Retention Bot

Surfacing and Response Time Tuning

Overview

Objective:

- To revamp the static response time of mp_bots in order to give a real player experience to the player
- To have a optimal distribution of instant bots and mp_bots in the game to increase rolling retention

Business Goals:

- To increase LTV by 3.22%
- To increase Mp_bot starts per dau from 1.95 to 2.28 (31% dau are doing 6.3 starts against mp_bot -> expecting this to increase to 35% dau doing 6.5 starts as the completes and surfacing will increase)
- 100 bps rolling retention upside from 35% of dau (who are engaging with mp_bots) resulting in 35 bps upside in D1 RR

New Bot Response/ Categorization Logic (Var2 and Var3)

- There will be 2 categories of bots in the game in terms of response time:
  - Instant play bots (plays as soon as the player plays the turn)
  - Dynamic ranged bots (plays according to the logic defined here)

- Categorisation to each category happens on every new game start with the bot based on the following conditions
  - There will be at least 1 instant bot in the game
  - If instant bot is already present, next priority will be Dynamic ranged bot
  - 3rd bot onwards, bots are selected based on probabilities which will be a runtime (bot_category_prob_exp)

- Once a game gets over with a bot, the category resets and the behavior changes as per the new category

Dynamic Ranged Bot Response Logic (Var2 and Var3)

- Dynamic ranged bot will respond based on the number of open games player has in that moment, number of moves player has played in that session and “threshold” value of the player.

- Value of “threshold” for each player will be calculated using the below formula:
  - Threshold Value = (1 + 0.2) * (avg moves played in the last 7 sessions)
  - This value will rounded off to the ceil value
  - If the threshold value is less than 10 for any player then it will be set to 10 for that session and if the threshold value is more than 40 for any player then it will be set to 40 for that session for the player.

- Here, 0.2 is multiplier, 10 is lower limit and 40 is upper limit of threshold value which will be runtime controlled (bot_ret_threshold)

- **Session Definition:** If the game is paused and player resumes/ launch the game after 300s then the session is considered as a new session

- **Note:** For player who are from the previous build or are new installs and don’t have a value of avg moves played in the last 7 sessions, the value of threshold will be set to 40 in this case

Dynamic Ranged Bot Response Logic (Var2 and Var3)

- If player moves played in that session is less than or equal to “threshold” value then :
  - Response Time of bot = (0.2 * No. of Open games + 0.8 * Moves played in the session) * 10 (in secs)

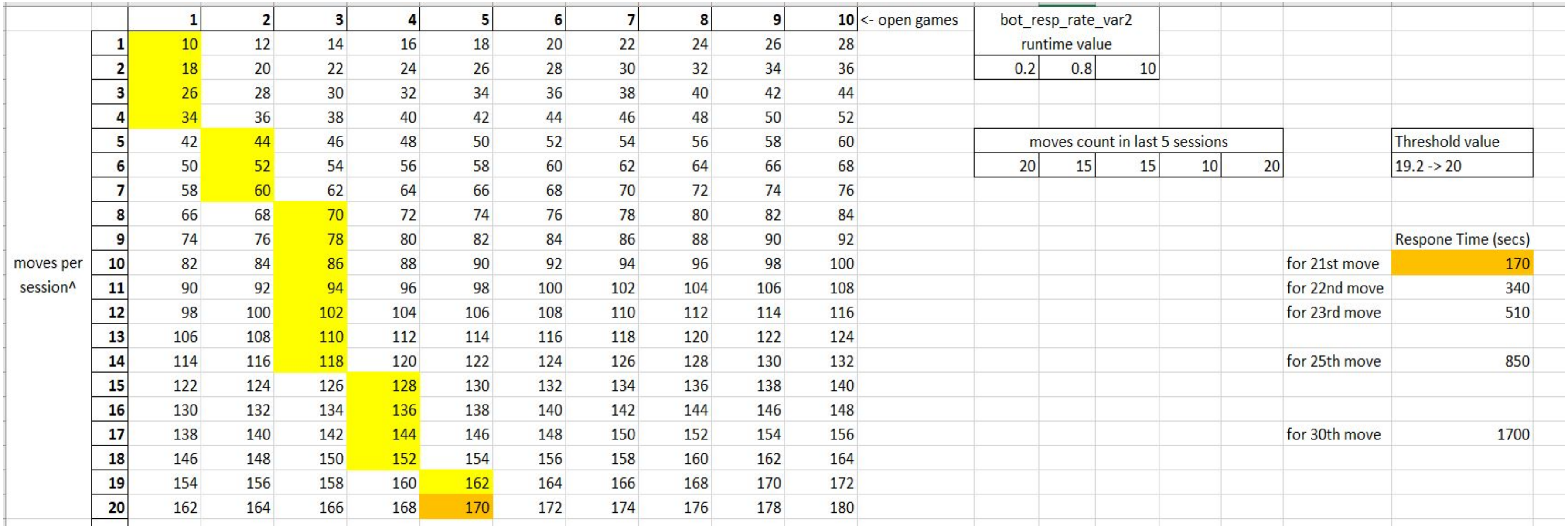
- If player moves played in that session is more than the “threshold” value then:
  - Response Time of bot = (Moves played in the session - Threshold value) * (Threshold value) * 10 (in secs)

