
Module 7: Product Analytics and Metrics

Choose the right metrics, calculate them correctly, run experiments you can trust, and turn data into decisions.



AUTHOR

Varun Kumar

Product Management Master Handbook
v0.1 · 23 October 2026

Built from the 55-lecture course study guide and extended to the 12-module Product Core curriculum.
Hypothetical examples are labelled; no real company results are invented.

Contents

7.1 Product analytics fundamentals	3
7.2 Choosing KPIs	4
7.3 AARRR (Pirate Metrics)	6
7.4 Funnel analysis and drop-off	8
7.5 Engagement: DAU, WAU, MAU and stickiness	9
7.6 Retention, cohorts and churn	10
7.7 Monetisation metrics: GMV, AOV, ARPU, ARPPU	12
7.8 Unit economics: contribution margin, CAC, LTV, payback	13
7.9 Experiment design and A/B testing	16
7.10 Hypothesis testing, p-values and confidence intervals	17
7.11 Correlation versus causation	20
7.12 Segmentation in analytics	21
7.13 Event instrumentation and data quality	21
7.14 SQL for Product Managers	22
7.15 Excel for product analytics	25
7.16 Dashboards and tools: Power BI, Mixpanel, Amplitude, GA4	26
7.17 Experiment readout template (worked example)	27
7.18 Source-flagged corrections	27
7.19 Exercises with model answers	27
7.20 Interview perspective	29
7.21 Key takeaways	29
7.22 Check your understanding	30
7.23 Recommended resources	31

MODULE 7

Product Analytics and Metrics

Choose the right metrics, calculate them correctly, run experiments you can trust, and turn data into decisions.

≈ 12 h study

About the data in this module

Every number is **hypothetical or synthetic**. Hand-worked examples use round numbers so you can follow the arithmetic. SQL examples run against the synthetic files in `datasets/` (`users.csv`, `events.csv`, `orders.csv`, ...) and every output shown was produced by actually running the query. None of it is real company data, and no benchmark here should be quoted as an industry norm.

7.1 Product analytics fundamentals

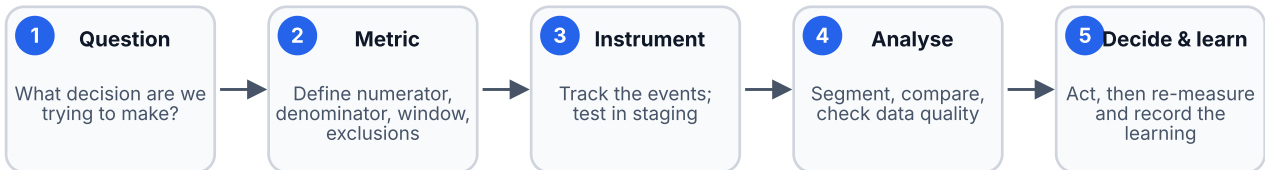
Product analytics is the practice of measuring how people use a product so that teams can find problems, size opportunities, test changes and verify outcomes.

Level 1 • Beginner

A metric is a number that describes something about your product. Analytics is choosing the *right* numbers, calculating them the *same way every time*, and using them to decide what to do next.

Every useful metric needs five parts defined — otherwise two people will calculate it differently and argue:

```
Metric definition = event/population + numerator + denominator + time
window + exclusions
Example: Checkout completion = users who placed an order ÷ users who started
checkout,
        same 7-day window, excluding internal test accounts and bots.
```



From product behaviour to decision. Start from the decision, not from the data you happen to have. Dashboards that answer no decision become wallpaper.

From your course notes (Lecture 21)

Define what is being measured, for whom and over what period, and compare against a relevant baseline or target. Users and revenue alone are not sufficient measures of every product's success. **Correlation alone does not prove that a change caused an outcome.** If an outcome misses its target, check data quality, user feedback, usability and context before removing a feature; revisit a metric when it no longer measures the intended outcome — not merely to make results look better.

7.2 Choosing KPIs

A **KPI** (key performance indicator) is a metric chosen to assess progress toward an important goal. Not every metric is a KPI.

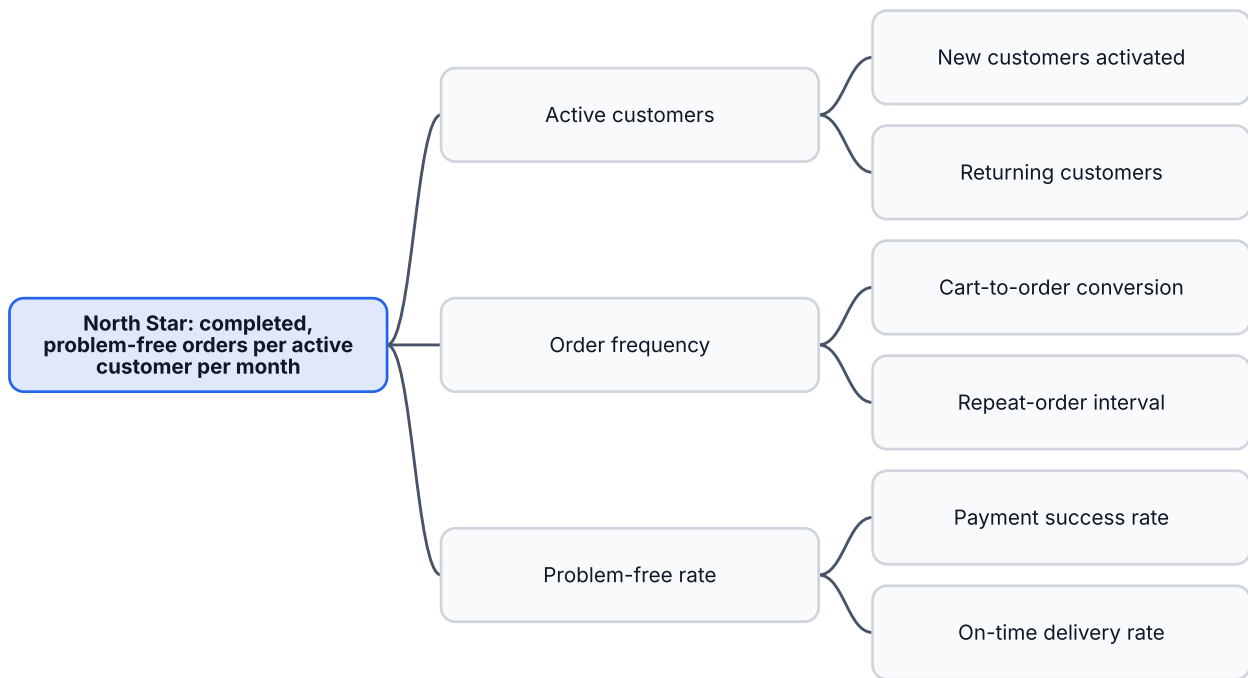
Concept	Meaning	Example (food ordering)
Leading indicator	Early signal that predicts a later result; you can influence it quickly	Share of new users who place a first order within 7 days
Lagging indicator	Result that confirms what already happened	90-day retention, quarterly revenue
Input metric	A driver the team directly controls	Payment success rate, time-to-first-order
Output / outcome metric	The result the inputs are meant to move	Orders per active customer
Guardrail metric	Must not get worse while you optimise something else	Complaint rate, refund rate, cancellation rate
Vanity metric	Looks good, drives no decision	Total registered users (all time)

7.2.1 The North Star Metric and metric trees

A **North Star Metric (NSM)** is a single measure of the recurring value users get from the product that also supports sustainable business results. It focuses teams; it never replaces guardrails.

From your course notes (Extra topic 3)

A north star is a team's chosen measure of recurring value delivered, not a universal number. A **metric tree** breaks a high-level outcome into smaller drivers so a team can investigate changes. Pair it with guardrails such as refunds and delivery complaints. *A single metric never tells the entire story. Do not claim causality merely from a breakdown.*



Metric tree for a food-ordering product (hypothetical). Each branch is a lever a team can own. Guardrails (refunds, complaints, cancellations) sit beside the tree. "Problem-free" must be defined precisely, e.g. no refund and no complaint within 48 hours.

