

A thick, dark purple L-shaped bar in the top-left corner of the slide.

GRITTY SYSTEMS DESIGN FOR RETENTION

Raph Koster
Game Designer, among other things

A thick, dark purple L-shaped bar in the bottom-right corner of the slide.

Some basic assumptions

- Most service games use RPG-style advancement models these days

Some basic assumptions

- Most service games use RPG-style advancement models these days
- People who leave because of a cap or running out of content often return where there is more to do.

Some basic assumptions

- Most service games use RPG-style advancement models these days
- People who leave because of a cap or running out of content often return where there is more to do.
- People who quit because you pissed them off do not.

Why this matters

Consistency of reward is the secret for
engagement

The Metrics Trap

- There is this idea that you need to gather metrics on player behavior in order to arrive at values for your game.

The Metrics Trap

- There is this idea that you need to gather metrics on player behavior in order to arrive at values for your game.
- But good systems design can *create* metrics you want.

EXPLAINING CHURN

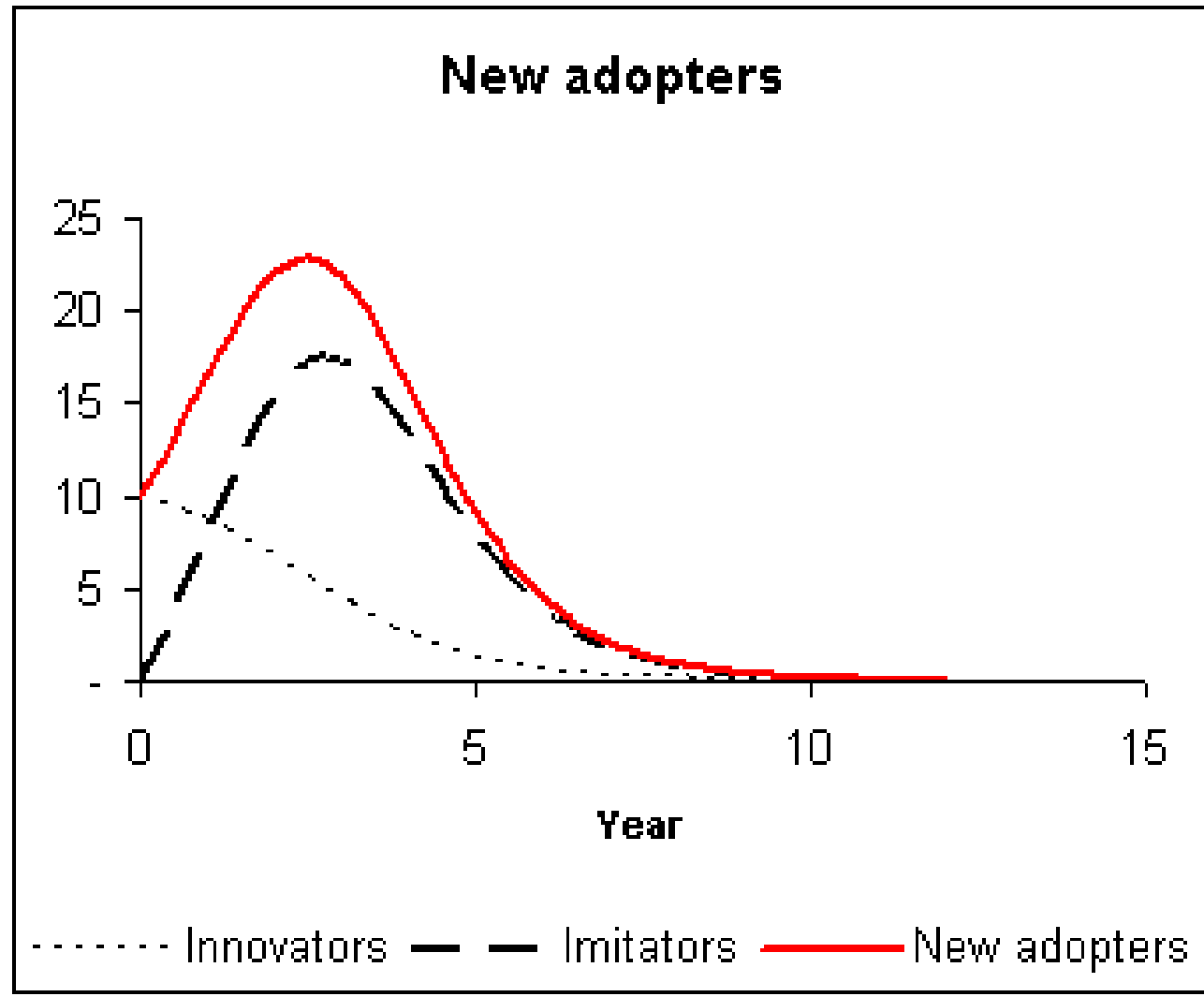


The Bass Diffusion Model

$$\frac{f(t)}{1 - F(t)} = p + qF(t)$$

Let's do some simple calculus...

Just kidding



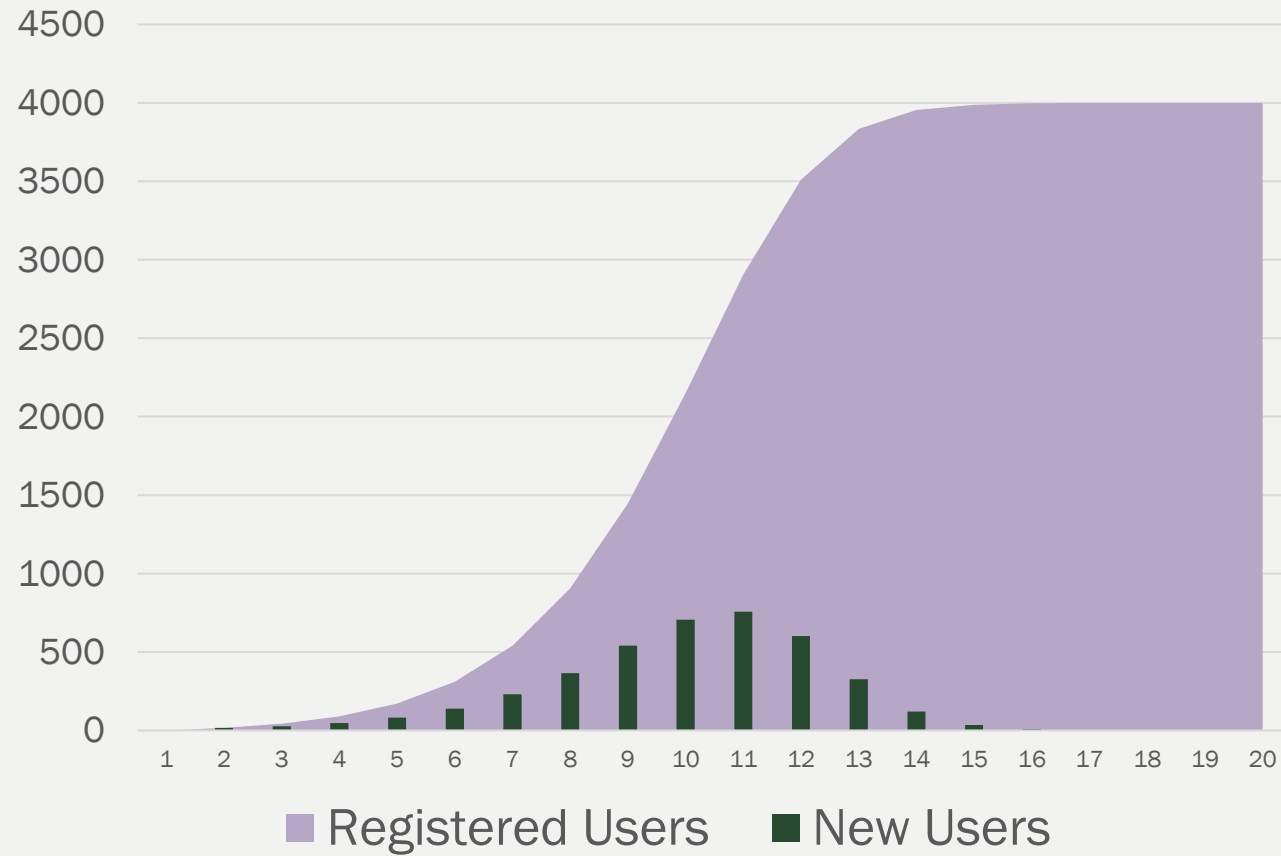
■ **Marketing power** makes this TALLER.

- *Brand*
- *Advertising*
- *Word of mouth*

■ **Market size** makes this WIDER.

- *Audience*
- *Territories*

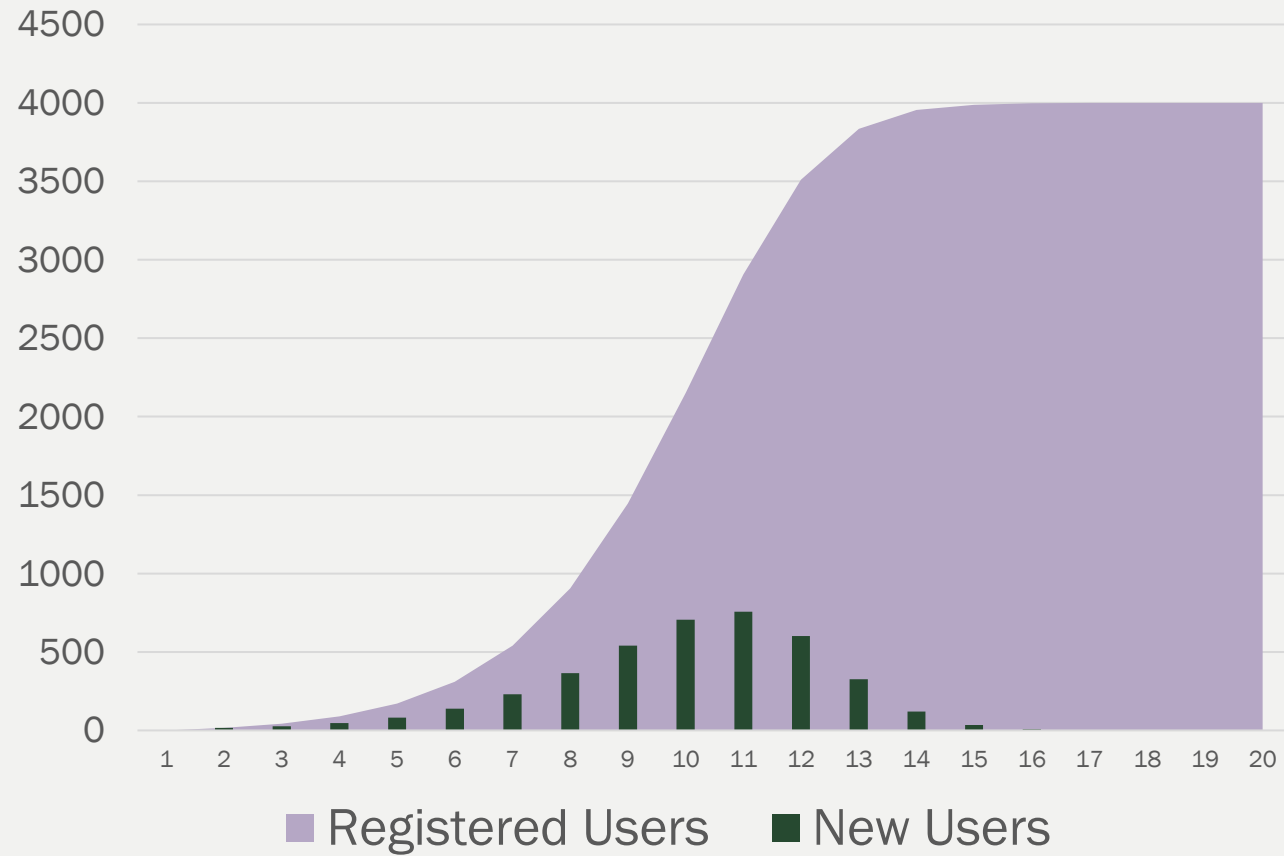
A toy sample



■ One market, size 4000.

■ The larger curve includes all previous adopters

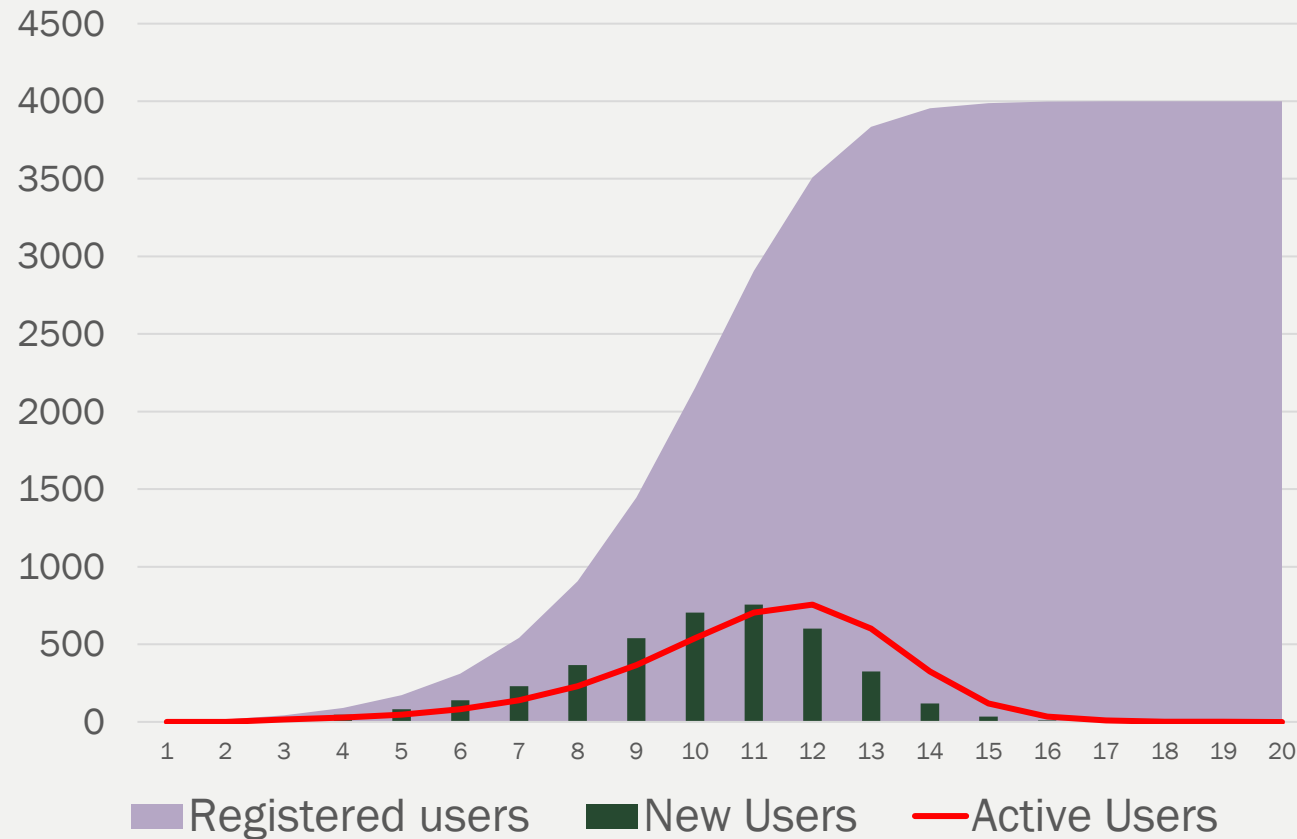
But services have churn



■ One market, size 4000.

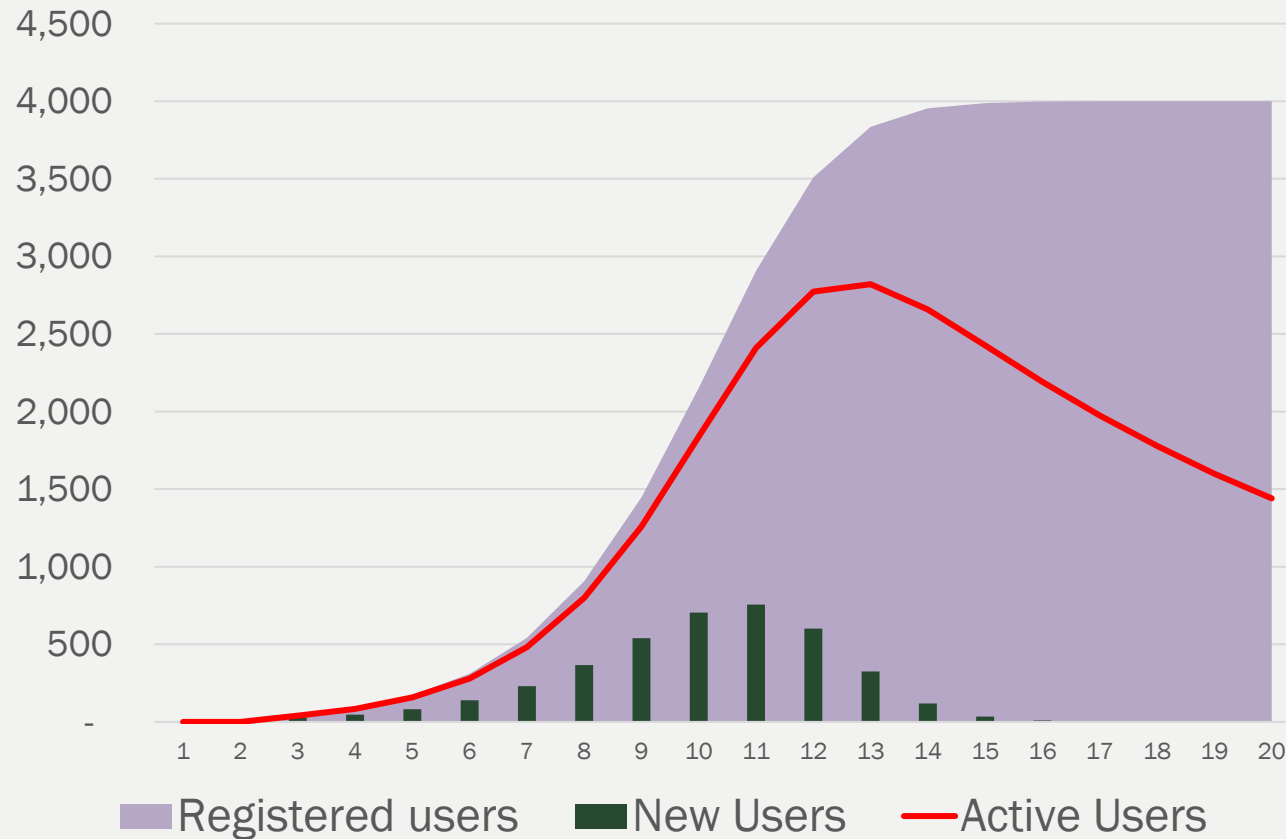
■ The larger curve includes all previous adopters

But services have churn



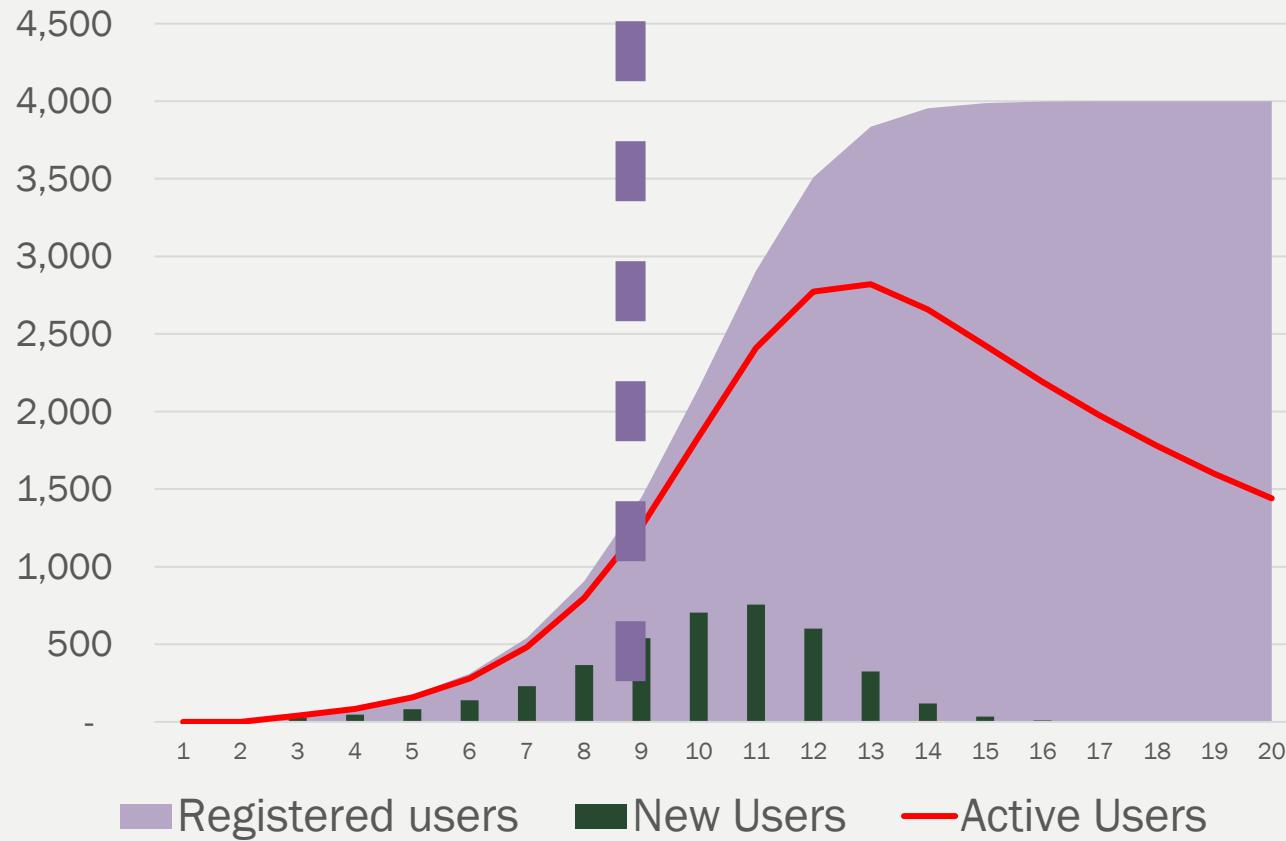
- If a user only lasts for 24 hours...
- Churn affects the WIDTH of a diffusion curve's results

But services have churn



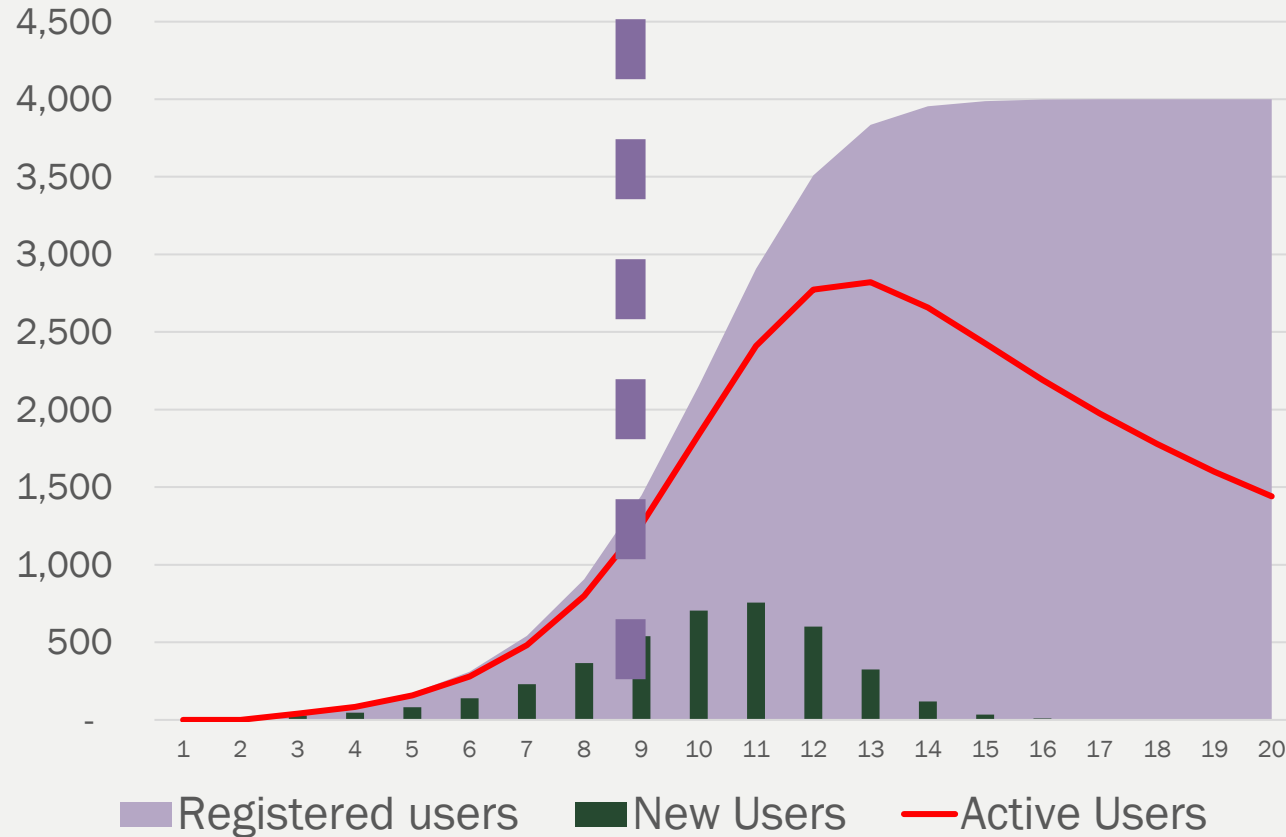
- If a user only lasts for 24 hours...
- Churn affects the WIDTH of a diffusion curve's results
- Here's losing 10% of your active users each period instead of 100%

Launch day



■ Based on marketing in advance, you may well start well up on this curve.