- The values 0.2, 0.8 and 10 will be runtime controlled (bot_resp_rate)

- **Open Game Definition:** It will be the number of games in your move section (excluding: MOTD and solo_series)

- **Note:** Response time of bot will be capped to 86400 secs. (Any response time more than 86400 will be changed to the capped value)

Example



		1	2	3	4	5	6	7	8	9	10	<- open games
	1	10	12	14	16	18	20	22	24	26	28	
	2	18	20	22	24	26	28	30	32	34	36	
	3	26	28	30	32	34	36	38	40	42	44	
	4	34	36	38	40	42	44	46	48	50	52	
	5	42	44	46	48	50	52	54	56	58	60	
	6	50	52	54	56	58	60	62	64	66	68	
	7	58	60	62	64	66	68	70	72	74	76	
	8	66	68	70	72	74	76	78	80	82	84	
	9	74	76	78	80	82	84	86	88	90	92	
moves per session^	10	82	84	86	88	90	92	94	96	98	100	
	11	90	92	94	96	98	100	102	104	106	108	
	12	98	100	102	104	106	108	110	112	114	116	
	13	106	108	110	112	114	116	118	120	122	124	
	14	114	116	118	120	122	124	126	128	130	132	
	15	122	124	126	128	130	132	134	136	138	140	
	16	130	132	134	136	138	140	142	144	146	148	
	17	138	140	142	144	146	148	150	152	154	156	
	18	146	148	150	152	154	156	158	160	162	164	
	19	154	156	158	160	162	164	166	168	170	172	
	20	162	164	166	168	170	172	174	176	178	180	

bot_resp_rate_var2 runtime value		
0.2	0.8	10

moves count in last 5 sessions				
20	15	15	10	20

Threshold value
19.2 -> 20

	Respone Time (secs)
for 21st move	170
for 22nd move	340
for 23rd move	510
for 25th move	850
for 30th move	1700

Dynamic Ranged Bot Response Logic (Var2 and Var3)

- If player has 0 open games and there are more than 'x' (runtime bot_inst_resp) dynamic ranged bot in the their moves section who are timed to response after t1, t2, … , tx secs then out of those, the bot from the most progressed game (in terms of round) will respond after 3s (runtime bot_inst_resp).

- This will happen only if players move count is less than threshold value.

- This will be checked
  - when player views the main screen
  - when more games panel on gameboard pop ups.
  - when player views the game over dialog

- **Note:** If there are multiple games which are on round 3 (say), and this is the most progressed round among all the dynamic ranged bot games in their move section then, we can randomly choose one game out the games which are on round 3 to respond quickly.

Quick Game Button Logic Change (Var3 Only)

- **UI Change:**
  - The text of quick game button on main screen will change from “Quick Game” to “Random Opponent”
  - On create screen and friends screen also we will be changing text to “Random Opponent” from “Start Quick Game”

- **Logic Change:**
  - Clicking on quick game button on main/create or friends screen will start a bot game
  - Category of the bot will be decided by the logic mentioned here.

New Game Button Logic Change (Var3 Only)

- **Logic Change:**
  - Clicking on new game button on gameover will start a bot game for “x” number of times.
  - Category of the bot will be decided by the logic mentioned here.
  - The value “x” will be controlled by an existing runtime ibsc.
  - For this we will be changing the the logic of this runtime
    - Old Logic: “Limit of *instant bot games* that can be started from gameover screen before starting best match games”
    - New Logic: “Limit of *bot games* that can be started from gameover screen before starting best match games”

Bot Image Changes (Build Change)

- We will be updating the image of the existing bots
- The images of the bot will be updated for all of the users (Build Change)
- ~~For existing open games image can remain the same, images will be updated on new game start on the new build~~
- Dimension Suggested: 128*128
- Link for new bot images:
- Drive Link -> https://drive.google.com/drive/folders/1aFeUoV9mwXyA4rLGwZL7-nbH3TRuye9w

Dev Requirements

- We will be removing ibs_opt experiment and ibs_opt runtime from the code side (it is ramped down to 0% from product side).

- We will be adding these changes into var2 and var3 of an existing experiment bot_response

- **Experiment Name:** bot_response
  - Control: No Changes
  - Var1: No Changes
  - Var2: Categorization and Response logic change
  - Var3: Categorization, Response logic and Surfacing change

- **Runtimes required:**
  - bot_resp_rate: Will contain 2 fields
    - Instant: default value would be 0#0 (response time of bot in sec)
    - Dynamic: default value would be 0.2#0.8#10 (multiplier of open games # multiplier of moves played in that session # time in secs used in the formula mentioned in above slide)

Dev Requirements

- **Runtimes Required (Continued) :**
    - bot_category_prob_exp: It will have 2 field which will control the distribution of bots
        - in var2
            - Instant: Will control the probability on instant bot (default value: 70)
            - Dynamic: Will control the probability on dynamic bot (default value: 30)
        - in var3
            - Instant: Will control the probability on instant bot (default value: 70)
            - Dynamic: Will control the probability on dynamic bot (default value: 30)