How to choose an NSM — a checklist

1. Does it reflect value *delivered to the user* (not just value captured by the company)?
2. Is it tied to a repeatable behaviour (not a one-off)?
3. Is it sensitive enough that a good team can move it in a quarter?
4. Can it be measured reliably today?
5. Does it predict long-term business health (retention, revenue)?
6. Can you name two guardrails that stop gaming it?

Level 3 · Senior PM

NSMs fail when they become targets without guardrails (Goodhart's law): optimising "time in app" can produce addictive design and wellbeing harm. Senior PMs pressure-test an NSM by asking "*how could a team hit this number while harming users or the business?*" and add the guardrail that blocks that path. They also revisit the NSM when strategy or product stage changes.

7.3 AARRR (Pirate Metrics)

AARRR examines the path from discovering a product to recommending it: **Acquisition → Activation → Retention → Revenue → Referral**.



AARRR with example metrics. Stages can run in parallel, and the order is not a company-stage sequence. AARRR is a diagnostic lens, not a substitute for product-specific definitions.

From your course notes (Lecture 25)

- **Activation** should be an early action showing meaningful value — *first payment is only one example*; the right event depends on the product.
- **Retention** can mean returning or continuing to receive value; recurring subscription payment is one example, not the only one.
- Funnel arithmetic: if **800 of 1,000** visitors complete a step, conversion is **80%** and drop-off **20%**. If **600 of 800** return, retention is **75%** and loss **25%**.
- **Churn** needs a defined starting group, behaviour and time period; non-payment alone does not mean every free user has churned.
- The lecture's "20–30% ideal activation" and one-to-one referral claims are **not universal benchmarks**.
- Analytics shows *where* behaviour changes; interviews and tests help explain *why*.

7.3.1 Acquisition and activation

- **Acquisition metrics:** visits, sign-ups, installs, cost per acquisition by channel, share of traffic by channel. Always compare channels on *quality* (activation and retention), not just volume.
- **Activation:** pick an event that **correlates with long-term retention** and represents real value. Test it by comparing retention of users who did vs did not reach it — remembering correlation is not causation (see below).
- **Time-to-value:** how long from sign-up to the activation event. Shortening it usually raises activation.

Worked example · defining activation (hypothetical)

For a reading app, candidate events: *signed up*, *opened a book*, *finished a first chapter*, *read on 3 different days in week one*. The team compares 4-week retention of users who did each. Hypothetically, "finished a first chapter" predicts retention far better than "opened a book," so it becomes the activation event. The next step is an *experiment* to see if nudging users toward finishing a first chapter actually improves retention — not an assumption.

7.4 Funnel analysis and drop-off

A **funnel** counts how many distinct users (or sessions) get through an ordered series of steps. **Step conversion** compares each step with the previous one; **overall conversion** compares with the top.

Step conversion = users at step n ÷ users at step n-1

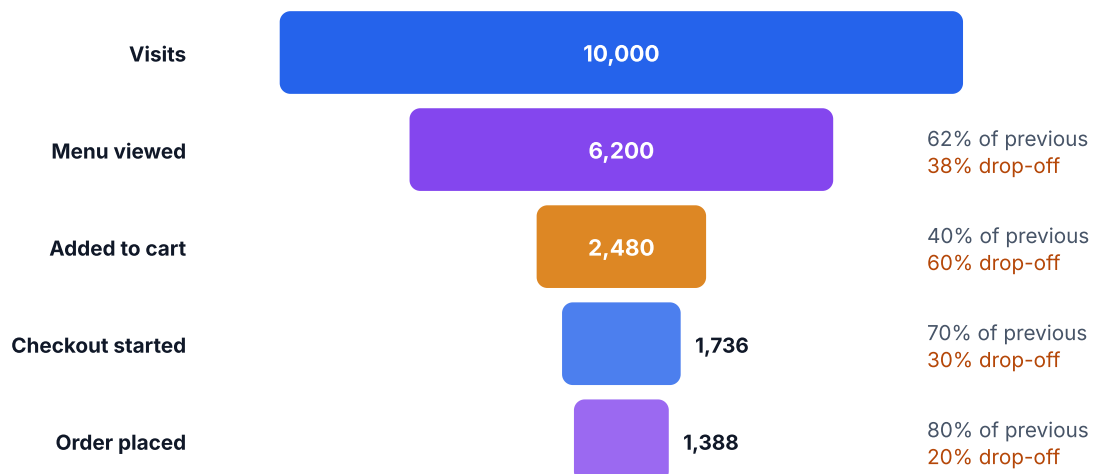
Drop-off = 1 - step conversion

Overall conversion = users at last step ÷ users at first step

Hypothetical worked funnel (10,000 visits)

Step	Users	Step conversion	Drop-off	% of top
Visits	10,000	–	–	100%
Menu viewed	6,200	62.0%	38.0%	62.0%
Added to cart	2,480	40.0%	60.0%	24.8%
Checkout started	1,736	70.0%	30.0%	17.4%
Order placed	1,388	80.0%	20.0%	13.9%

Interpretation: the **largest absolute loss** is visit → menu (3,800 users) and the **lowest step conversion** is menu → cart (40%). The best place to investigate is not automatically the biggest percentage drop; weigh how fixable each step is and how many users it affects.



Checkout funnel (hypothetical, 10,000 visits). Hypothetical numbers. Percentages on the right are relative to the previous step. A steep step is a prompt to investigate, not a diagnosis.

From your course notes (Companion ch. 5)

Of 300 checkout starters, 240 complete: **80% completion, 20% abandonment**. Cart-to-order of $240 \div 400 = \mathbf{60\%}$. These use different starting populations, so *their percentages should not be mixed*.

Funnel pitfalls

- **Sessions vs users:** counting sessions can inflate steps for repeat visitors. State which one you use.
- **Strict vs loose order:** must a user hit steps in order, within X minutes?
- **Time window:** a user who adds to cart today and buys tomorrow — is that a conversion?
- **Segmenting is essential:** an overall funnel can hide that mobile-web users convert at half the rate of app users.

Interview perspective — “Checkout conversion dropped. What do you do?”

1. Clarify the metric definition and window.
2. Check data quality (tracking change? outage?).
3. Segment: platform, app version, geography, payment method, new vs returning, channel.
4. Compare to seasonality.
5. Look at internal changes (releases, price/fee, promo, partner outage) and external ones (competitor, holiday).
6. Form hypotheses, rank by likelihood $\times$ impact, propose the first checks. Always say what you would look at *first* and why.

