

DATA INFORMED GAME DESIGN

YASIN HATIBOČLU

Understanding the player behavior and designing experiences

About me

- Born in Turkey, studied in Austria, living in Finland
- Education
 - BSc in Econometrics
 - MSc in Quantitative Finance
- Industry experience
 - Data Analyst at *Sproing* (Austria)
 - Data Analyst at *Seriously* (Finland)
 - Product Manager at *Rovio* (Finland)
 - Data Analyst at *Supercell* (Finland)

Overview

How can we make better puzzle levels with the help of data?

Overview

How can we make better puzzle levels with the help of data?

- I. How can we identify good levels? (the million \$ question!)
- II. Which mechanics should we have in the levels?
- III. How should we design the level objectives?
- IV. How difficult should the levels be?
- V. How long should the levels be?

Problem

How can we make better puzzle levels?

How can we make better levels?

User perspective

Fun levels where players neither fail miserably, nor get stuck or win constantly.

Business perspective

Core content that generates high resource sink which will lead to IAPs, while not causing high churn.

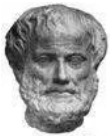
How can we make better levels?

User perspective

Fun levels where players neither fail miserably, nor get stuck or win constantly.

Business perspective

Core content that generates high resource sink which will lead to IAPs, while not causing high churn.



Aristotle

“What makes a level good?”

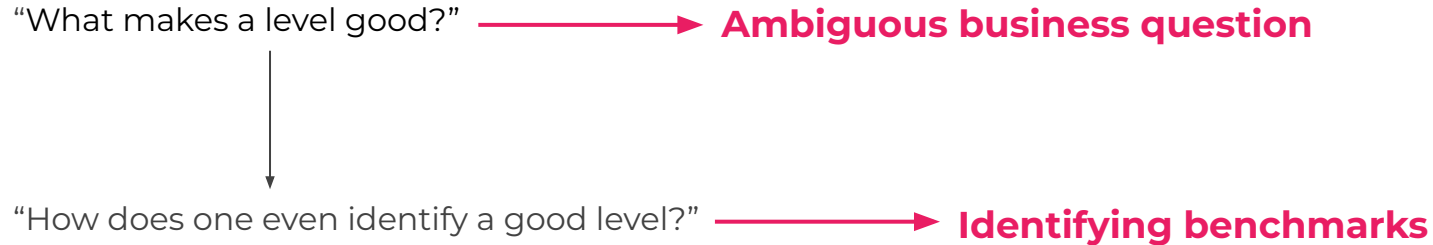
How can we make better levels?

Asking specific, measurable and actionable business questions

“What makes a level good?” —————> **Ambiguous business question**

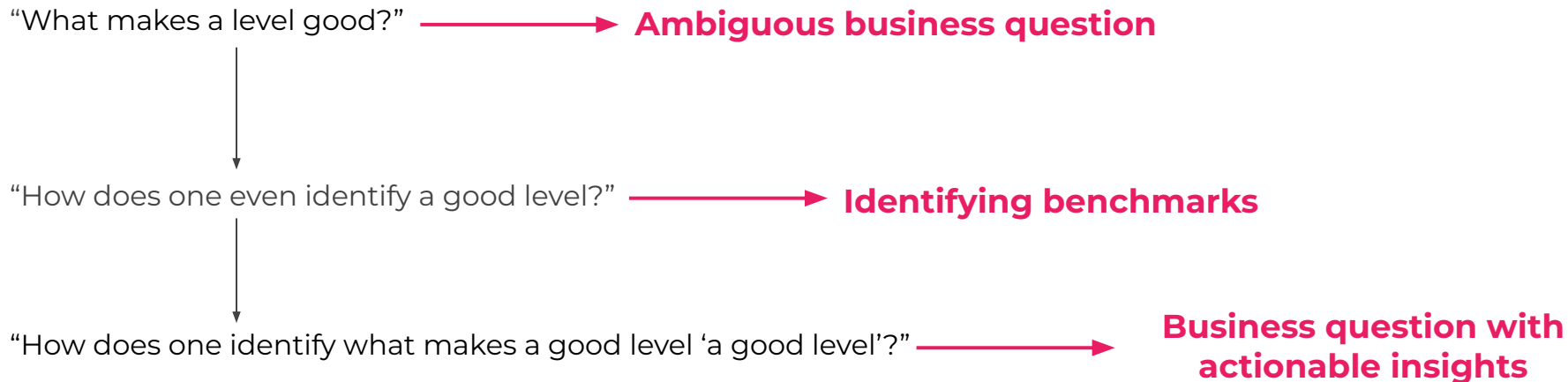
How can we make better levels?

Asking specific, measurable and actionable business questions



How can we make better levels?

Asking specific, measurable and actionable business questions



Problem

How can we identify good levels?

Measuring the level performance

- We want to identify levels that are performing better and make more of them.
- We also want to identify poorly performing levels and make less of them.

Measuring the level performance

- We want to identify levels that are performing better and make more of them.
- We also want to identify poorly performing levels and make less of them.



Descartes

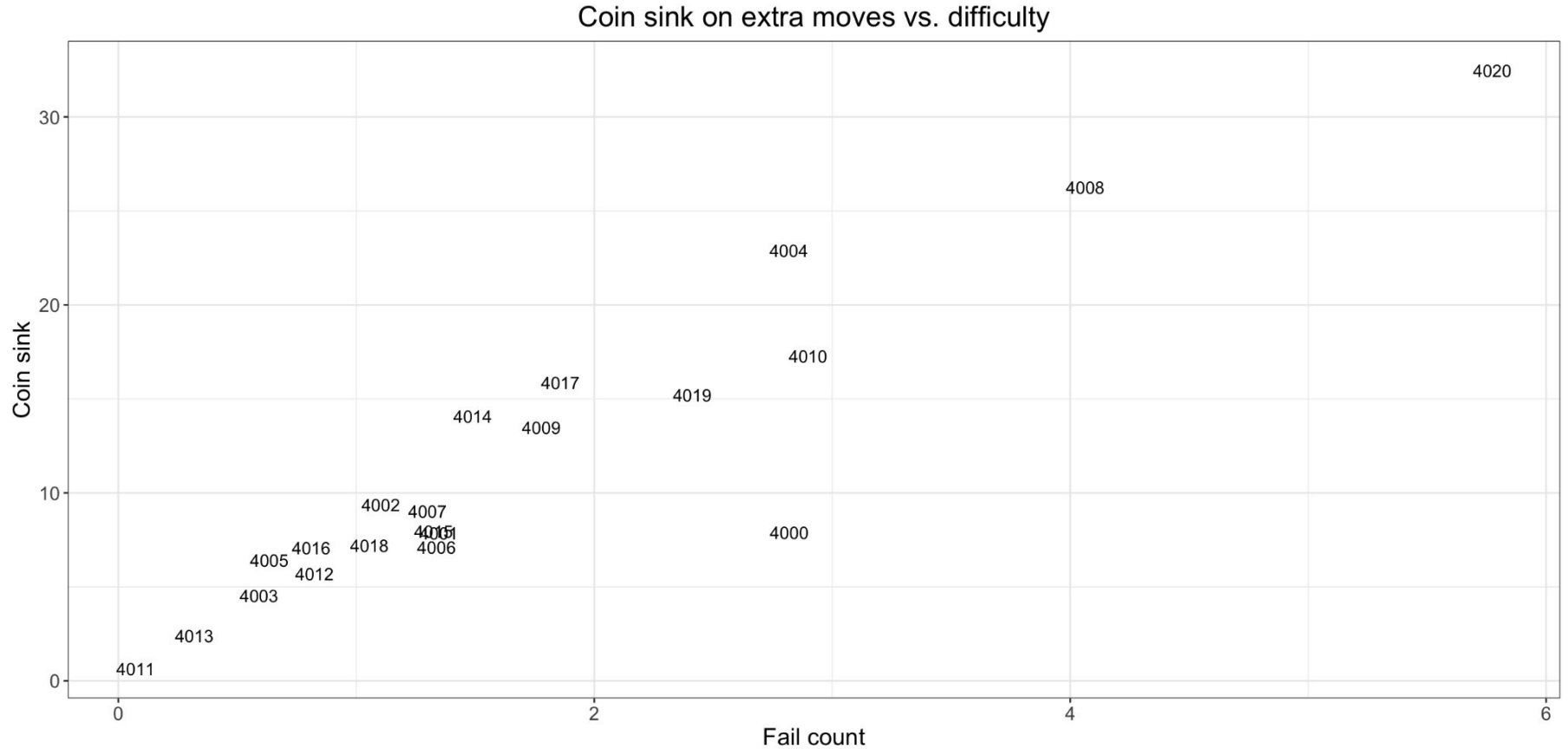
“How does one even identify a good level?”

