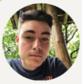
Balancer

WHAT 961 TOKEN LAUNCHES TAUGHT US

A DATA-DRIVEN RESEARCH ON LIQUIDITY BOOTSTRAPPING POOLS

NEO EMPRESARIAL | Dune

DATA POWERED BY DUNE



Fábio Mendes
Head of Product

We analyzed nearly a thousand LBPs and found a clear pattern: you can't engineer success, but you can engineer away the most common failures. This report gives crypto teams the tools to bootstrap liquidity with confidence instead of guessing.
The token launch landscape is shifting: projects are raising again, communities want real ownership, and fair price discovery matters more than ever. As this new wave takes shape, Balancer has the infrastructure and the data to lead what comes next.

Alsie Liu
Marketing lead at Dune

Blockchain data is inherently transparent. Every swap, every bot transaction, every liquidity event is recorded and publicly accessible. What's been missing is a platform that brings it all together so you can actually work with it at scale. Dune is excited to power the most comprehensive LBP analysis to date, covering 961 launches, four years of data, all in one place.

Balancer

WHAT 961 TOKEN LAUNCHES TAUGHT US

A data-driven research about
Liquidity Bootstrapping Pools

Contents

1. Context and Motivation

Most LBP configurations are chosen arbitrarily. Teams copy what others did or guess based on intuition.

We analyzed 961 LBPs on Balancer to identify which parameters affect outcomes and which do not. The dataset spans Balancer V1 and V2 across Ethereum, Polygon, and Arbitrum, covering launches from 2021 through 2024.

The result is a framework that separates configuration decisions into two categories: parameters that mechanically influence risk, and parameters that are preference only. Configuration cannot guarantee success. Marketing, community, and project quality dominate outcomes. But configuration can guarantee failure. This playbook shows you which settings create mechanical failure modes and how to avoid them.

2. The Dataset

The foundation of this Proof of Concept comes from onchain transactions queried via Dune. We extracted every LBP deployment event, swap, and liquidity operation from Balancer smart contracts.

Our extraction targeted both Balancer V1 and V2 smart contracts, capturing the full lifecycle of token launch events across major networks including Ethereum Mainnet, Polygon, and Arbitrum.

Raw blockchain data contains noise: test deployments, abandoned pools, failed transactions. We filtered out pools with zero volume, missing liquidity, or duration under 12 hours. Final sanitized dataset: 961 unique LBPs.

Data Structure

To effectively model the relationship between configuration and success, we architected the data into two distinct logical schemas: **Table A (Configuration)** and **Table B (Performance)**. This separation is the core of our analytical strategy.

It treats the LBP event as a controlled experiment, where Table A represents the "Input Variables" (the levers a project team can pull) and Table B represents the "Output Variables" (the market's mechanical response). By isolating these two domains, we prevent the analysis from being contaminated by factors outside a team's control.

Table A is designed to capture the structural "DNA" of every pool. It moves beyond basic metadata by including engineered features that describe the physics of the launch. For instance, rather than simply recording the starting and ending weights, we calculated the **weight_slope,** a derived metric that quantifies the velocity of downward price pressure.

Similarly, we normalized (made easier to compare) the **duration_hours** and **swap_fee_pct** to identify standard operating ranges. Crucially, this table also enriches the raw data with contextual tags, such as the collateral type (Stablecoin vs. Volatile Asset) and timing factors (Weekend vs. Weekday launches).

This allows the model to understand that a steep slope might behave differently on a volatile weekend market compared to a stable weekday launch.

Table A (Configuration)		
Column Name	**Source**	**Calculation / Logic**
pool_address	Raw	Unique Contract Address (Primary Key).
chain	Raw	Blockchain network (Ethereum, Arbitrum, Polygon, etc.).
version	Raw	Balancer V1 (Legacy) or V2 (Standard).
start_timestamp	Raw	UTC Timestamp of the LBP creation/launch.
duration_hours	Calc	(end_timestamp - start_timestamp) / 3600.
start_weight_proj	Calc	Initial weight (0.0-1.0) of the Project Token (e.g., 0.99).
end_weight_proj	Calc	Final weight (0.0-1.0) of the Project Token (e.g., 0.20).
start_weight_reserve	Calc	Initial weight of the Collateral Token (e.g., 0.01).
end_weight_reserve	Calc	Final weight of the Collateral Token (e.g., 0.80).
weight_slope	Calc	Abs(end_weight_proj - start_weight_proj) / duration_hours.
swap_fee_pct	Raw	Trading fee charged to swappers (e.g., 0.01 for 1%).
collateral_is_stable	Calc	1 if collateral is USDC/DAI/USDT, 0 if volatile (e.g., WETH).
is_weekend	Calc	1 if launch day is Saturday or Sunday, 0 otherwise.
weekend_pct	Calc	Percentage of total duration that overlaps with a weekend.