7.5 Engagement: DAU, WAU, MAU and stickiness

DAU = distinct users with a qualifying action on a given day

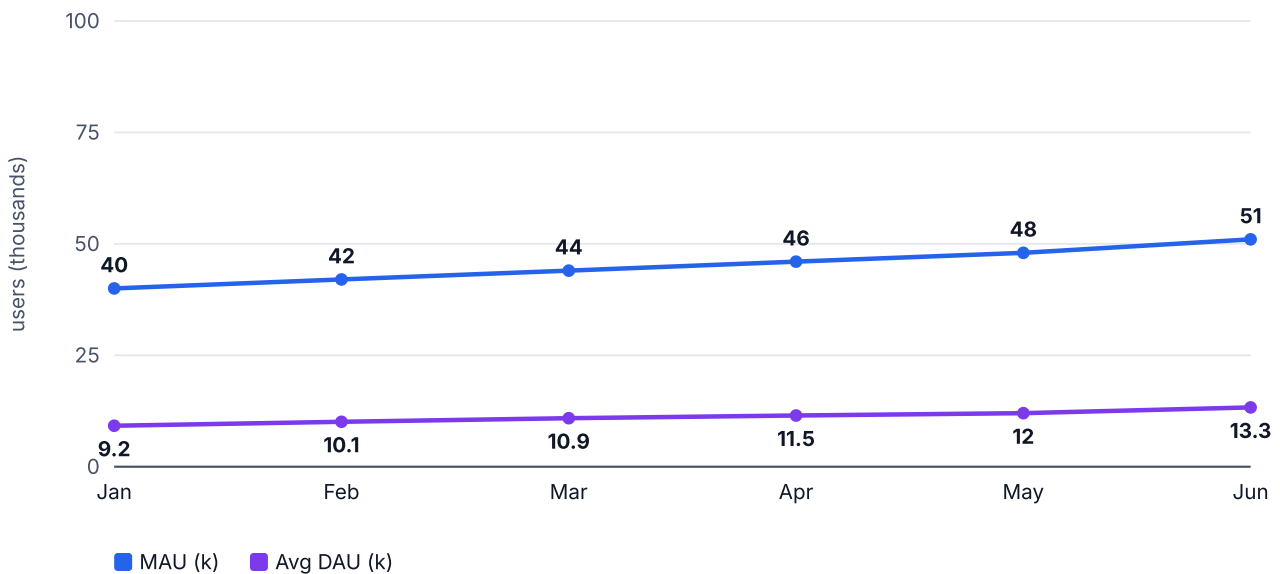
WAU / MAU = same over a 7-day / 28-30-day window

Stickiness = $DAU \div MAU$ (average DAU over the period $\div$ MAU for the same period)

“Active” must mean a *meaningful* action (e.g., viewed a menu or placed an order), not merely “opened the app”.

Worked example

Average DAU = 12,000; MAU = 48,000 $\rightarrow$ **stickiness = 25%** — on average a monthly user is active about 7.5 days a month (0.25×30). Whether 25% is good depends entirely on how often the product's job occurs: a banking app opened monthly differs from a messaging app used daily.



DAU and MAU trend (hypothetical, thousands). Hypothetical. Stickiness ($\text{DAU} \div \text{MAU}$) moves from 23% in January to 26% in June — engagement is deepening, not just growing.

7.6 Retention, cohorts and churn

Retention measures how many users in a defined **cohort** are still active in a later window. A cohort groups users by a shared starting event (e.g., signup week).

$\text{Retention (week } k) = \text{cohort users active in week } k \div \text{cohort size}$

$\text{Churn rate (period)} = \text{customers lost during the period} \div \text{customers at start of the period}$

$\text{Net revenue retention (NRR)} = (\text{start MRR} + \text{expansion} - \text{contraction} - \text{churned MRR}) \div \text{start MRR}$

From your course notes (Extra topic 4)

Of 100 users who signed up in week one, 25 order again in a specified later seven-day window → **25% for that window**. Users who ordered on day one but never returned are not “retained” unless your definition says so. State cohort start, qualifying action, time zone and exact return window. A correlation between activation and retention does not prove the activation action causes retention.

7.6.1 Reading a cohort heatmap

The heatmap below uses the **synthetic dataset** `cohort_retention_weekly.csv`: each row is a signup-week cohort; each column is the share of that cohort that placed a delivered order k weeks

after signup.

Signup cohort	Week 0	Week 1	Week 2	Week 3	Week 4
Wk 1 (n=51)	17.6%	7.8%	11.8%	3.9%	3.9%
Wk 2 (n=67)	11.9%	13.4%	4.5%	9.0%	4.5%
Wk 3 (n=50)	14.0%	12.0%	2.0%	4.0%	6.0%
Wk 4 (n=54)	13.0%	9.3%	11.1%	1.9%	7.4%
Wk 5 (n=75)	10.7%	8.0%	4.0%	8.0%	8.0%
Wk 6 (n=74)	18.9%	5.4%	6.8%	5.4%	1.4%

Weekly cohort ordering rate (synthetic dataset). Synthetic data. Darker = higher share ordering. Read across for one cohort's decay; read down a column to see whether newer cohorts retain better. With cohorts of 50–75 users, one or two users move a cell by 2–4 points — noise is large, so avoid over-reading differences.

How to read it: (1) *Across a row* — the shape of decay and whether it flattens. (2) *Down a column* — are newer cohorts better after a product change? (3) *Diagonals* show calendar effects (holiday, outage). Small cohorts produce noisy cells.

Retention pitfalls

- **Survivorship:** averaging only active users hides churn.
- **Mixed definitions:** “active” in Week 1 vs “ordered” in Week 1.
- **Blended cohorts:** comparing a fresh paid-ads cohort to an organic one without segmenting.
- **Small samples:** see the caption above.

7.7 Monetisation metrics: GMV, AOV, ARPU, ARPPU

GMV = Σ value of orders in the period (define: before/after discounts, refunds, cancellations)

AOV = total order value $\div$ number of orders (same definition, same period)

ARPU = revenue in the period $\div$ active users in the period

ARPPU = revenue in the period $\div$ paying users in the period

From your course notes (Lecture 31)

AOV = total order value $\div$ number of orders for a chosen period; it does not by itself measure profit or user happiness. Example: ₹200, ₹300, ₹400 $\rightarrow$ ₹900 $\div$ 3 = **₹300**.

Compare similar customers; check discounts and delivery costs; **a higher AOV alone does not prove improved profitability or experience.**

Worked example — GMV is not revenue

A marketplace processes 100,000 delivered orders at an AOV of ₹400 $\rightarrow$ **GMV ₹40,000,000**.

The platform's *net revenue* is only what it keeps: commission, fees, ad income less the discounts it funds. Quoting GMV as “revenue” inflates the picture.

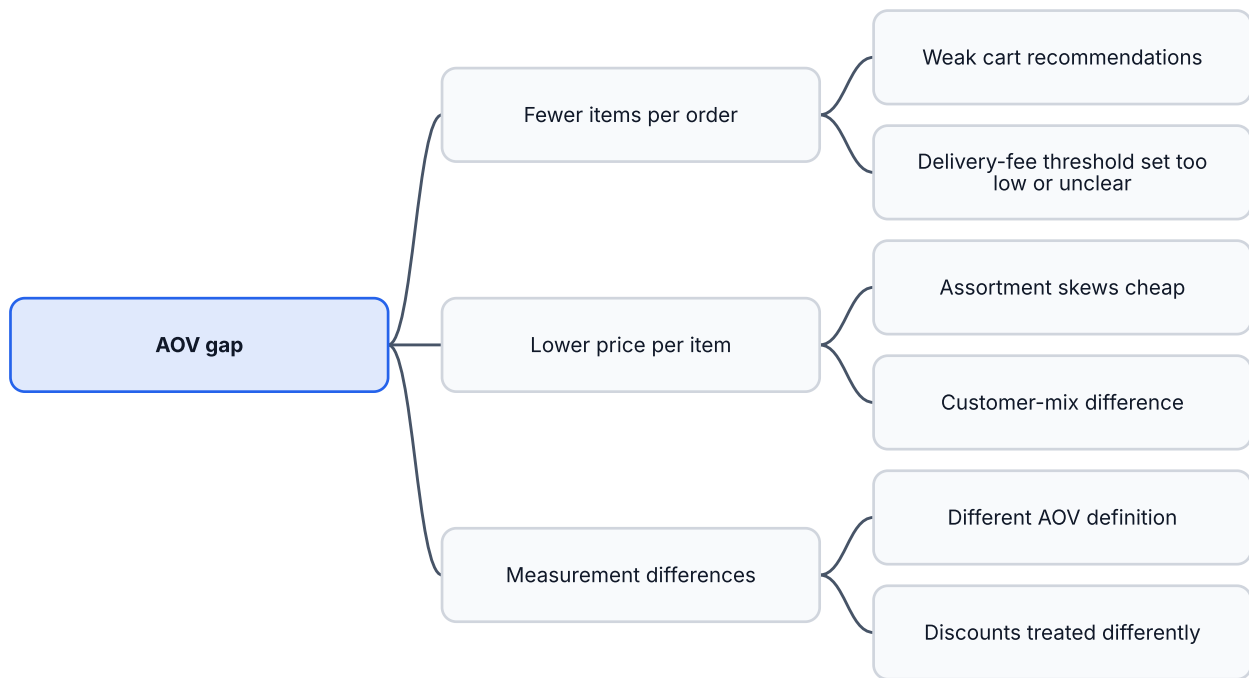
Worked example — ARPU vs ARPPU

Revenue ₹2,400,000; monthly active users 48,000; paying users 9,600. ARPU = $2,400,000 \div 48,000$ = **₹50**. ARPPU = $2,400,000 \div 9,600$ = **₹250**. Payer conversion = $9,600 \div 48,000$ = **20%**. Note ARPU = payer conversion $\times$ ARPPU ($0.20 \times 250 = 50$). A freemium PM can raise ARPU by converting more users *or* by raising what payers spend — different levers, different risks.