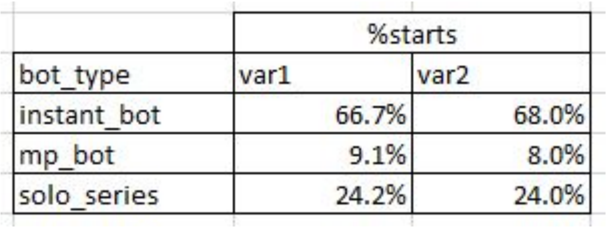
    - bot_ret_threshold: Will contain 3 values separated by ‘#’
        - Default value will be 0.2 # 10 # 40 (multiplier # lower limit # upper limit)

    - bot_inst_resp: Will contain 2 values separated by ‘#’
        - Default value will be 3 # 3
        - First value is number of open games player has in their move section
        - Second value is time after which bot should reply instantly

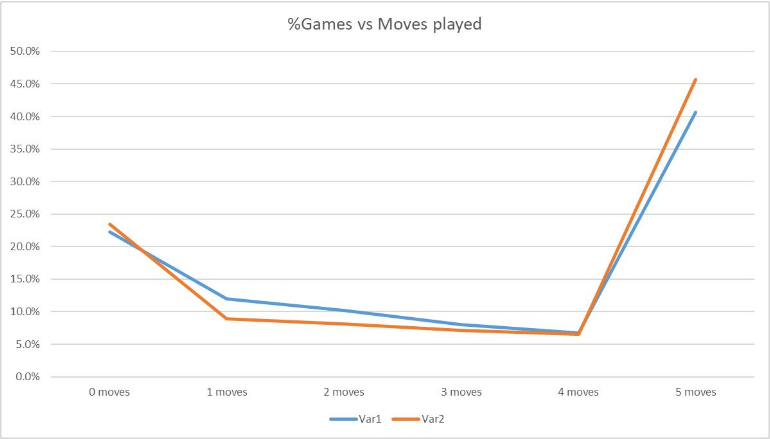
Next Steps

Overview:

- Due to a faster response time logic, we saw ~500 bps increase in completion rate of games against mp_bots in Var2 of bot response experiment which had dynamic bots only
- Also, having a gradual increase in response time vs static response time per move lead to an increase in more number of moves played per game against mp_bots
- However, we saw ~11% decrease in mp_bot game starts
    - This was happening due to increase in completion rate which lead to an increase in views of game outro
    - From game outro players were starting more instant bot games as ‘new game’ button on outro screen starts instant bot games mostly.



	%starts	
bot_type	var1	var2
instant_bot	66.7%	68.0%
mp_bot	9.1%	8.0%
solo_series	24.2%	24.0%



Overview:

- In Var2 there is very small upside of 36 bps and 9 bps in D1 retention & D1 rolling retention respectively
- Possible reason for low retention upside:
  - The changes we did to response time is affecting only 5% of the games played by the players
  - Currently 90% of the bot games started are instant bots, rest all games are started against mp_bots.
  - 3 major sources of bot start games are mainly starting instant bot games
- We will be changing logic of these start sources
  - in order to have a unified bot game start logic across all major sources
  - and to increase mp_bots engagement which will lead to retention upside

Start source	**Surfacing**	**Bot Type (If bot game is started)**	**% Bot Game Starts**
Quick Game Button	Main/ Create/ Friends Screen or More Games Panel	Instant bot	~10%
New Game Button	Game Over Dialog	Instant bot (Runtime controlled)	~30%
Auto Game Starts	Main Screen	Instant bot	~20%

New Game Button Logic Change (Var2 Only)

- **Logic Change:**
  - Clicking on new game button on gameover will start a bot game for “x” number of times.
  - Category of the bot will be instant or dynamic which will be controlled by this runtime bot_category_prob_exp.
  - The value “x” will be controlled by an existing runtime ibsc.
  - For this we will be changing the the logic of this runtime
    - Old Logic: “Limit of *instant bot games* that can be started from gameover screen before starting best match games”
    - New Logic: “Limit of *bot games* that can be started from gameover screen before starting best match games”

Auto Game Start Logic Change (Var2 Only)

- **Logic Change:**
    - Current auto game start logic is mentioned here.
    - Instead of starting a game against instant bot in auto game start logic, a bot game is to be started
    - Category of the bot will be instant or dynamic which will be controlled by this runtime bot_category_prob_exp.

Quick Game Button Logic/UI Change (Var2 Only)

- **UI Change:**
  - The text of quick game button on main screen will change from “Quick Game” to “Random Opponent”
  - On create screen and friends screen also we will be changing text to “Random Opponent” from “Start Quick Game”

- **Logic Change:**
  - Clicking on quick game button on main/create or friends screen will start a bot game
  - Category of the bot will be decided by the logic mentioned here.

- This is already implemented in Var3, we can replicate the changes from there

Dev Requirements

- Control & var3 is to be removed from the code side
- Var3 field is to be removed from bot_category_prob_exp runtime
- All the changes mentioned in the next steps section will be implemented in Var2 of bot response experiment.