Measuring the level performance

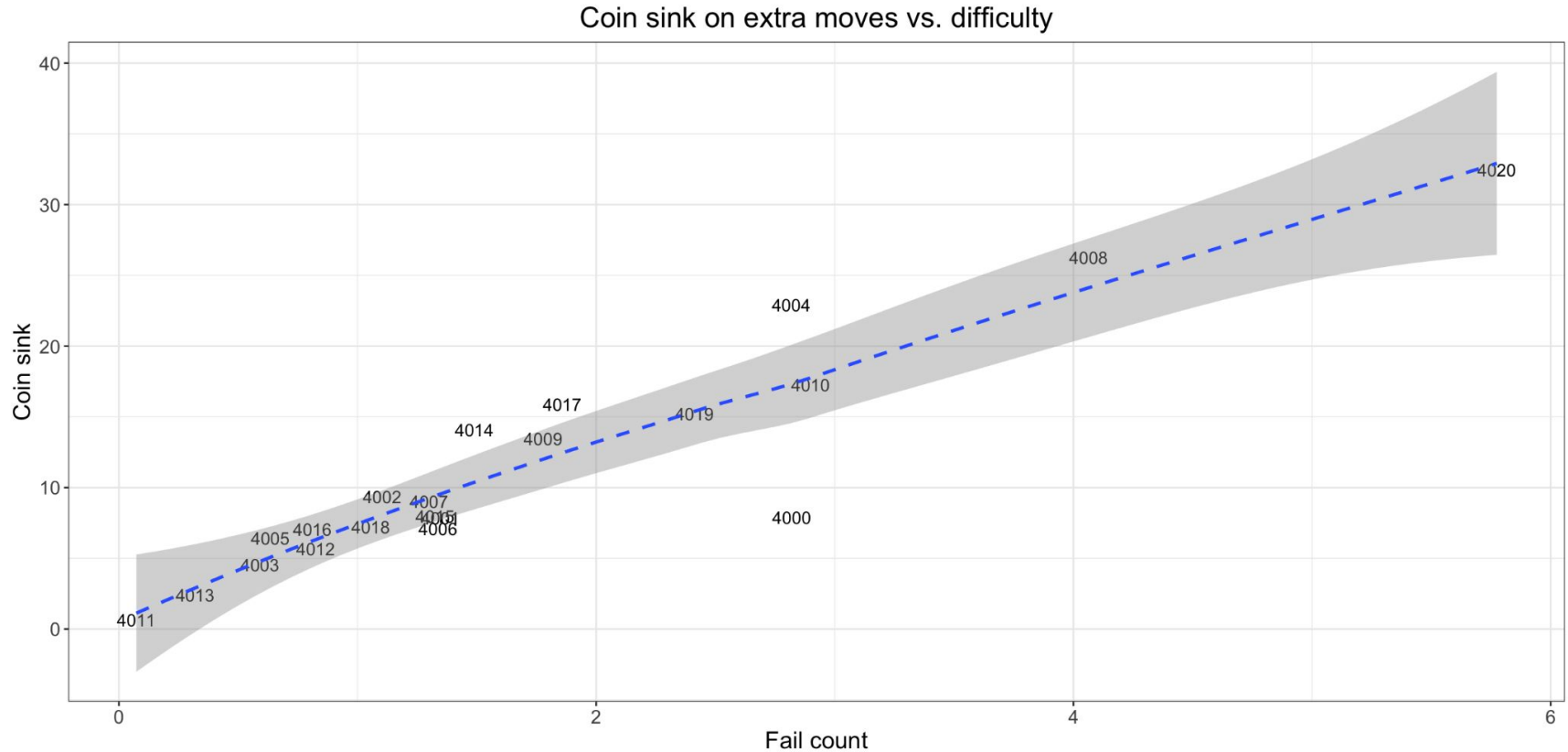
Key level metrics

- Extra moves usage
 - Extra moves can be purchased when players run out of moves on a level.
- Difficulty
 - How many attempts does it take to win a level?
- Churn
 - What percentage of players become inactive after starting a level?