S-curves and J-curves



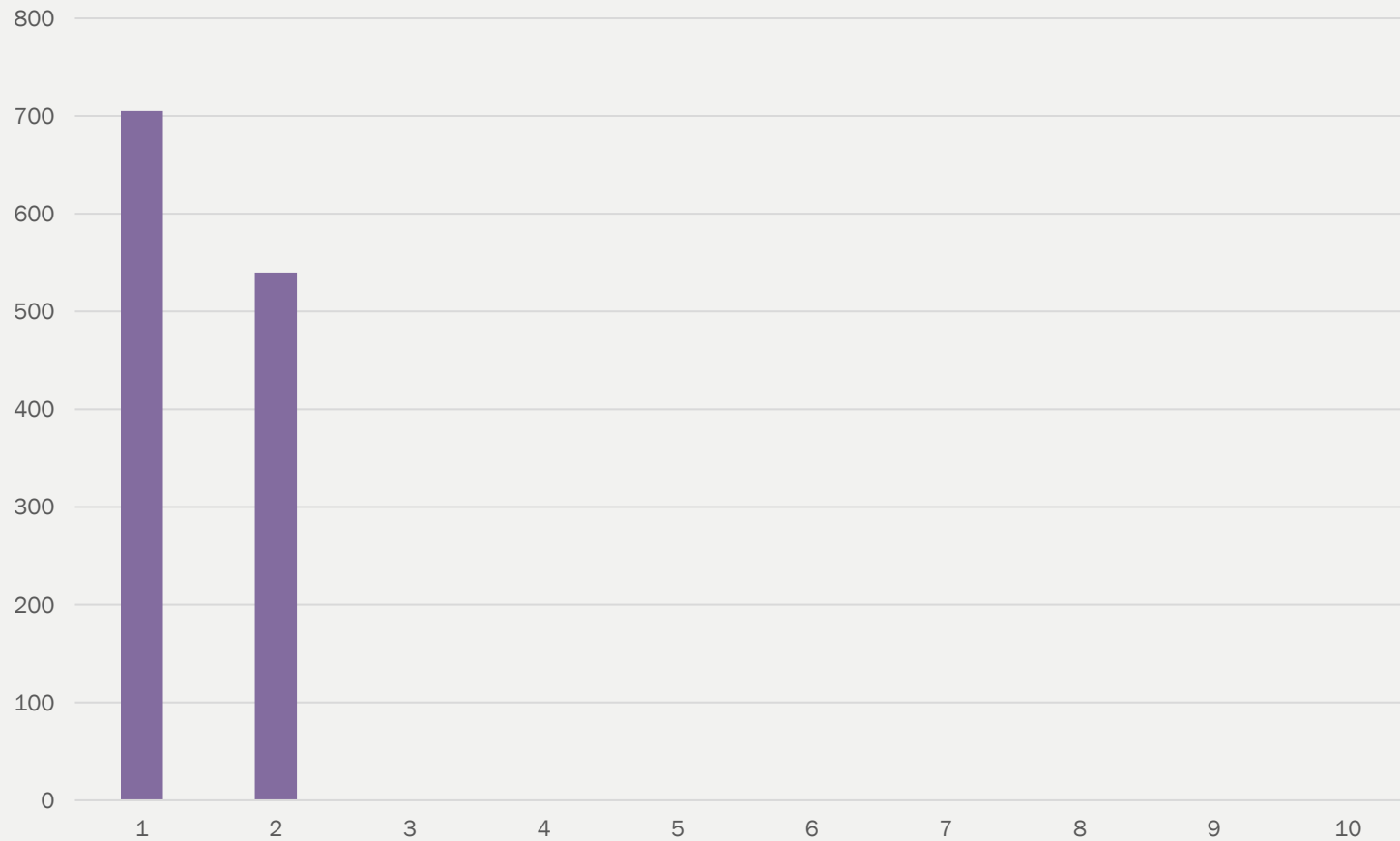
- S curve is the common name for the registered user graph
- J curve just means there's no launch day line, churn is good, we haven't hit the peak, and people are surprised at the slope of the red line.
- But everything hits peak someday.

Users move through



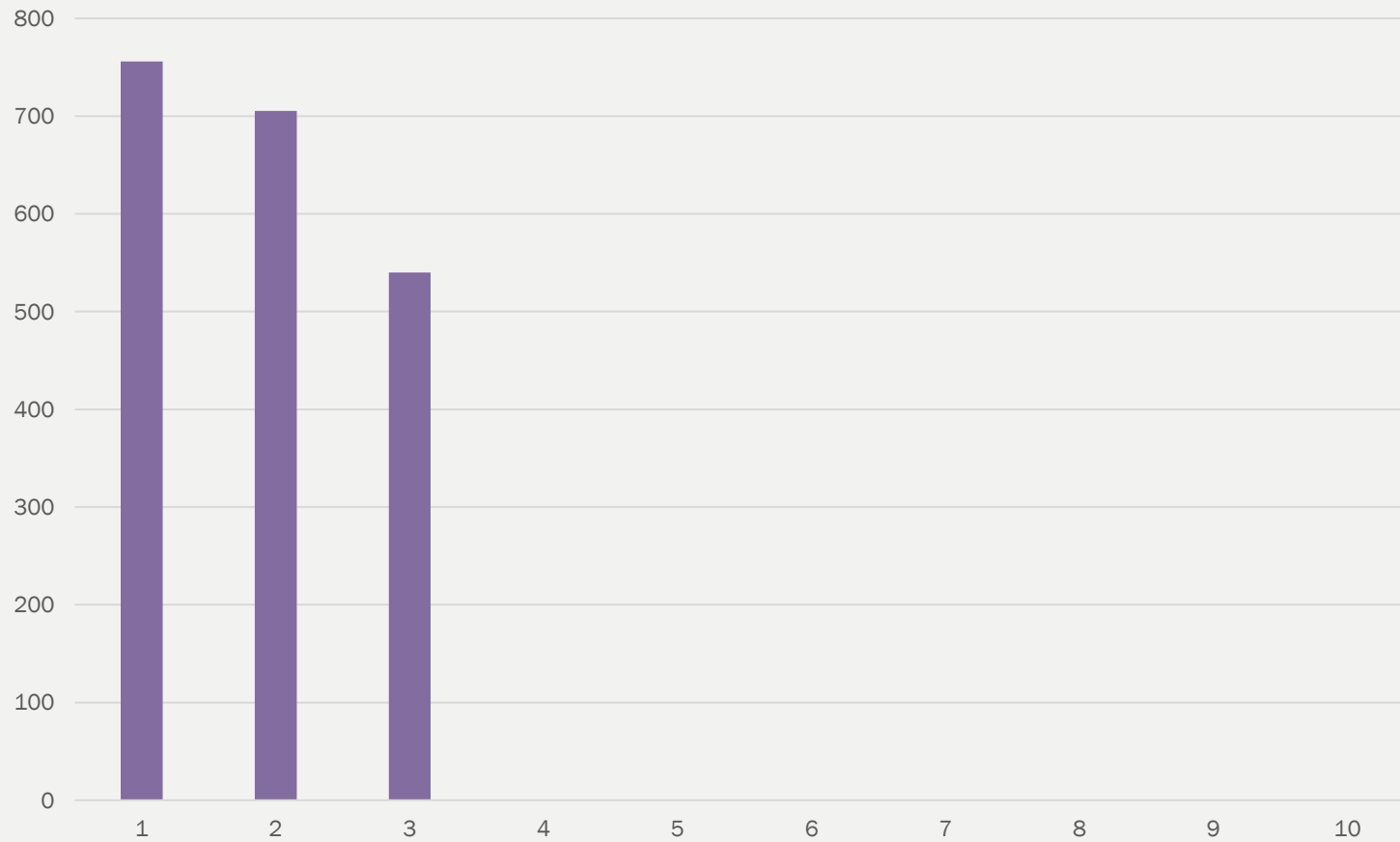
- Let's assume players go up one level per unit (unlikely!)
- You get a wave that does this

Users move through



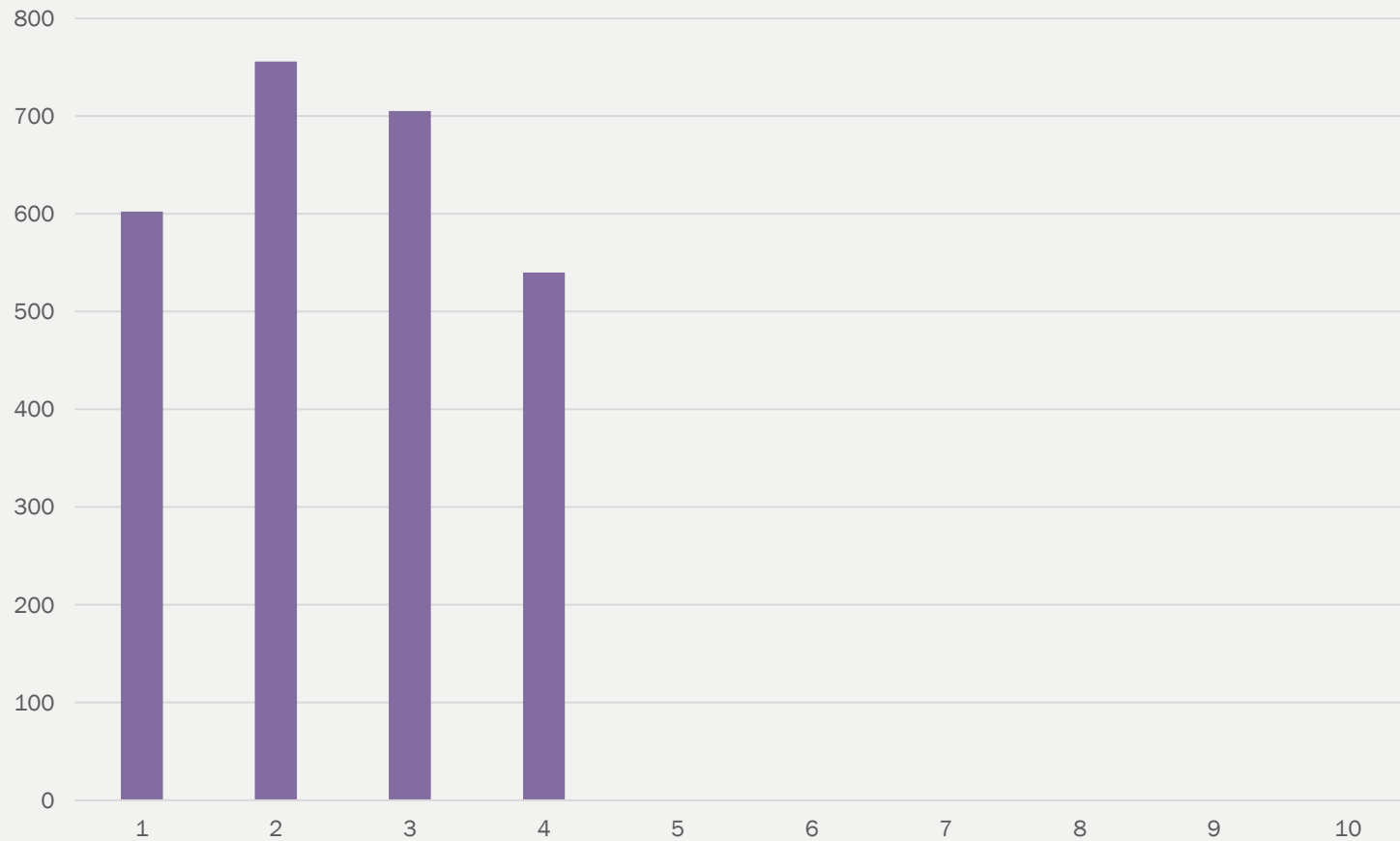
- Let's assume players go up one level per unit (unlikely!)
- You get a wave that does this

Users move through



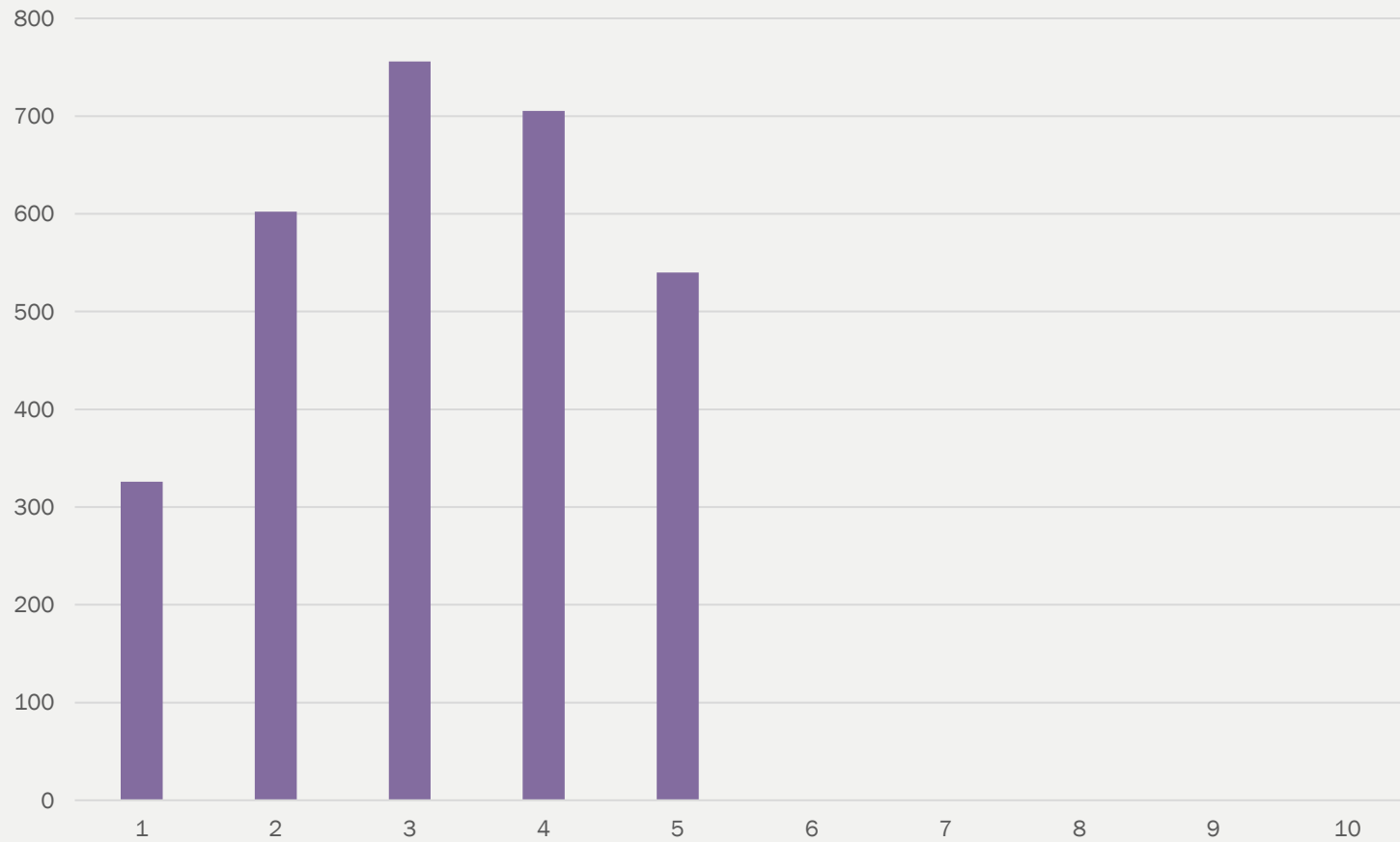
- Let's assume players go up one level per unit (unlikely!)
- You get a wave that does this

Users move through



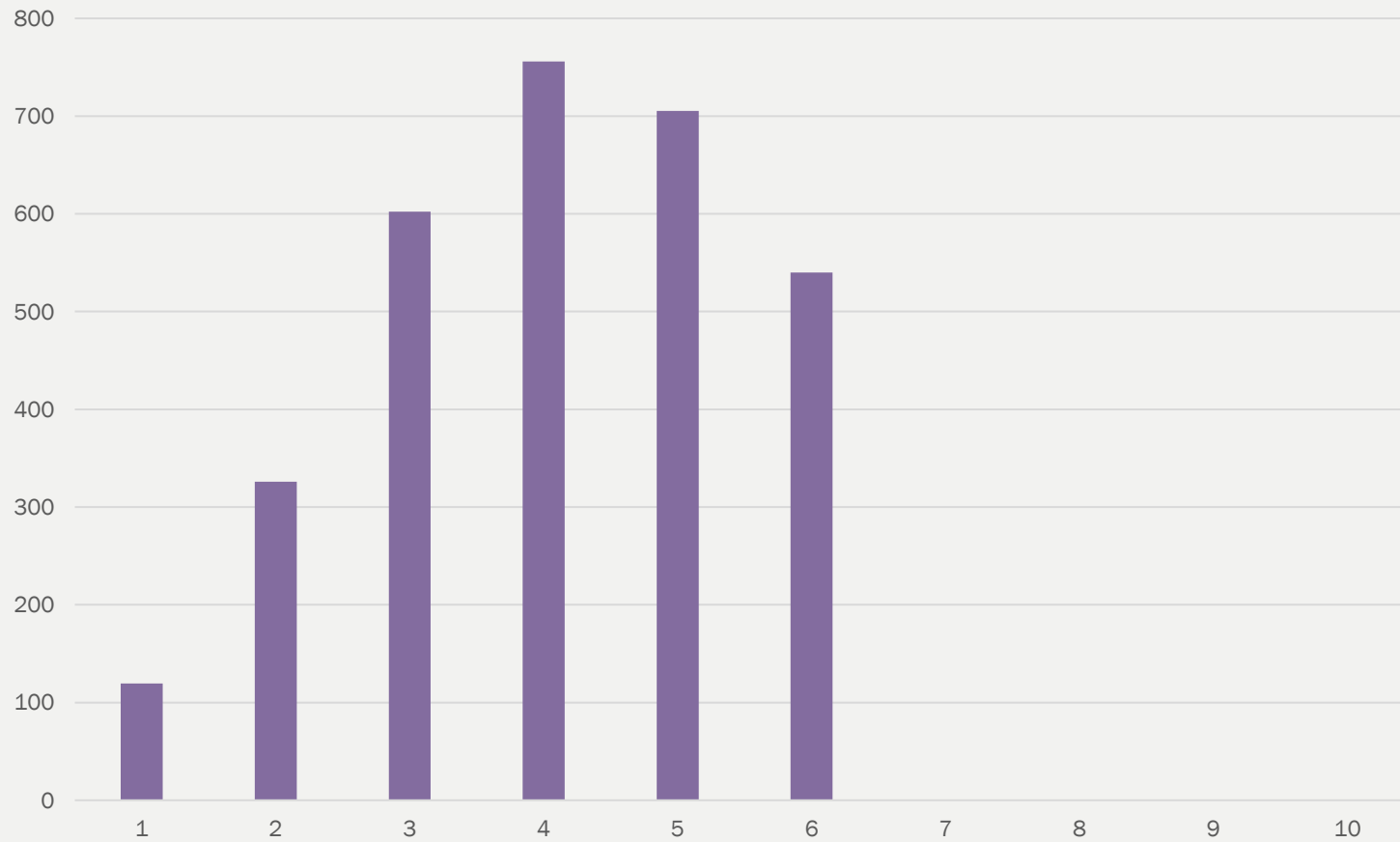
- Let's assume players go up one level per unit (unlikely!)
- You get a wave that does this