RESEARCH

Sources of bot game starts

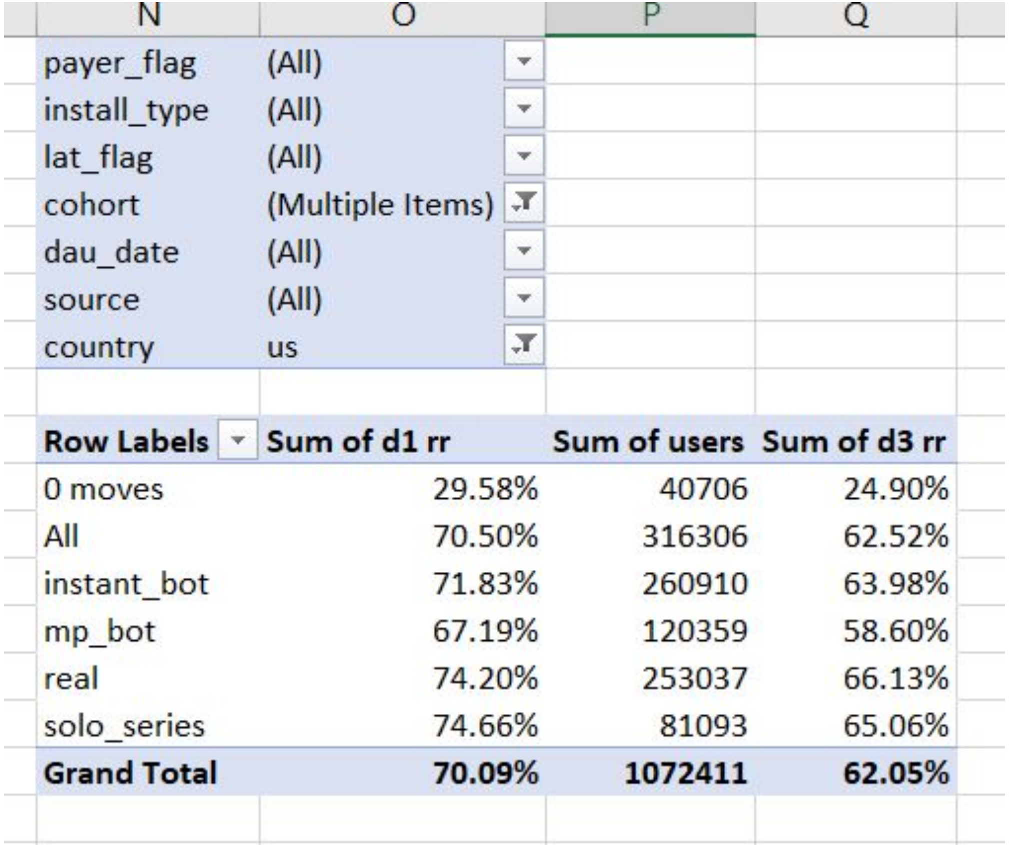
Start source	Surfacing	Bot Type (If bot game is started)	%Bot Game Starts
Quick Game Button	Main/ Create/ Friends Screen or More Games Panel	Instant bot	10.1%
New Game Button	Game Over Dialog	Instant bot (Runtime controlled)	29.0%
Best Match	Main/ Create/ Friends Screen	Instant/ Short/ Long bot (Bot logic)	9.6%
Auto Game Starts	Main Screen	Instant	19.7%
Bot Start Notifs	Notifs	Instant/ Short/ Long bot (Bot logic)	6.4%
Rematch	Game Over Dialog	Instant/ Short/ Long bot (Bot logic)	8.6%
MOTD	Main Screen/ More Games Panel	Instant/ Short/ Long bot (Bot logic)	2.1%
Play CTA	Leaderboard Screen	Instant bot	0.7%
Play CTA	Daily_Quest Carousel	Instant bot	0.2%
Solo Series	Main Screen	Instant bot	13.3%

Current Bot Response/ Categorization Logic

- There are 3 categories of bots in the game in terms of response time:
  - Instant play bots (plays as soon as the player plays the turn)
  - Short ranged bots (plays between 2 minutes and 5 minutes after player's move)
  - Long ranged bots (plays between 120 minutes and 1440 minutes after the player's move)
- Categorisation to each category happens on every new game start with the bot based on the following conditions
  - There will be at least 1 instant bot in the game
  - If instant bot is already present, next priority will be short ranged bot
  - If both are present, next bot will be long ranged bot
  - 4th bot onwards, bots are selected based on probabilities which is a runtime
- Once a game gets over with a bot, the category resets and the behavior changes as per the new category

- **Experiment:**
  - Name: bot_response
  - Control: bot responding with runtime values contro, var1: bots responding with runtime values var1
- **Runtimes required:**
  - bot_resp_rate_control: instant:short:long (0#1:2#5:2#5)
  - bot_resp_rate_var1:instant:short:long (0#1:2#5:120#1440)
  - Bot_category_prob: instant#short#long:instant#short#long (control:var1)