Table B was engineered to solve the “Vanity Metric” problem. In early iterations, we found that raw metrics like Total Volume were heavily influenced by external marketing hype, which cannot be predicted by onchain data. To address this, we developed a set of “Physics-based” performance metrics:

- **dump_pressure:** a ratio comparing sell volume to buy volume, to measure the net sentiment of traders.
- **turnover_ratio:** calculated by cross-referencing trading volume with the Balancer Vault's initial liquidity events, providing a true measure of capital efficiency.
- **bot_tx_ratio:** calculated through block-by-block transaction analysis, identifying how susceptible a pool configuration was to sniper bots in its opening moments.

Table B (Performance)		
Column Name	**Source**	**Calculation / Logic**
pool_address	Raw	Foreign Key (Links to Table A).
volume_usd	Calc	Total USD value of all swaps during the LBP.
unique_buyers	Calc	Count of unique wallet addresses that executed a BUY.
price_retention	Calc	(Avg Price Last 5 Blocks) / (Avg Price First 5 Blocks). (Measures if price held up).
volatility_score	Calc	Mean Price / Standard Deviation of Price. (Measures turbulence).
dump_pressure	Calc	Total Buy Volume (USD) / Total Sell Volume (USD). (>1.0 means net selling).
volume_time_skew	New	Time-weighted center of volume (0.0 = Start, 1.0 = End). Ideal is ~0.5.
whale_dominance_pct	New	Volume of Top 1% Trades / Total Volume. (Measures centralization risk).
turnover_ratio	New	Total Volume / Initial Liquidity. (Measures capital efficiency).
bot_tx_ratio	New	Trades in First 5 Blocks / Total Trades. (Measures sniper activity).
bot_extraction_usd	New	Bot Sells - Bot Buys (during first 5 blocks). Negative means bots are holding.
price_discovery_stability	New	Volatility calculated only on the last 10% of trades. (Did price settle?).

These metrics provide a granular view of an LBP's structural health, unrelated to the hype surrounding the project. To our knowledge, this is the largest structured analysis of LBP configuration and performance to date.

2.1. Success Score Distribution

The histogram below visualizes the distribution of the final_success_score across all 961 analyzed pools. This composite metric aggregates four performance indicators into a single value ranging from 0.0 (Total Failure) to 1.0 (Perfect Launch).

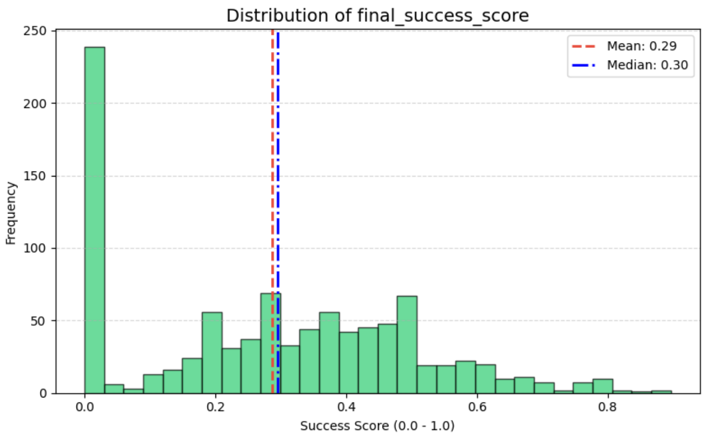


Figure 1 — Distribution of Final Success Scores across LBPs.

To interpret this distribution, it is crucial to understand how the score was calculated. We utilized a weighted average that prioritizes Price Retention (40%), reflecting our core philosophy that a successful LBP must sustain its value. The remaining weight is distributed among Unique Buyers (20%) and Dump Pressure (20%) to measure demand quality, followed by Volatility (20%) to penalize instability and manipulation.