Users move through



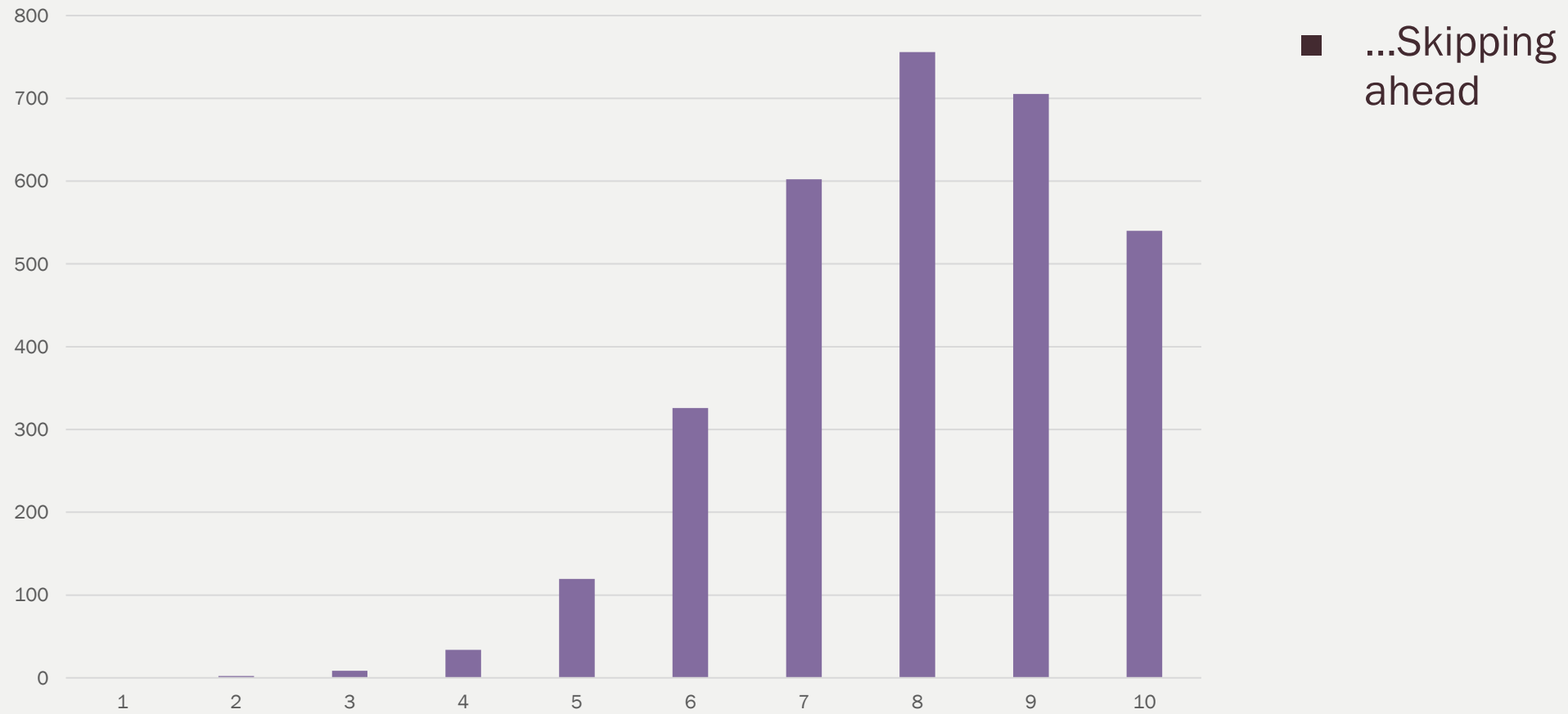
- Let's assume players go up one level per unit (unlikely!)
- You get a wave that does this

Users move through

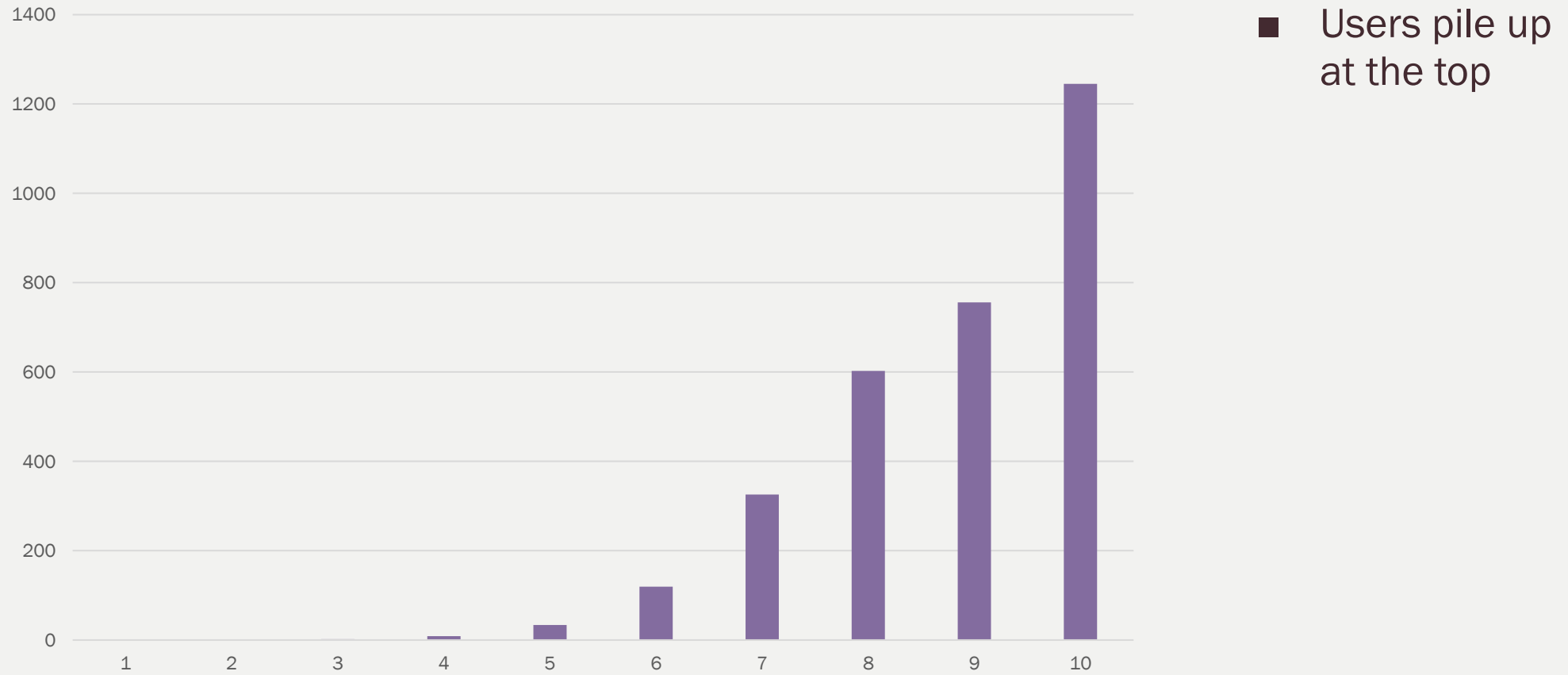


- Let's assume players go up one level per unit (unlikely!)
- You get a wave that does this

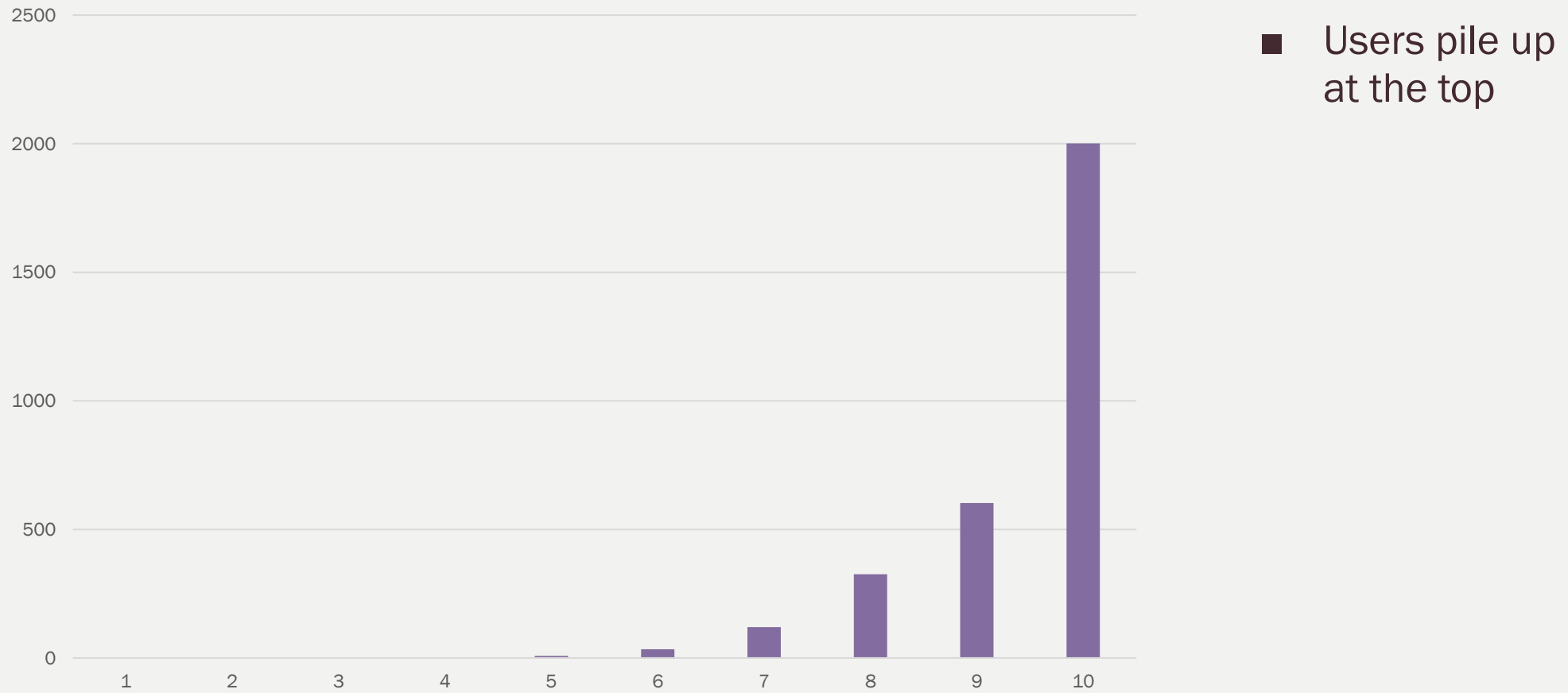
Users move through



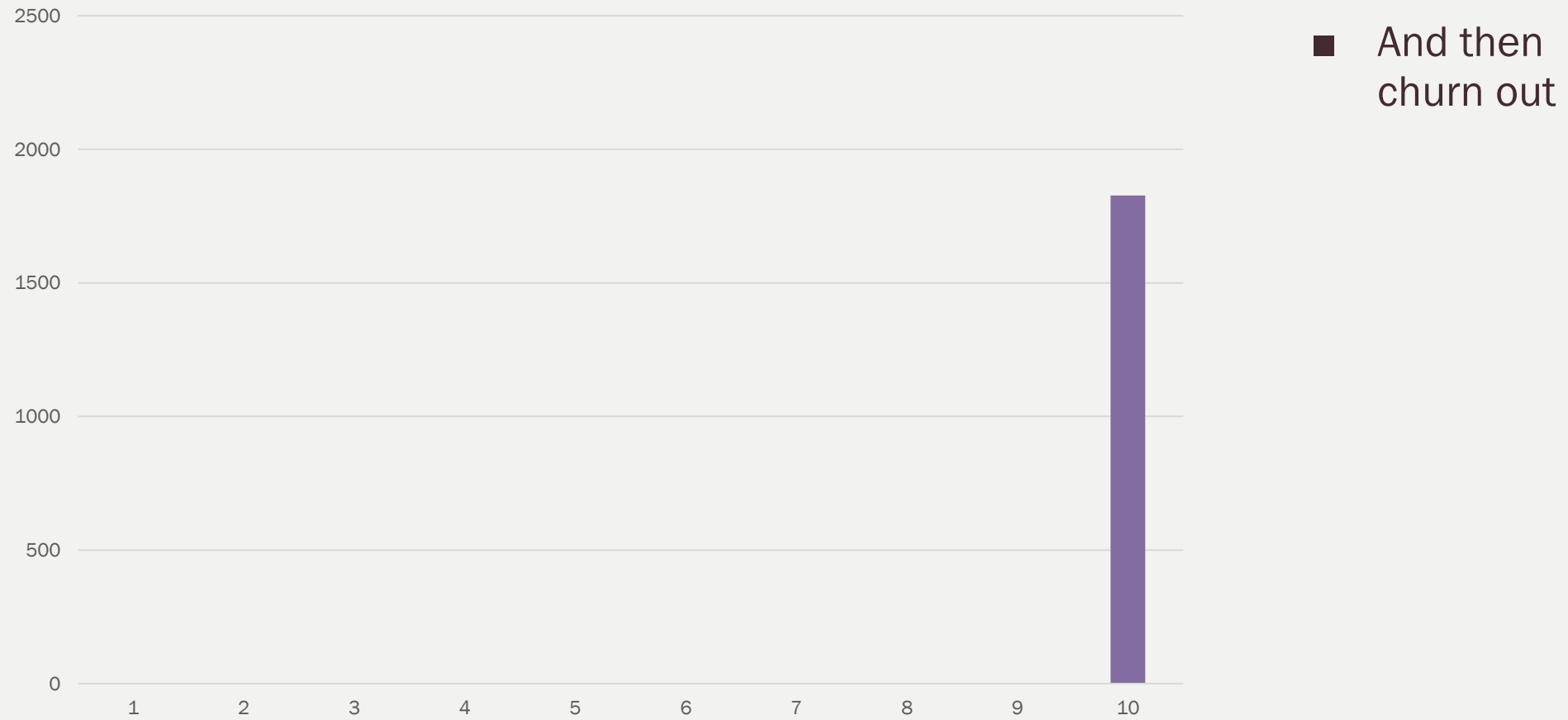
Users move through



Users move through

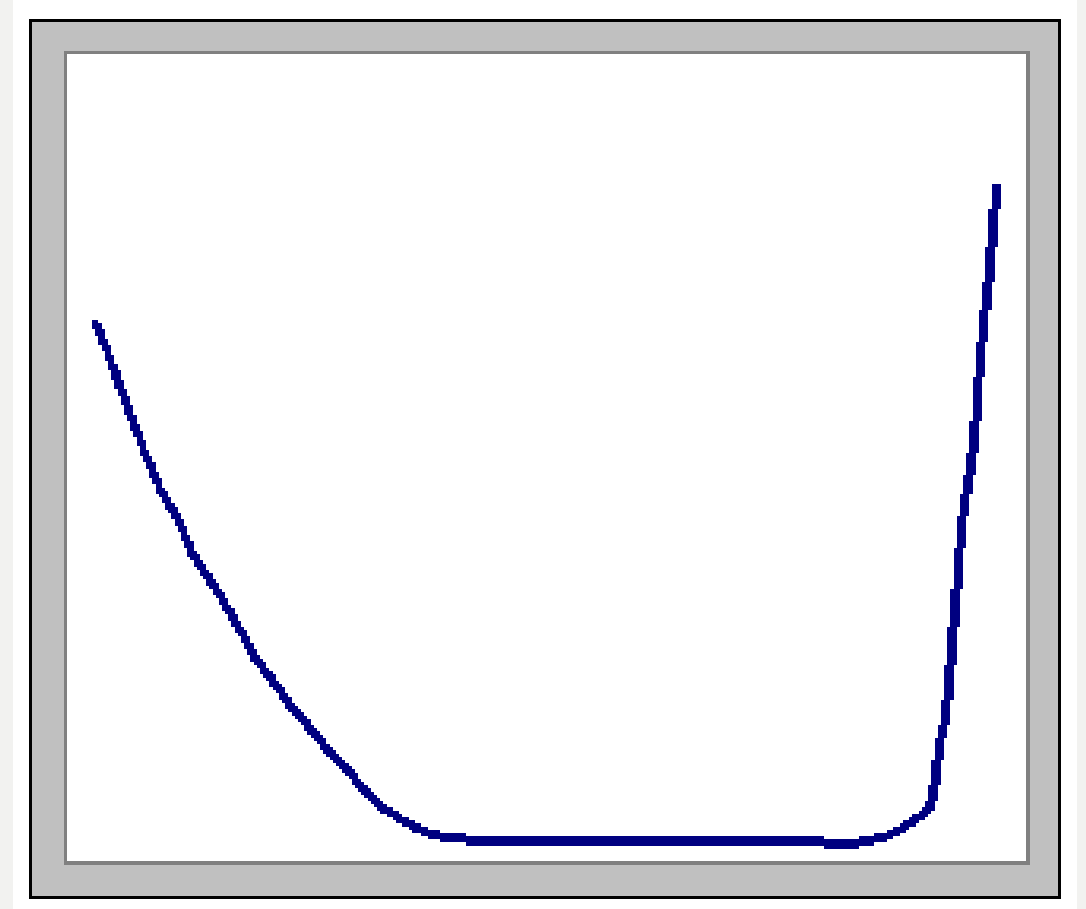


Users move through



But this is oversimplified

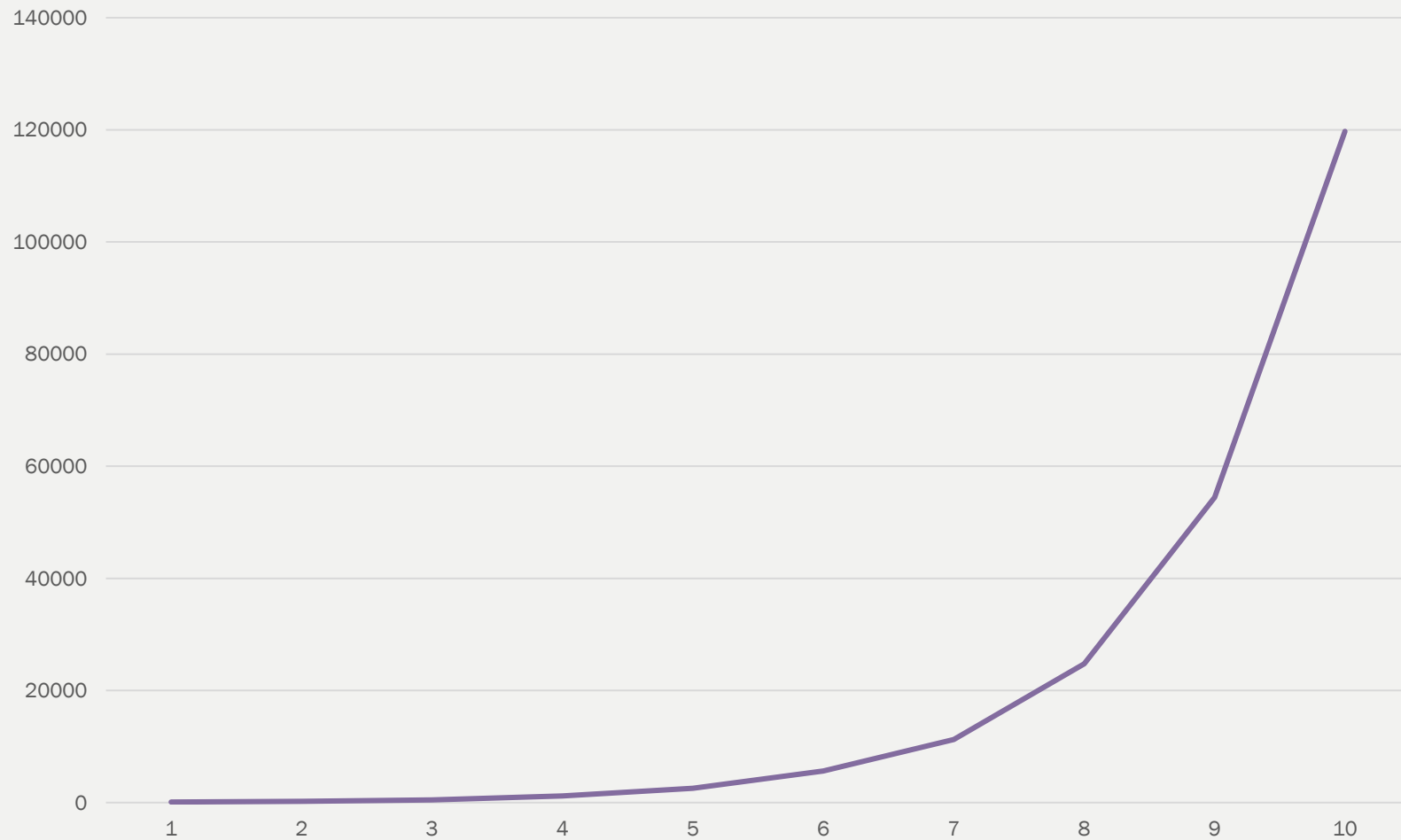
- Not everyone advances at the same rate
- Abandoned accounts pile up on lower levels
 - *And abandonment is likelier at lower levels*
- So it actually should look like a sort of U



