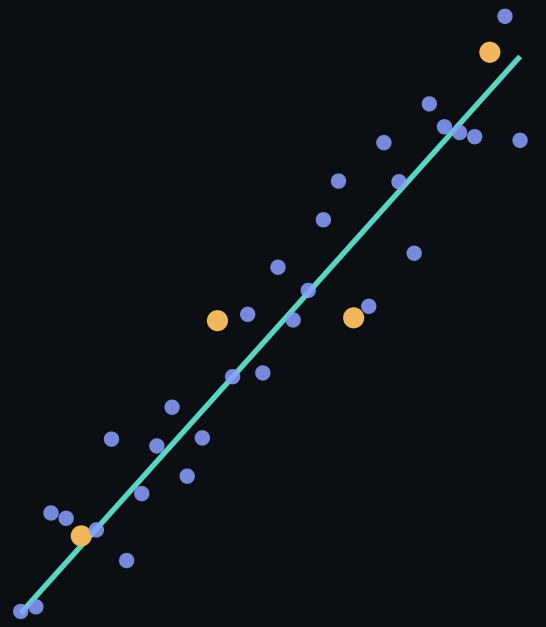
STRATUM · CASE STUDY

Retail Intelligence



From a million messy transactions to decisions a business owner can act on.

Data engineering · Statistics · Machine learning · Analytics

THE BRIEF

A wholesaler with data, but no answers

A UK online giftware wholesaler selling mainly to small shops. Two years of sales exports: **1,067,371 invoice lines** in a spreadsheet, and four questions the owner couldn't answer.

Q1

Is the business growing?

Revenue looks up on some charts and down on others.

Q2

Who are our valuable customers?

And which of them are quietly slipping away?

Q3

Who should we call this quarter?

The account team can't contact everyone.

Q4

What should we plan for?

Stock and staffing for the next quarter.

THE ANSWERS

Four questions, four evidence-backed answers

+2.2%

IS IT GROWING?

No: revenue is flat. The year-over-year change is not statistically significant.

69%

WHO MATTERS?

of revenue comes from 22% of customers, the “Champions”.

2.1×

WHO TO CALL?

more buyers reached by the model's top 10% than by random calling.

±10%

WHAT TO PLAN FOR?

weekly error on a forecast that beats the standard benchmark.

THE APPROACH

One pipeline, from raw export to decision

01

Profile

Explore the raw data before writing any cleaning rule

02

Engineer

Layered ETL into a DuckDB star schema

03

Verify

15 automated data quality checks

04

Analyse

Hypothesis tests, segments, lifetime value, propensity, forecasts

05

Deliver

Interactive dashboard, notebooks and documentation

One command (uv run retail all) rebuilds everything from the public source in about a minute.