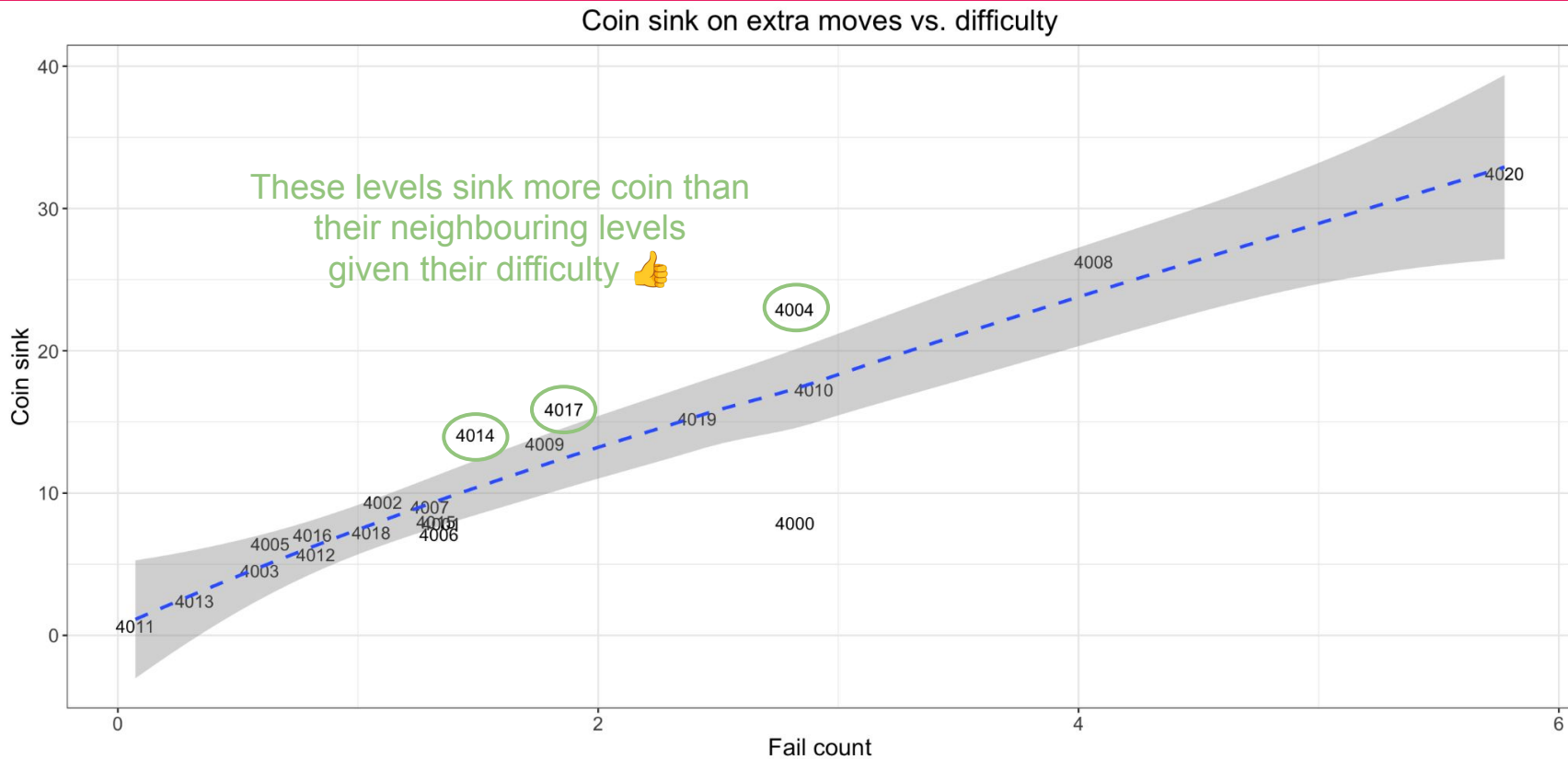
Measuring the level performance



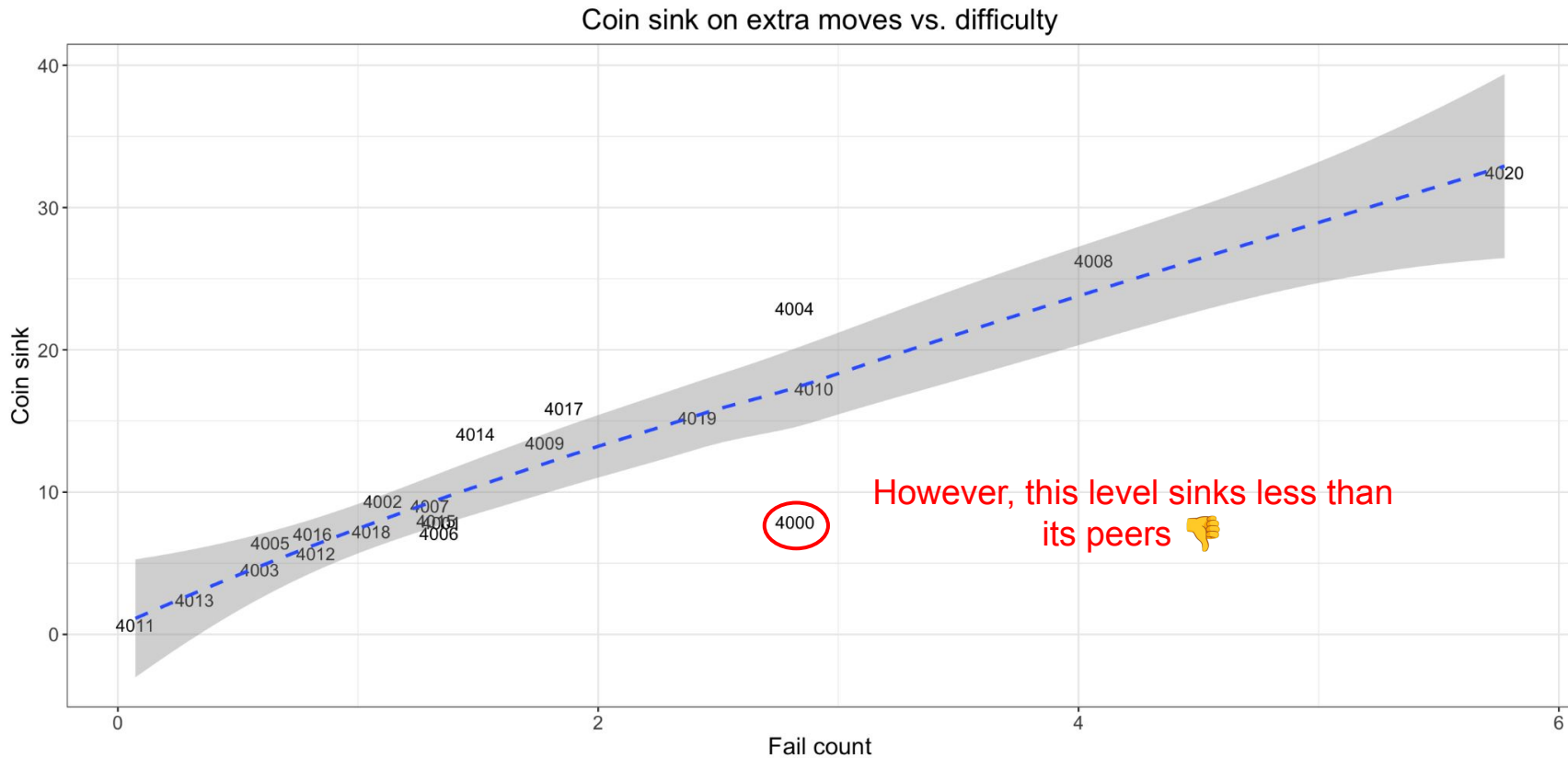
Measuring the level performance



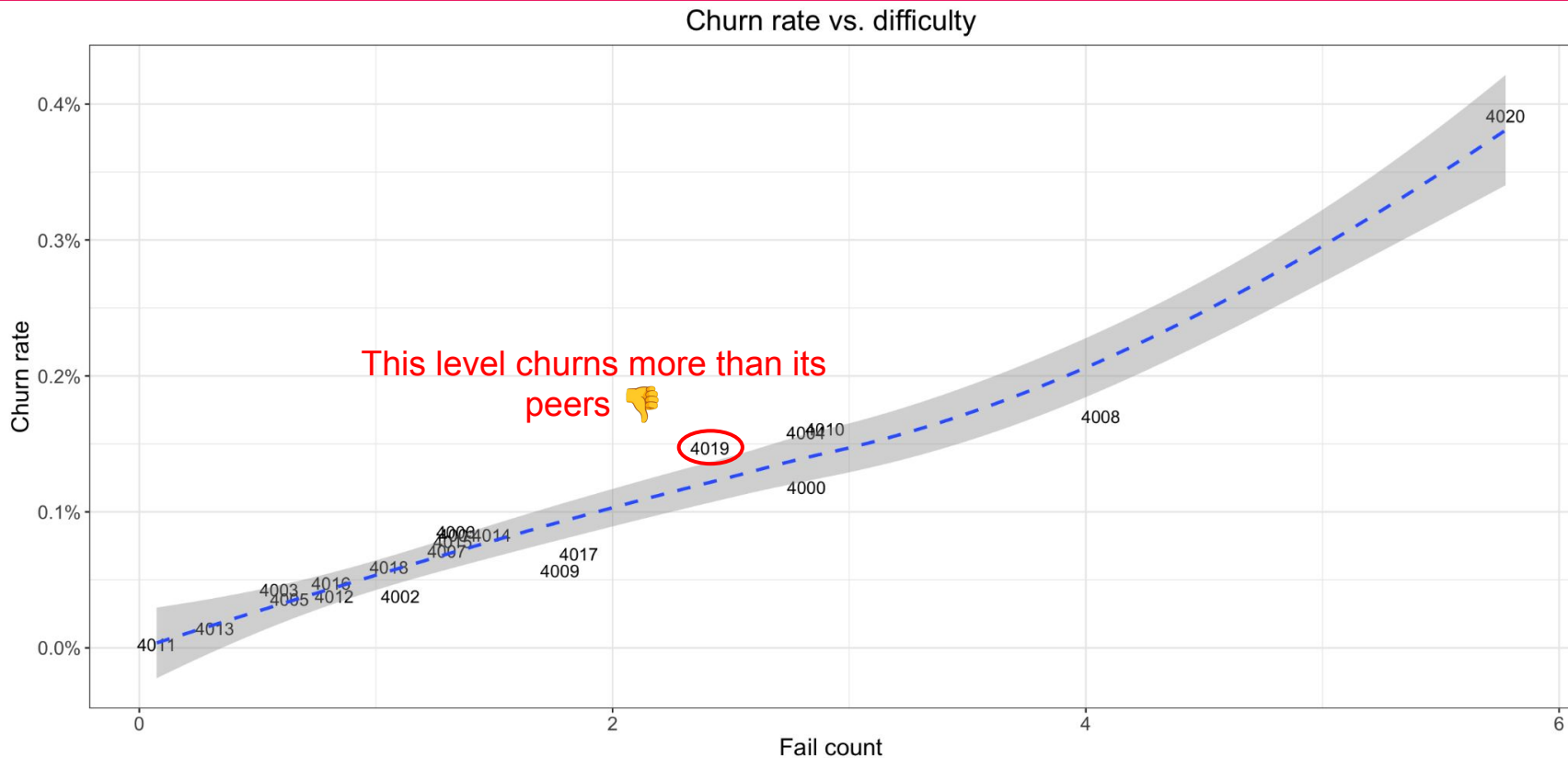
Measuring the level performance



Measuring the level performance



Measuring the level performance







Measuring the level performance



Act based on the content performance

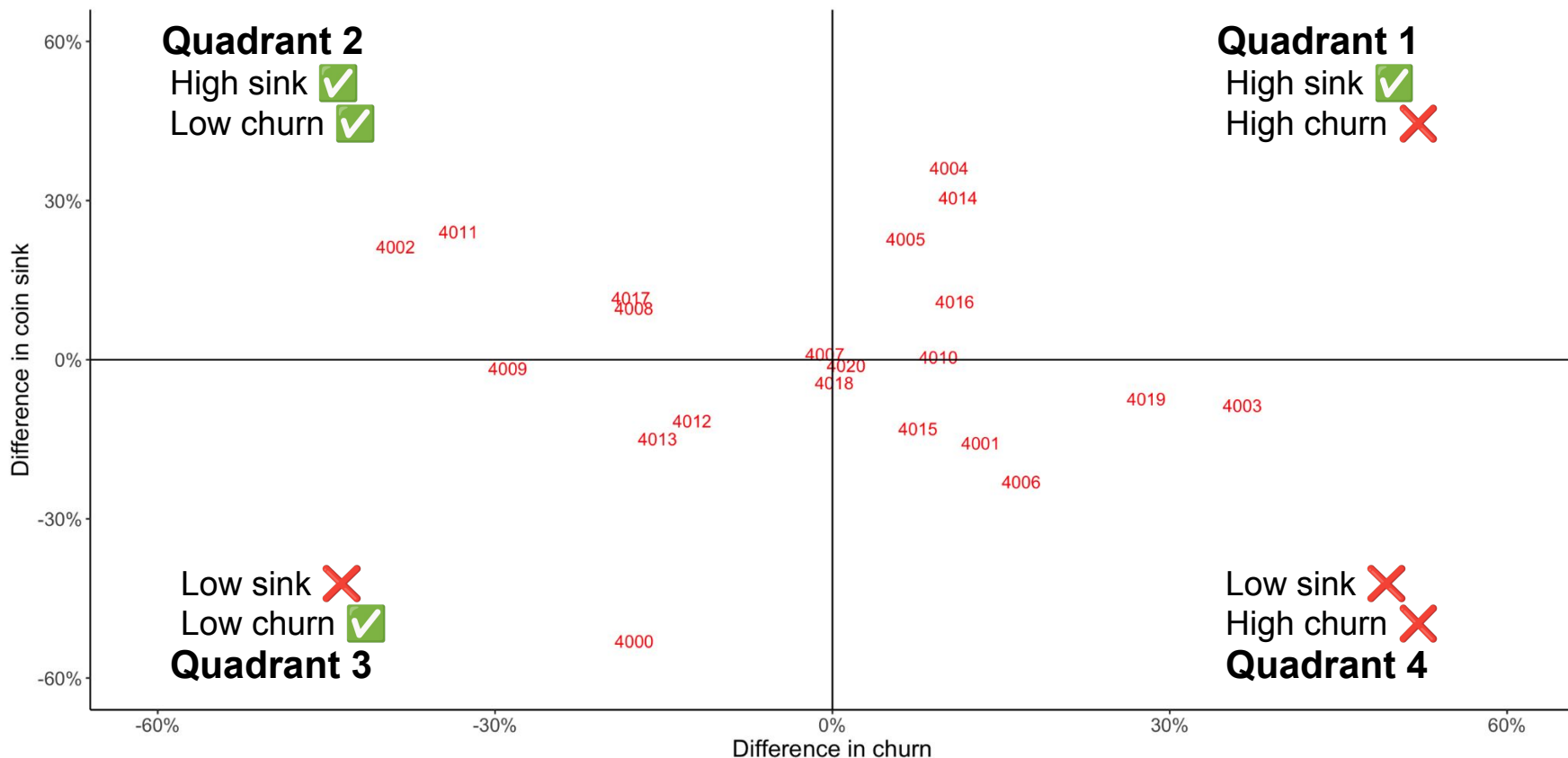
Content performance matrix

	Low churn	High churn
Low sink		Worst levels 
High sink	Best levels 	

Possible action points

- 1) Identify the “best” and “worst” content/levels. Analyze and look for the common patterns. What could make these levels better or worse than others? Update your design guidelines based on your findings.
- 2) Increase the difficulty of best performing levels. These levels sink more resources while having a lower churn. These levels will bring LTV growth.
- 3) Decrease the difficulty of worst performing levels. These levels potentially have a negative impact on the LTV growth as they churn more than what they sink. It is also recommended to **replace/redesign** this kind of underperforming content with better ones.