Statistical analysis: the distribution exhibits a positive skew (Skewness ≈ 1.18), with the majority of the data concentrated towards the lower end of the spectrum. This structure manifests in three patterns:

The “mediocrity trap”: the mean score is 0.38, and the median is remarkably lower at 0.32. This indicates that the vast majority of LBPs struggle to achieve high marks across all categories simultaneously.

The “elite” tail: only the top 10% of pools achieved a score above 0.60, and the maximum recorded score was 0.91. This sharp drop-off confirms that achieving a “perfect” launch (one with high volume, low dump pressure, and excellent price retention) is a rare statistical outlier.

Component drivers: decomposing the score reveals that price retention and unique buyers are the primary drag factors (average sub-scores of ~0.16 and ~0.14 respectively). While many pools manage to be technically stable (low volatility, high skew scores), most fail to attract significant unique users or hold their price post-launch.

The table below presents the **five highest-scoring LBPs according to the success score metric**, together with the parameters used in each configuration:

	1º	2º	3º	4º	5º
chain	ethereum	ethereum	ethereum	ethereum	ethereum
duration_hours	72	48	72.18	68	120
start_weight_proj	0.99	0.99	0.99	0.99	0.99
end_weight_proj	0.01	0.01	0.01	0.01	0.01
start_weight_reserve	0.01	0.01	0.01	0.01	0.01
end_weight_reserve	0.99	0.99	0.99	0.99	0.99
weight_slope	0.01	0.02	0.01	0.01	0.01
swap_fee_pct	0.03	0.03	0.03	0.03	0.03
collateral_is_stable	0	1	1	1	0
is_weekend	1	0	0	0	1
weekend_pct	0.26	0	0	0.19	0.26
final_success_score	0.9	0.87	0.84	0.83	0.83

The question becomes: what configurations are teams actually using, and do any of them reliably produce better outcomes?

2.2. Configuration Patterns

This section analyzes the design choices used in Liquidity Bootstrapping Pools, focusing on duration, timing, and weight configuration. By examining their distributions, we identify prevailing market standards and the strategic intent behind common parameter selections.

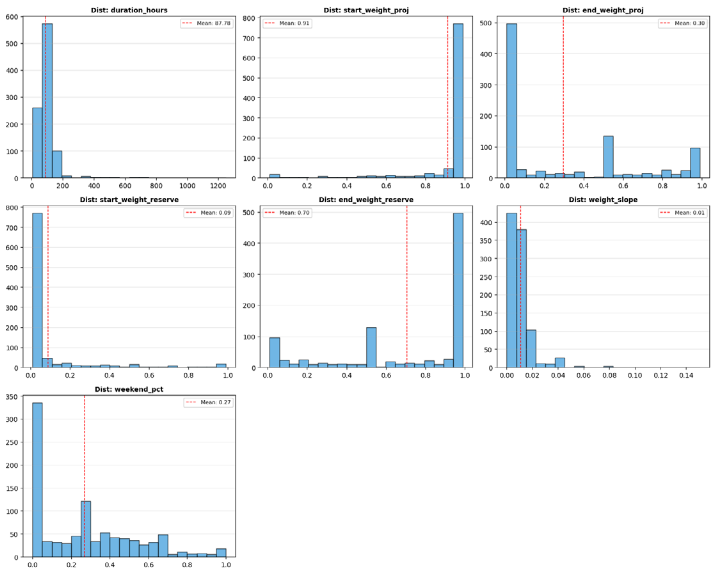


Figure 2 — Empirical distributions of LBP configuration parameters.

Duration and timing: the duration of LBPs (duration_hours) is heavily right-skewed, with a mean of approximately 88 hours (roughly 3.5 days). The high concentration of values between 0 and 100 hours indicates that the market standard is a short, concentrated bootstrapping event rather than a prolonged campaign.

This is further supported by the **weekend_pct** distribution, where a significant spike at 0.0 suggests that many issuers deliberately schedule their pools to avoid weekends, likely to maximize engagement during standard trading hours.