STEP 1 · PROFILE

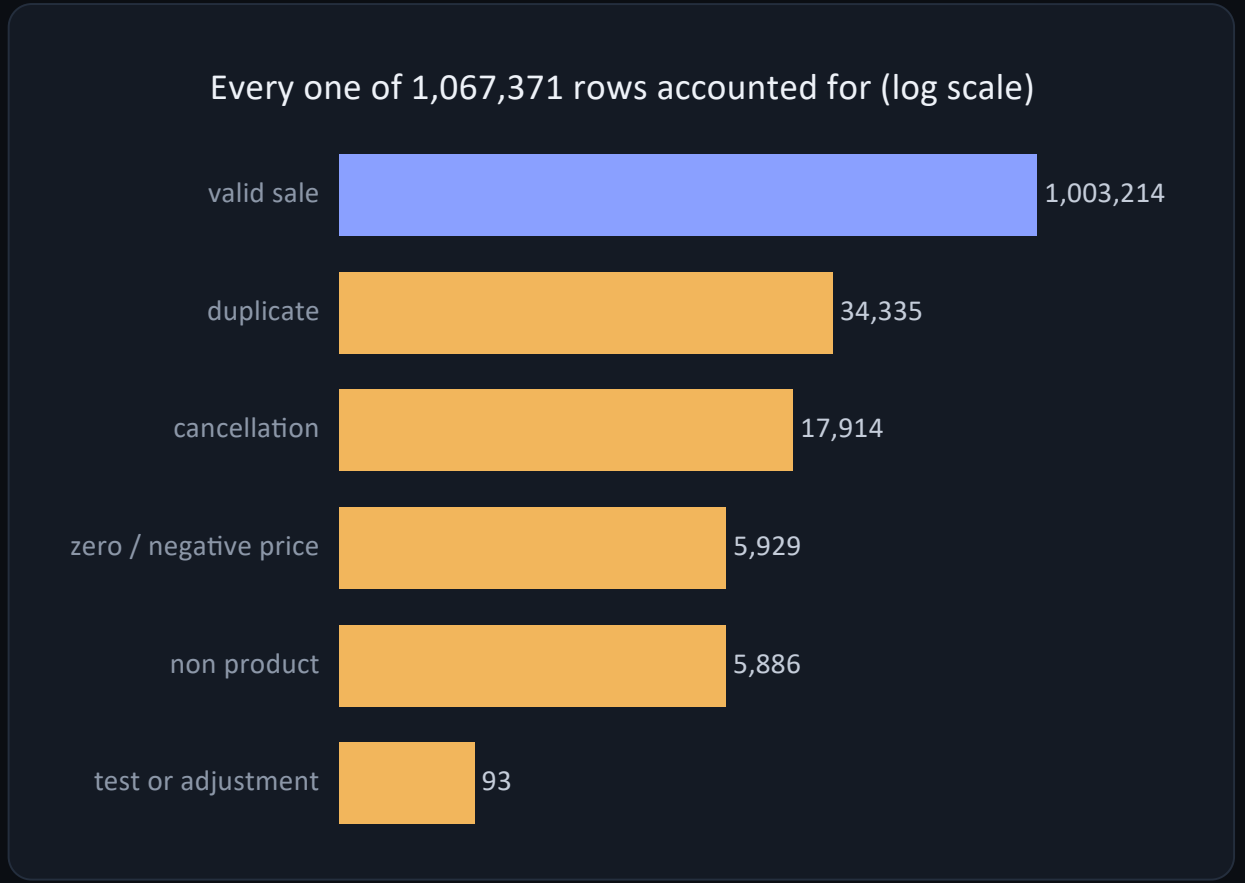
The raw data was messier than it looked

23% of lines have no customer ID (guest checkouts)

34,335 exact duplicate lines

19,494 cancellation lines, plus postage and fees mixed in with products

80,995 units in one order (£168k), cancelled 12 minutes later



STEP 2 · ENGINEER & VERIFY

A pipeline built to be trusted

Bronze

Untouched copy of the source

- Checksum-verified download
- Lineage columns: sheet, row, load time
- Re-run without the network

Silver

Typed, standardised, tagged

- Every row gets a status
- One canonical name per product
- Robust outlier flags

Gold

DuckDB star schema

- Date, product, customer dimensions
- Sales, returns, order facts
- KPI, cohort and country marts

