

Scoring who sees the offer

A live mobile fighting game showed its starter pack to everyone. We scored players on thirty minutes of behaviour, showed it to one player in twenty, and sold the same number of packs.

CAUSAL A/B • 64,302 DECISIONS • 18 DAYS • RANDOMISED CONTROL

The situation

The starter pack fired for every player crossing a threshold. There was no control group, so there was no way to know whether the offer was doing anything, or whether the players who bought would have bought regardless.

What we did

Each player was scored from their first thirty minutes of in-game behaviour. The pack was shown only to players the model flagged. A randomised control was held back so the outcome could be attributed rather than inferred.

Results

8.7×

conversion per offer shown, against control

95%

fewer interruptions, same packs sold

Pooled, conversion moved from 1.0% to 8.7%. If the model made no difference, a gap that size would turn up by chance less than once in a million tests. It replicated independently across both arms, at 8.0× and 9.4×. Half of all buyers sat within the model's top 10% of players.

Retention, sessions, playtime and average revenue per user (ARPU) were flat. Removing 95% of offer impressions is exactly the kind of change that can cost something elsewhere; the flat metrics are the evidence it did not.

Throughput

- 64,302 individual decisions across 18 days, one every 24 seconds
- Approximately 30 minutes from install to a scored decision
- 2,215 players re-scored mid-session as behaviour changed

Caveat

How impressions were counted. The game's telemetry had no popup-impression event, so an impression means the player was eligible and flagged, not a confirmed rendered popup. The conversion result is measured against a randomised control and does not rely on that stand-in. The 95% efficiency figure does.

Client identifying details removed. Figures are those cleared for external use; where a result is not statistically powered it is described as such rather than quoted. hello@ilara.ai