Measuring the level performance



Measuring the level performance

Possible action points

- Remove the underperforming levels and replace them with levels that resemble the well performing ones.
- Now that we can classify levels into 4 performance quadrants, we can research what level design patterns and mechanics correlate with each quadrant.
- This will help us
 - make more of the good levels (high sink & low churn)
 - avoid using underperforming mechanics (low sink & high churn).

Problem

Which mechanics should we have in the levels?

What makes a level good?

- Now we know which performance quadrant each level belongs to.
- Next, we will use a decision tree to map level mechanics to performance quadrants.

What makes a level good?

- Now we know which performance quadrant each level belongs to.
- Next, we will use a decision tree to map level mechanics to performance quadrants.

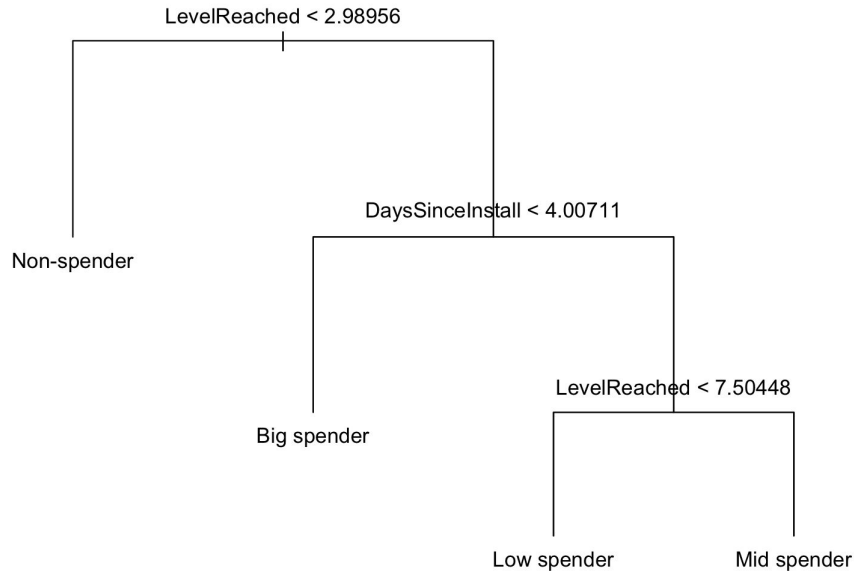


Nietzsche

“How does one identify what makes a good level ‘a good level’?”

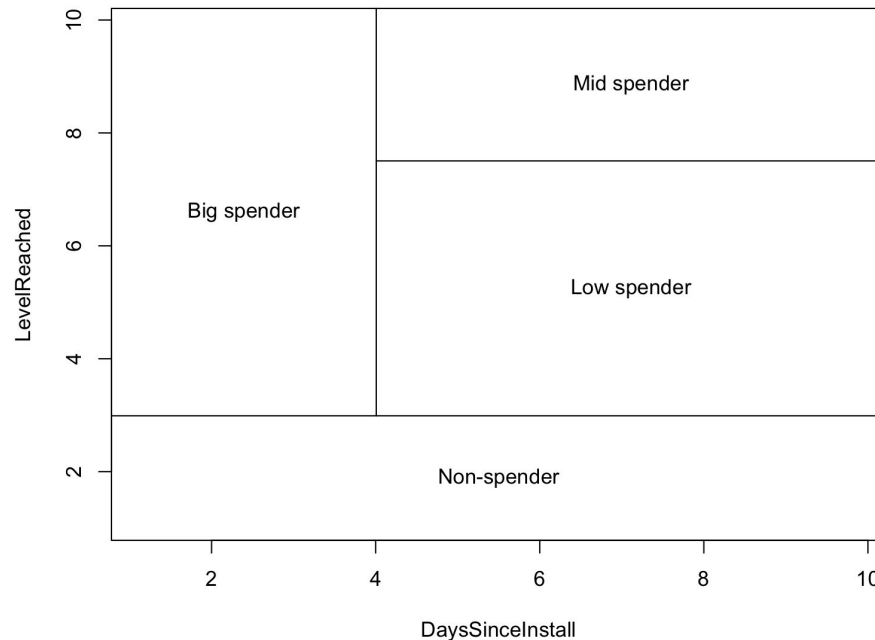
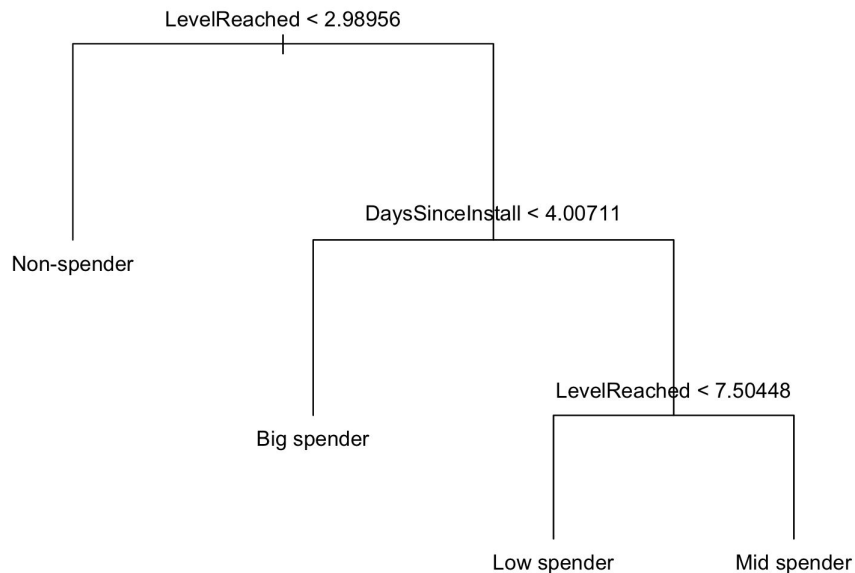
What makes a level good?