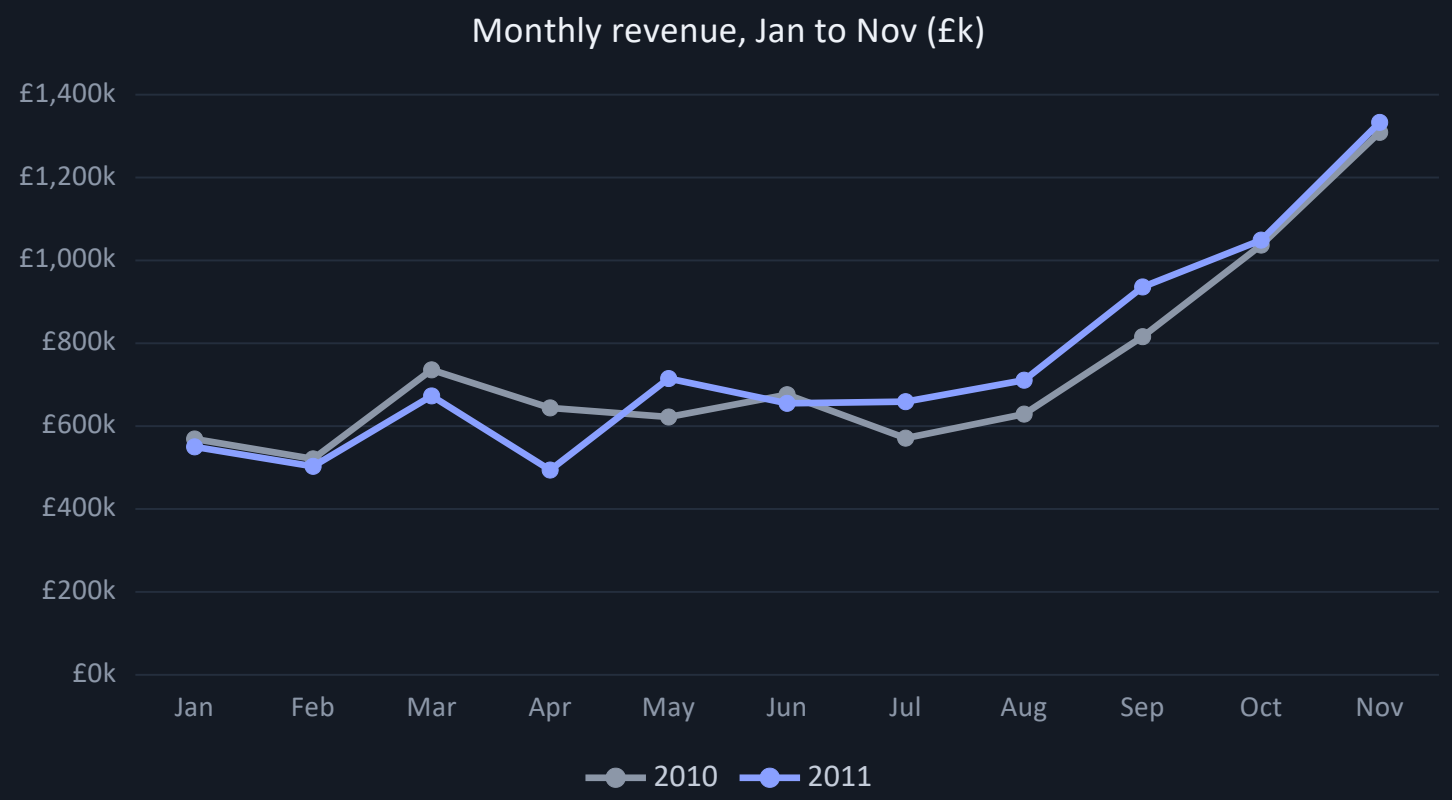
15 / 15 data quality checks pass

100% of rows and revenue reconcile end to end

< 1 min full rebuild with one command

FINDING 1 · GROWTH

Revenue is flat. The monthly chart hides it



+2.2%

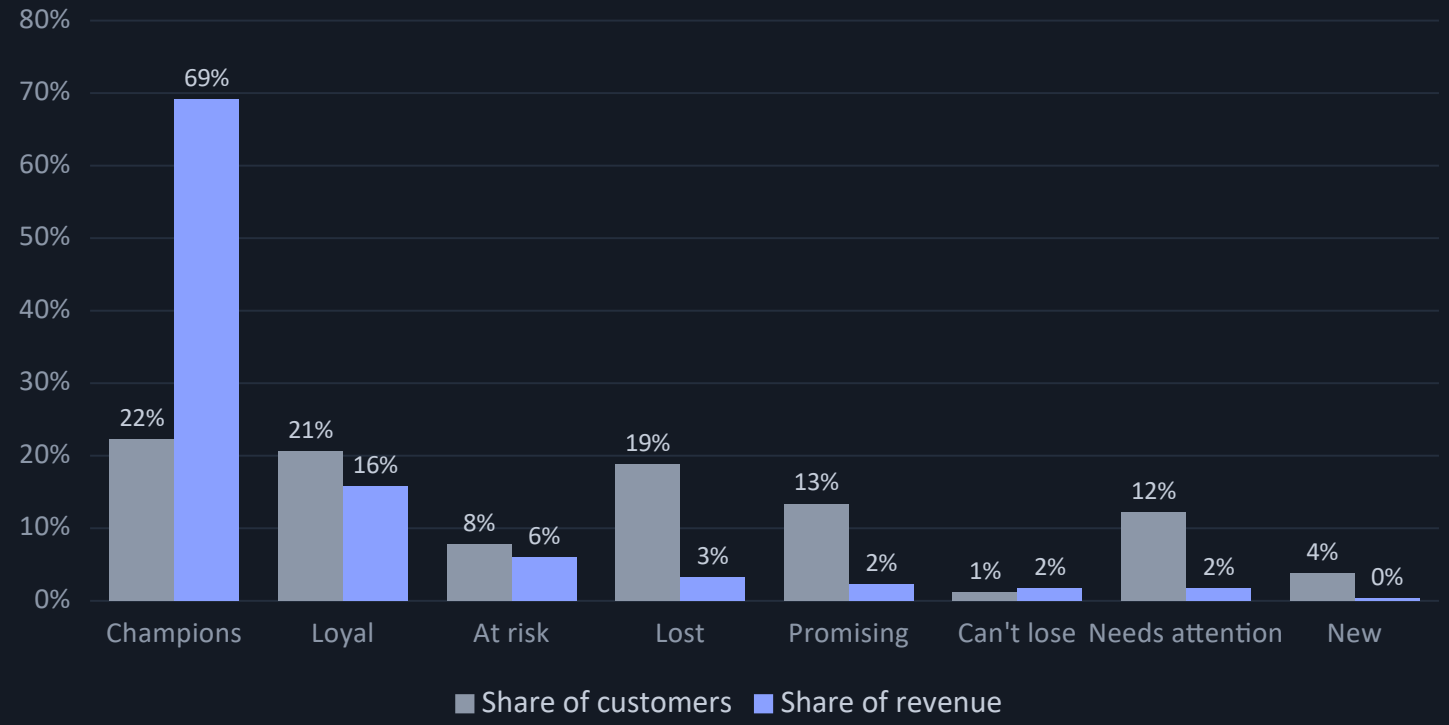
average daily revenue, Jan–Nov 2011 vs 2010