D1 RR vs opp_type



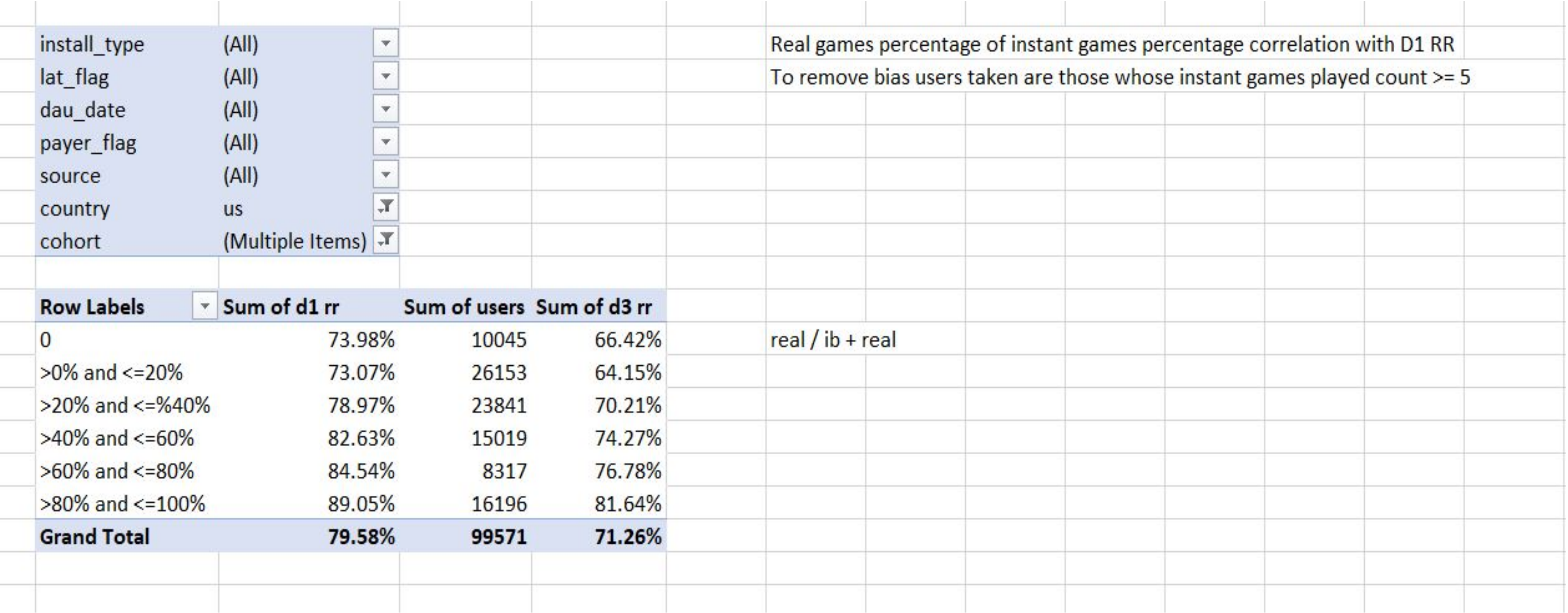
payer_flag	(All)
install_type	(All)
lat_flag	(All)
cohort	(Multiple Items)
dau_date	(All)
source	(All)
country	us

Row Labels	Sum of d1 rr	Sum of users	Sum of d3 rr
0 moves	29.58%	40706	24.90%
All	70.50%	316306	62.52%
instant_bot	71.83%	260910	63.98%
mp_bot	67.19%	120359	58.60%
real	74.20%	253037	66.13%
solo_series	74.66%	81093	65.06%
Grand Total	**70.09%**	**1072411**	**62.05%**

- We saw player playing at least one move against mp bots (short or long) have lower D1 RR
- Where as, in case of real player it is more
- Our hypothesis is that static response time for every turn is not making the mp_bot experience same as real

%Real games vs D1 RR

- From this data we can say player playing more % of games against real players have better D1 RR.



install_type	(All)
lat_flag	(All)
dau_date	(All)
payer_flag	(All)
source	(All)
country	us
cohort	(Multiple Items)

Row Labels	Sum of d1 rr	Sum of users	Sum of d3 rr
0	73.98%	10045	66.42%
>0% and <=20%	73.07%	26153	64.15%
>20% and <=%40%	78.97%	23841	70.21%
>40% and <=60%	82.63%	15019	74.27%
>60% and <=80%	84.54%	8317	76.78%
>80% and <=100%	89.05%	16196	81.64%
Grand Total	**79.58%**	**99571**	**71.26%**