Starting weight strategy: the starting weights exhibit a near-universal consensus on strategy. The **start_weight_proj** (Project Token) is massively concentrated between 0.90 and 1.0, while the **start_weight_reserve** (Collateral Token) is conversely concentrated between 0.0 and 0.10.

This confirms that the primary mechanism for price discovery is to commence with high sell pressure (high project weight) to discourage early bot sniping and allow the market to determine the fair price as weights adjust downward.

Ending weight strategies (bimodal distribution) Unlike the starting weights, the ending weights reveal a divergence in strategy. The end_weight_proj histogram is bimodal, featuring two distinct peaks:

1. **The full flip (Near 0.0):** a large cluster of pools ends with the project token weight near roughly 10-20% (or lower), effectively transferring the majority of liquidity into the reserve asset.
2. **The soft landing (Near 0.5):** a secondary but significant cluster ends at a 0.5 (50%) weight. This suggests a strategy where the LBP is designed to transition smoothly into a standard 50/50 pool for long-term trading immediately after the bootstrapping phase.

Rate of change: finally, **the weight_slope** distribution is tightly clustered around a 0.01 mean, indicating that although durations and end-weights vary, the hourly rate of weight change tends to be kept low and gradual. This consistency suggests that issuers favor a smooth price decay curve over abrupt shifts that could introduce volatility or confuse participants.

2.3. Parameter Correlations

Building on the analysis of individual parameters, you may ask yourself: how do parameters relate to each other? And to final_success_score? The correlation heatmap matrix reveals three findings.

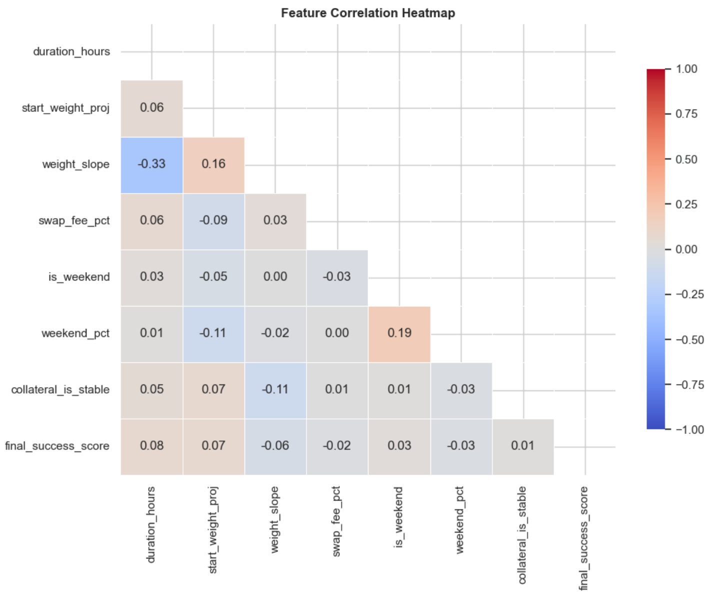


Figure 3 — Correlation structure of LBP configuration parameters and launch outcomes.

1. **duration_hours and weight_slope are coupled:** the strongest relationship in the dataset is the negative correlation (-0.33) between duration_hours and weight_slope. This indicates a necessity rather than a preference: shorter LBPs generally require a steeper weight slope (a faster rate of change) to achieve the target price discovery within a limited timeframe.

Conversely, longer pools tend to adopt a gentler slope to prevent price volatility from becoming too aggressive over an extended period. This confirms that slope and duration cannot be set in isolation, they are coupled variables where one constrains the other.

2. **Parameter independence:** aside from the duration-slope dynamic, the matrix is predominantly populated by near-zero coefficients (gray squares). This suggests that most configuration choices, such as **swap_fee_pct** or whether the collateral is a stablecoin (**collateral_is_stable**), are made independently of the duration or starting weights.

   For the optimization framework, this is a positive finding: it implies high modularity. An issuer can select a swap fee appropriate for their specific liquidity goals without it statistically forcing them into a specific duration or weight strategy.

3. **No single parameter predicts success:** crucially, the bottom row (**final_success_score**) shows extremely low correlations with all individual configuration parameters (ranging from -0.06 to 0.08). This "negative result" implies that **there is no single "magic number"** for any individual parameter that guarantees a successful LBP. A specific duration or starting weight alone does not drive success. Instead, optimal performance is likely a result of:

   a. **External factors:** marketing momentum, project quality, and community engagement.
   b. **Complex combinations:** the interplay of parameters (e.g., a specific slope given a specific duration) rather than their individual values.