95% CI includes zero (p = 0.58)

Welch's t-test with a bootstrap interval. The busy monthly chart suggests momentum; the test shows the business is standing still.

FINDING 2 · CUSTOMERS

22% of customers drive 69% of revenue



69%

of revenue from the Champions segment

55%

of next year's predicted value sits with the top 10% of customers

0.74

Gini coefficient of customer revenue

FINDING 3 · ASSUMPTIONS, TESTED

Two common beliefs that turned out to be wrong

BELIEF

“Gift-season customers are one-off buyers”

66% vs **59%**

return within 180 days if first bought in Nov–Dec

if first bought the rest of the year

Two-proportion z-test, $p < 0.001$. Q4 acquisition is worth investing in, not discounting.

BELIEF

“Customers who return items are a problem”

£2,072 vs **£479**

median lifetime spend of customers who return items

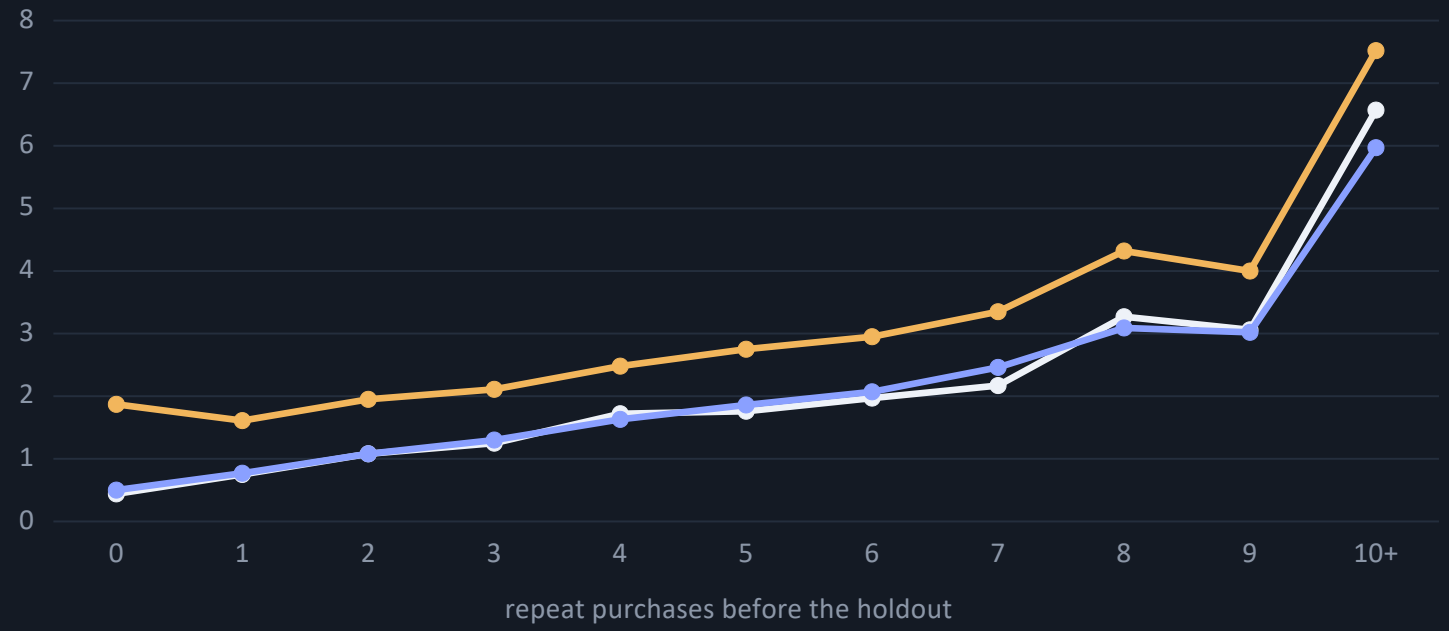
for customers who never return anything

Mann-Whitney U with bootstrap CI. Returns come with volume: don't deprioritise these accounts.