7.7.1 Decomposing AOV (the quick-commerce case)

AOV = items per order $\times$ average price per item

To raise AOV, a team can raise items per order (relevant add-on suggestions, bundles, free-delivery thresholds) or price per item (assortment, premium options). The course case compared a quick-commerce AOV with a competitor's.



Issue tree for "Why is our AOV lower than a competitor's?" (hypotheses, not findings). Every leaf is a hypothesis to test with data. Start with definitions (the cheapest cause to rule out), then segment baskets by customer type.

From your course notes (Lecture 31)

Participants discussed fee thresholds, inexpensive add-ons, near-expiry discounts, bundles and loyalty rewards. **These were hypotheses, not validated causal findings.** Check contribution margin, discount and delivery costs, conversion, repeat purchase and customer trust — not AOV alone. Assess whether each idea genuinely helps customers rather than encouraging unwanted spending.

7.8 Unit economics: contribution margin, CAC, LTV, payback

Unit economics looks at the money earned and spent for *one unit* — an order, a customer, a subscription.

Contribution margin = revenue - variable costs attributable to the unit

CAC = acquisition spend ÷ new customers acquired (define which costs)

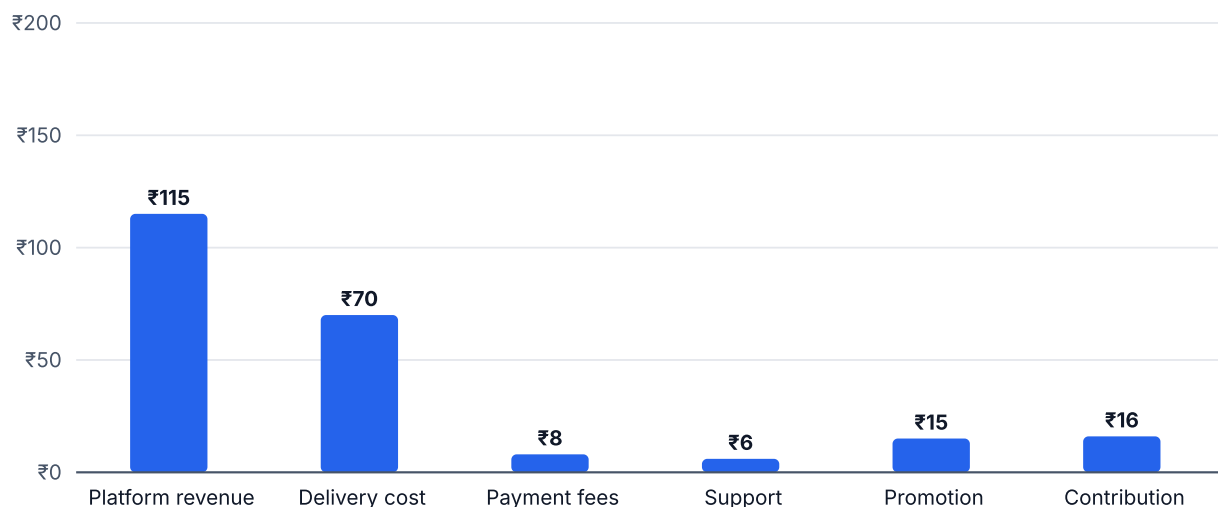
LTV (simple) = monthly contribution per customer ÷ monthly churn rate

Payback period = CAC ÷ monthly contribution per customer

Worked example — contribution per order

Line	₹ per order
Average order value	400
Commission (20% of AOV)	80
Delivery fee collected	30
Platform fee	5
Platform revenue per order	115
Delivery partner cost	-70
Payment processing (2% of AOV)	-8
Support cost allocation	-6
Promotion funded by the platform	-15
Contribution per order	16 (4.0% of AOV; 13.9% of revenue)

Without the ₹15 promotion, contribution would be ₹31. If the promotion lifts orders by less than the contribution it burns, it destroys value. *More orders at negative contribution make the business worse.*



From order value to contribution (hypothetical ₹ per order). Revenue of ₹115 less four costs (₹70 + ₹8 + ₹6 + ₹15 = ₹99) leaves ₹16. Bars for costs are shown as positive magnitudes.

From your course notes (Extra topics 6, 7)

Contribution margin is *not* net profit. With ₹100 revenue and ₹65 attributable cost, contribution is ₹35 (35%); with a ₹15 promotion it falls to ₹20. Campaign example: ₹10,000 spent to acquire 100 qualified customers → simple **CAC ₹100**; at ₹40 contribution per month, simple undiscounted **payback is 2.5 months only if that contribution persists and other costs are excluded**. Avoid blindly copying LTV-to-CAC benchmarks.

Worked example — LTV, CAC and payback

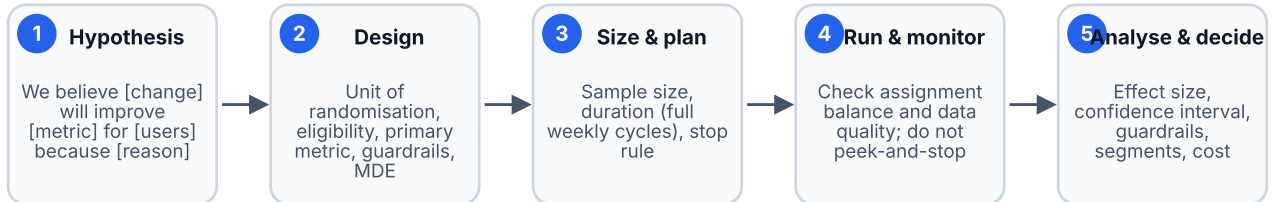
Monthly contribution per paying customer ₹120; monthly churn 5%; CAC ₹800. LTV (simple) = $120 \div 0.05 = \text{₹2,400}$. LTV : CAC = $2,400 \div 800 = 3.0$. Payback = $800 \div 120 = 6.7$ months. Sensitivity: if churn is 8% instead of 5%, LTV = $120 \div 0.08 = \text{₹1,500}$ and the ratio falls to **1.9** — churn is the lever the whole calculation hangs on. Real LTV models discount future cash and use cohort retention curves rather than a constant churn.

Level 3 · Senior PM

Executives care about **payback period** and **cash** as much as LTV : CAC, because LTV is a forecast while cash is real. Present base, low and high scenarios, state which costs are in CAC (paid media only vs fully loaded), and show how sensitive the answer is to churn and to retention flattening. Never present a precise LTV to two decimals from three months of data.

7.9 Experiment design and A/B testing

An **A/B test** randomly assigns eligible users to **control** (current experience) and **variant** (change), then compares a pre-agreed metric. Random assignment is what makes the groups comparable, so differences can be attributed to the change.



A/B test lifecycle. Decide the metric, duration and guardrails before launch. The most common failure is deciding afterwards.