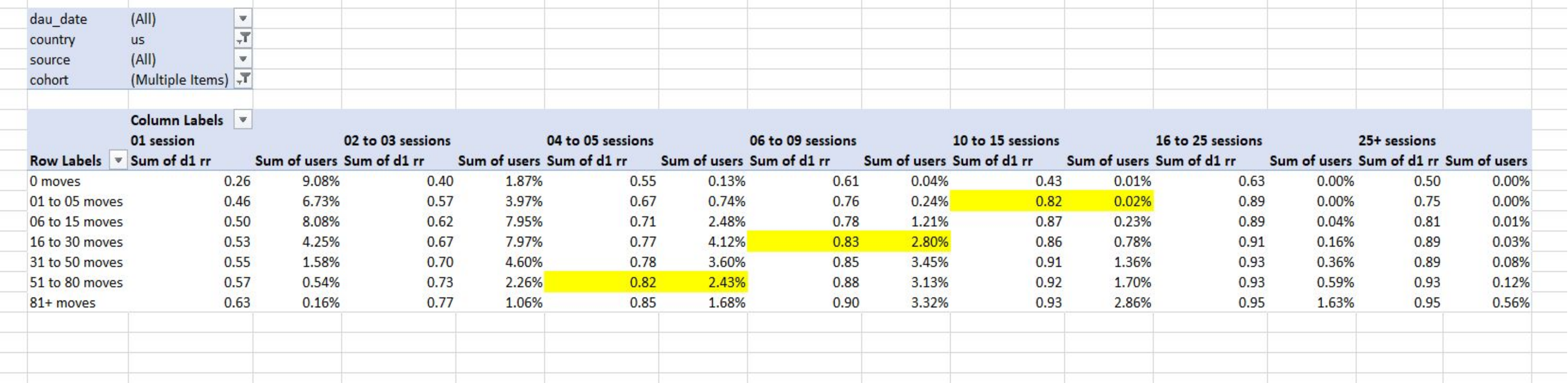
Real games percentage of instant games percentage correlation with D1 RR

To remove bias users taken are those whose instant games played count >= 5

real / ib + real

Sessions vs Moves vs D1 RR

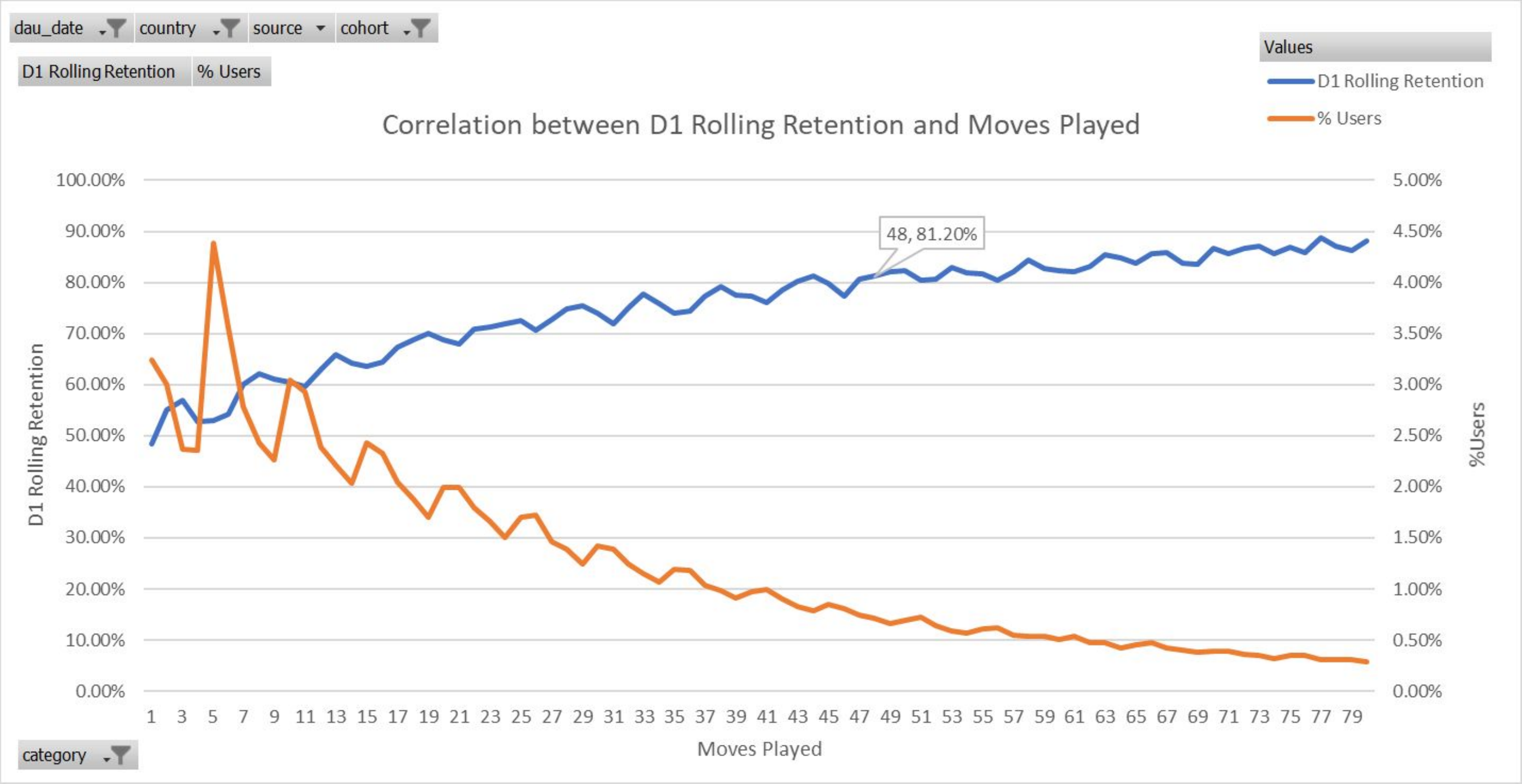
- Most of the players in our game have low sessions
- Even if player is engaging more his D1 RR is low if his session count is less than 3
- We should be pushing players for more sessions (4-5 each day)