Also tested: international orders are larger (median £418 vs £299), and weekday strongly drives revenue. All five tests are Holm-corrected for multiple comparisons.

MODEL 1 · CUSTOMER LIFETIME VALUE

Predicting what each customer is worth



−1.7%

error in total predicted purchases, on six months the model never saw

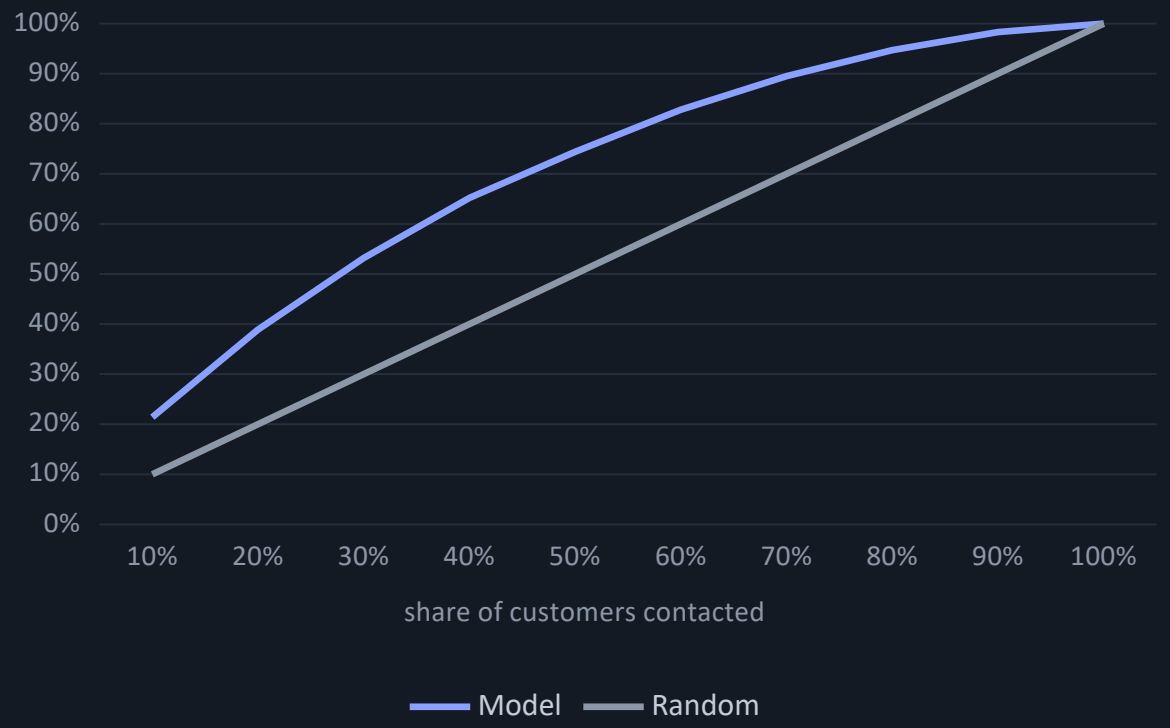
+69%

the naive “keep buying at your usual rate” baseline

BG/NBD + Gamma-Gamma, implemented from the original papers.