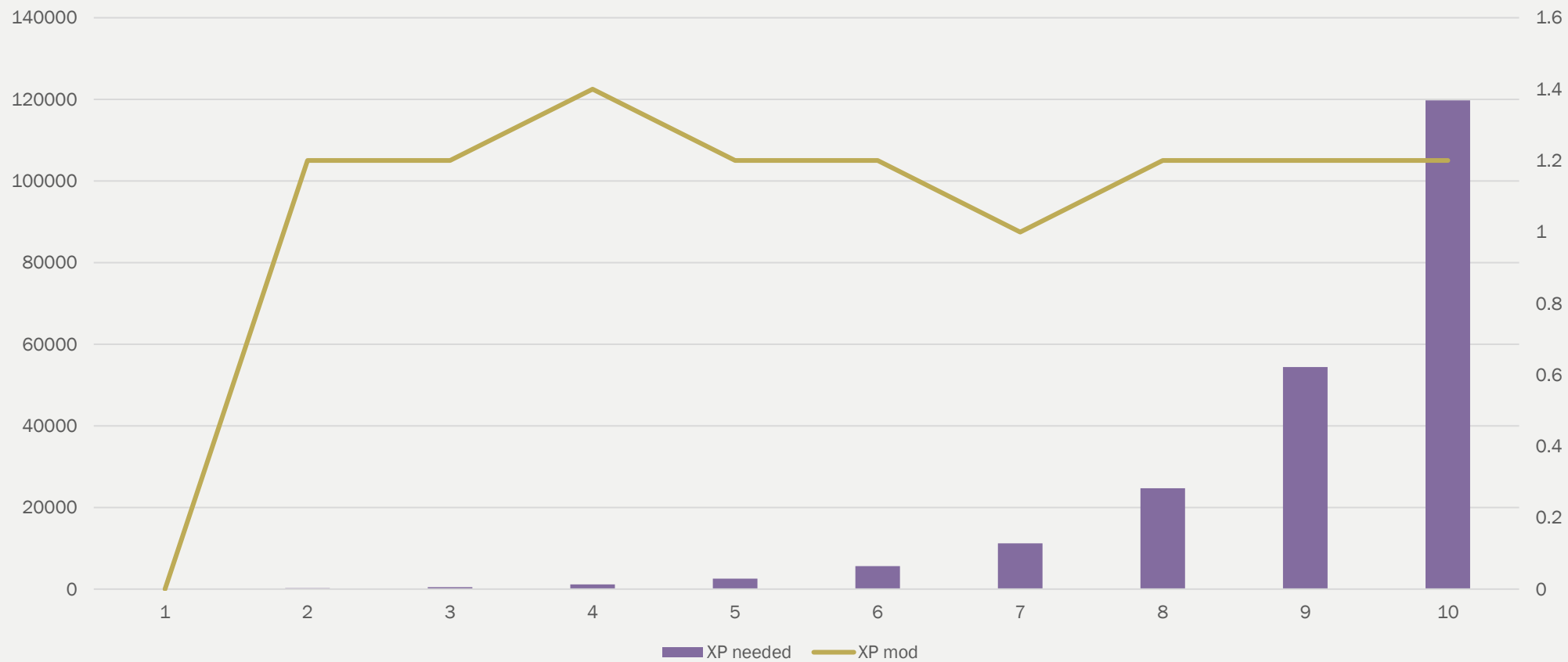
Movement through levels

- And of course, **not all levels are created equal!**
- A level whose difficulty is not in line with the levels around it is termed a **Hell Level**
- It will result in increased churn for the cohort hitting that level

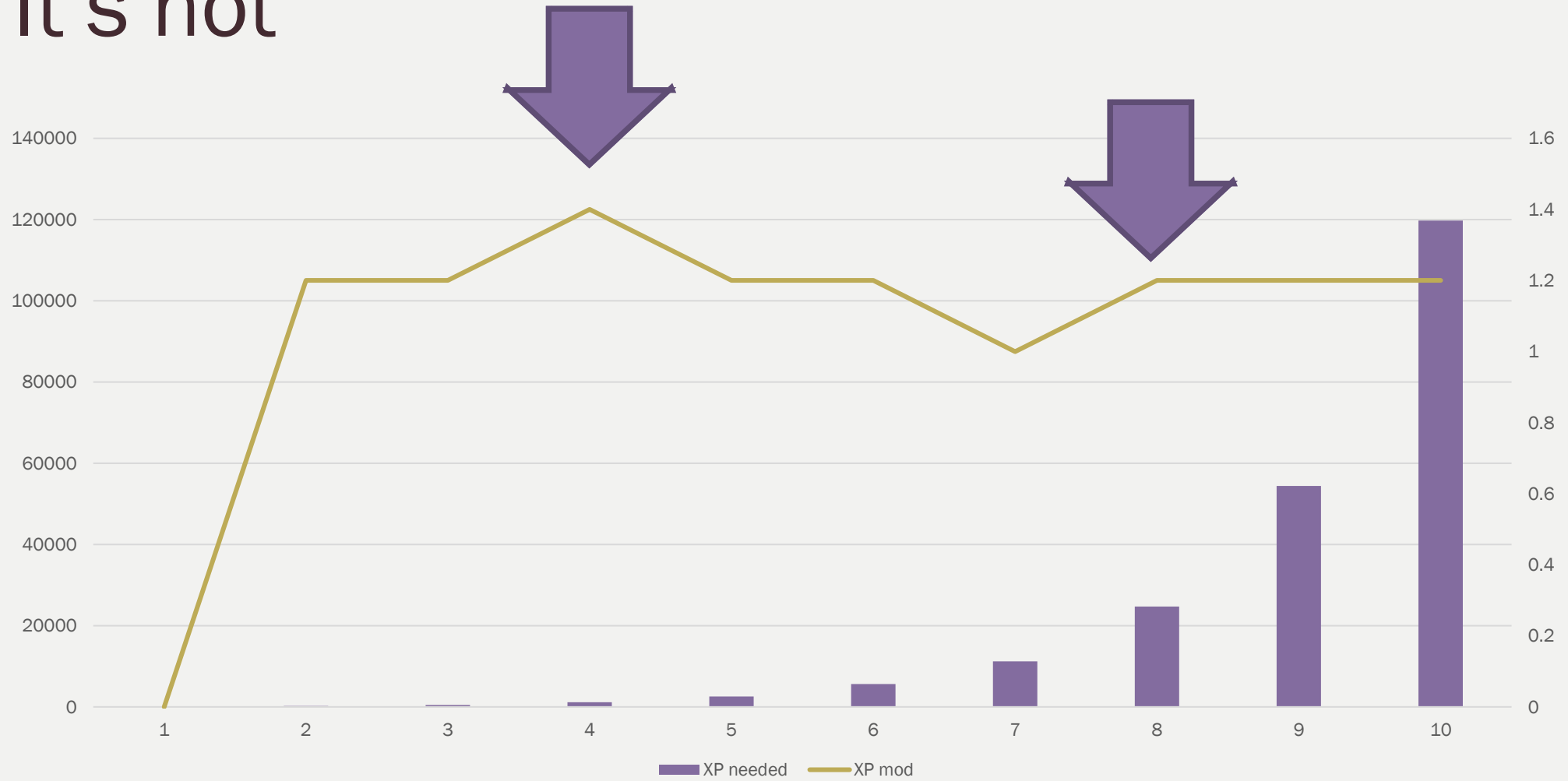
Look good?



It's not

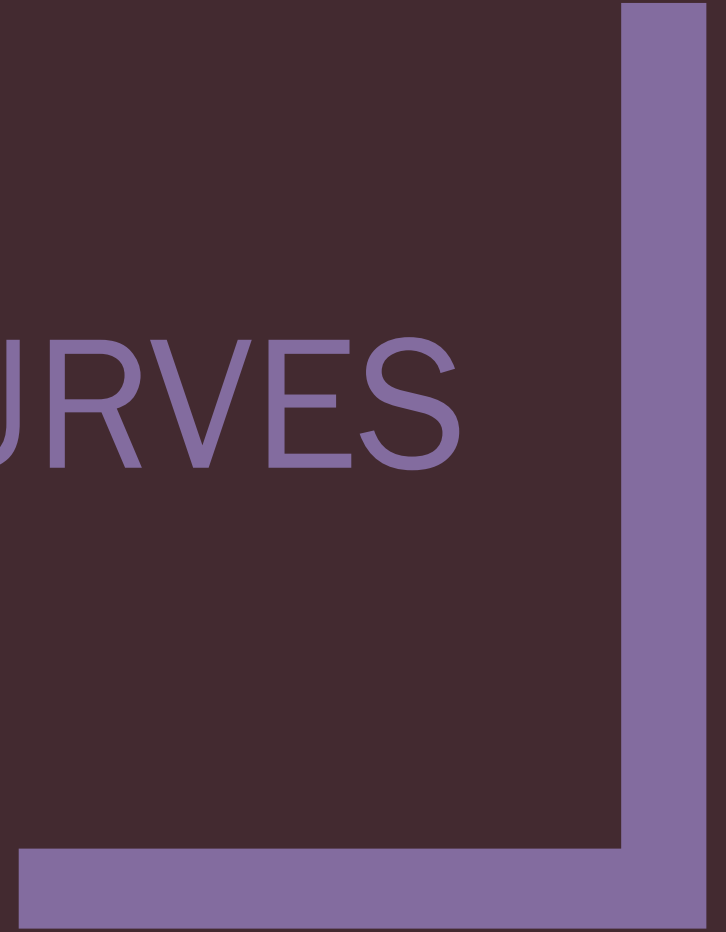


It's not



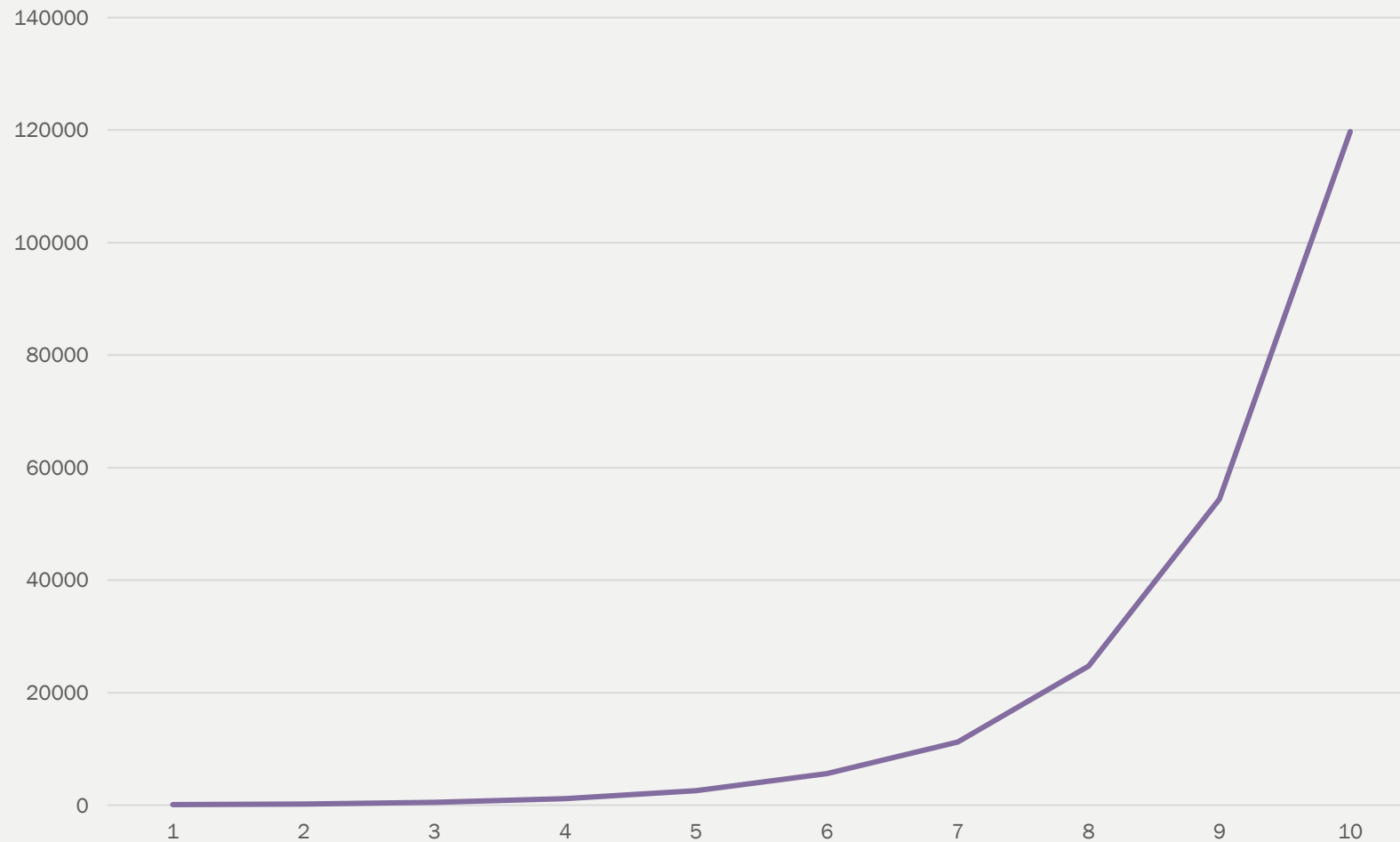
■ These two levels will have increased churn

LEVEL CURVES



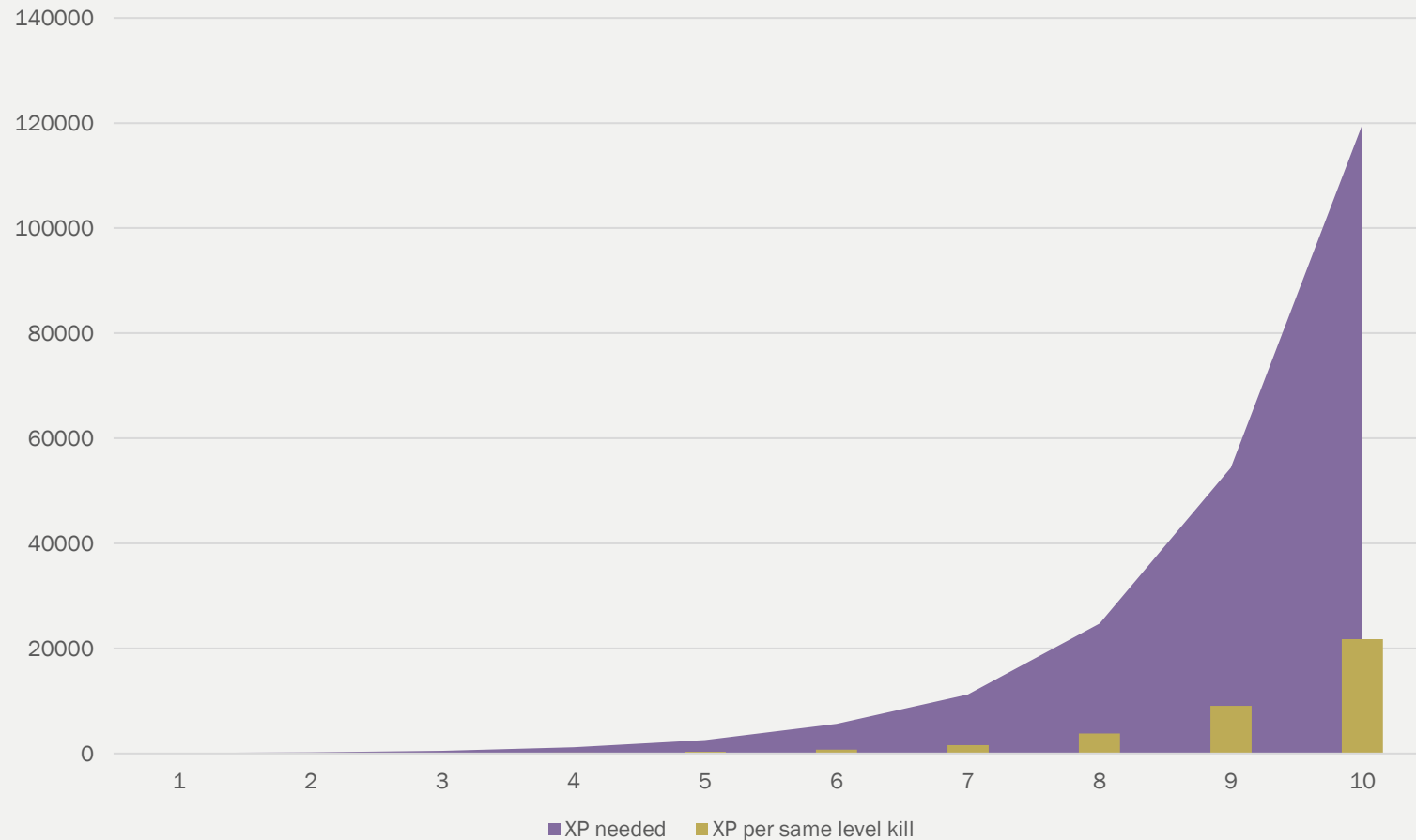
Exponential curve

- We tend to think of this as a steep curve... but it might not be



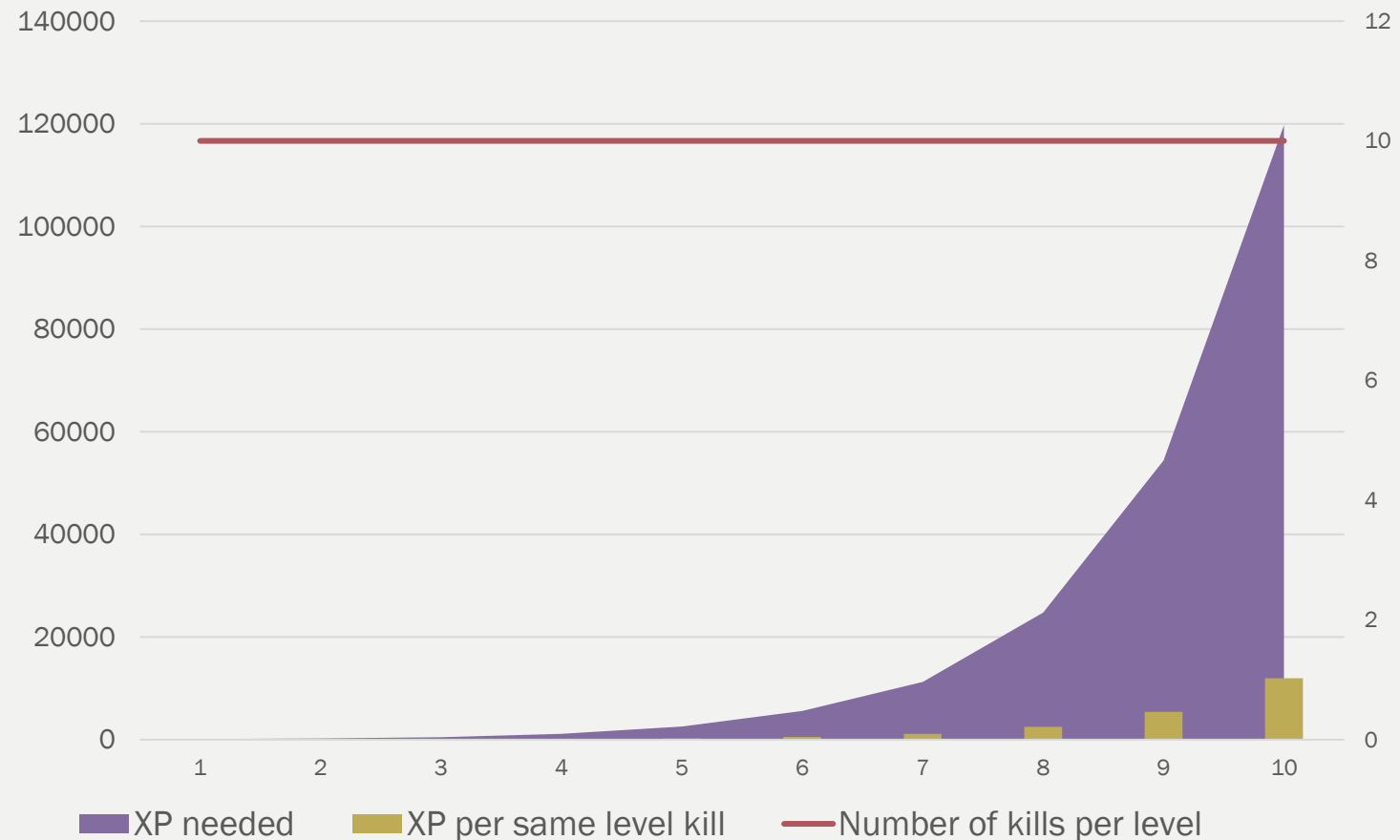
How many “steps” (encounters) to level?

- Here's with increased XP per encounter



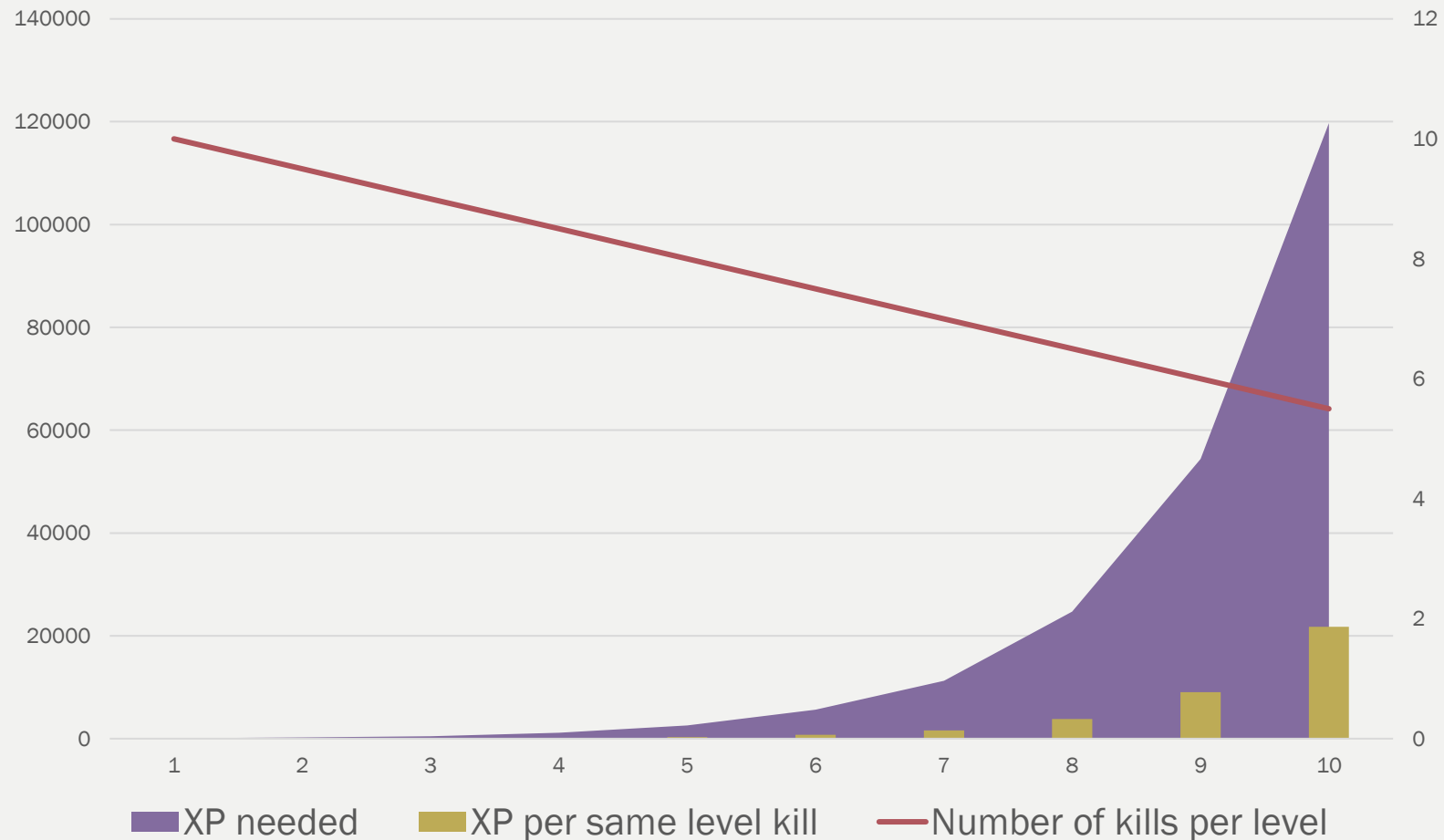
Number of encounters is the *real* pace

- Here's with increased XP per encounter
- Look at the actual curve... it's really flat