From your course notes (Lecture 27, Companion ch. 6)

- A/B testing compares variants against a defined outcome; it differs from alpha/beta release testing.
- Specify hypothesis, target users, control and treatment, primary metric, guardrails, sample requirements and duration; randomly assign eligible users; compare rates using consistent definitions.
- Choose the **unit of randomisation** and make sure one person is not repeatedly switching groups. Compare **completed orders per eligible user, not just clicks**.
- A higher observed conversion is not automatically a reliable winner: consider uncertainty, sample size, practical impact, implementation cost, accessibility, safety and compliance.
- **A preference poll is not an A/B test.** Confidence level does not mean the probability a variant will work, and no test proves safety with 100% certainty.
- If traffic is too low, run usability research or a different appropriately sized test instead of claiming proof.

When not to A/B test: too little traffic; irreversible or safety/legal changes; strong network effects (a marketplace treatment can contaminate control — consider geo or switchback tests); changes whose value only appears over months.

7.10 Hypothesis testing, p-values and confidence intervals

7.10.1 The idea

- **Null hypothesis (H_0):** there is **no real difference** between control and variant.
- **Alternative (H_1):** there is a difference.
- **p-value:** if H_0 were true, how often would you see a difference *at least this large* purely by chance? It is **not** the probability that the variant works, and it does not measure the size or importance of the effect.
- **Significance level α :** the threshold you fix in advance (commonly 0.05). If $p < \alpha$, you reject H_0 .
- **Confidence interval (CI):** a range of effect sizes consistent with the data; a 95% CI is produced by a procedure that captures the true effect in 95% of repeated experiments. It tells you *how big* the effect might be.

Fully worked example — early fee display

Control: 10,000 eligible users, 7,000 complete an order (**70.0%**). Variant (fee shown early): 10,000 users, 7,250 complete (**72.5%**).

Step 1 — difference: $72.5\% - 70.0\% = +2.5 \text{ percentage points}$ (relative lift $2.5 \div 70 = 3.6\%$).

Step 2 — pooled proportion: $\hat{p} = (7,000 + 7,250) \div 20,000 = 0.7125$.

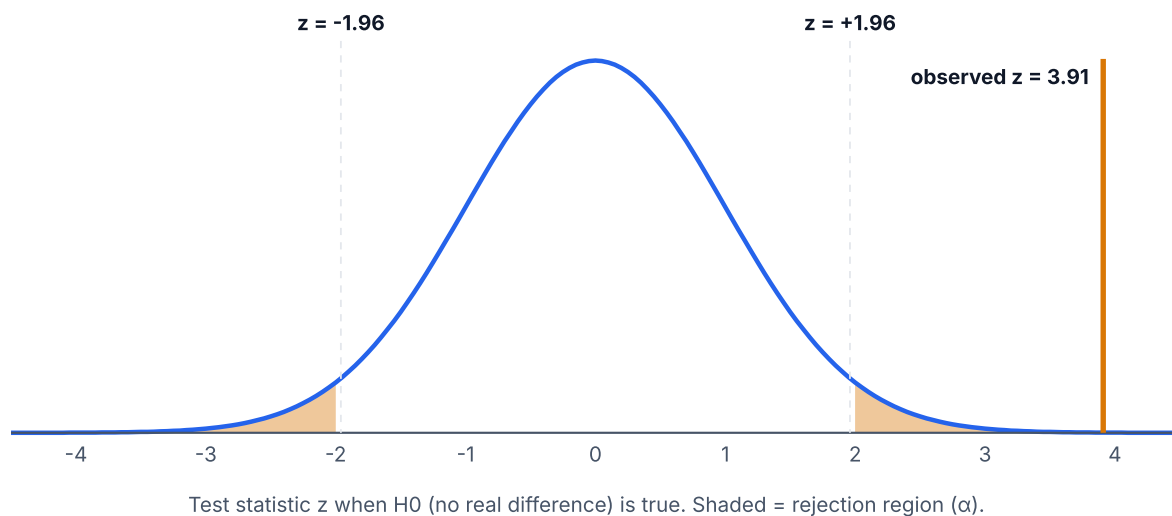
Step 3 — standard error (pooled): $SE = \sqrt{[\hat{p}(1 - \hat{p}) \times (1/10,000 + 1/10,000)]} = \sqrt{(0.7125 \times 0.2875 \times 0.0002)} = \mathbf{0.006401}$.

Step 4 — test statistic: $z = 0.025 \div 0.006401 = \mathbf{3.906}$.

Step 5 — p-value (two-sided): $p \approx \mathbf{0.00009} \rightarrow$ far below 0.05, so we reject H_0 .

Step 6 — 95% CI for the difference (unpooled $SE = 0.006398$): $0.025 \pm 1.96 \times 0.006398 = \mathbf{[+1.25, +3.75] \text{ percentage points}}$.

Decision: the effect is statistically distinguishable from zero and the plausible range is positive. Check the guardrail: complaints were 118 vs 121 per 10,000 users (1.18% vs 1.21%) — no meaningful change. Then ask about practical significance: is +1.2 to +3.8 points of completion worth the cost of shipping and maintaining the change? (Dataset: `ab_test_checkout.csv`.)



The observed $z = 3.91$ falls far inside the rejection region. If there were truly no difference, a z -statistic this extreme would be very rare ($p \approx 0.00009$). Shaded tails = $\alpha = 0.05$ (two-sided).

Two-proportion z -test

$$\hat{p} = (x_1 + x_2) / (n_1 + n_2)$$

$$SE = \sqrt{\hat{p} (1 - \hat{p}) (1/n_1 + 1/n_2)}$$

$$z = (p_2 - p_1) / SE$$

$$95\% \text{ CI for difference} = (p_2 - p_1) \pm 1.96 \times \sqrt{p_1(1-p_1)/n_1 + p_2(1-p_2)/n_2}$$

From your course notes (Lecture 27)

The classroom example compared **23/100 with 37/100** and showed how a stricter significance threshold changes the conclusion. Check it: $\hat{p} = 0.30$, $SE = \sqrt{0.3 \times 0.7 \times 0.02} = 0.0648$, $z = 0.14 \div 0.0648 = \mathbf{2.16}$, two-sided **$p \approx 0.031$** . That is significant at $\alpha = 0.05$ but **not at $\alpha = 0.01$** — the same data, a different threshold, a different conclusion. With only 100 users per arm, the 95% CI is very wide (about +1.4 to +26.6 points), so the data is also consistent with a small effect. *Statistical significance is not a guarantee, and a small sample gives imprecise estimates.*

7.10.2 Type I and Type II errors, and power

	H_0 true (no real effect)	H_0 false (real effect)
Reject H_0 ("we found an effect")	Type I error (false positive), probability α	Correct — probability = power ($1 - \beta$)
Do not reject H_0	Correct	Type II error (false negative), probability β

- **Power** is the probability of detecting a real effect of a given size. Conventional target: **80%**.
- **MDE (minimum detectable effect)** is the smallest effect worth detecting; smaller MDE → much bigger sample.