Decision tree example



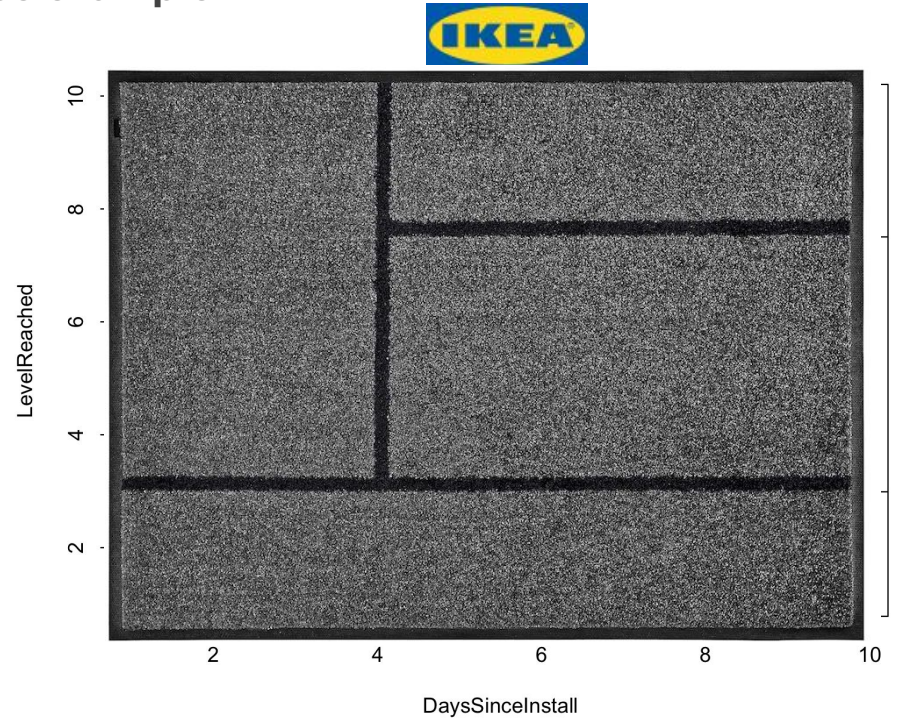
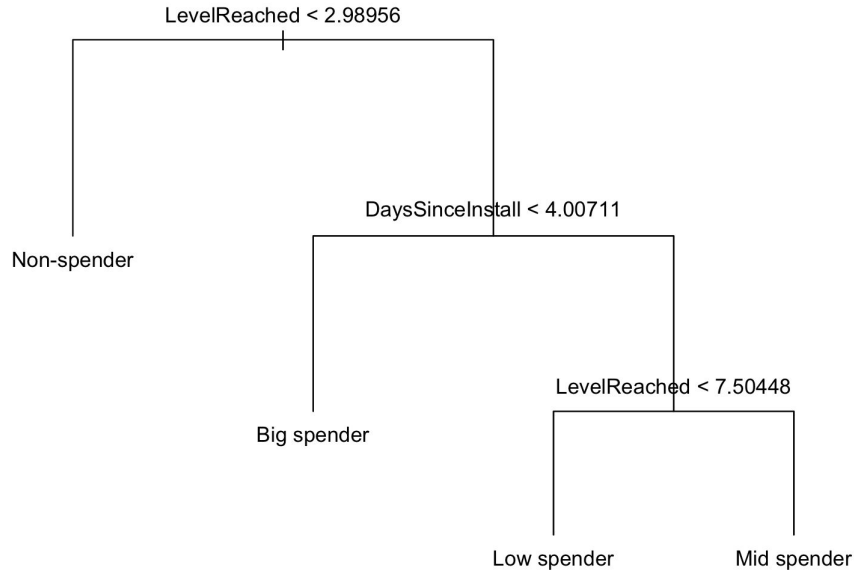
What makes a level good?

Decision tree example



What makes a level good?

Decision tree example



Problem

Can we use level design practices to predict the level performance?

What makes a level good?