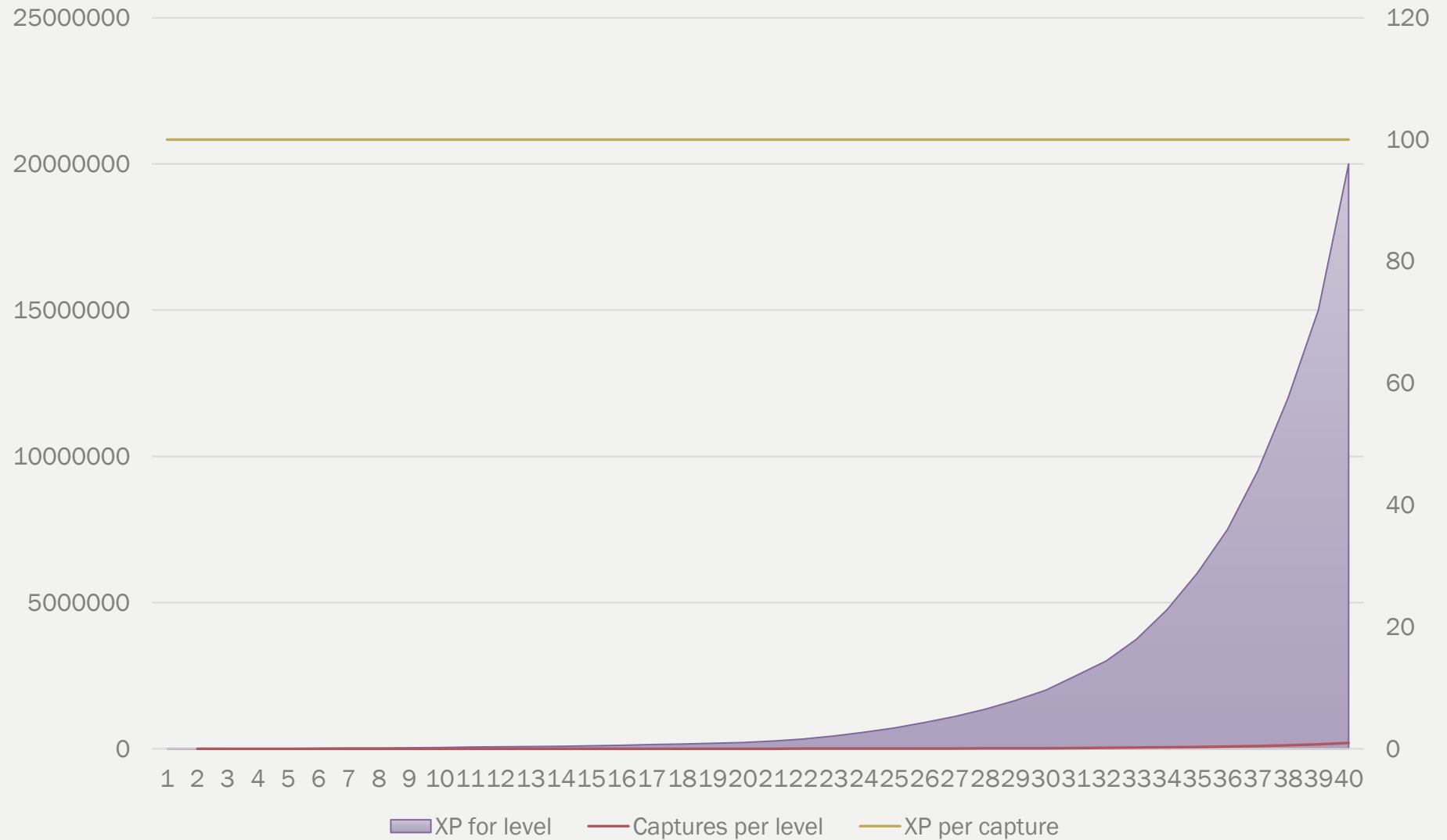


In fact, you can make the game get easier as you advance

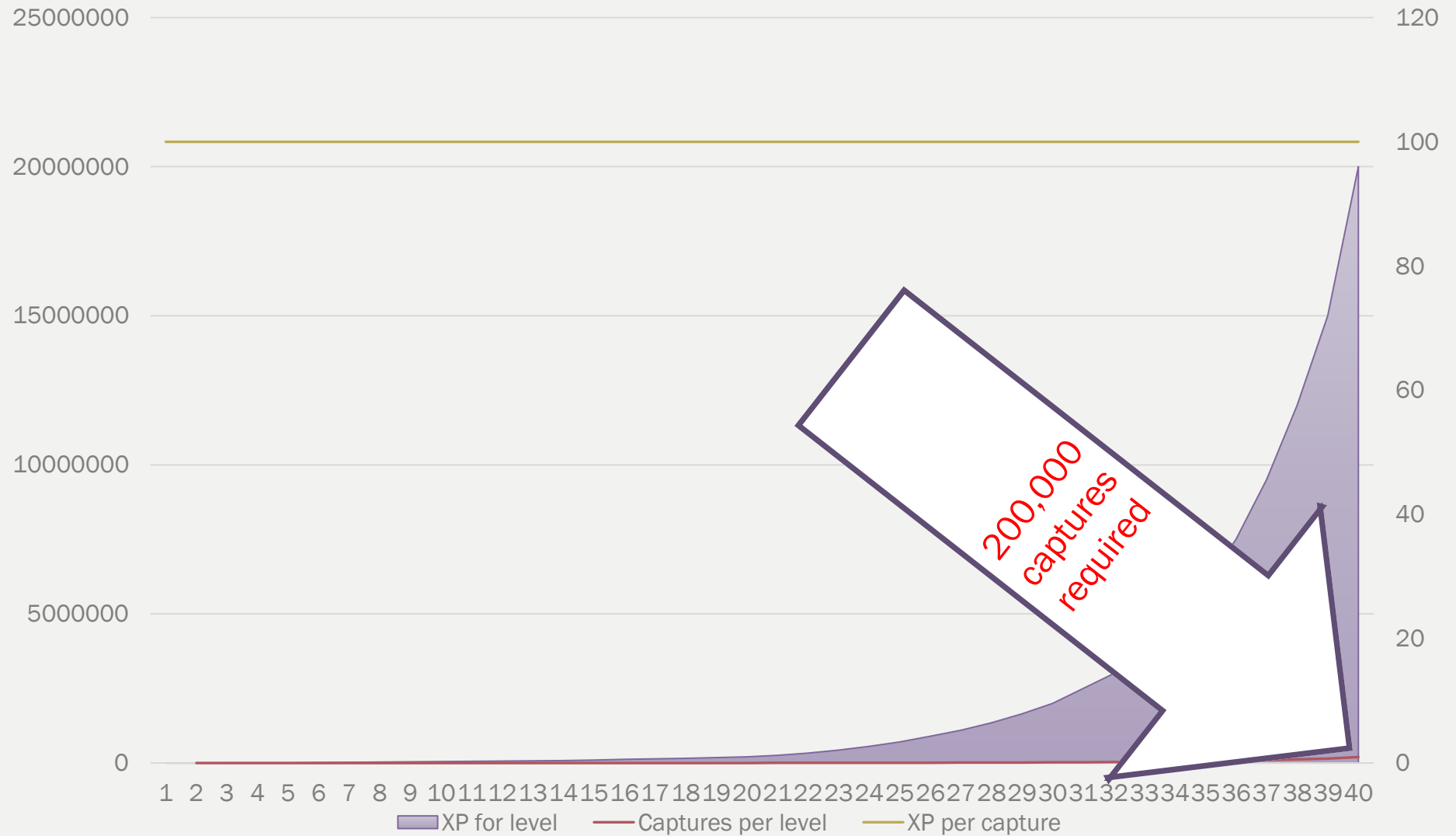
- With steps that get bigger than the rate of the XP curve



Pokemon Go



Pokemon Go



Actually, it's worse than that

- You are shown a greater spread of Pokemon, so ones with tougher catch rates and higher flee rates (first way)
- AND flee rates get worse as you level (second way)
- AND catch rates drop as you level (third)
- AND XP needed for next level rises dramatically
- AND Pokeball supplies stay flat or even decrease (because the 3-4 drops you get start including other stuff) (fourth way)
- AND XP awarded stays flat regardless of level.

What's the success rate?

- If a given encounter at level 10 is twice as hard as one at level one, here's the REAL difficulty pacing; on average, the step size will start to fall
- If an encounter is harder to find, they are spaced farther apart
- If an encounter costs something to take on, then access to those is part of the difficulty

Outcome:

- Less spawn availability of appropriate type
- Less rewards per minute
- 20 year time to max assuming optimal play
- Rampant cheating
- Massive churn
- Compressed retention curve
- Literal billions of lost dollars

Exploring fixes

- Increase pokeball drop rates... this hits revenue!
- Increase XP value on monsters... will ripple down to newbies too, affecting your play pacing.
- Reduce flee rates or increase catch rates... also hits revenue!

The right way

- Define level distances as a smooth ramp of number of successful encounters.
 - Pick whatever visual curve looks good – a 1.2x increase per level, or whatever
 - XP value per encounter is defined as XP needed divided by number of encounters
 - Walk through your encounter logic to ensure steady feedback to the player via successful fights
 - If you need to tune, change number of encounters and cascade the values back
-
- XP curve won't change
 - XP per monster will
 - Most importantly, **player moment to moment pacing won't**

FINDING YOUR TARGET

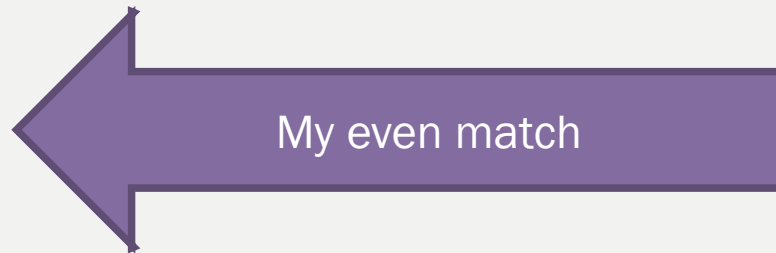


Let's start with a table of possible encounters

Level	Monster
1	Mouse
2	Hamster
3	Gerbil
4	Rat
5	Squirrel
6	Guinea Pig
7	Chinchilla
8	Marmot
9	Capybara
10	ROUS

What are the odds of getting my even match?

Level	Monster
1	Mouse
2	Hamster
3	Gerbil
4	Rat
5	Squirrel
6	Guinea Pig
7	Chinchilla
8	Marmot
9	Capybara
10	ROUS



You guessed...

1/10

Roll, roll roll...

[illegible]

Roll, roll roll...

[illegible]

Roll, roll roll...

[illegible]

Roll, roll roll...

[illegible]

Roll, roll roll...

[illegible]

Roll, roll roll...

[illegible]

Roll, roll roll...

Try number...	Rolled a...	Got a...	What I did:
1	3	Gerbil	Wasted ammo
2	3	Gerbil	Wasted ammo
3	3	Gerbil	Wasted ammo
4	10	ROUS	Died
5	8	Marmot	Fled
6	3	Gerbil	Wasted ammo
7	6	Guinea pig	Fled

Roll, roll roll...

Try number...	Rolled a...	Got a...	What I did:
1	3	Gerbil	Wasted ammo
2	3	Gerbil	Wasted ammo
3	3	Gerbil	Wasted ammo
4	10	ROUS	Died
5	8	Marmot	Fled
6	3	Gerbil	Wasted ammo
7	6	Guinea pig	Fled
8	10	ROUS	Died instantly

Roll, roll roll...

Try number...	Rolled a...	Got a...	What I did:
1	3	Gerbil	Wasted ammo
2	3	Gerbil	Wasted ammo
3	3	Gerbil	Wasted ammo
4	10	ROUS	Died
5	8	Marmot	Fled
6	3	Gerbil	Wasted ammo
7	6	Guinea pig	Fled
8	10	ROUS	Died instantly
9	10	ROUS	Died on sight

Roll, roll roll...

Try number...	Rolled a...	Got a...	What I did:
1	3	Gerbil	Wasted ammo
2	3	Gerbil	Wasted ammo
3	3	Gerbil	Wasted ammo
4	10	ROUS	Died
5	8	Marmot	Fled
6	3	Gerbil	Wasted ammo
7	6	Guinea pig	Fled
8	10	ROUS	Died instantly
9	10	ROUS	Died on sight
10	4	Rat	Got pissed off

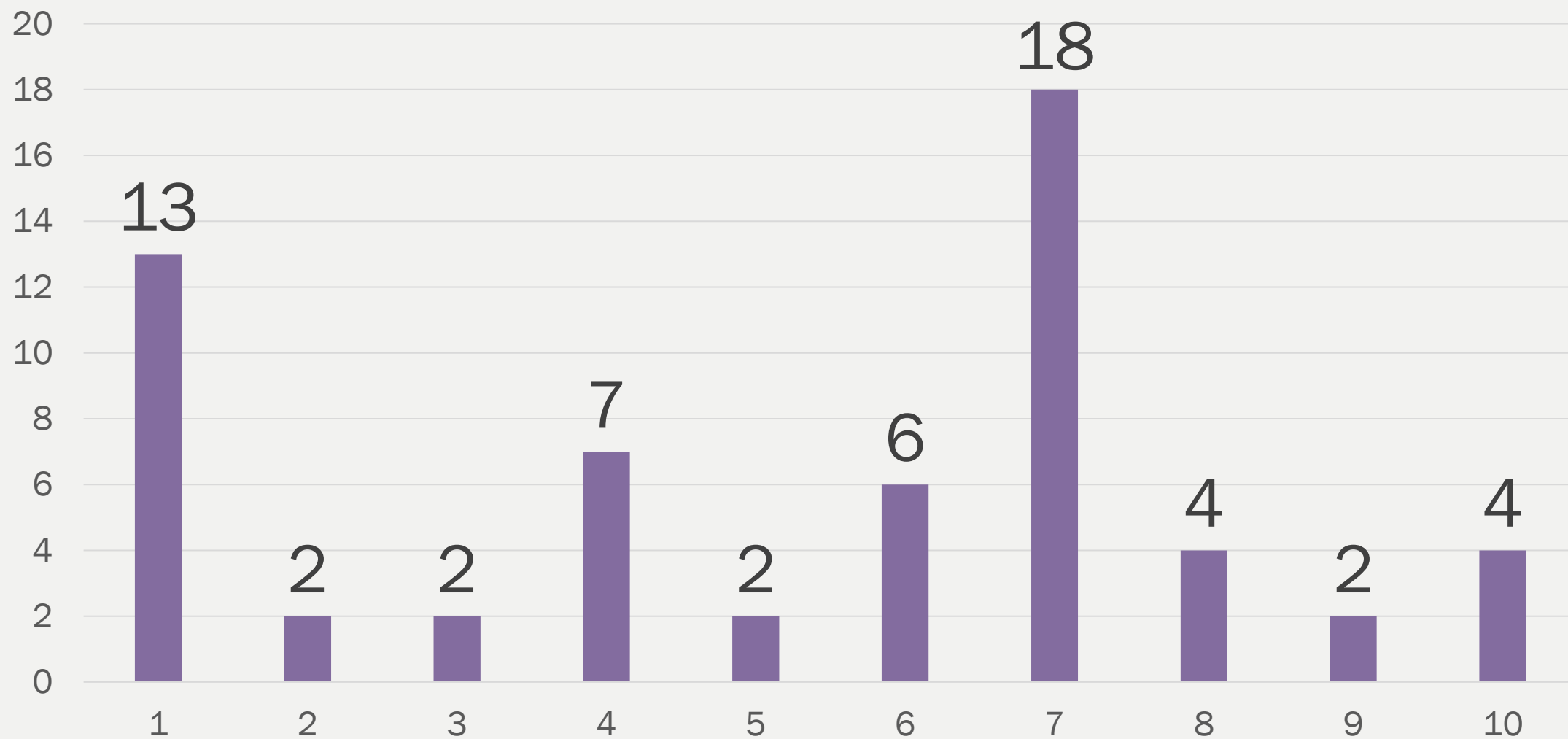
Roll, roll roll...

Try number...	Rolled a...	Got a...	What I did:
1	3	Gerbil	Wasted ammo
2	3	Gerbil	Wasted ammo
3	3	Gerbil	Wasted ammo
4	10	ROUS	Died
5	8	Marmot	Fled
6	3	Gerbil	Wasted ammo
7	6	Guinea pig	Fled
8	10	ROUS	Died instantly
9	10	ROUS	Died on sight
10	4	Rat	Got pissed off
11	9	Capybara	Cussed at developer

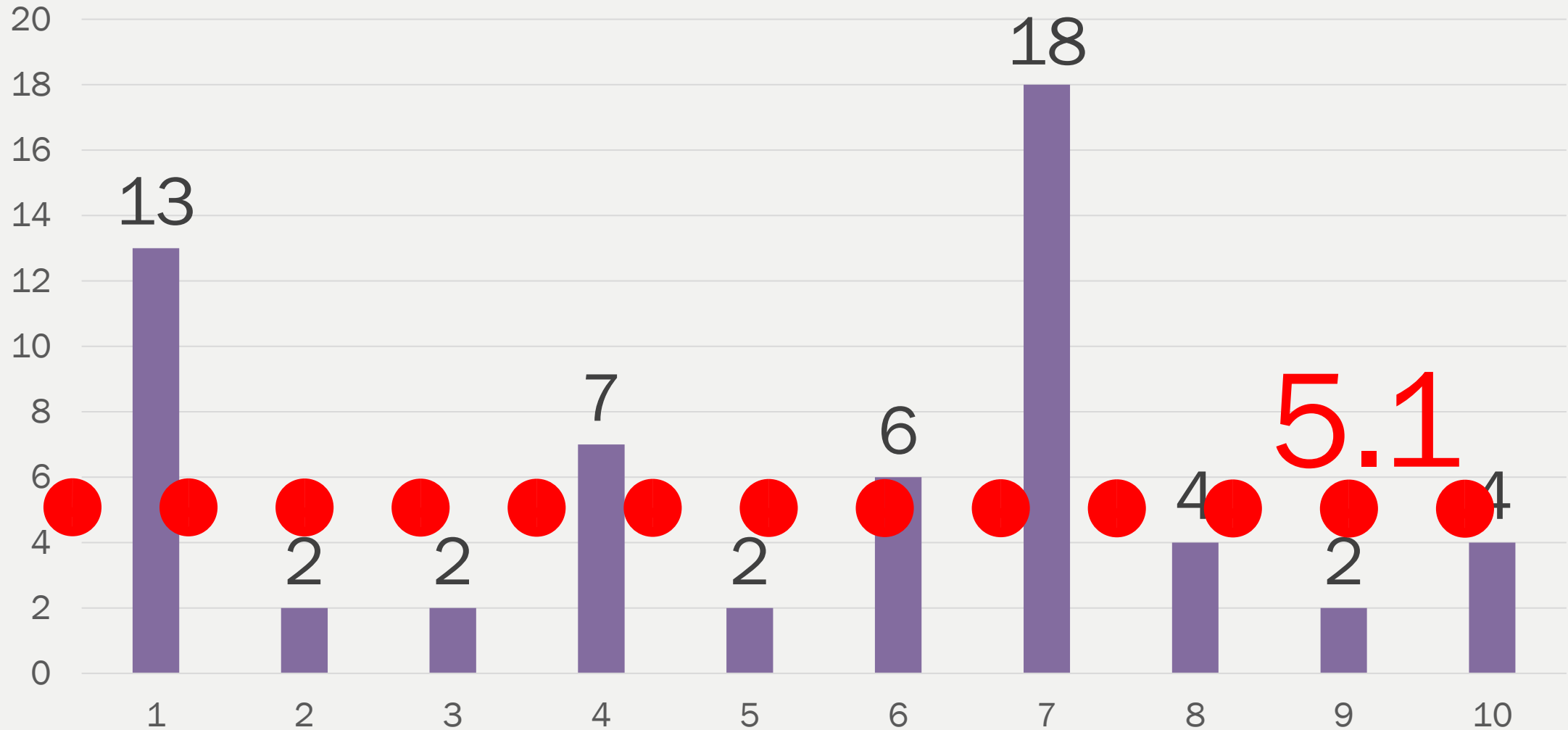
Roll, roll roll...