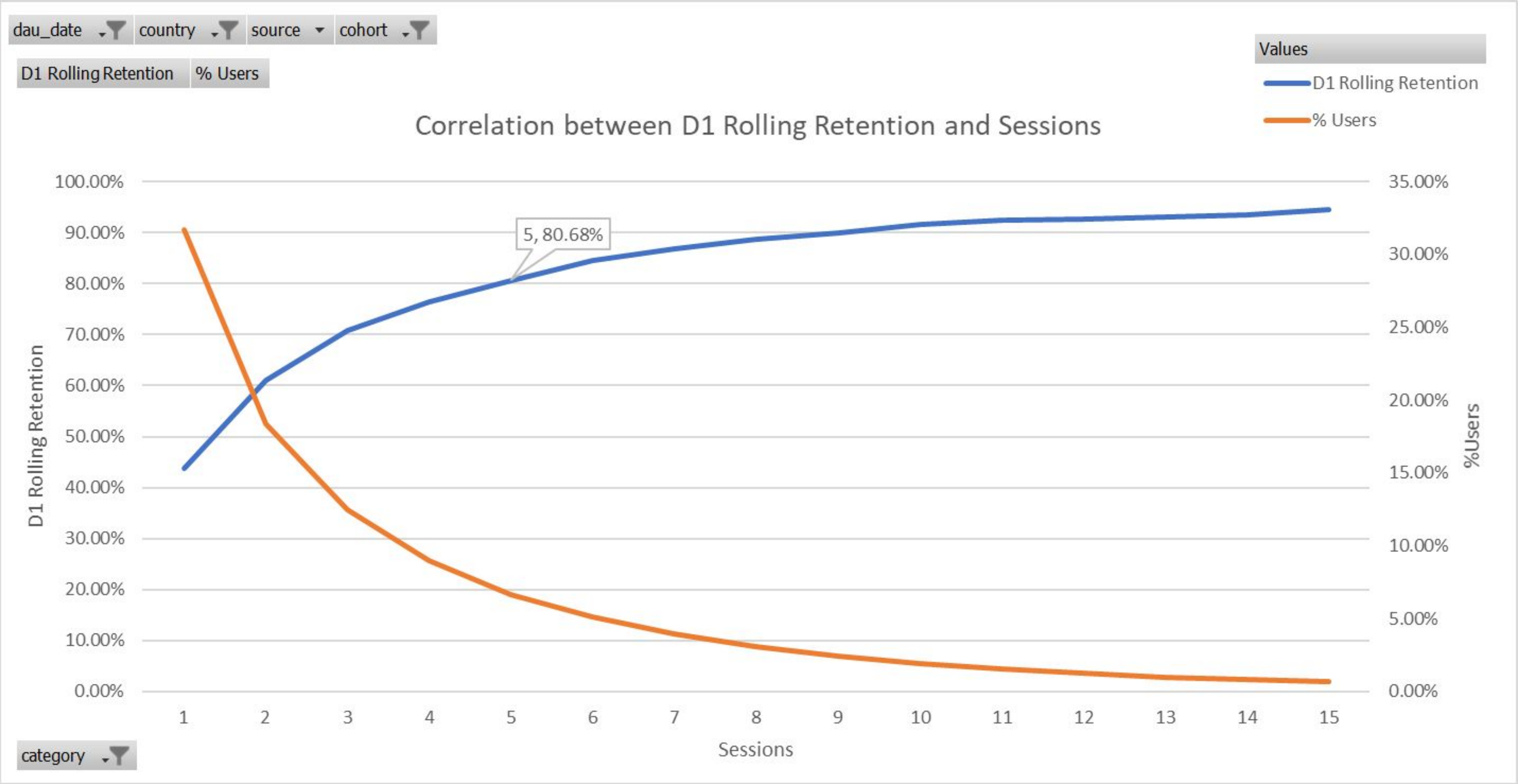
dau_date	(All)
country	us
source	(All)
cohort	(Multiple Items)