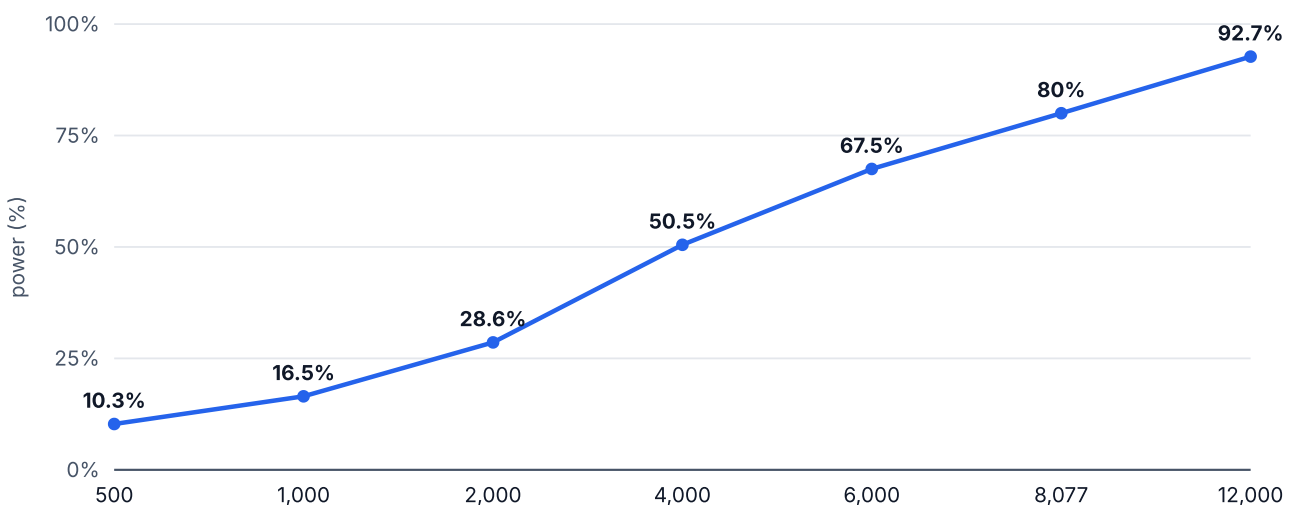
7.10.3 Sample size

$$n \text{ per arm} = (z_{\{1-\alpha/2\}} + z_{\{\text{power}\}})^2 \times [p_1(1-p_1) + p_2(1-p_2)] \div (p_2 - p_1)^2$$

$\alpha = 0.05$ two-sided → $z = 1.96$; power 80% → $z = 0.8416$

Worked example — sizing the checkout test

Baseline 70%; we care about detecting +2 points (72%). $n = (1.96 + 0.8416)^2 \times [0.70 \times 0.30 + 0.72 \times 0.28] \div 0.02^2 = 7.849 \times 0.4116 \div 0.0004 \approx \mathbf{8,077 \text{ users per arm}}$. Other MDEs (same baseline, $\alpha = 0.05$, power 80%): +1 point needs $\approx \mathbf{32,600 \text{ per arm}}$; +5 points needs $\approx \mathbf{1,250 \text{ per arm}}$. Halving the MDE roughly quadruples the sample.



Power to detect a +2-point lift (70% → 72%) by sample size per arm. Computed from the normal approximation. Below ~8,000 users per arm, a real +2-point effect is more likely missed than found. Running underpowered tests wastes time and teaches teams false lessons.

7.10.4 Practical rules for trustworthy experiments

- **Fix the sample size or use a sequential method designed for peeking.** Stopping the moment $p < 0.05$ inflates false positives (the “peeking problem”).
- **Run full weekly cycles** to cover weekday/weekend behaviour.
- **Check sample ratio mismatch (SRM):** if you planned 50/50 and observed 52/48 with large counts, assignment may be broken and results untrustworthy.

- **Pre-register one primary metric;** treat others as secondary. Testing many metrics or segments without correction creates false positives (multiple comparisons).
- **Separate statistical from practical significance.** A tiny, "significant" lift may not justify the cost; a larger, "non-significant" one may merit a bigger test.
- **Novelty effects:** an early lift may fade. Where stakes are high, hold out a group longer.

Interview perspective

"How would you know if an A/B result is real?" → fixed hypothesis and metric; adequate power; random assignment verified; check guardrails and SRM; look at the CI, not just the p-value; consider practical impact, cost and risk; and if traffic is too low, use qualitative research or a longer or larger test.

7.11 Correlation versus causation

Two things moving together does not mean one causes the other. Causes of misleading correlation: **confounding** (a third factor drives both), **reverse causation**, **selection bias**, and **Simpson's paradox**.

Worked example — Simpson's paradox from an unbalanced split

A team compares two checkout designs but, due to a faulty rollout, the variant gets mostly desktop traffic.

Segment	Control (conv / users)	Variant (conv / users)
Desktop (high converting)	60 / 200 = 30.0%	232 / 800 = 29.0%
Mobile (low converting)	64 / 800 = 8.0%	15 / 200 = 7.5%
Overall	124 / 1,000 = 12.4%	247 / 1,000 = 24.7%

The variant looks **twice as good overall** yet is **slightly worse in both segments**. The overall gap comes from the *traffic mix*, not the design. The cure is correct randomisation (and segment-aware analysis), not trusting the aggregate.

From your course notes (Lecture 21; Extra topic 4)

Correlation alone does not prove causation. A relationship between activation and retention may be useful but does not prove the activation action *causes* retention. To test causality, run an experiment.

7.12 Segmentation in analytics

Always ask “for whom?” Useful cuts: acquisition channel, platform/app version, geography, new vs returning, plan/tier, cohort, device class, behavioural segments (power users vs casual). Use the synthetic data to see it:



Share of signups that became buyers, by acquisition channel (synthetic dataset). Synthetic data. Channels with high volume are not necessarily high quality; compare conversion and, in real life, retention and CAC before reallocating spend. These are small samples; do not generalise.

7.13 Event instrumentation and data quality

Instrumentation is adding tracking so product behaviour becomes data. Do it from a **tracking plan** written *before* development.

Event name	Fires when	Key properties	Owner
checkout_started	User opens the checkout screen	cart_value, item_count, platform	Checkout team
fee_displayed	A delivery fee/estimate is shown	fee_amount, is_estimate, source	Pricing
payment_attempted	User taps Pay	method, amount	Payments
payment_succeeded / payment_failed	Gateway responds	method, error_code (no card data)	Payments
order_placed	Backend confirms the order	order_id, amount	Orders

A well-formed event (illustrative):

```
{
  "event": "checkout_started",
  "timestamp": "2026-03-14T18:22:05+05:30",
  "anonymous_id": "a1b2-c3d4",
  "properties": { "cart_value": 520, "item_count": 3, "platform": "ios" }
}
```

Data quality checklist

- **Naming convention** (`object_action`, snake_case) and a single owner of the schema.
- **Test in staging** before release; compare dashboard totals with the authoritative system (orders table).
- **Deduplicate** (retries, double fires) and **handle bots/internal traffic**.
- **Time zones** and clock skew: define the reporting zone.
- **Late and lost events**: mobile events batch and can arrive late; define a lag window.
- **Privacy**: minimise personal data; never put card details, emails or phone numbers into analytics events; honour consent and retention rules.

From your course notes (Lecture 35; Companion ch. 9)

Do not send personal identifiers such as email addresses or phone numbers to Google Analytics — privacy and platform rules apply. Request a tracking plan before development, test events in staging, compare dashboards with the authoritative order system, minimise personal data, limit access, honour retention rules, and never send card details to an analytics event. Decide what you want to learn **before** setting up the tool; low use alone is not enough reason to remove a feature.

7.14 SQL for Product Managers

Many PM interviews and day-to-day questions need basic SQL. These queries run against the synthetic dataset (`users`, `events`, `orders`). They use portable SQL; date functions differ by database (SQLite uses `strftime`, PostgreSQL uses `DATE_TRUNC`, BigQuery uses `DATE_TRUNC / FORMAT_DATE`).

From your course notes (Lecture 34)

Relational databases organise data in tables of rows and columns; SQL lets an authorised user select, filter and combine data. `SELECT * FROM Customers WHERE Country='UK' AND City='London'` — `AND` requires both conditions, `OR` either. Check whether **cancelled orders, time zones or missing values** change the answer, and do not expose personal details unnecessarily. *Contrary to a claim in the lecture, many PMs do query data and some interviews test SQL or joins.*

1 · Basics: GMV, AOV and counts (delivered orders only)

```
SELECT COUNT(*)           AS delivered_orders,  
       SUM(amount_inr)    AS gmv_inr,  
       ROUND(AVG(amount_inr),2) AS aov_inr  
FROM orders  
WHERE status = 'delivered';
```

Result: 320 | 122874 | 383.98 . (GMV here excludes cancelled orders — a definition you must state.)