Try number...	Rolled a...	Got a...	What I did:
1	3	Gerbil	Wasted ammo
2	3	Gerbil	Wasted ammo
3	3	Gerbil	Wasted ammo
4	10	ROUS	Died
5	8	Marmot	Fled
6	3	Gerbil	Wasted ammo
7	6	Guinea pig	Fled
8	10	ROUS	Died instantly
9	10	ROUS	Died on sight
10	4	Rat	Got pissed off
11	9	Capybara	Cussed at developer
12	5	Squirrel!	Thanked the gods

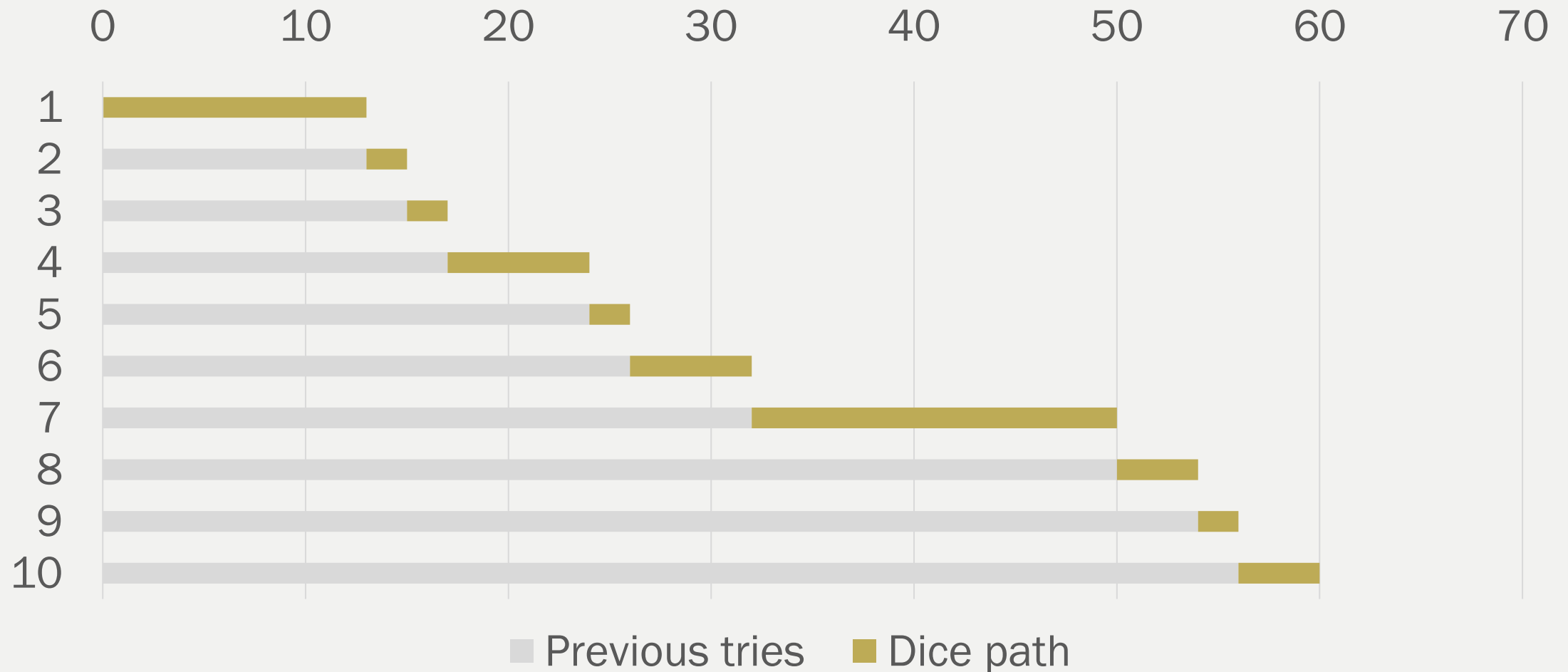
Tries it took to get ten squirrels



Tries it took to get ten squirrels



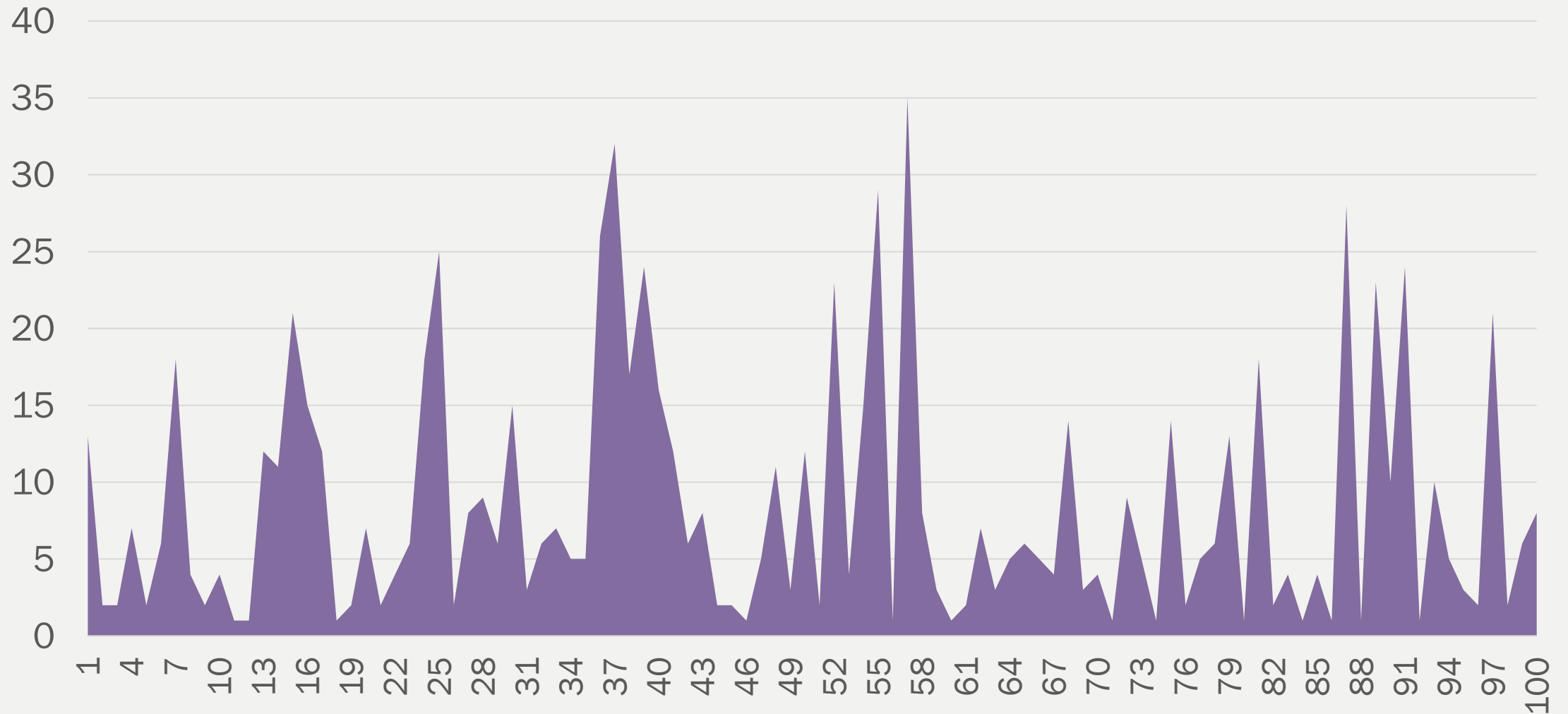
The player pacing was very unpredictable



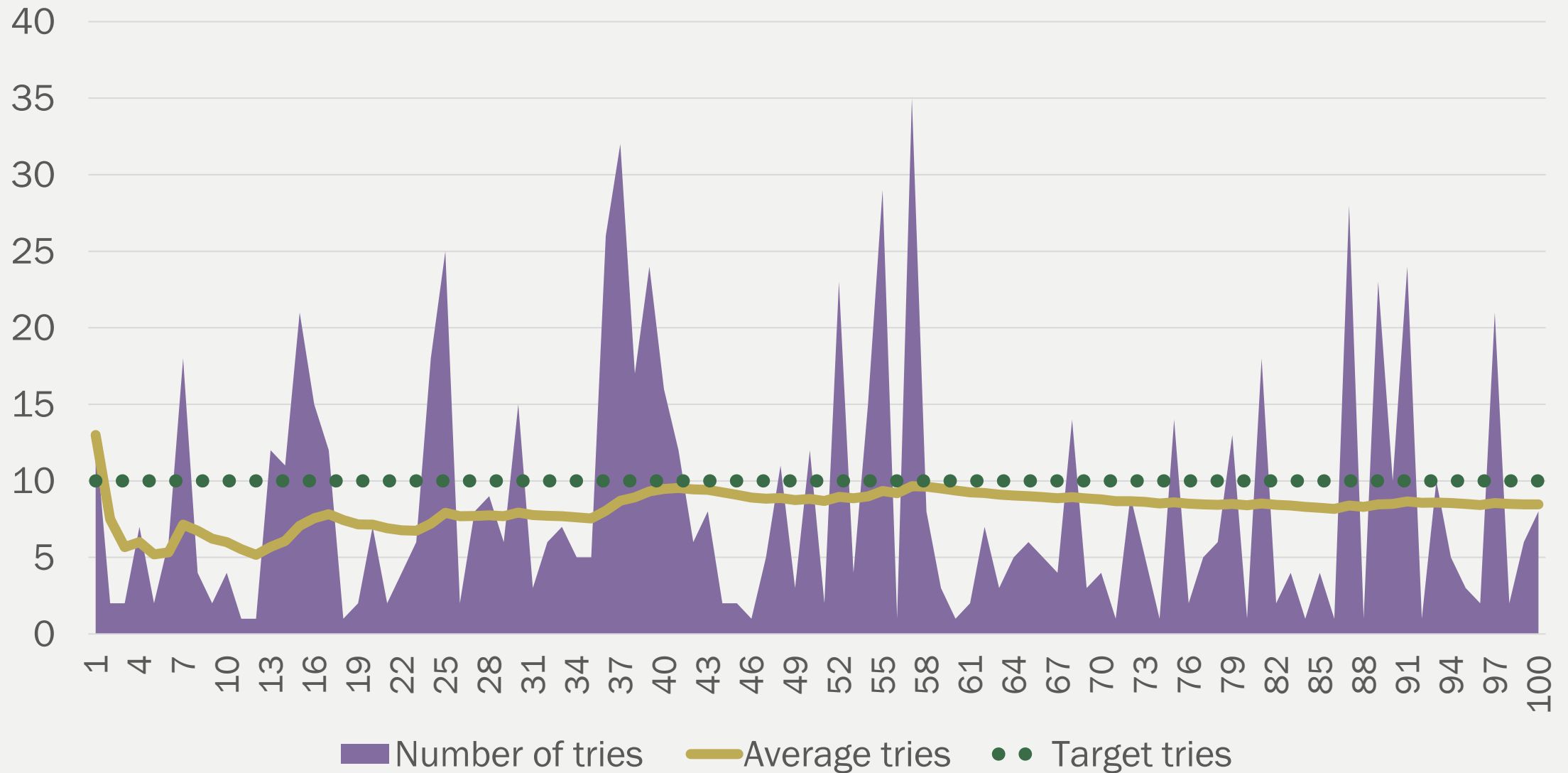
Let's do way more to see it smooth out

1,000 tries!!

Let's do way more to see it smooth out

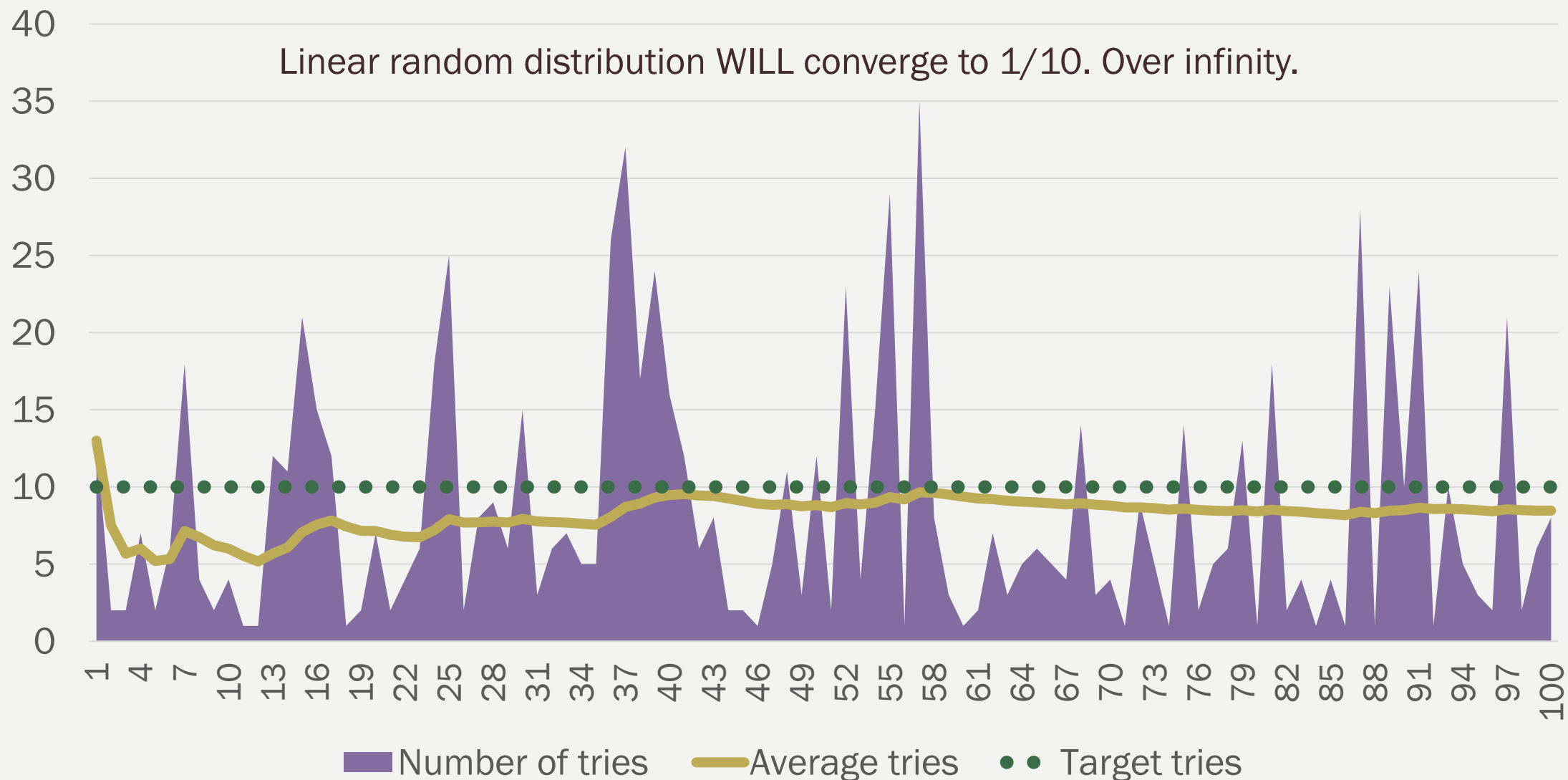


That didn't work



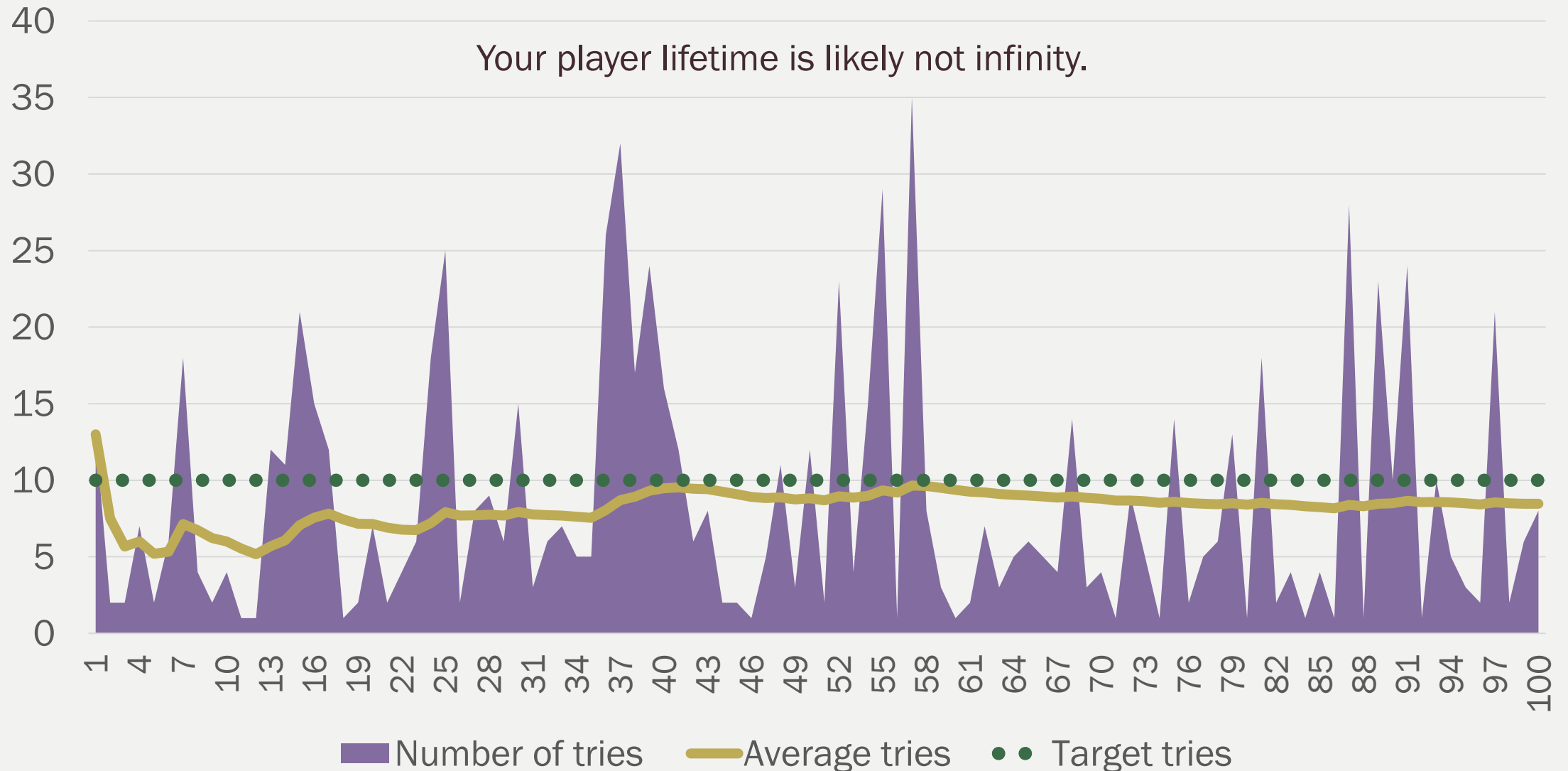
That didn't work

Linear random distribution WILL converge to 1/10. Over infinity.



That didn't work

Your player lifetime is likely not infinity.



In fact

The percentage chance of a player
failing to get a squirrel
100 times in a row is...

In fact

The percentage chance of a player
failing to get a squirrel
100 times in a row is...

$$0.9 * 0.9 * 0.9 * 0.9 \dots$$

$$= (0.9)^{100}$$

$$= 0.003\%$$

In fact

The percentage chance of a player
failing to get a squirrel
100 times in a row is...

Not too bad, right?

= 0.003%

In fact

The percentage chance of a player
failing to get a squirrel
100 times in a row is...

Not too bad, right?

1 of every 333 players quitting.

In fact

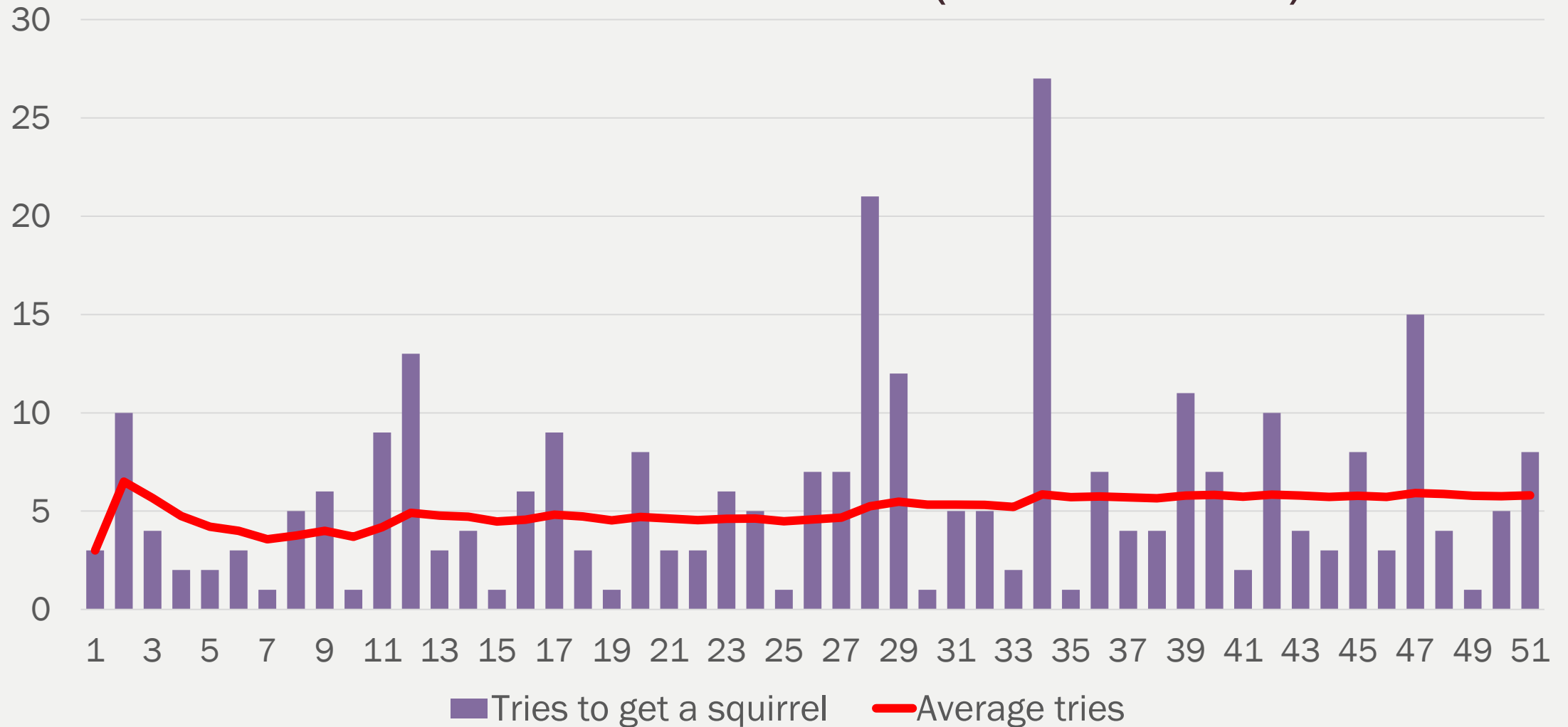
The percentage chance of a player
failing to get a squirrel
100 times in a row is...

Not too bad, right?