MODEL 2 · PURCHASE PROPENSITY

Who to call: twice the buyers for the same effort



CALL 20% OF CUSTOMERS

889

buyers reached with the model

457

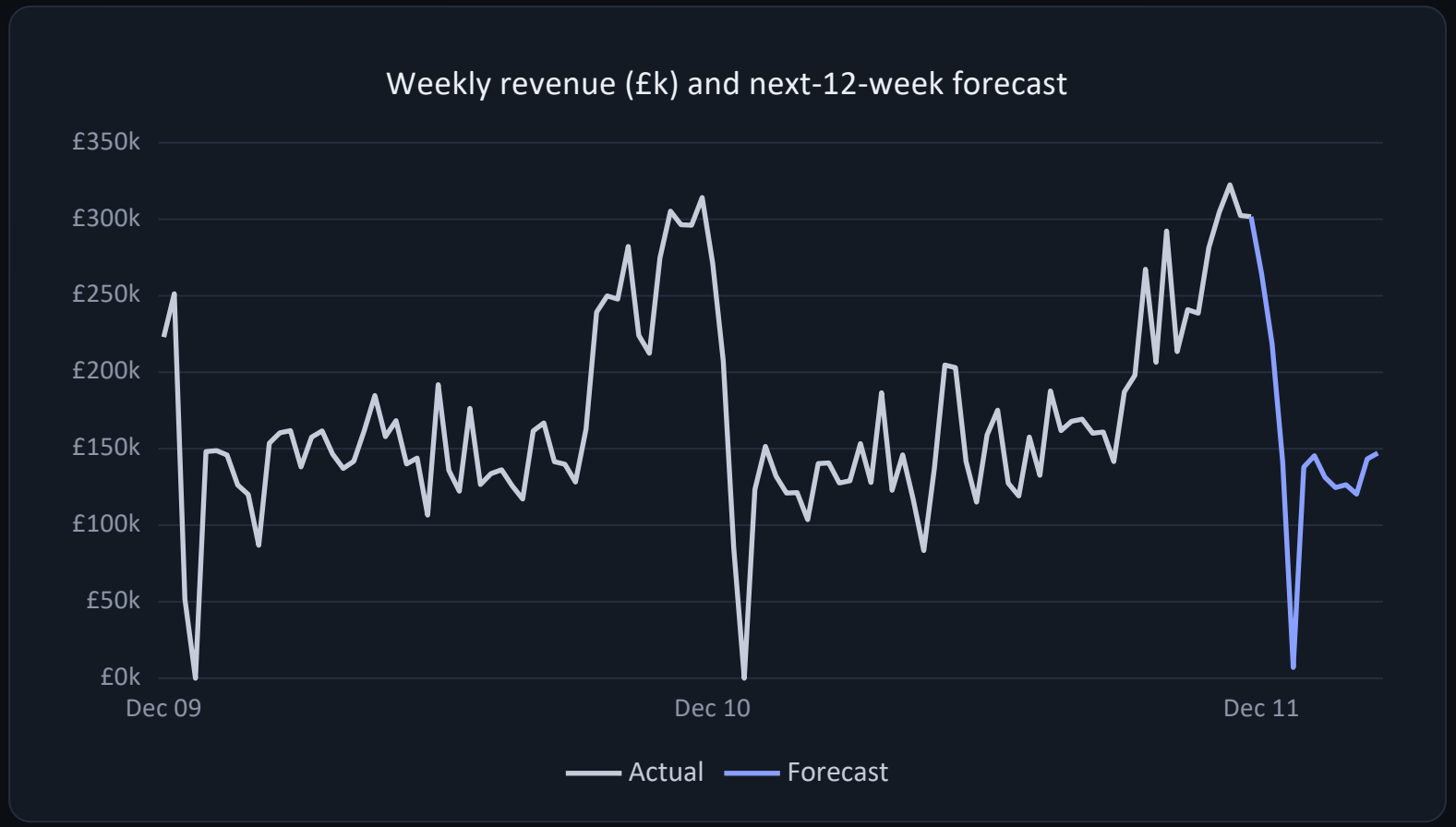
buyers reached calling at random

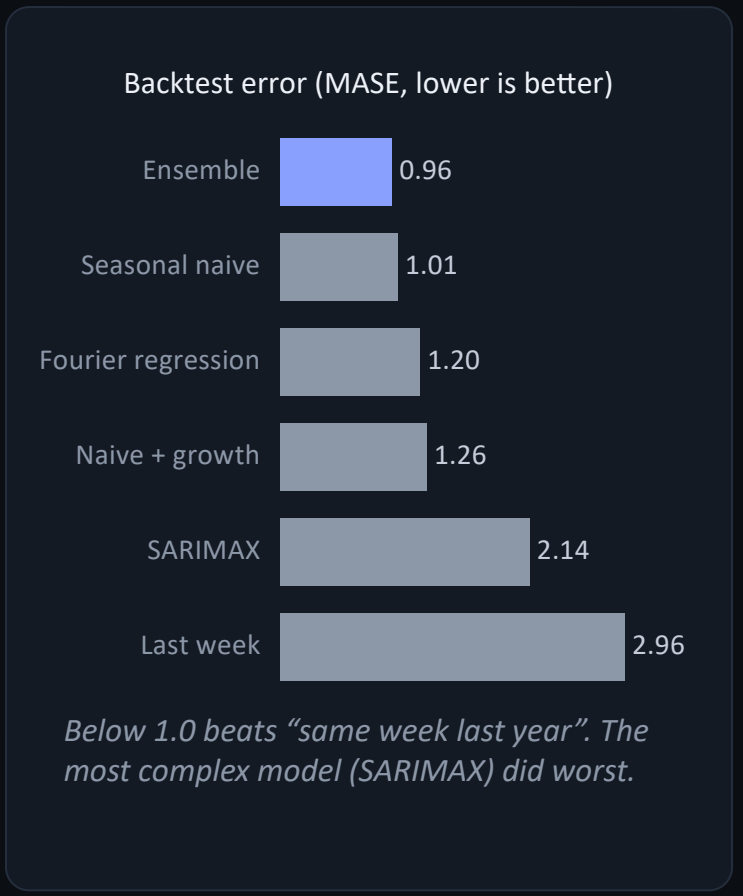
Model	ROC-AUC	Lift @10%
Recency rule	0.763	1.78×
Logistic regression	**0.804**	**2.13×**
Gradient boosting	0.801	2.14×