This reinforces the need for a holistic framework rather than a rigid template, configuration parameters must be tailored to the specific context of the project rather than strictly following a universal standard.

3. From Prediction to Risk Assessment

The initial goal was to predict final_success_score from configuration parameters. If successful, teams would know before launch whether their settings would produce good outcomes.

The attempt failed. But the failure revealed something useful: while success cannot be predicted, mechanical failure can. The model became a risk auditor rather than a crystal ball.

3.1. Why Success Prediction Failed

Our modeling phase began immediately following the remediation of the data extraction pipeline. After rectifying the event duplication bug, where initialization events corrupted the dataset, and enforcing a strict duration floor to purge test pools, we established a pristine training set of 961 unique LBP launches. The initial objective was ambitious: to predict the **final success score** (a composite of Volume, Retention, and Liquidity) using only the technical configuration parameters defined in Table A.

We subjected this hypothesis to a battery of regression architectures, ranging from linear baselines to complex ensemble methods. We first applied **Ridge Regress**ion to test for linear relationships, which failed to converge on a positive result ($R^2 < 0$). We then escalated to **Random Forest** and finally **XGBoost**, utilizing Randomized Search for hyperparameter tuning. Despite these techniques, the predictive performance hit a distinct signal ceiling, with the coefficient of determination plateauing at approximately **$R^2 \approx 0.056$.**

	RMSE	MAE	R^2
Ridge Regression	0.2155	0.1773	-0.0110
Random Forest	0.2090	0.1692	0.0492
XGBoost	0.2083	0.1716	0.0558

This led to a critical conclusion: the financial success of a token launch is dominated by **extrinsic variables** (marketing, community sentiment, and hype) which are invisible to onchain data.
A perfect configuration cannot force a bad token to succeed, but a bad configuration can mechanically force a good token to fail. Consequently, we determined that building a price predictor was theoretically impossible with this dataset, necessitating a strategic pivot.

3.2. Predicting Failure Instead

Recognizing that we could not predict the magnitude of success (upside), we reframed the problem to predict the probability of structural failure (downside). We abandoned the regression approach and restructured the machine learning architecture as a **binary classification** problem. Instead of asking the model “what will the volume be?”, we asked, “is this configuration likely to result in high selling pressure?”

This yielded immediate results. By isolating specific physics-based failure modes, specifically **dump pressure** (selling volume outweighing buying volume) and **capital inefficiency** (low turnover), we pierced the noise floor that had plagued the earlier models.

Our updated Random Forest classifier achieved an Area Under the Curve (**AUC**) score of **0.759** for predicting high Dump Pressure and **0.752** for Capital Efficiency. These scores are well above the 0.70 threshold typically required for production-grade financial models, validating that while market sentiment is random, the mechanical leverage of an LBP is deterministic and predictable.

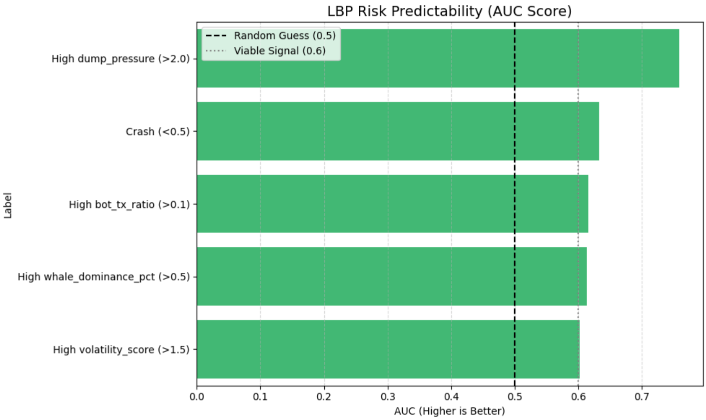


Figure 4 — Predictability of structural risk factors in LBPs (AUC scores)

3.3. Isolating Parameter Effects

To transition from abstract probability scores to actionable engineering rules, we utilized Partial Dependence Plots (PDPs). This technique isolates the marginal impact of a single feature on the predicted outcome by averaging out the effects of all other variables.