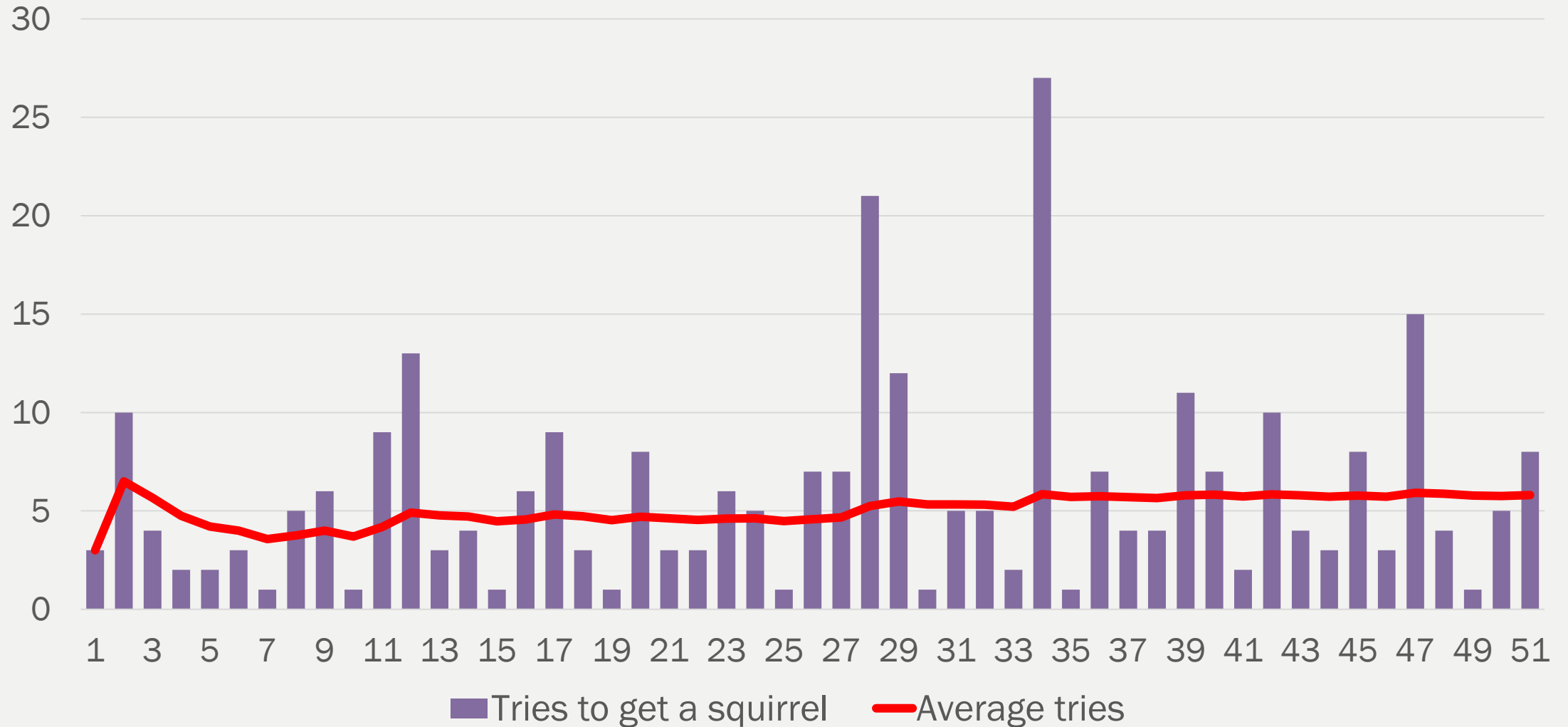
1 of every 333 players quitting.

(worse if you assume they will also quit if it takes 99 times)

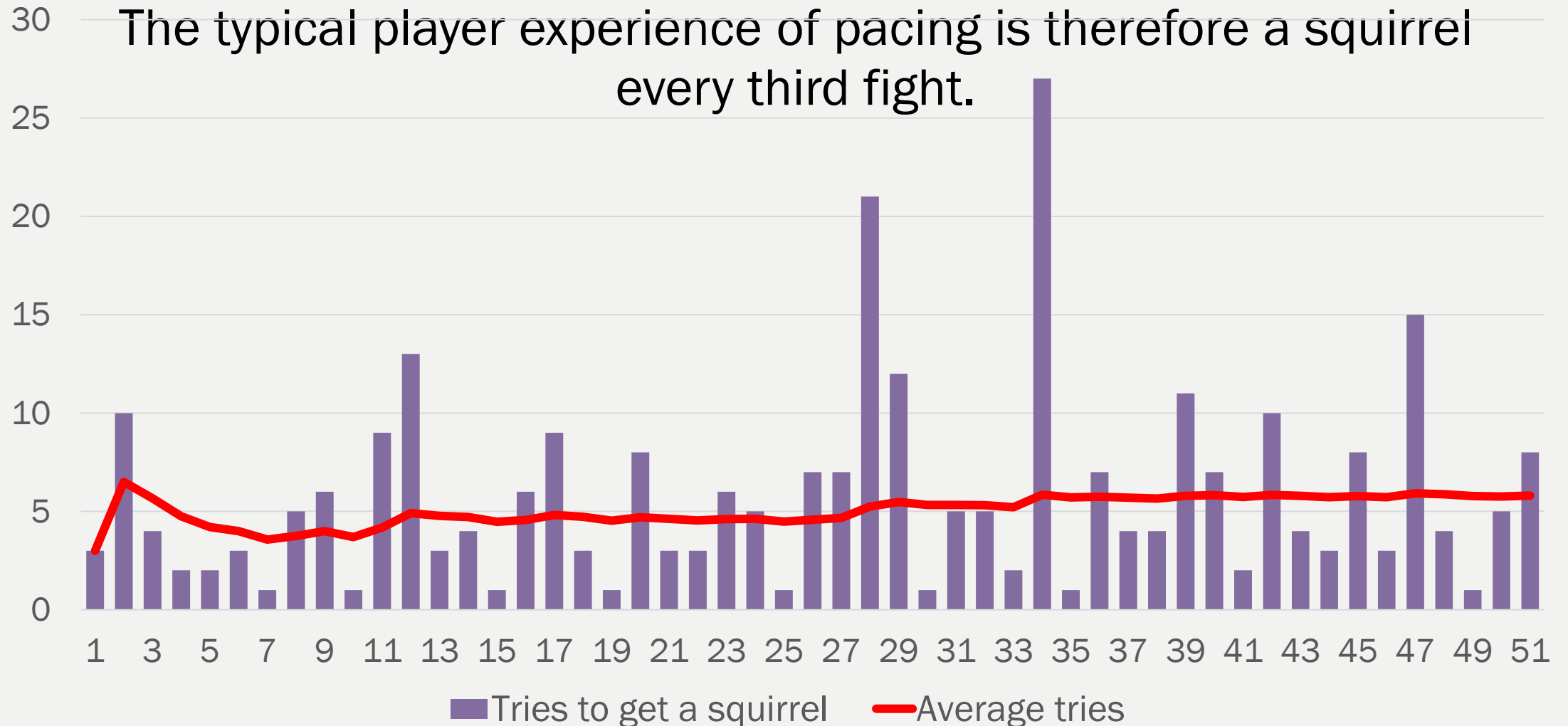
Multiple dice give a bell curve, but these can still have real outliers (this is 3d3)



More dice added means it converges instead to the center of the bell curve.



Also, note that the mode here is 3



Better is to use a deck of cards as a randomization method

Level	Monster
1	Mouse
2	Hamster
3	Gerbil
4	Rat
5	Squirrel
6	Guinea Pig
7	Chinchilla
8	Marmot
9	Capybara
10	ROUS

Draw a card; what are the odds of pulling a specific card as the next card?

Level	Monster
1	Mouse
2	Hamster
3	Gerbil
4	Rat
5	Squirrel
6	Guinea Pig
7	Chinchilla
8	Marmot
9	Capybara
10	ROUS

Draw a card; what are the odds of pulling a specific card as the next card?

Level	Monster
1	Mouse
2	Hamster
3	Gerbil
4	Rat
5	Squirrel
6	Guinea Pig
7	Chinchilla
8	Marmot
9	Capybara
10	ROUS

=1/10

Level	Monster
1	Mouse
2	Hamster
3	Gerbil
4	Rat
5	Squirrel
6	Guinea Pig
7	Chinchilla
9	Capybara
10	ROUS

Draw a card; what are the odds of pulling a specific card as the next card?

Level	Monster
1	Mouse
2	Hamster
3	Gerbil
4	Rat
5	Squirrel
6	Guinea Pig
7	Chinchilla
8	Marmot
9	Capybara
10	ROUS

=1/10

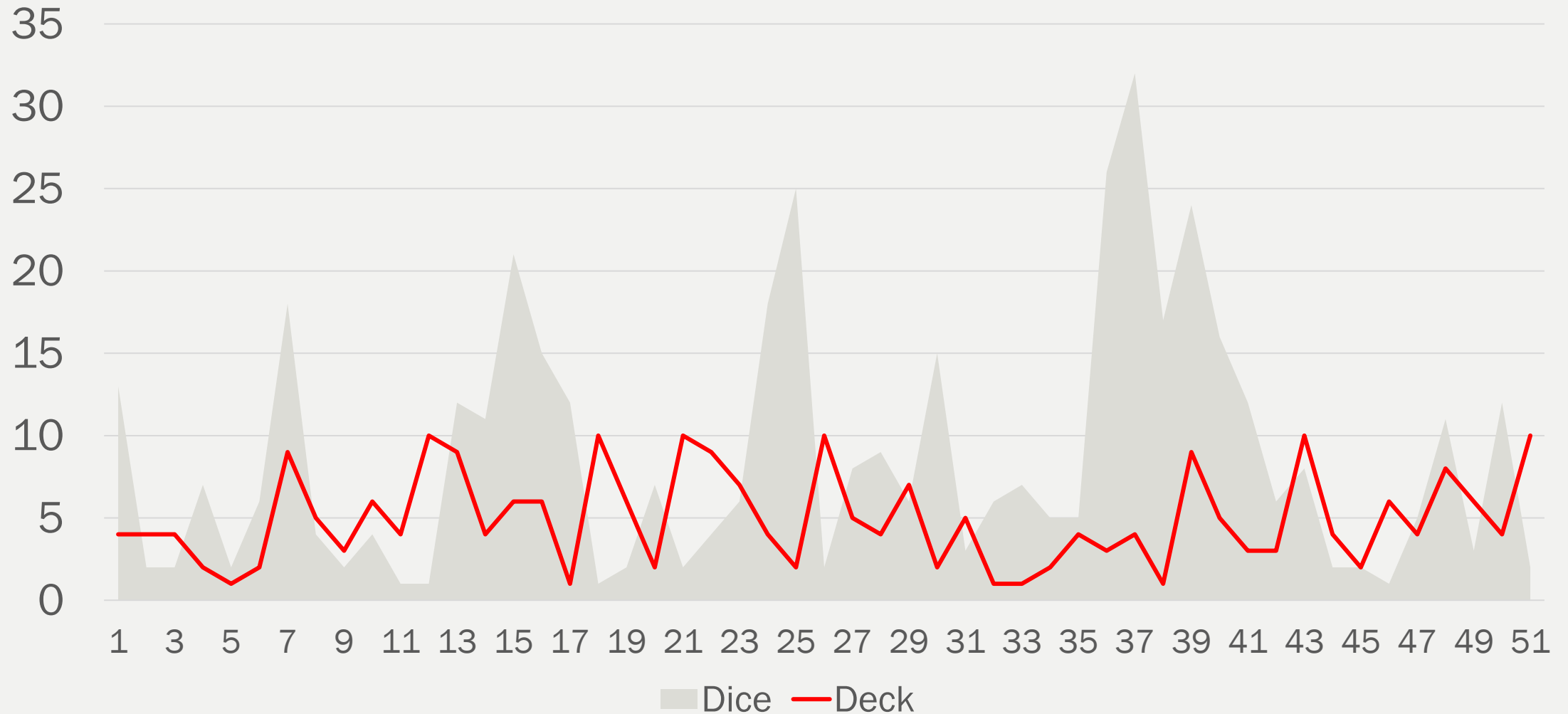
Level	Monster
1	Mouse
2	Hamster
3	Gerbil
4	Rat
5	Squirrel
6	Guinea Pig
7	Chinchilla
9	Capybara
10	ROUS

=1/9

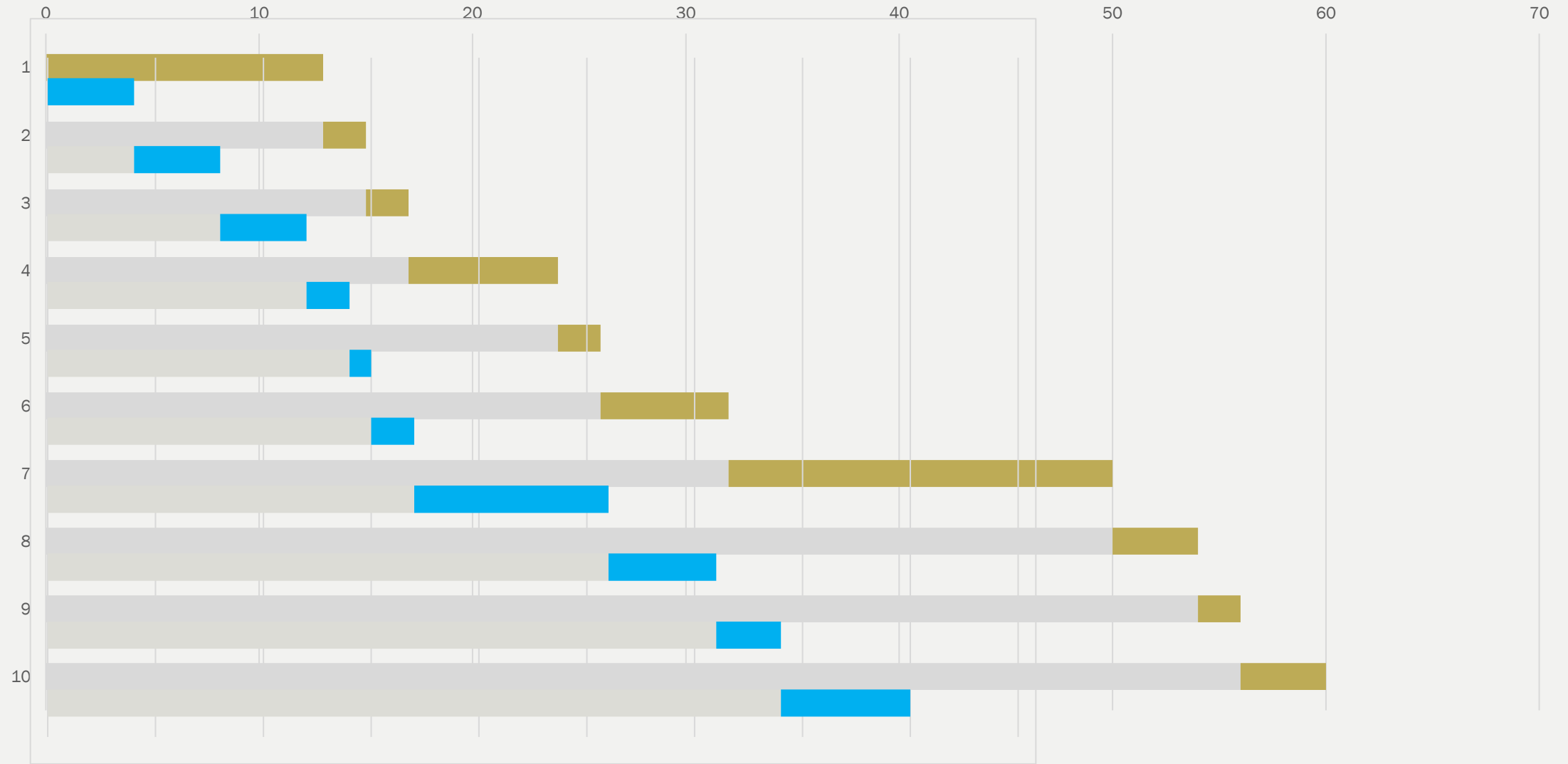
Odds are always $1/n$

Ten is now a limit

Dramatically more predictable!



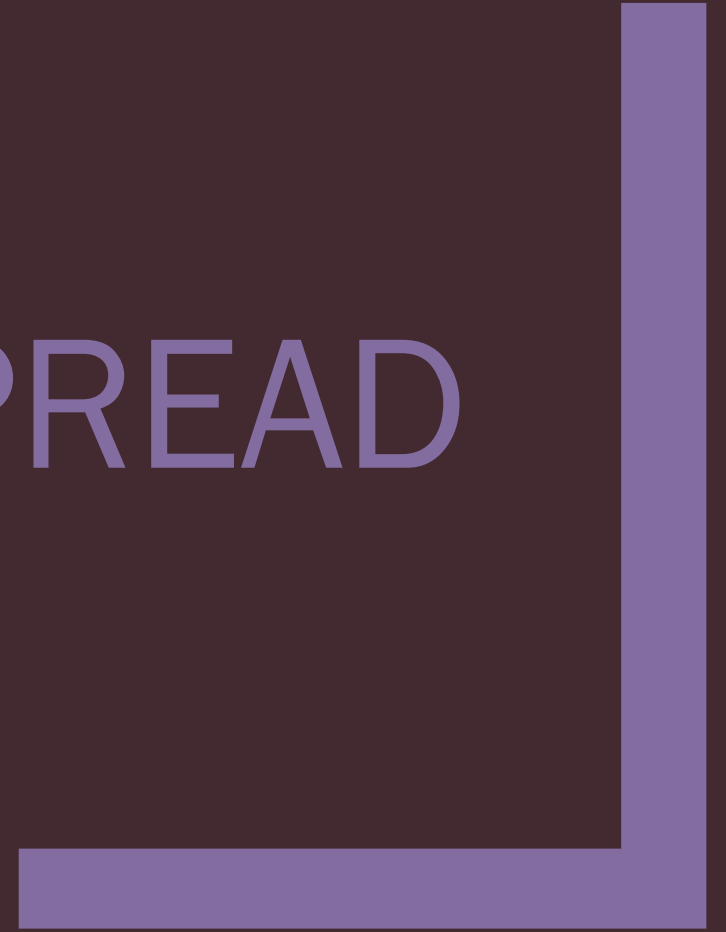
Player experience is much smoother



Encounter pacing

- What is the time delay between encounters?
- This is effectively another curve set by things like
 - *Travel time*
 - *Spawn rates*
 - *Spawn frequency*

REWARD SPREAD



But of course you can't only fight squirrels

- “Bottom feeding” is when you go fight things with high ROI – killing squirrels when you can handle a ROUS.
 - *Your odds of success are like 99.9%*
 - *This causes grinding, because players are smart.*

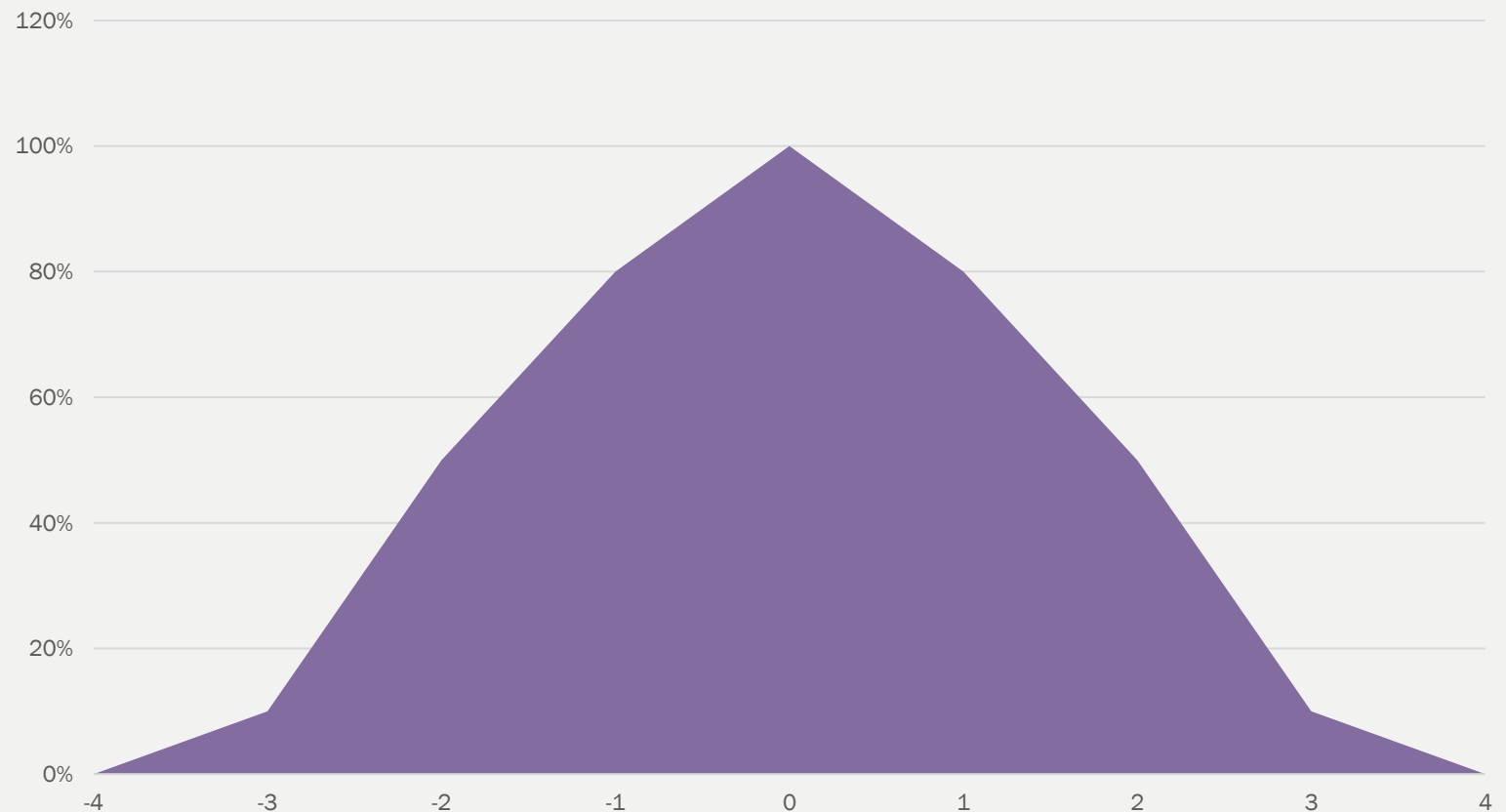
But of course you can't only fight squirrels

- One form of “twinking” is to get help in tackling stuff higher than you should be able to
 - *You get a reward calculated for a different point on the curve*
 - *The result is better than same level matches despite the lower success odds*
 - *You can do this solo too (over time, it's a self-reinforcing behavior unless there are notable loss penalties)*

Bell curve the reward

- Scaling factors should actually be based on % chance of success and may not match for higher and lower matches

Level spread	Scaling factor
-4	0%
-3	10%
-2	50%
-1	80%
0	100%
1	80%
2	50%
3	10%
4	0%



