = Exhibit 2.

PART 5 OF 5

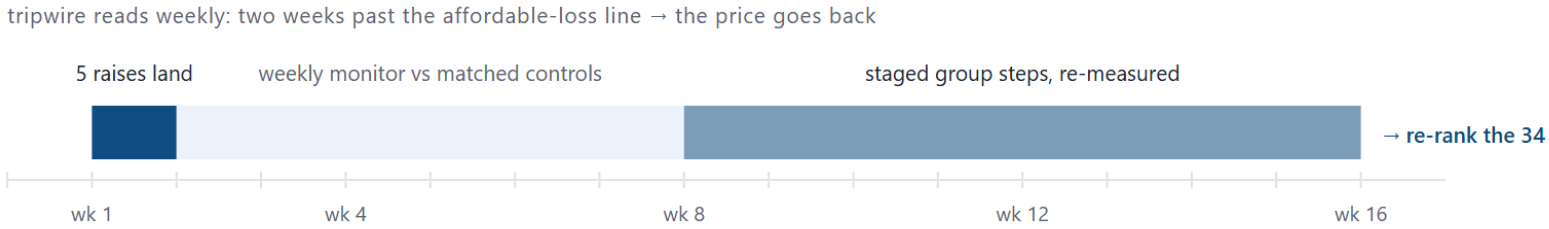
The rollout

Reversible by design: staged steps, a weekly tripwire, and a matched-store test for everything history cannot answer.

5

PART 5 · THE ROLLOUT

Five raises land in weeks 1–2 — a weekly tripwire reverses any breach



IMMEDIATE — WEEKS 1–2

MOVE 1 PL Thin Spaghetti $0.58 → $0.64 (+10%); Angel Hair moves with it in the same window

MOVES 2–5 Ragu Trad $1.53 · AJ Syrup $3.20 · AJ Mix $2.15 · Prego $1.82 (+5%)

TRIPWIRE two weeks past the affordable-loss line → the price goes back. That product only. No meeting. Exposure capped in weeks, not quarters.

BASKET WATCH sauce and pasta complements sit inside the weekly monitor — a pasta move that taxes sauce trips the same wire

THEN — AND WHO DOES THE WORK

WEEKS 3–8 monitor volume, revenue, margin against matched controls, weekly

WEEKS 9–16 clean read → the staged group steps, re-measured at each new level; then re-rank the remaining 34

THE 192 PAIRS matched on size and revenue, sharing trade areas so local competition is held constant — uncertain SKUs get an 8-week paired test

THE WORKLOAD the monitor is a delivered workbook — refreshed in ~15 minutes a week; execution is a re-tag of 39 SKUs, no system change

Nobody jumps to an optimum. Every rung re-measures before the next.

YOUR AUDIT

The same document, on your numbers

WHAT WE NEED One export: 12–24 months of sales — product × store × week, units and revenue — plus your price list and your cost file. A one-page data checklist follows the first call. No system access, no IT project.

WHAT YOU GET This same set on your products: the deck, the long-form report, the numbers workbook with every interval and formula, and the store-test protocol ready to run. Your team can check every figure.

HOW IT STARTS A 10-day audit at a fixed fee, quoted on the first 30-minute call. A written data-readiness verdict comes before anything else is billed. Reply: kevin.larretche@gmail.com

> *The method survived a file with no costs and no price list. Yours is better — and 1.3 points of margin scales with your book, not with ours.*

CLOSING

Nothing here is a black box

THE DECK	editable source + locked PDF — change a number six months from now
THE REPORT	the long-form version of every finding, with the method annex
THE WORKBOOK	elasticities with intervals and observation counts, the 39-row price list, live scenario formulas
DATA & CODE	the shaped datasets, the cost model marked ASSUMED, and the scripts that reproduce every check — including the ones on this deck's own numbers

Kevin Larretche · kevin.larretche@gmail.com

ANNEX A · METHOD

How the elasticity is measured

Estimated separately for each product, on its own store-week panel — up to 33,325 cells for a single product (Ragu Traditional: 29,715).

THE ESTIMATING EQUATION

$$log\ units(store, week) = \beta \cdot log\ price + \delta_1 \cdot feature + \delta_2 \cdot display + \alpha(store) + \gamma(week) + \varepsilon$$

B the price elasticity. This file carries one price — what was paid — so β is one blended response per product; a file with both list and paid price splits it into a list response and a promo response.