By plotting the feature value (X-axis) against the predicted probability of a risk event (Y-axis), we can visualize the precise “response curve” of the market to specific configuration changes. This allows us to distinguish between parameters that merely correlate with success and those that mechanically drive it.

The graphical output of the PDP analysis should be read as a direct function of sensitivity. The vertical axis represents the model’s confidence, expressed as a probability between 0.0 and 1.0, that a specific configuration will result in a failure mode, such as high dump pressure.

A flat line indicates that the parameter has zero marginal impact on risk. Changes to this value are statistically irrelevant to the safety of the pool.

Conversely, a curve with high variance (steep vertical movement) identifies a “Dominant Driver.” For these parameters, we look for “inflection points,” specific thresholds where the risk probability accelerates rapidly. These mathematical elbows serve as the boundaries for our optimization guidelines.

Affected feature: dump pressure

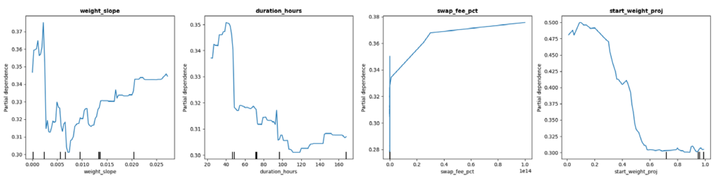


Figure 5 — Partial dependence analysis for dump pressure risk.

Affected feature: turnover ratio

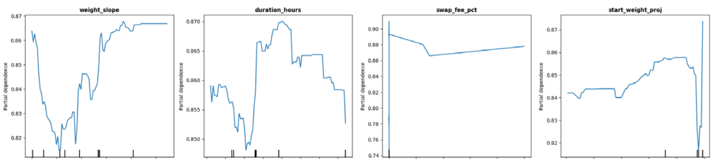


Figure 6 — Partial dependence analysis for turnover ratio (capital efficiency).

3.3.1. Weight slope

The analysis identifies **weight_slope** as the single most significant determinant of LBP safety, exhibiting the highest variance of any tested feature. The risk curve for this parameter is non-linear. In the lower range (0.0 to 0.4), the probability of generating excessive dump pressure remains relatively stable and low. However, the curve exhibits a sharp inflection point at approximately **0.6.**

Beyond this threshold, the probability of a “High Dump” outcome rises sharply, approaching certainty as the slope increases. This confirms that an aggressive slope functions as a mechanical forcing mechanism, mathematically driving a price collapse regardless of buy-side demand. Consequently, the Auditor must flag any slope exceeding 0.6 as a critical “Red Flag,” as the sell pressure overwhelms organic price discovery.

3.3.2. Duration

The **duration_hours** parameter displays an inverse relationship with capital inefficiency, characterized by diminishing returns. The curve begins with high risk at short durations (12—24 hours), indicating that extremely short pools often fail to achieve sufficient capital turnover, likely due to insufficient time for market participants to react.

The risk probability declines steadily as duration increases, stabilizing around the 72-hour mark. Beyond that point, the curve flattens, indicating that extending a pool to 4—5 days provides minimal additional safety benefits.

This data provides a clear optimization window: **the ideal duration for maximizing capital efficiency lies between 48 and 72 hours.** Durations outside this window either incur unnecessary volatility risk (too short) or waste time (too long) without improving performance.

3.3.3. Neutral factors (swap fees and initial weights)

In contrast to the dominant levers, the curves for **swap_fee_pct** and **start_weight_proj** are remarkably flat, exhibiting a variance of less than 5% across their entire functional range. This lack of sensitivity implies that these parameters do not dictate the success or failure of a launch in the same way slope or duration do. Therefore, the “LBP Safety Auditor” should classify these as preference parameters.

Users can be advised that adjusting their swap fee from 1% to 3% will not statistically alter their risk profile, granting them the flexibility to align these settings with their specific marketing or revenue goals without compromising the integrity of the pool.

3.4. Failure Analysis

To validate the model's findings, we compared configurations of failed LBPs against successful ones. "Failed" pools are those with high bot_tx_ratio, high dump_pressure, or severe price collapse. "Successful" pools are those in the top quartile of final_success_score.