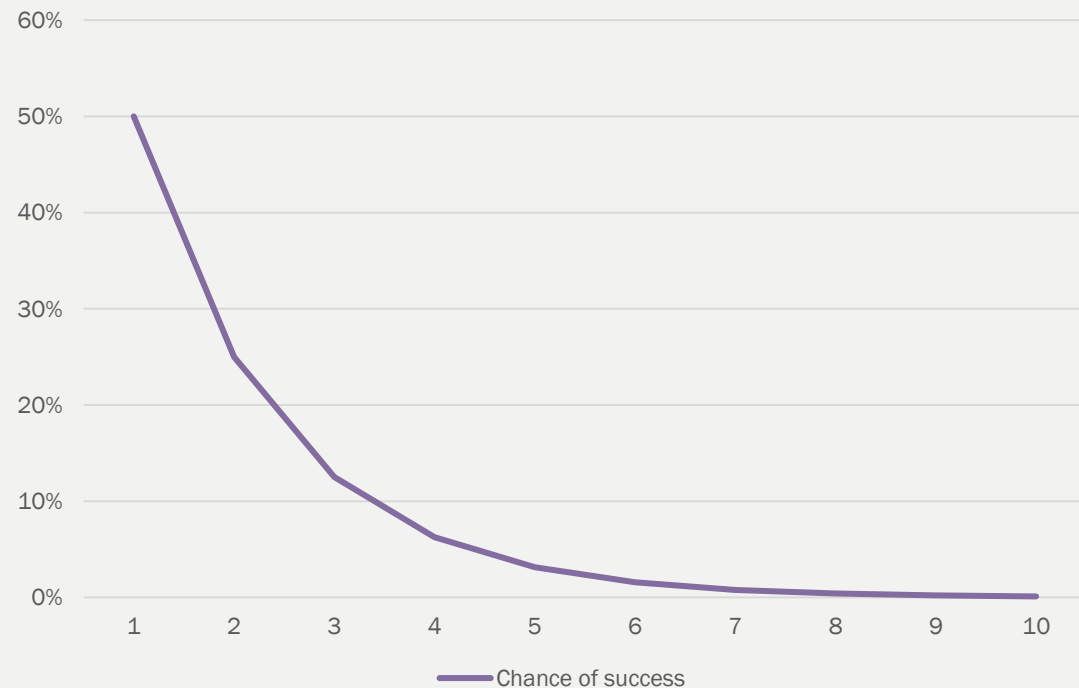
FUN FIGHTS



Even matches shouldn't be

- Push your luck is an important quality in RPG design

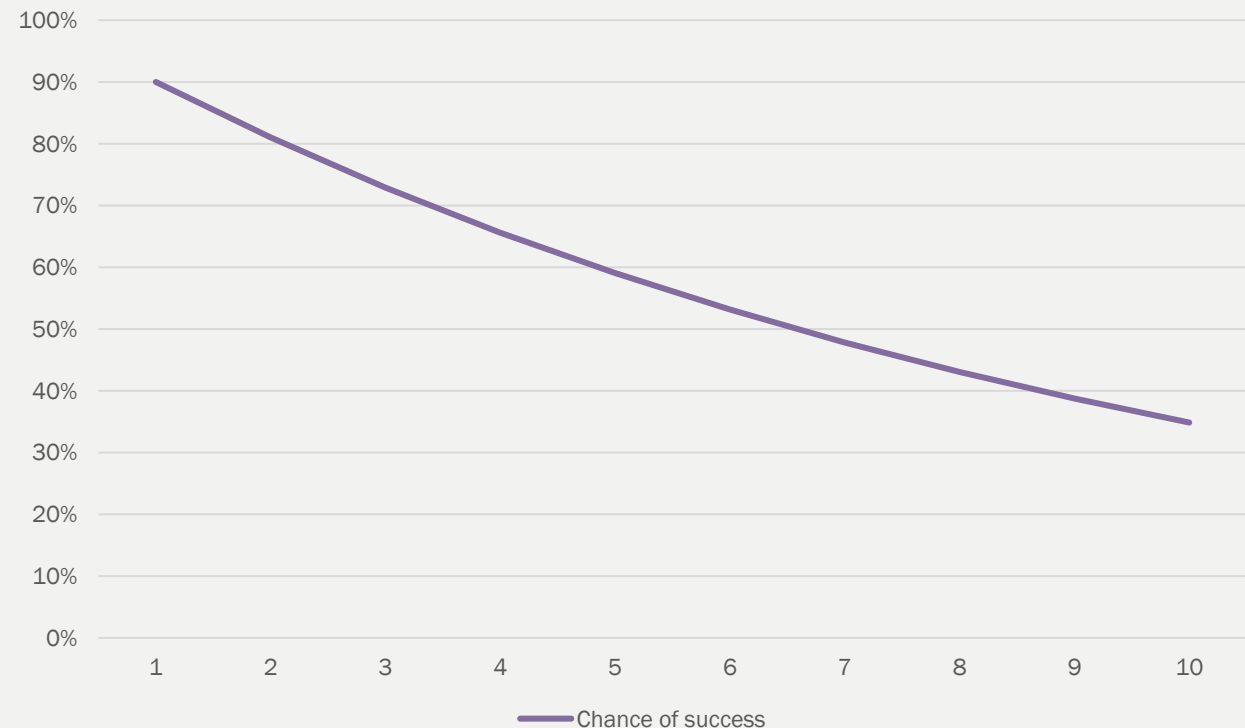
Opponent Number	Chance of success
1	50%
2	25%
3	13%
4	6%
5	3%
6	2%
7	1%
8	0%
9	0%
10	0%



90% is a better starting point for an even match

- You can scale this based on number of opponents per level, and decide how many “stopping points” there are in a level

Opponent Number	Chance of success
1	90%
2	81%
3	73%
4	66%
5	59%
6	53%
7	48%
8	43%
9	39%
10	35%



Single opponent

- Note, I can handle one more squirrel after this. But it'll be close.

	You	Squirrel One	
DPS	HP	HP	DPS
2	10	10	1
	9	8	
	8	6	
	7	4	
	6	2	
	5	Dead	

Multiple opponents

- Two 90% matches simultaneously is not an 80% match!

	You	Squirrel One	Squirrel Two	
DPS	HP	HP	HP	DPS
2	10	10	10	1
	8	8	10	
	6	6	10	
	4	4	10	
	2	2	10	
	Dead	Dead	10	

Make the enemy wimpier, but...

- Even halving their HP makes a very close thing. Add randomness and you die half the time.

	You	Squirrel One	Squirrel Two	
DPS	HP	HP	HP	DPS
2	10	5	5	1
	8	3	5	
	6	1	5	
	4	Dead	3	
	3		1	
	2		Dead	
	1			

Watch all the curves!

- Just like in the case of XP curves, overlapping curves have results you have to work through.
- An example of a subtle difficulty ramp:
 - *Level 1:*
I have 10hp, the monster has 5hp, my attack does 2hp. Time to kill is 3 hits.
 - Let's give me more HP and double my attack!
 - But let's make the monster WAY tougher.
 - *Level 2:*
I have 18hp, monster has 14hp, my attack does 4hp.
 - *Time to kill is now 4 hits, even though all the numbers went up.*

Monte Carlo modeling PvP

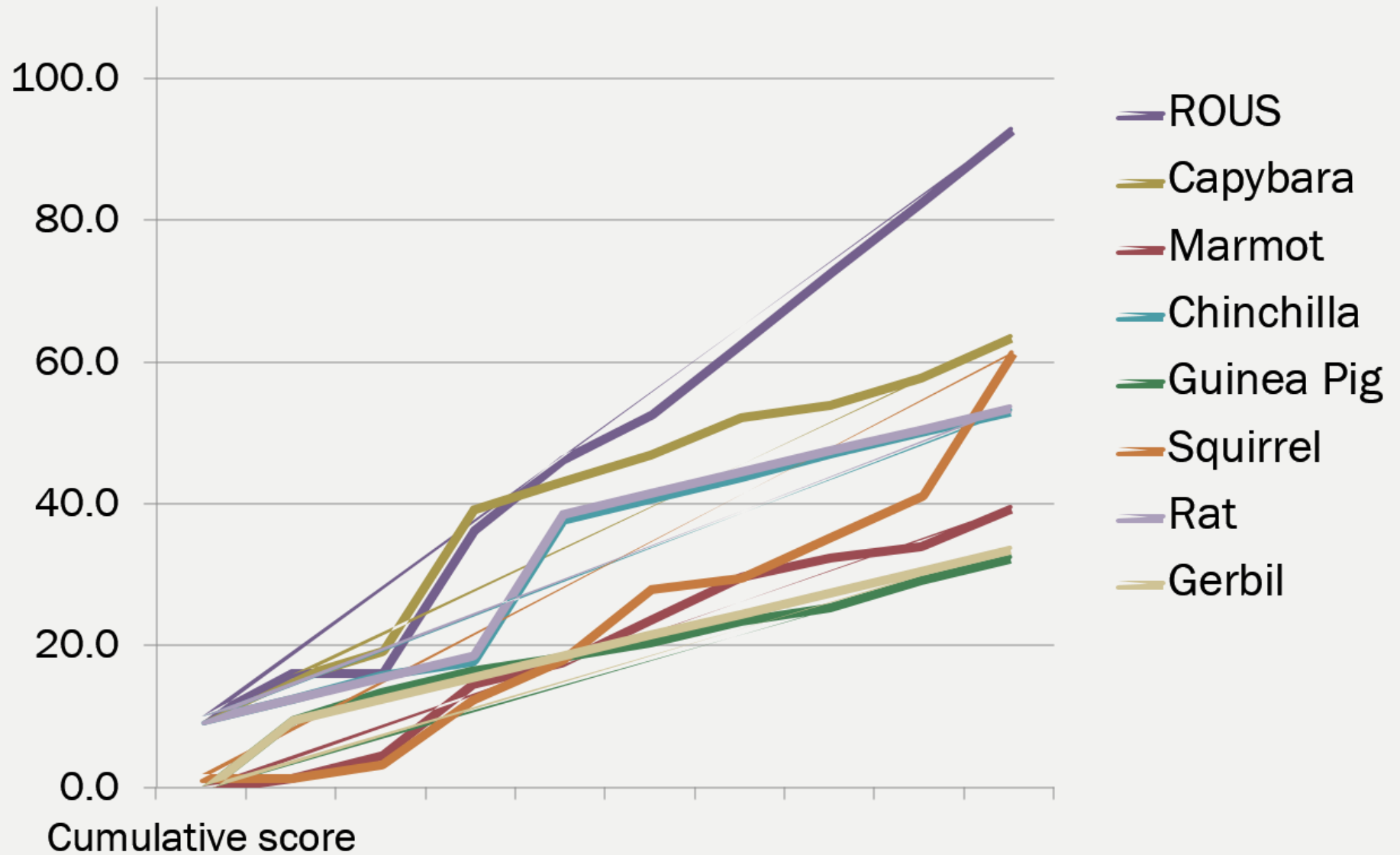
- The Monte Carlo method is basically running a pile of simulations
- Steps:
 - *Build a model of how your players act*
 - *Run through matches*
 - *Look at the graphs of progress through the match*

A model of rodent combat

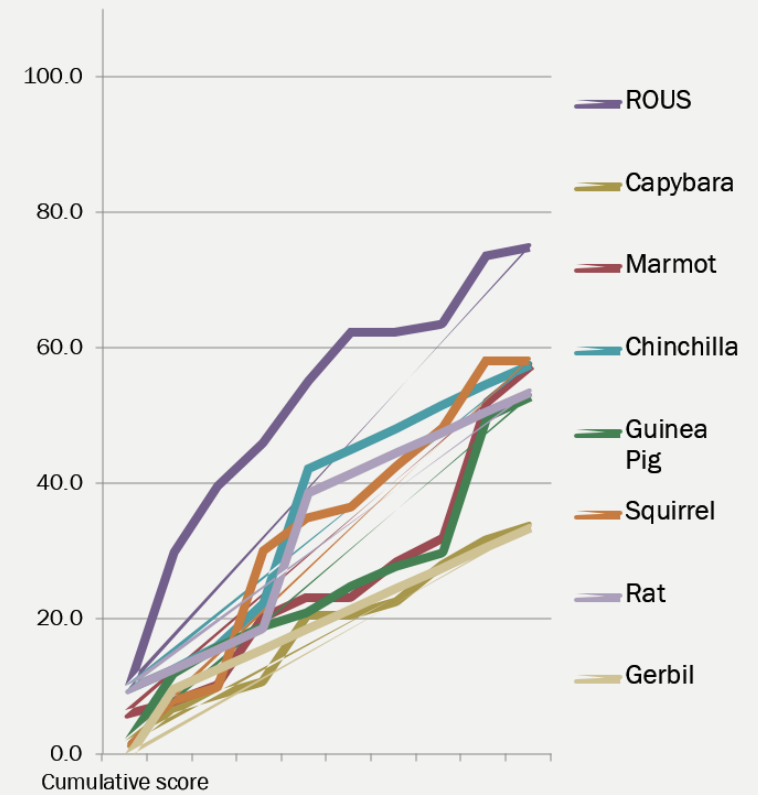
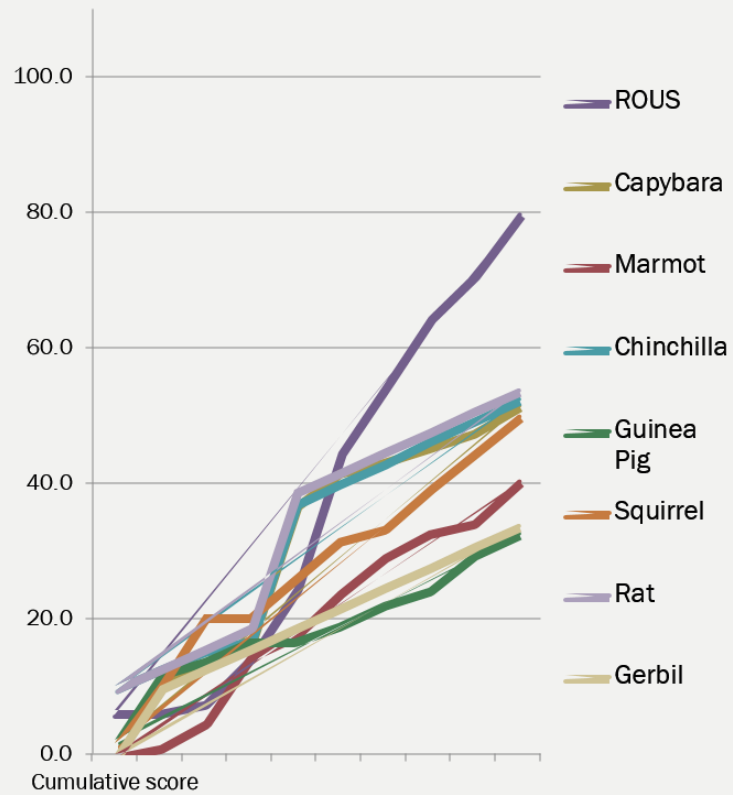
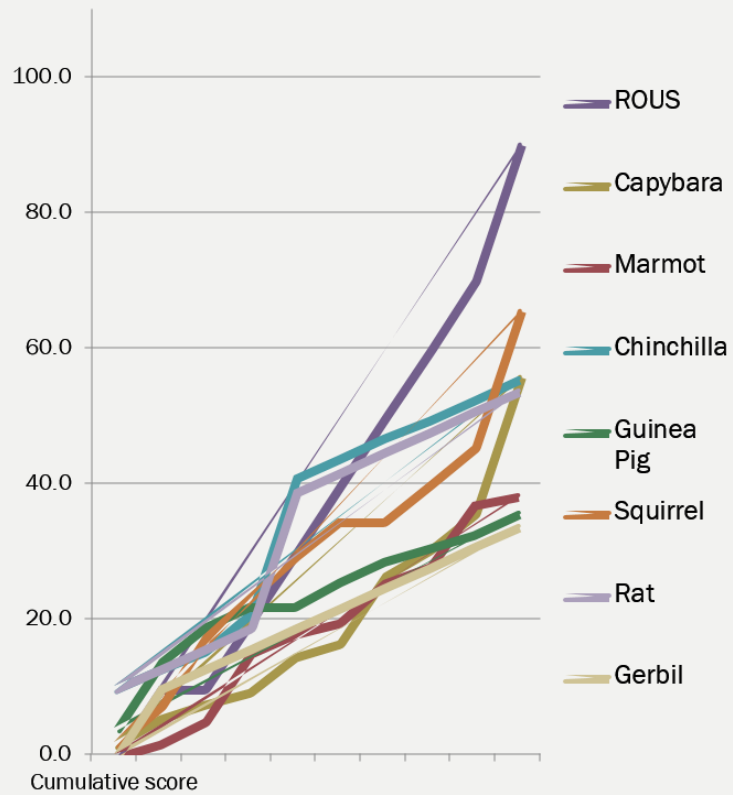
	Odds of move				
	A	B	C	D	E
ROUS	100%	100%	100%	100%	100%
Capybara	50%	50%	50%	50%	50%
Marmot	80%	60%	20%	0%	0%
Chinchilla	0%	0%	20%	60%	80%
Guinea Pig	40%	0%	40%	0%	40%
Squirrel	50%	75%	50%	75%	50%
Rat	0%	0%	0%	0%	100%
Gerbil	0%	0%	0%	0%	0%

Sample Game											
Round	1	2	3	4	5	6	7	8	9	10	
Monster type	A	B	C	D	E	A	B	C	A	B	
Opponent difficulty -- F9 to reroll. Used to account for the fact that experts still blow it all the time. Lower means tougher.											* difficulty is different for every user every encounter, to cover luck
ROUS	100%	62%	87%	88%	92%	64%	86%	65%	61%	87%	
Capybara	67%	61%	53%	89%	85%	93%	55%	67%	89%	100%	
Marmot	76%	69%	58%	85%	78%	52%	97%	76%	73%	83%	
Chinchilla	59%	69%	83%	92%	84%	56%	53%	90%	87%	66%	
Guinea Pig	99%	80%	60%	69%	72%	59%	89%	91%	64%	97%	
Squirrel	96%	74%	91%	98%	71%	55%	71%	69%	85%	75%	
Rat	52%	80%	59%	68%	81%	65%	91%	60%	87%	59%	
Gerbil	77%	88%	56%	96%	75%	78%	81%	64%	76%	64%	

Sample match results



Trial runs



Fun matches are tight

- Might want to turn down the ROUS in this instance
- You can use this same technique for modeling playstyles or other player behaviors, progression through levels, churn estimates, and much more

SOME CONCLUSIONS



Some benefits

- If you model out how long one fight takes, you can calculate pretty solid “time played between levels.”
 - *And also “ammo needed per kill”*
- In energy-based systems, you can supply exactly enough to leave the player just before a level up, or just past one – whatever tests better.
- If you set a monetary value for energy or for ammo consumed per kill, you now know the skipped playtime’s value.

Trading: a side note

- Enabling player to player trade of purchased items (even *indirectly* purchased ones) can be up to **30% hit on revenue** as markets seek a more optimal level.
 - *There's probably an equation in an econ textbook that gives the precise number.*
- Pokémon are effectively priced on the market by the cost of a Pokéball and the average number of throws it takes to catch a given Pokémon type.
 - *If it takes, on average, 20 throws to catch a dragonite, then a dragonite costs 99 cents (price of 20 Pokéballs), modded by a rarity value and labor value.*
- Frankly, I don't know how you balance a microtransaction economy without having these fundamentals worked out.

Most importantly...

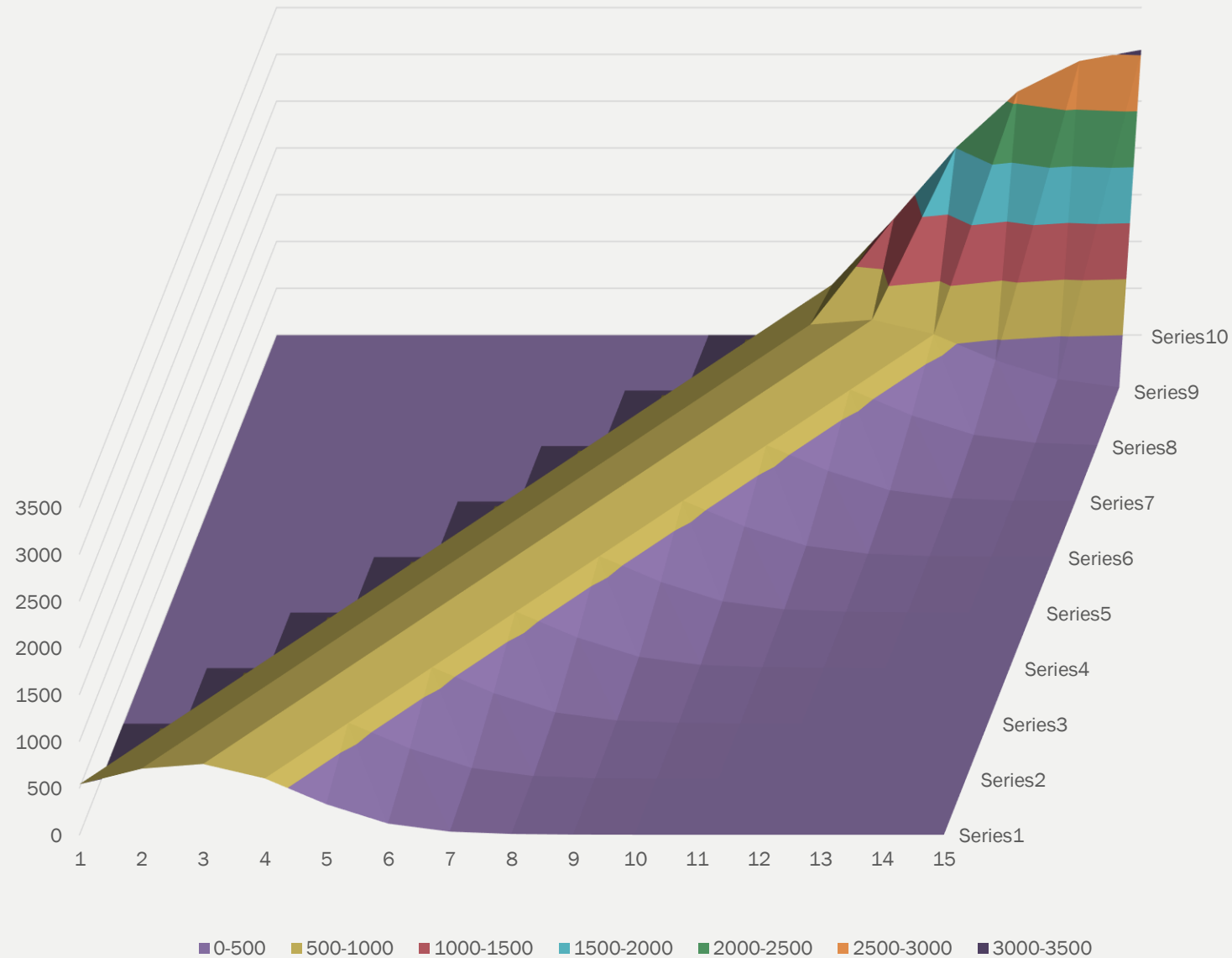
- We know that unpredictable advancement causes churn.
- We know that fairly frequent rewards increase player engagement.
- You can tune your game to be giving rewards at the pacing you desire
 - *Simply adjust the size of the deck*
 - *And the number of kills per level*
 - *And the cost for a minute of advancement*
 - *etc*

Quite a lot of fun is predictable

- And designable!

- *Pacing of play*
- *Grindiness*
- *Fun PvP*
- *Return hooks*
- *etc*

ARPD AU/LTV area under the curve



Carpentry

- The best reason to do this is craft.
 - *A self-respecting carpenter wouldn't build a table they knew was going to fall apart.*
- Take your craft seriously.
- And go add joy to the world.

<http://www.raphkoster.com>

Twitter: @raphkoster

LinkedIn: <http://www.linkedin.com/in/raphkoster>

Book: <http://www.theoryoffun.com>