2 · Cancellation rate

```
SELECT COUNT(*) AS orders,  
       SUM(CASE WHEN status = 'cancelled' THEN 1 ELSE 0 END) AS cancelled,  
       ROUND(100.0 * SUM(CASE WHEN status = 'cancelled' THEN 1 ELSE 0 END) / COUNT(*),2) AS cancellation_rate  
FROM orders;
```

Result: 334 | 14 | 4.19 .

3 · Monthly active buyers and GMV (GROUP BY)

```
SELECT strftime('%Y-%m', order_date) AS month,  
       COUNT(DISTINCT user_id)       AS active_buyers,  
       SUM(amount_inr)               AS gmv_inr  
FROM orders  
WHERE status = 'delivered'  
GROUP BY month  
ORDER BY month;
```

Result: 2026-01 | 55 | 23128 , 2026-02 | 107 | 45400 , 2026-03 | 136 | 54346 .

4 · Funnel by distinct users

```
SELECT event_name, COUNT(DISTINCT user_id) AS users  
FROM events  
GROUP BY event_name  
ORDER BY users DESC;
```

Result: app_open 792 → view_menu 716 → add_to_cart 423 → checkout_start 322 → payment_success 284. Step conversions: $716/792 = 90.4\%$, $423/716 = 59.1\%$, $322/423 = 76.1\%$, $284/322 = 88.2\%$. *This counts users who ever reached a step, so it is not a strict ordered funnel.*

5 · JOIN: which acquisition channels produce buyers?

```
SELECT u.acquisition_channel,
       COUNT(DISTINCT u.user_id) AS users,
       COUNT(DISTINCT o.user_id) AS buyers,
       ROUND(100.0 * COUNT(DISTINCT o.user_id) / COUNT(DISTINCT u.user_id), 1) AS bu
FROM users u
LEFT JOIN orders o ON o.user_id = u.user_id AND o.status = 'delivered'
GROUP BY u.acquisition_channel
ORDER BY buyer_pct DESC;
```

Result: organic 370/142 (38.4%), search_ads 130/49 (37.7%), referral 106/32 (30.2%), paid_social 194/51 (26.3%). A *LEFT JOIN* keeps users with no orders; an *INNER JOIN* would silently drop them and overstate conversion.

6 • Repeat purchase (subquery)

```
SELECT COUNT(*) AS buyers,
       SUM(CASE WHEN n ≥ 2 THEN 1 ELSE 0 END) AS repeat_buyers,
       ROUND(100.0 * SUM(CASE WHEN n ≥ 2 THEN 1 ELSE 0 END) / COUNT(*), 1) AS repea
FROM (SELECT user_id, COUNT(*) AS n
      FROM orders WHERE status = 'delivered' GROUP BY user_id);
```

Result: 274 | 44 | 16.1 .

7 • Window function: rank customers by spend

```
SELECT user_id, SUM(amount_inr) AS spend,
       RANK() OVER (ORDER BY SUM(amount_inr) DESC) AS rnk
FROM orders WHERE status = 'delivered'
GROUP BY user_id
ORDER BY rnk
LIMIT 5;
```

Result: U0648 (1,229), U0483 (1,210), U0246 (1,160), U0665 (1,040), U0751 (1,039).

SQL habits PMs should build: state the metric definition in a comment; check row counts before and after JOINS (duplicates inflate sums); filter to the right status; be explicit about time zones; never

`SELECT *` on personal data; sanity-check totals against an authoritative dashboard.

Practice: write the query (answers)

Q: Average discount per delivered order? `SELECT ROUND(AVG(discount_inr),2) FROM orders WHERE status='delivered';` Q: GMV by city, highest first? `SELECT city, SUM(amount_inr) AS gmv FROM orders WHERE status='delivered' GROUP BY city ORDER BY gmv DESC;` — Result: Mumbai 40,089; Delhi 34,279; Bengaluru 34,022; Pune 14,484. Q: Users who started checkout but never paid? Use `checkout_start` users minus `payment_success` users (e.g. `NOT IN` or `LEFT JOIN ... WHERE p.user_id IS NULL`).

7.15 Excel for product analytics

Excel is enough for quick funnels, cohorts and A/B maths.

Task	Formula
Step conversion	<code>=C3/C2</code>
AOV	<code>=SUM(Amount)/COUNT(OrderID)</code>
Conditional sum / count	<code>=SUMIFS(Amount, Status,"delivered", City,"Mumbai"),</code> <code>=COUNTIFS(Status,"cancelled")</code>
Retention %	<code>=ActiveInWeek/CohortSize</code>
z for two proportions	<code>=(p2-p1)/SQRT(pp*(1-pp)*(1/n1+1/n2))</code> where <code>pp=(x1+x2)/(n1+n2)</code>
Two-sided p-value	<code>=2*(1-NORM.S.DIST(ABS(z),TRUE))</code>
Pivot table	Insert → PivotTable: rows = channel, values = count of orders, average of amount
Lookup	<code>=XLOOKUP(user_id, Users[user_id], Users[city])</code>

Build a **tracking sheet** per experiment: hypothesis, dates, n, conversions, z, p, CI, guardrails, decision. It doubles as a decision log.

7.16 Dashboards and tools: Power BI, Mixpanel, Amplitude, GA4

Tool	Best for	Notes
Mixpanel / Amplitude	Event-based product analytics : funnels, retention, cohorts, user paths, segmentation, often with experimentation add-ons	Needs a good tracking plan; self-serve for PMs
Google Analytics 4 (GA4)	Web/app traffic and acquisition; events with parameters; “key events” (formerly called conversions)	Do not send personal identifiers; sampling and thresholds can apply to reports
Power BI	Business dashboards over many sources (SQL, Excel, APIs) with a data model and DAX measures	Good for exec reporting; refresh schedule and access control matter
SQL + notebooks	Custom analysis, joins, reproducible work	Needs data access and review
Spreadsheet	Fast calculations and small tests	Easy to break silently; document formulas

From your course notes (Lecture 35)

The instructor demonstrated setting up an analytics property and web data stream, adding a tracking tag and viewing recent users and page activity. Compare activity across pages and time periods but **validate definitions, data quality and sample size** before deciding what to change. **Reconcile metric definitions when reports disagree**. Setup screens change; follow current official documentation.

A simple Power BI measure (DAX), to show the idea:

```
Conversion Rate = DIVIDE( DISTINCTCOUNT(Orders[user_id]), DISTINCTCOUNT(CheckoutStar
```

Dashboard design rules: start with the decision; one headline metric per question; show the trend *and* the target; include definitions and “last refreshed”; put guardrails next to the metric they protect; avoid pie charts for more than 3–4 parts; annotate releases and incidents.

7.17 Experiment readout template (worked example)

Readout: Early fee display (hypothetical)

Decision requested: Ship early fee display to 100% of users? **Recommendation:** Yes, with a 5% holdout for 4 weeks. **Hypothesis:** Showing the delivery fee in the cart (not only at payment) will increase checkout completion by reducing surprise. **Design:** randomised by user; 10,000 eligible per arm; 14 days; primary = completed orders ÷ eligible users; guardrails = complaint rate, payment failures, AOV. **Result:** 70.0% → 72.5% (+2.5 pts; 95% CI +1.25 to +3.75; $z = 3.91$; $p < 0.001$). Complaints 1.18% vs 1.21% (n.s.). No SRM detected (10,000 vs 10,000). **Segments (exploratory, not powered):** directionally positive on new and returning users. **Risks / caveats:** novelty effect possible; estimate accuracy depends on the fee service; two-week window. **Next steps:** staged rollout; monitor the estimate error rate; re-run on high-fee cities.