Out-of-time test. Simple logistic regression matched gradient boosting, so the explainable model ships.

MODEL 3 · FORECASTING

A forecast that beats the standard benchmark

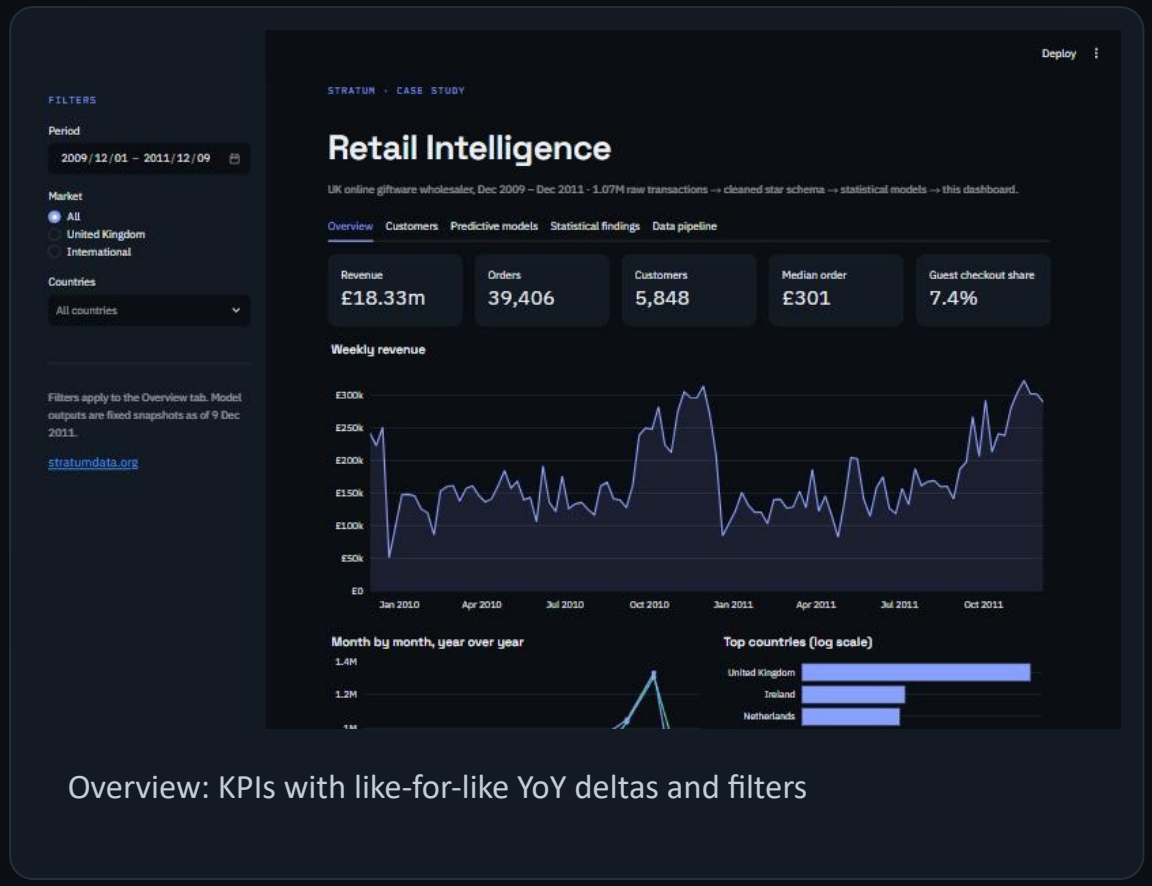




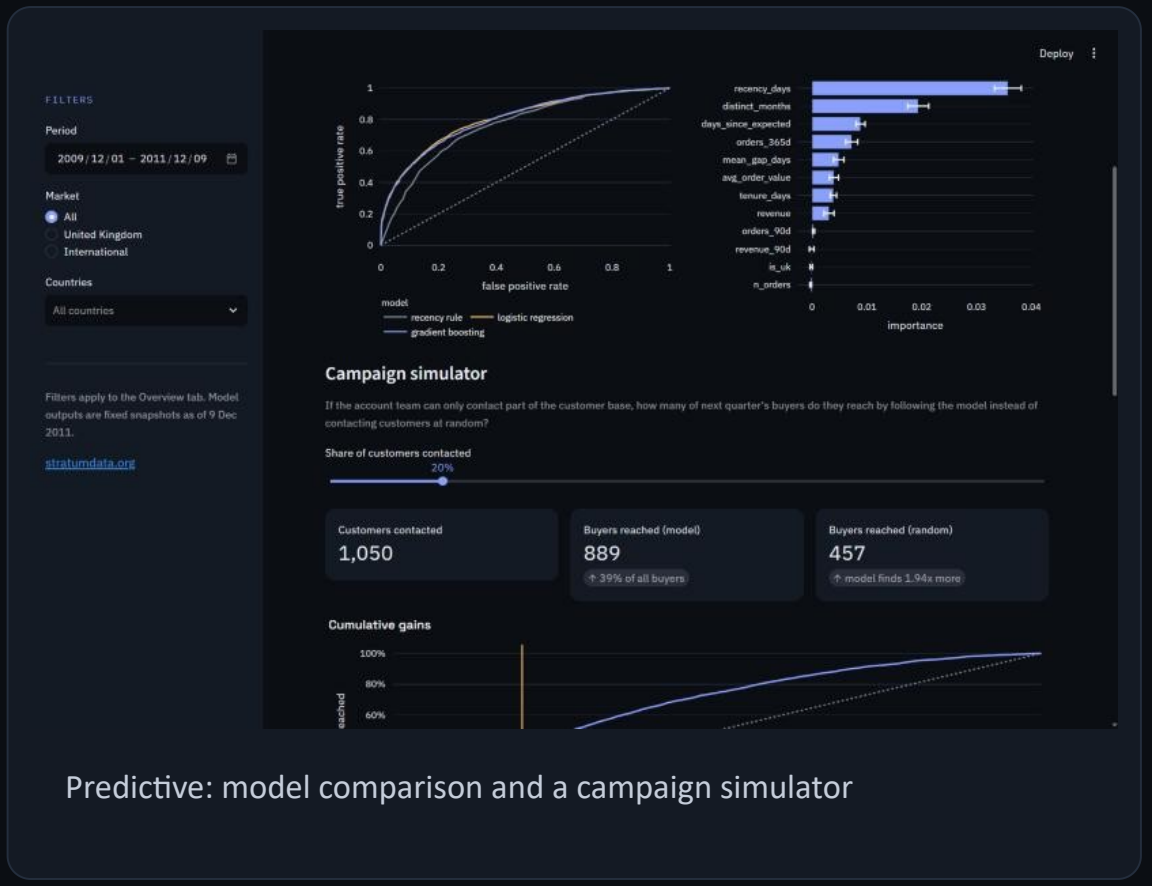
Below 1.0 beats "same week last year". The most complex model (SARIMAX) did worst.

STEP 5 · DELIVER

An interactive dashboard the team can use



Overview: KPIs with like-for-like YoY deltas and filters



Predictive: model comparison and a campaign simulator

Five tabs: Overview · Customers · Predictive models · Statistical findings · Data pipeline (with live quality checks)

CONCLUSIONS

What I'd tell the business owner

Protect the Champions

01

22% of customers bring 69% of revenue. Give them account management and early access to ranges.

Call by propensity score

02

Each quarter, call the top-scored customers first: about twice the buyers per call.

Invest in Q4 acquisition

03

Gift-season customers come back more often. Don't treat them as one-off buyers.

Plan for a flat year

04

Stock and staff to the forecast; growth needs a deliberate plan, it isn't happening on its own.

HOW IT'S BUILT

Production habits, not a notebook demo

Validated against baselines

Every model is tested against a simple alternative on data it never saw

18 automated tests

ETL rules, model correctness on simulated data, no-leakage checks, dashboard smoke test

CI on every push

GitHub Actions lints, tests and rebuilds the full pipeline from source

Reproducible end to end

One command from the public download to the dashboard

TOOLKIT

Python · SQL · DuckDB · pandas · SciPy · statsmodels · scikit-learn · Plotly · Streamlit · pytest · GitHub Actions · uv

STRATUM

Have data sitting unused?

I'll take the messy data your business already has and turn it into pipelines, models and dashboards your team can actually use.

stratumdata.org

stratumfreelance@gmail.com