Quadrant 2

High sink ✓
Low churn ✓

Quadrant 1

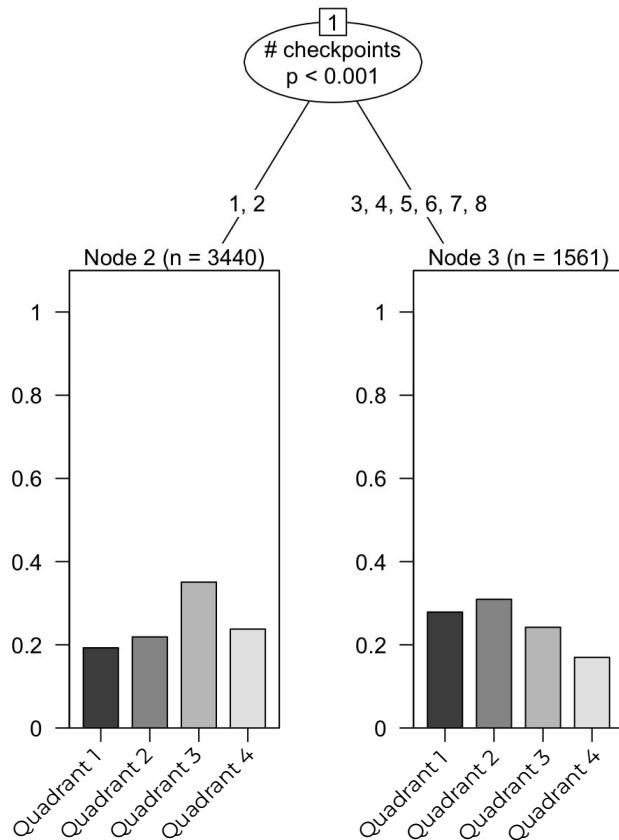
High sink ✓
High churn ✗

Quadrant 3

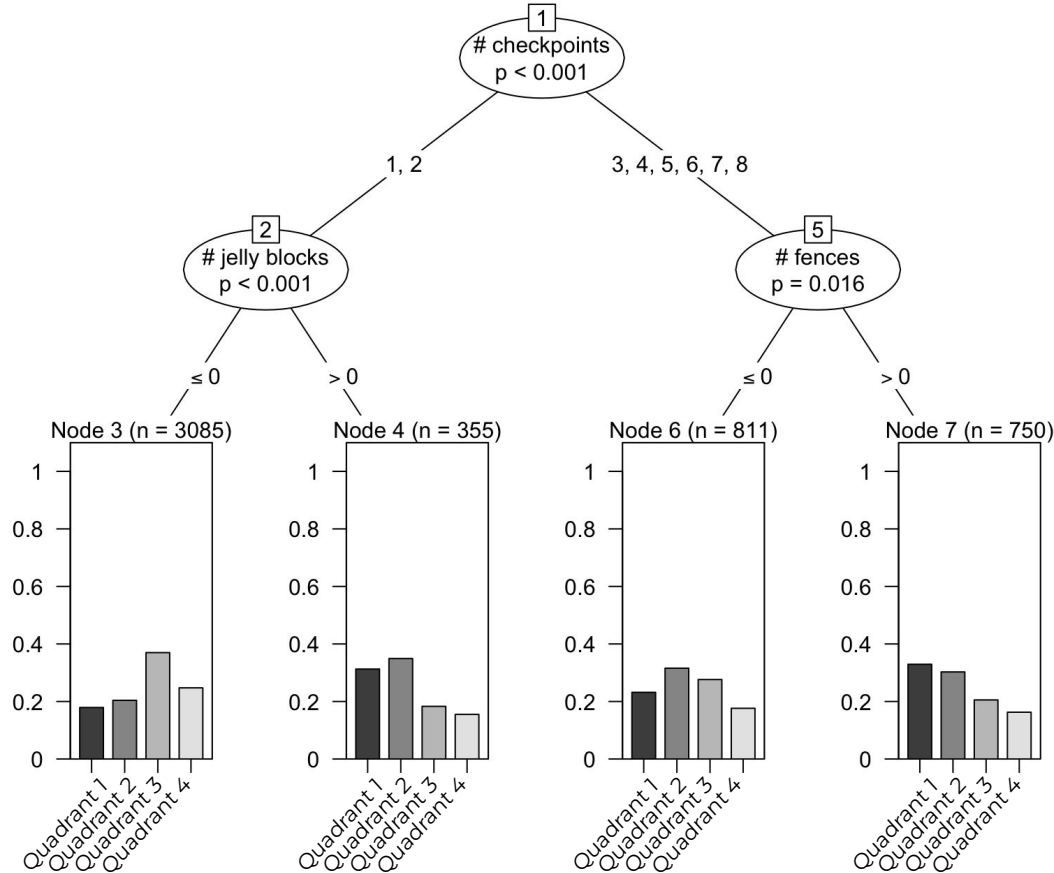
Low sink ✗
Low churn ✓

Quadrant 4

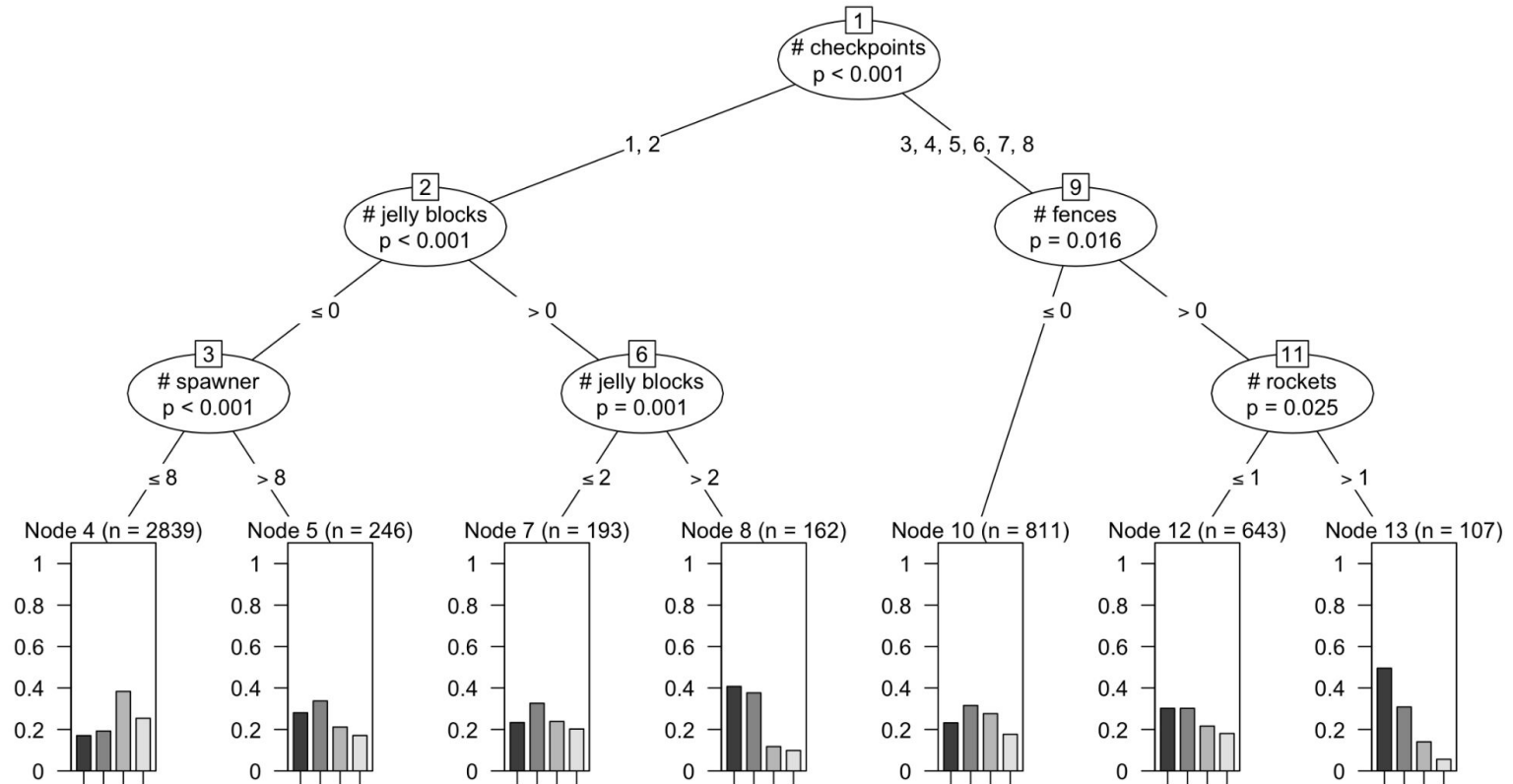
Low sink ✗
High churn ✗



What makes a level good?

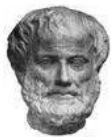


What makes a level good?



Recap

Recap



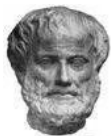
Aristotle

“What makes a level good?”



Core content that generates high resource sink which will lead to IAPs, while not causing high churn.

Recap



Aristotle

“What makes a level good?”



Core content that generates high resource sink which will lead to IAPs, while not causing high churn.



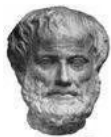
Descartes

“How does one even identify a good level?”



Compare resource sink and churn per difficulty bracket;
estimate the benchmark and identify over- and underperforming levels accordingly.

Recap



Aristotle

“What makes a level good?”



Core content that generates high resource sink which will lead to IAPs, while not causing high churn.



Descartes

“How does one even identify a good level?”



Compare resource sink and churn per difficulty bracket, estimate the benchmark and identify over- and underperforming levels accordingly.



Nietzsche

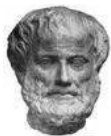
“How does one identify what makes a good level ‘a good level’?”



Gather data about level mechanics and try mapping them into level performance data. Identify the common patterns on well and poorly performing levels.

Statistical models can be used for initial exploration.

Recap



Aristotle

"What makes a level good?"



Core content that generates high resource sink which will lead to IAPs, while not causing high churn.



Descartes

"How does one even identify a good level?"



Compare resource sink and churn per difficulty bracket, estimate the benchmark and identify over- and underperforming levels accordingly.



Nietzsche

"How does one identify what makes a good level 'a good level'?"



Gather data about level mechanics and try mapping them into level performance data. Identify the common patterns on well and poorly performing levels.

Statistical models can be used for initial exploration.



Lil Jon:
"What?"

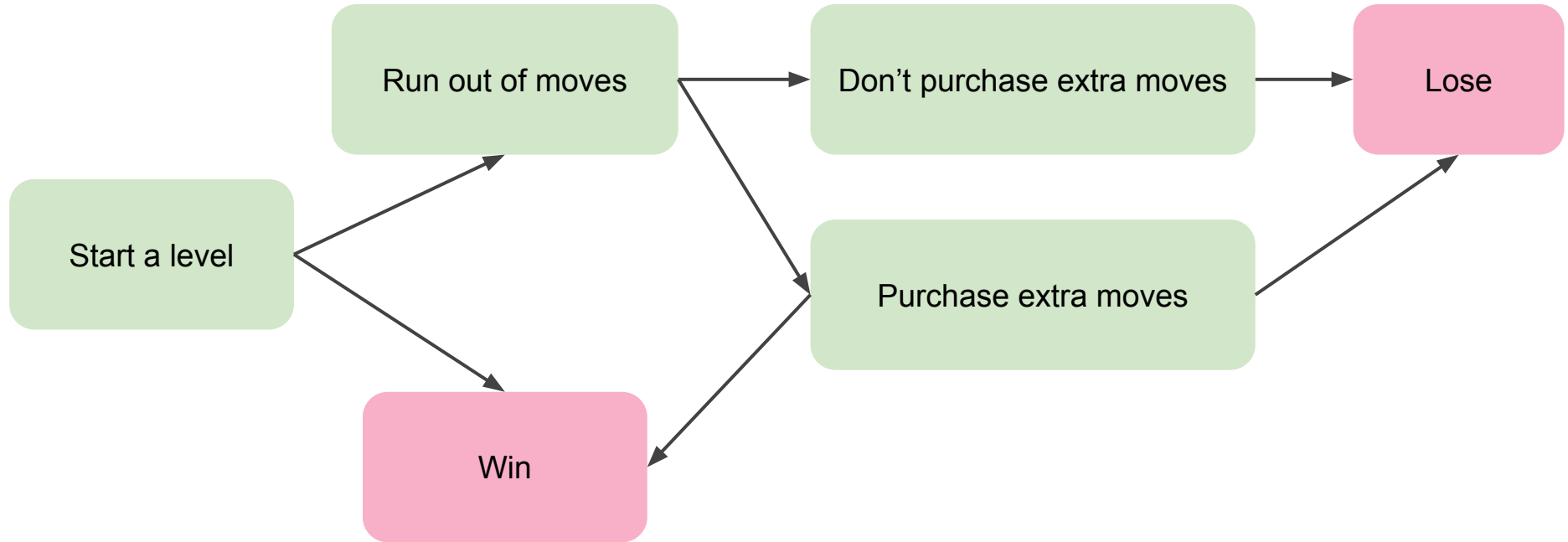
Problem

How should we design the level objectives?