$\Delta_1 \cdot \Delta_2$ the merchandising effects — leaflet and display, recorded per product, store and week, so exposure never reads as a price effect.

A (STORE) one constant per store. Absorbs everything permanent about that shelf. “Each product compared against itself”, made literal.

Γ (WEEK) one constant per week. Absorbs what hits every store that week — season, holidays, the economy.

E errors, clustered by store; intervals at 80%. Ragu: −0.526 ± 0.06 — and every estimate re-runs the monthly-vs-weekly grain check before it ships.

ANNEX B · THE LAUNCH READ

Barilla Plus: consistent with cannibalisation — and why no causal number is claimed

THE EVENT, WEEKS 80–81

THE LAUNCH six new Barilla Plus SKUs land in two weeks, against 22 incumbent Barilla pasta SKUs with 79 weeks of prior history

THE RAW READ incumbent Barilla volume ran −8.5% after the launch while the whole pasta category moved −2.9% — consistent with the launch eating its own brand first

THE LIMIT — IN FULL

NO CLEAN CONTROL the rollout reached 367 of 387 stores; the 6 usable control stores give a DiD of +27% with t = 0.7 — pure noise, which is exactly why the raw −8.5% is reported as directional, not causal

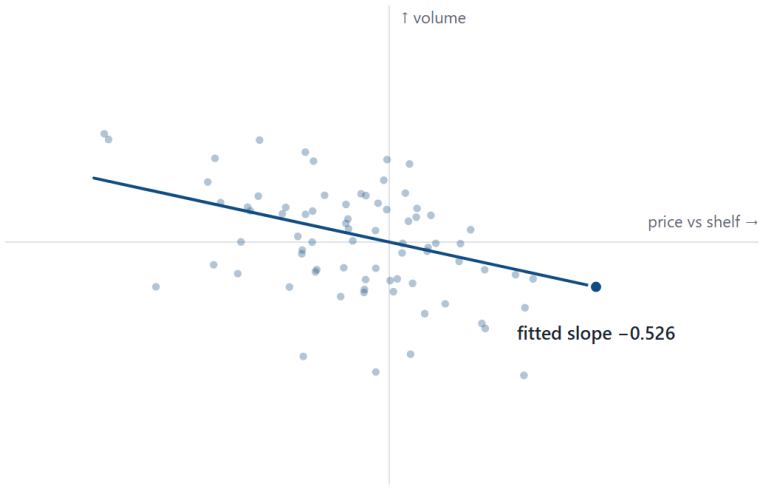
ON YOUR DATA launches that roll out region by region, or staggered shelf resets, give the controlled read this file cannot — the design is built and ready to run

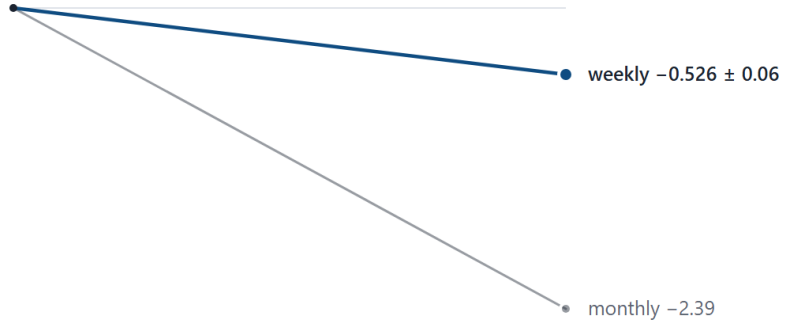
> *A limit stated is worth more than a number invented.*

Source: launch cohort from the product-week panel (README); before/after and DiD from cross_cannibal_forecast.py.

ANNEX C · THE ESTIMATE, DRAWN

What one elasticity looks like before it becomes a number





- **Exhibit A1 —** Ragu Traditional Plain: each point is one store in one week, after store and week effects are removed. The cloud is a sketch; the slope is the real estimate: −0.526.

- **Exhibit A2 —** the grain check: 24 monthly points read −2.39; 29,715 store-weeks read −0.526 ± 0.06. Months mix promo weeks into full-price weeks — the coarse read is 4.5× steeper. This comparison runs on every estimate before it ships.