3.4.1. Defining Adversarial Activity

Before analyzing the impact of specific parameters on market participants, it is essential to define how adversarial actors are identified.

Bot transactions: Any transaction in the first 5 blocks (~60 seconds on Ethereum). Humans cannot physically observe a new pool, analyze its parameters, and execute a transaction in this window. Activity in this period is programmatic by definition.

Therefore, the **bot_tx_ratio** metric represents the percentage of total trades that occurred during this high-velocity, non-human timeframe.

Whale transactions: Transactions above the 99th percentile of volume for that specific pool. Using a relative threshold rather than absolute value ensures the measure works regardless of pool size.

The **whale_dominance_pct** metric is then derived by summing the volume of these outlier trades and dividing it by the total pool volume. This provides a normalized measure of market concentration, revealing how heavily the price discovery process was dictated by the top 1% of capital deployment, regardless of the pool's absolute size.

To validate our model, we conducted a statistical analysis of the three primary configuration levers: duration, weight structure, and failure categories to distinguish between parameters that provide real protection and those that are merely cosmetic.

3.4.2. The Duration Paradox

Our analysis of the **duration_hours** bucket reveals a non-linear relationship between pool length and adversarial bot participation. The data forms a "U-Curve":

- **Short Pools (<24h):** attract moderate bot activity (~11.8%).
- **Medium Pools (24h—72h):** minimize bot activity (~10.4%).
- **Long Pools (>72h):** attract the highest bot activity (~16.8%).

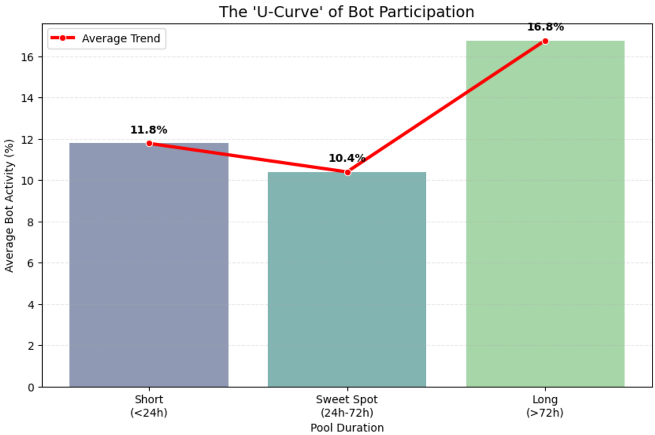


Figure 7 — The U-shaped relationship between LBP duration and bot participation.

Longer launches do not mean safer launches. Extended pools become targets for arbitrage bots exploiting slow price decay. The **optimal window is 24 to 72 hours,** minimizing both capital inefficiency and bot exposure.

3.4.3. The placebo effect of weight ratios

We tested the hypothesis that the **starting weight ratio** (e.g., starting at 95:5 vs. 80:20) acts as a buffer against price crashes. The correlation analysis returned a value of **0.049** for price retention and **-0.098** for Dump Pressure.

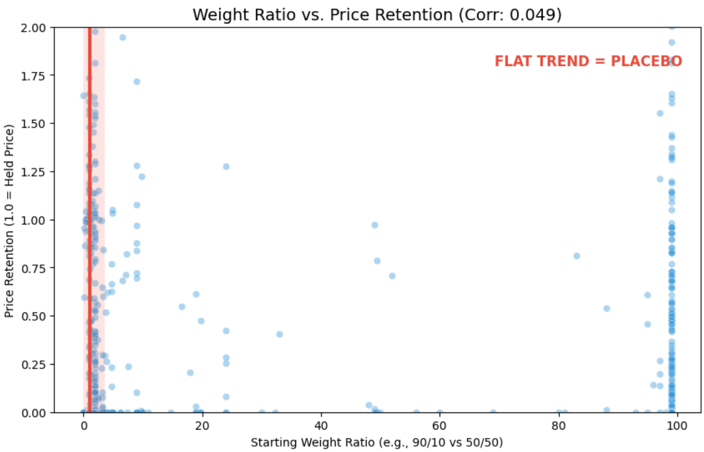


Figure 8 — Starting weight ratio versus price retention (placebo effect).