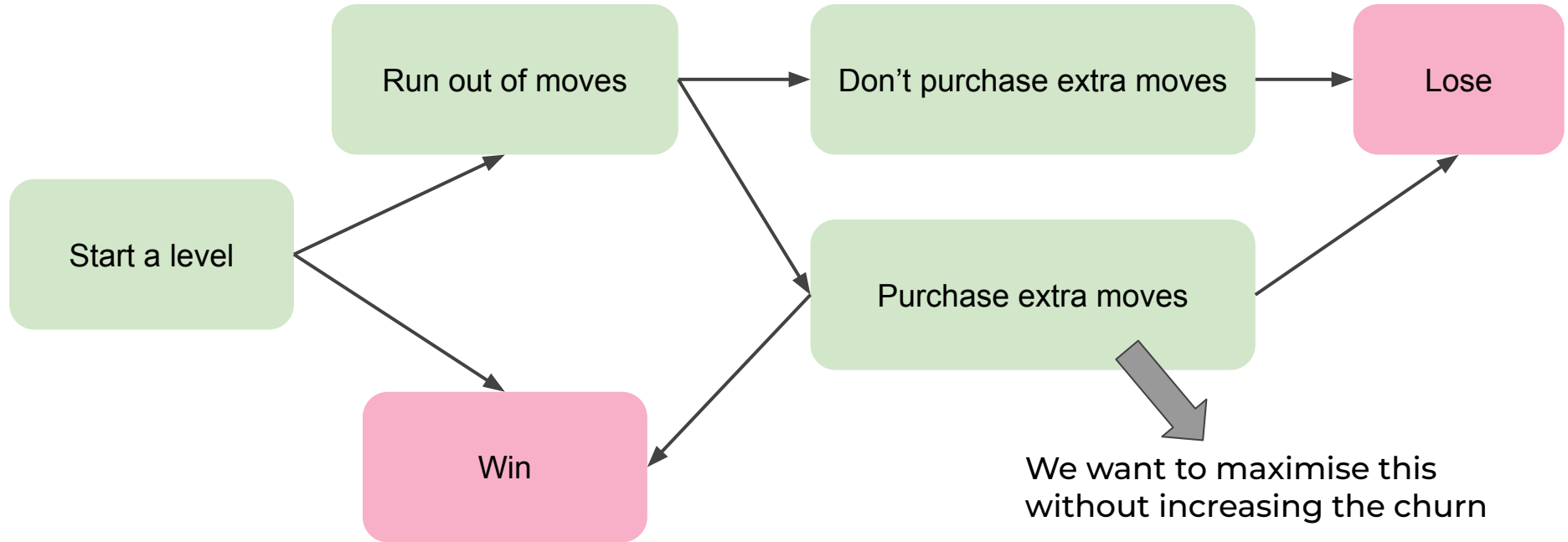
Designing level objectives



Designing level objectives



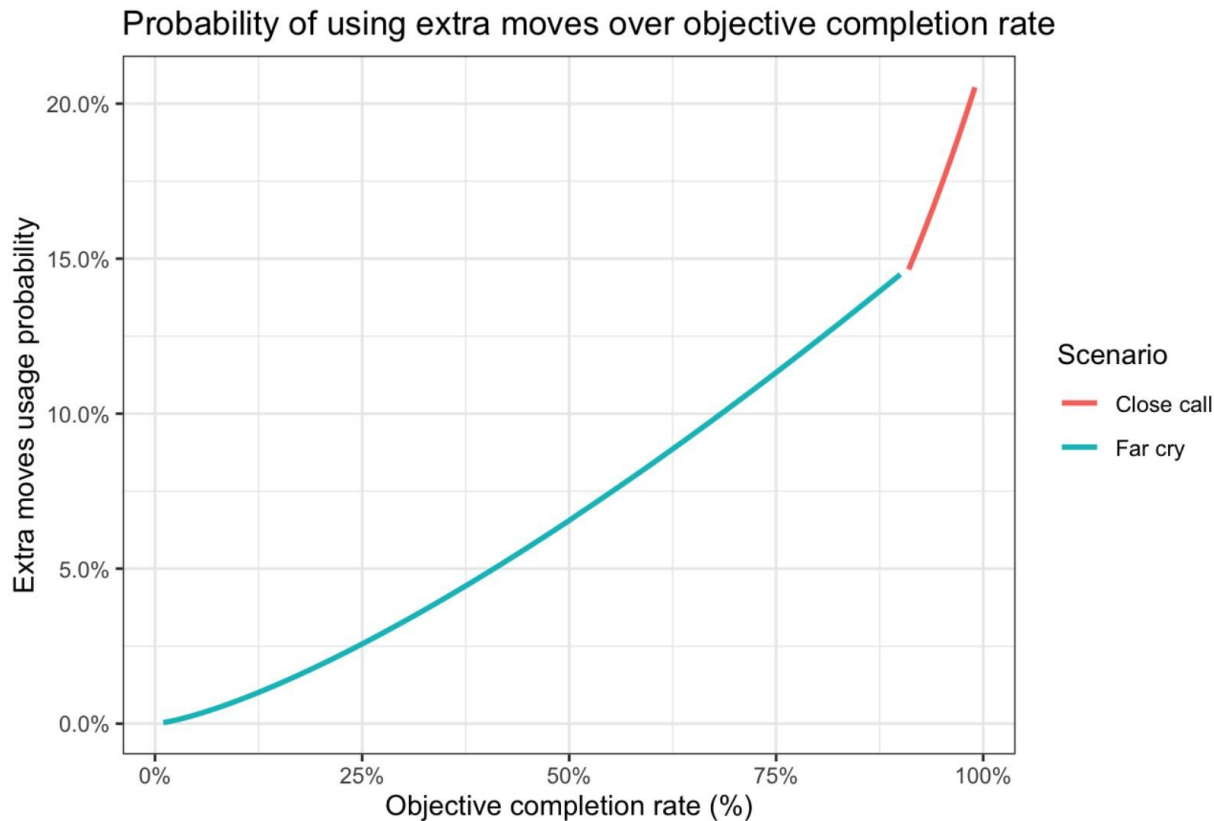
Designing level objectives



Connection between level objectives and sink

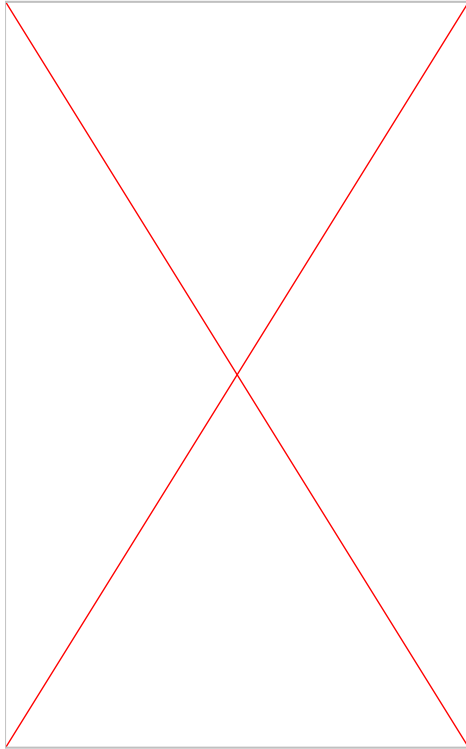
- If we want players to purchase extra moves, we first need them to run out of moves.
- Players' probability of purchasing extra moves is affected by how much players progressed with level objectives.

Connection between level objectives and sink



Connection between level objectives and sink

Near miss



Far miss



Connection between level objectives and sink



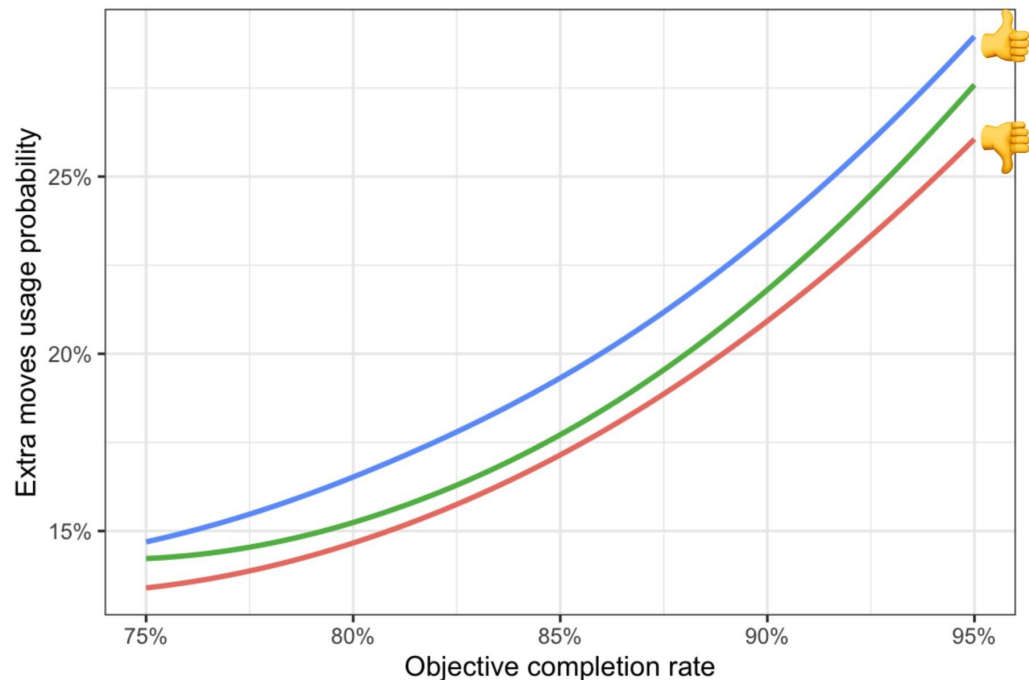
- With a simple UI change, players can get a better feeling of progression.
- Having multiple slugs with varying HPs made the progression less predictable, hence lowered extra moves usage at out of moves screen.
- Replacing multiple slug levels with a single boss slug where the HP was visualized as a percentage scale improved the experience greatly (less churn, higher sink).

Connection between level objectives and sink

Extra moves usage probability over objective completion rate

Breakdown by number of completed objectives

Completed objectives status 0/3 1/3 2/3

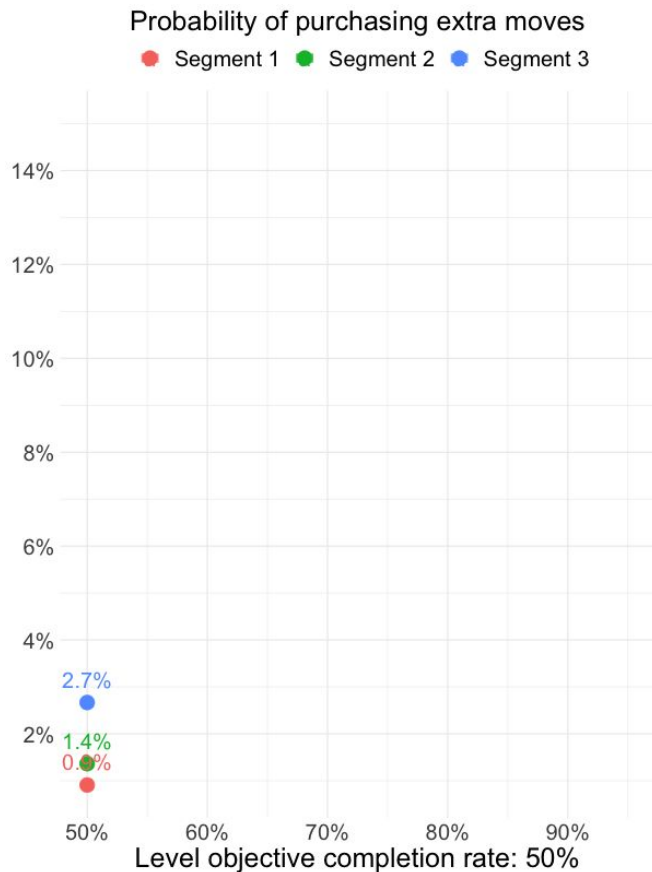


Players don't know what percentage of the objectives they have completed.