7.18 Source-flagged corrections

- The lecture's 20–30% *"ideal"* activation and referral claims are not universal benchmarks.
- A single page-load time cannot by itself establish API latency; browser rendering, network and other components also contribute.
- A **preference poll is not an A/B test**; **confidence level is not the probability a variant will work**.
- A confidence interval is not a probability statement about a single experiment's true value (frequentist interpretation).
- **GMV ≠ revenue ≠ profit**. Define each.

7.19 Exercises with model answers

Exercise 1 • Funnel diagnosis

A funnel shows: Visit 20,000 → Product page 12,000 → Cart 3,000 → Checkout 2,100 → Order 1,680. (a) Compute step conversions. (b) Which step is lowest? (c) Name three hypotheses and the data you would pull for each.

Model answer

(a) 60.0%, 25.0%, 70.0%, 80.0%; overall 8.4%. (b) Product page → Cart at **25%**. (c) H1: add-to-cart is hard to find on mobile — segment by device and check click heatmaps/events. H2: prices or fees not clear until cart — compare with sessions where a fee estimate was shown earlier; run a usability test. H3: wrong traffic mix (new channel bringing low-intent users) — split by acquisition channel. State the largest absolute loss is Visit→Product page (8,000), but the product page is a browsing step; the cart step is more actionable.

Exercise 2 · Is the test result real?

Control: 4,000 users, 600 conversions. Variant: 4,000 users, 660 conversions. Compute the lift, z-statistic, p-value and 95% CI. Would you ship?

Model answer

$p_1 = 15.0\%$, $p_2 = 16.5\%$; diff = +1.5 pts (relative +10%). Pooled $\hat{p} = 1,260/8,000 = 0.1575$; $SE = \sqrt{(0.1575 \times 0.8425 \times (2/4,000))} = 0.008141$; $z = 0.015/0.008141 = \mathbf{1.84}$; $p \approx \mathbf{0.066}$ (two-sided) → not significant at 0.05. Unpooled $SE = \sqrt{(0.15 \times 0.85/4,000 + 0.165 \times 0.835/4,000)} = 0.008155$ → 95% CI $\approx \mathbf{-0.1 \text{ to } +3.1 \text{ pts}}$, which includes zero. Do not claim a win; the effect might be real but the test is underpowered for a 1.5-point effect (roughly 9,250 per arm would be needed at 80% power). Options: extend the test to the planned size, or accept the uncertainty if the change is cheap and reversible.

Exercise 3 · Unit economics

Subscription: ₹300/month price; gross margin 70%; monthly churn 6%; CAC ₹1,500. Compute LTV (simple), LTV:CAC and payback. Which assumption would you stress-test?

Model answer

Monthly contribution $\approx 300 \times 0.70 = ₹210$. $LTV = 210 \div 0.06 = \mathbf{₹3,500}$. $LTV:CAC = 3,500 \div 1,500 = \mathbf{2.33}$. Payback = $1,500 \div 210 = \mathbf{7.1 \text{ months}}$. Stress-test churn (e.g., 9% → LTV ₹2,333, ratio 1.56) and whether churn is really constant — real cohorts usually churn faster in month 1 and flatten.

Exercise 4 · Pick a North Star

Choose an NSM for a language-learning app and name two guardrails and one way the team could game it.

Model answer

NSM: *learners who complete ≥ 3 lessons per week* (value = actual learning). Guardrails: lesson quality rating / test scores; 30-day retention; complaints about notification spam. Gaming: making lessons trivially short or sending aggressive streak-guilt notifications to inflate lesson counts — hence the quality and wellbeing guardrails.

7.20 Interview perspective

“How would you measure the success of a new feature?” — structure

1) Goal and user: what outcome and for whom. 2) **Primary metric** (outcome), 3) **input/leading metrics** that explain it, 4) **guardrails**, 5) **how you will measure causally** (A/B or phased rollout), 6) **timeframe** and success threshold, 7) **segments** to check, 8) **decision rule** (ship, iterate, stop). Close with what you would do if the result is inconclusive.

“Our retention is dropping. How do you investigate?” — 60-second answer

“I'd first confirm the definition and whether it's real — check tracking and seasonality. Then I'd see *which* cohorts: new vs old users, by channel, platform, geography. If new cohorts are weaker, I'd look at acquisition quality and onboarding changes; if everyone is weaker after a date, at releases or competitor moves. I'd pair the quantitative cut with 5–8 interviews of recently churned users to learn why, form two or three hypotheses, and test the cheapest first. I'd also check guardrails so a fix doesn't hurt revenue or satisfaction.”

7.21 Key takeaways

- A metric is only as good as its **definition** (event, numerator, denominator, window, exclusions).
- Use **leading + lagging** metrics, an NSM with a **metric tree**, and **guardrails**.
- Funnels, cohorts and segmentation show **where** things change; research and experiments show **why** and **whether it works**.
- **GMV $\neq$ revenue $\neq$ profit**. Check contribution margin and CAC payback before celebrating growth.
- Experiments need **pre-registered metrics, adequate power, clean randomisation and guardrails**. p-values are not the probability a variant works; CIs show effect size.
- **Correlation is not causation** — Simpson's paradox is the standard warning.
- Instrument from a **tracking plan**, validate data quality, and keep personal data out of analytics events.

7.22 Check your understanding

CHECK YOUR UNDERSTANDING

A funnel has 800 of 1,000 visitors completing a step. What are step conversion and drop-off?

- A. 80% and 20%
- B. 20% and 80%
- C. 100% and 0%
- D. Cannot tell without revenue

Why: Step conversion = $800 \div 1,000 = 80\%$; drop-off = $1 - 0.80 = 20\%$.

AOV is best defined as...

- A. Total revenue $\div$ number of users
- B. Total order value $\div$ number of orders
- C. Profit $\div$ orders
- D. GMV $\div$ MAU

Why: AOV = total order value $\div$ number of orders for the same period and definition. It says nothing by itself about profit.

A test gives $p = 0.03$. Which statement is correct?

- A. There is a 97% chance the variant is better.
- B. If there were truly no difference, results this extreme would occur about 3% of the time.
- C. The effect is large.
- D. The result will replicate 97% of the time.

Why: A p-value is a probability about the data under the null hypothesis. It is not the probability the variant works and says nothing about effect size.

Which change makes the required sample size per arm increase the most?

- A. Halving the minimum detectable effect
- B. Raising power from 80% to 85%
- C. Doubling traffic
- D. Using one more guardrail

Why: Sample size scales roughly with $1/(\text{effect size})^2$, so halving the MDE roughly quadruples the sample.

LTV (simple) = ₹120 ÷ 0.05 is ₹2,400. If churn rises to 8%, LTV becomes...

- A. ₹960
- B. ₹1,500
- C. ₹2,400
- D. ₹3,840

Why: $120 \div 0.08 = ₹1,500$. LTV is very sensitive to churn.

In Simpson's paradox, the overall result is reversed relative to every segment because of...

- A. Rounding
- B. Unequal segment mix between groups
- C. Too many users
- D. Using a two-sided test

Why: When the variant gets a different mix of high- and low-converting segments, the aggregate can mislead; proper randomisation and segmenting fix it.

Which should NOT be sent as an analytics event property?

- A. cart_value
- B. platform
- C. card_number
- D. item_count

Why: Card details (and typically emails and phone numbers) must never go into analytics tools.

7.23 Recommended resources

- [Google Analytics Help](#) — events, key events, privacy.
- [Mixpanel Docs](#) · [Amplitude Docs](#) — product analytics concepts and setup.
- [Power BI documentation](#) — dashboards and DAX.
- [Evan Miller: A/B testing tools and sample-size calculator](#) — practical references for sample size and sequential testing.

- [Kohavi, Tang, Xu — *Trustworthy Online Controlled Experiments*](#) — the standard book on online experimentation.
- [SQLBolt](#) — free interactive SQL lessons.