These near-zero coefficients statistically prove that the weight ratio is a **placebo parameter.** Adjusting the start/end weights does not prevent a dump or guarantee price retention. A token launching with a 50:50 ratio is just as likely to crash as one launching at 90:10 if the demand is absent. Optimization efforts should therefore focus on **slope** (the speed of the change) rather than the **ratio** (the magnitude of the weights).

3.4.4. What distinguishes a "healthy" pool?

1. **The high-weight shield:** the most significant difference lies in the start_weight_proj. Healthy pools had an average starting weight of **~94.1%**, while dumped pools averaged **~86.7%.** This suggests that a dominant initial project weight (e.g., 95% or 99%) creates a high-price buffer that absorbs early volatility. Lower starting weights appear to expose the pool to immediate selling pressure that the liquidity curve cannot dampen.
2. **The duration trap:** the sniped LBPs had the longest average duration (**~100 hours**), compared to **~87 hours** for healthy pools. This corroborates our earlier finding: excessive duration does not aid price discovery, it creates a prolonged window for value extraction by sophisticated arbitrageurs.

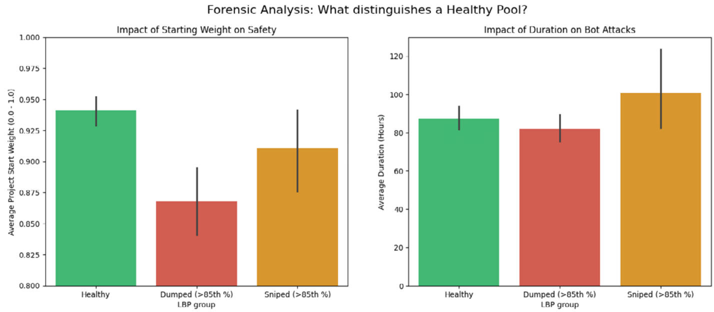


Figure 9 — Forensic comparison of healthy versus failed LBP configurations.

Based on this forensic evidence, the mathematically optimal LBP configuration is a **High-Weight (95%+), Medium-Duration (48h-72h) launch.** This combination maximizes the "Healthy" signal while statistically minimizing the surface area for bot attacks and early dumps.

4. Final Considerations

The rigorous statistical analysis conducted throughout this study leads to a fundamental shift in how Liquidity Bootstrapping Pools should be approached. Our machine learning experiments show that searching for a "Profit Formula" is futile, the market's price discovery mechanism is too heavily influenced by extrinsic factors like community sentiment and macro conditions to be predicted by onchain configuration alone.

What we did prove, however, is that while the upside is unpredictable, the downside is highly engineered. The value of our model lies not in forecasting token prices, but in auditing the structural integrity of the liquidity curve itself.

For a user defining their LBP parameters, the data distinguishes clearly between “Laws of Physics” and “Matter of Preference.” The Weight Slope emerges as the single most critical variable; it acts as a mechanical forcing function. Our optimization graphs reveal a safety cliff at a slope of 0.6. Crossing this threshold creates a mathematical inevitability of selling pressure that organic demand rarely overcomes.

Similarly, **Duration** exhibits a strict window of efficiency. Short pools (<24h) suffer from bot dominance, while long pools (>72h) bleed capital to arbitrageurs without improving price discovery. The “Sweet Spot” is located between **48 and 72 hours**, a window that maximizes human participation while minimizing algorithmic extraction.

Ultimately, the optimal LBP configuration is one that minimizes resistance rather than one that chases volatility. The forensic analysis of failed launches indicates that the most robust setup combines a **High Starting Weight (>95%)** to provide an initial valuation buffer, a **Medium Duration (approx. 3 days)** to ensure capital efficiency, and a **Flattened Slope (<0.6)** to prevent the pool from acting as a dumping mechanism against its own investors.

Parameters such as Swap Fees and specific Starting Ratios, previously thought to be decisive, were revealed to be cosmetic. Adhering to these “Red Line” thresholds shifts a project from gambling on market mechanics to leveraging them, ensuring the launch is defined by the strength of the project rather than the fragility of its configuration

The complete methodology, data extraction, and analysis process are available in [this repository.](#)

Balancer

AMMs MADE EASY

DATA POWERED BY DUNE