However, they can see **how many** of the objectives they completed.

Fewer incomplete objectives gives higher confidence for players that purchasing +5 moves will help them win the level.

Connection between level objectives and sink



● Segment 1 seems to be a very conservative player group, probably non-payer.

● Segment 3 is the least risk averse player group, probably a big spender.

Designing level objectives

Highlights and action points

- Levels should be designed in such a way that players are not too far from level completion when running out of moves.
- Having multiple slugs, scenes, checkpoints and multiple objectives emphasize the progression.
- Levels with multiple objectives and scenes usually perform very well...
 - ...**only if** players are at the last screen or last objective when running out of moves!
- Smart algorithms that carry players to higher completion rates can improve both retention and monetisation.

Problem

How difficult should the levels be?

Designing the difficulty

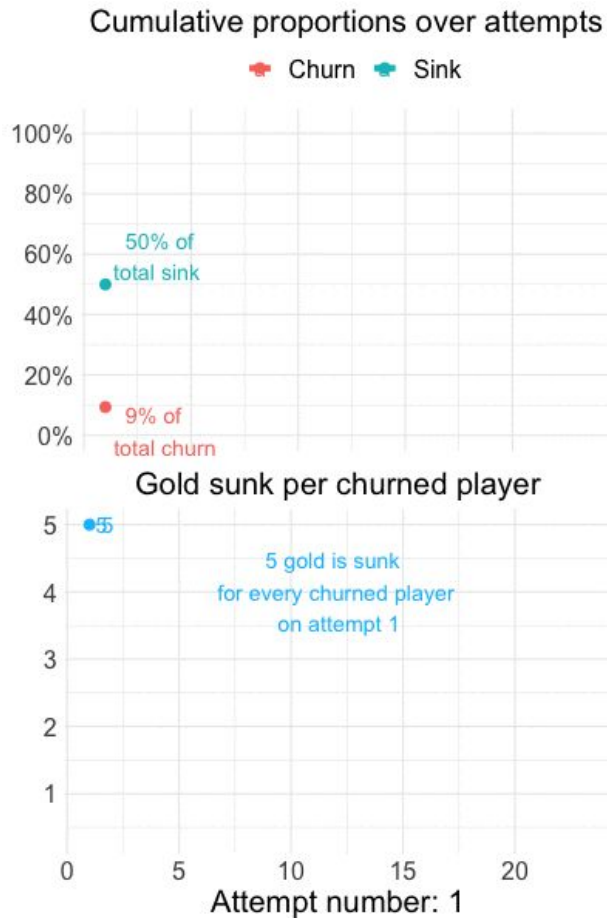
- There is a positive correlation between difficulty and resource sink.
- Harder levels are needed every now and then to increase the extra moves usage.
- It is however beneficial to lower the difficulty if players start getting stuck!

Designing the difficulty

We want

- higher number of out of moves screens (extra moves usage will increase)
- fewer cases where players get stuck in a difficult level (churn will decrease).

Designing the difficulty

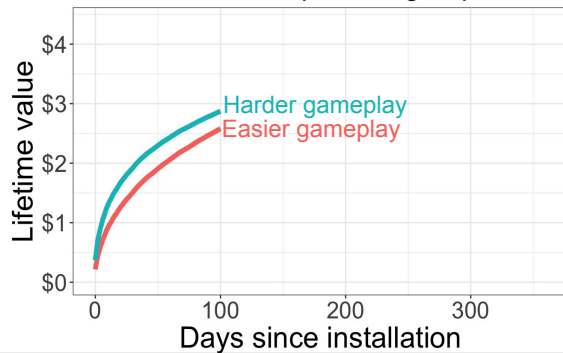


Right after the first attempt, we start seeing diminishing returns in terms of extra moves usage and churn trade-off.

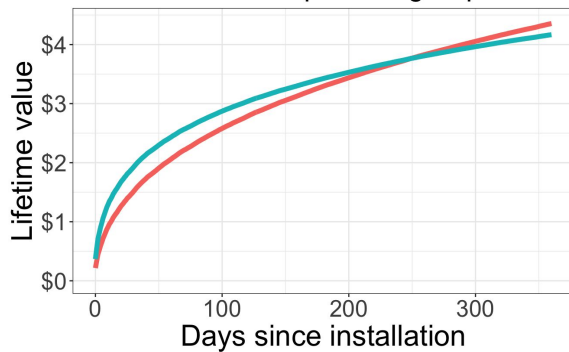
Most of the extra moves usage occurs at the first attempt, while churn starts escalating over attempts.

Designing the difficulty

LTV curve per test group



LTV curve per test group



Designing the difficulty

Highlights and action points

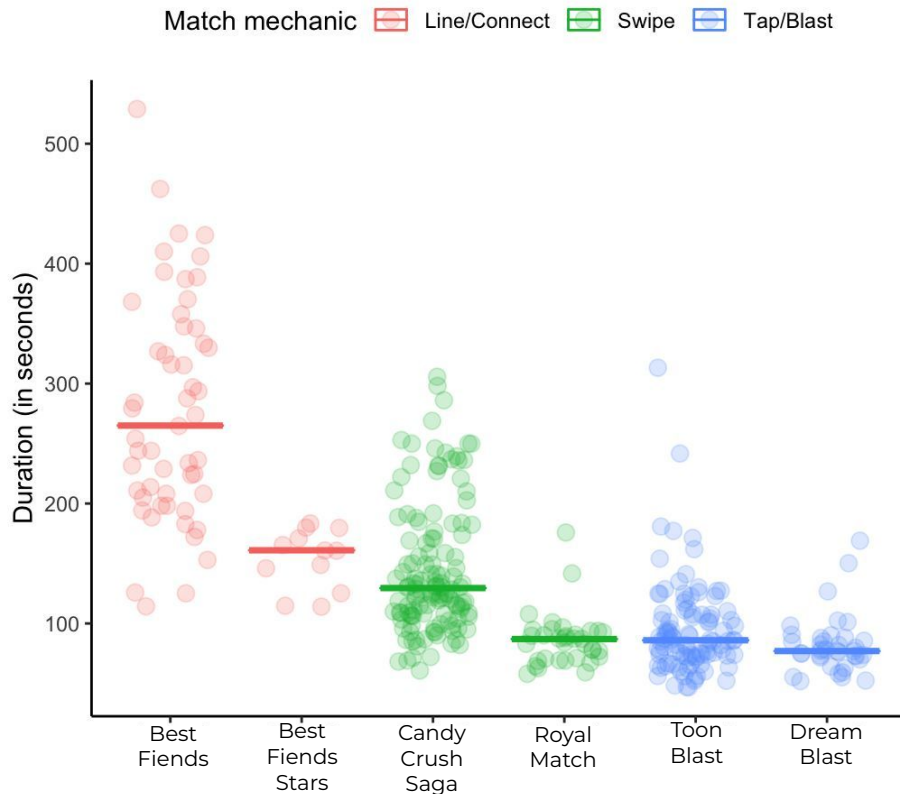
- It's important to have multiple easy levels in between very difficult levels, so that players can build up their win streak. Keeping the win streak going is one of the biggest motivators for purchasing extra moves.
- Variance of difficulty within a level should be low. Higher variance might mean that players either win too quickly or get stuck in the level.
- To avoid players getting stuck, a dynamic difficulty system can help players in a subtle way after a certain amount of failures.
- However, the helping system shouldn't be predictable, since players can exploit it by failing on purpose to receive help.

Problem

How long should the levels be?

Determining the level durations

Median level attempt duration per game (in seconds)



*Level durations are approximated by walkthrough video durations on Youtube.

Determining the level durations

Long vs. short level attempts

Long attempts

Higher “sunk cost feeling” drives more extra moves purchases **per level**.

Fewer levels played **a day***

Fewer out of moves screen impressions.

Short attempts

Lower “sunk cost feeling” drives less extra moves purchases **per level**.

More levels played **a day***

More out of moves screen impressions.

*daily playtime is not affected by average level durations

Determining the level durations

Highlights and action points

- AB testing levels with different durations to find the sweet spot.
 - Figure out if **many short levels** or a **few long levels a day** yield higher LTV.
- In Clash of Clans, one could easily sink \$100 worth of hard currency in one session, while in match 3 games, sink opportunities are bounded by level attempts.
 - In addition to the core level content, a meta layer can increase the possible sink points.

Thank you!

Questions?