	Column Labels													
	01 session		02 to 03 sessions		04 to 05 sessions		06 to 09 sessions		10 to 15 sessions		16 to 25 sessions		25+ sessions	
Row Labels	Sum of d1 rr	Sum of users	Sum of d1 rr	Sum of users	Sum of d1 rr	Sum of users	Sum of d1 rr	Sum of users	Sum of d1 rr	Sum of users	Sum of d1 rr	Sum of users	Sum of d1 rr	Sum of users
0 moves	0.26	9.08%	0.40	1.87%	0.55	0.13%	0.61	0.04%	0.43	0.01%	0.63	0.00%	0.50	0.00%
01 to 05 moves	0.46	6.73%	0.57	3.97%	0.67	0.74%	0.76	0.24%	0.82	0.02%	0.89	0.00%	0.75	0.00%
06 to 15 moves	0.50	8.08%	0.62	7.95%	0.71	2.48%	0.78	1.21%	0.87	0.23%	0.89	0.04%	0.81	0.01%
16 to 30 moves	0.53	4.25%	0.67	7.97%	0.77	4.12%	0.83	2.80%	0.86	0.78%	0.91	0.16%	0.89	0.03%
31 to 50 moves	0.55	1.58%	0.70	4.60%	0.78	3.60%	0.85	3.45%	0.91	1.36%	0.93	0.36%	0.89	0.08%
51 to 80 moves	0.57	0.54%	0.73	2.26%	0.82	2.43%	0.88	3.13%	0.92	1.70%	0.93	0.59%	0.93	0.12%
81+ moves	0.63	0.16%	0.77	1.06%	0.85	1.68%	0.90	3.32%	0.93	2.86%	0.95	1.63%	0.95	0.56%

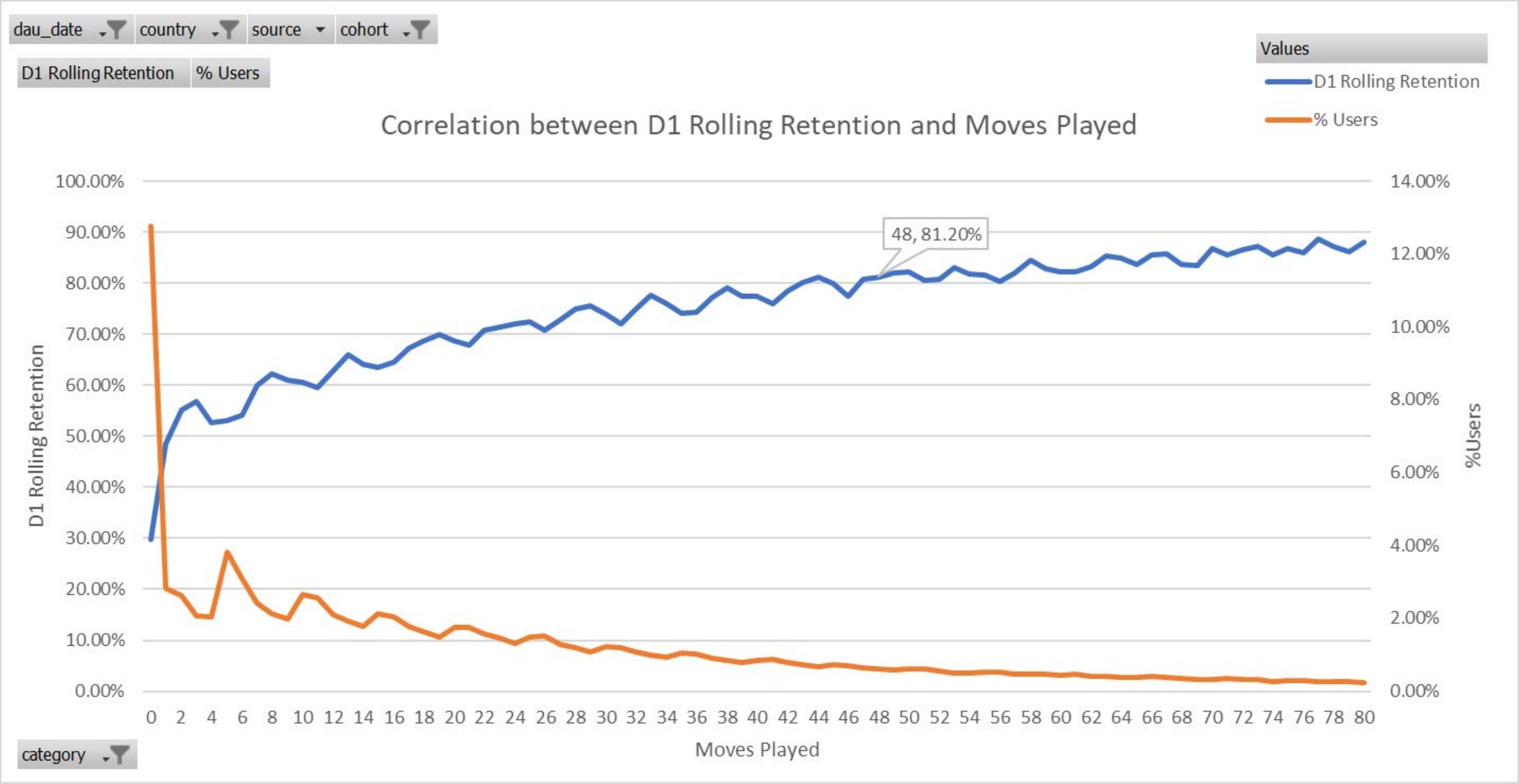
Correlation between D1 Rolling Retention and Moves Played



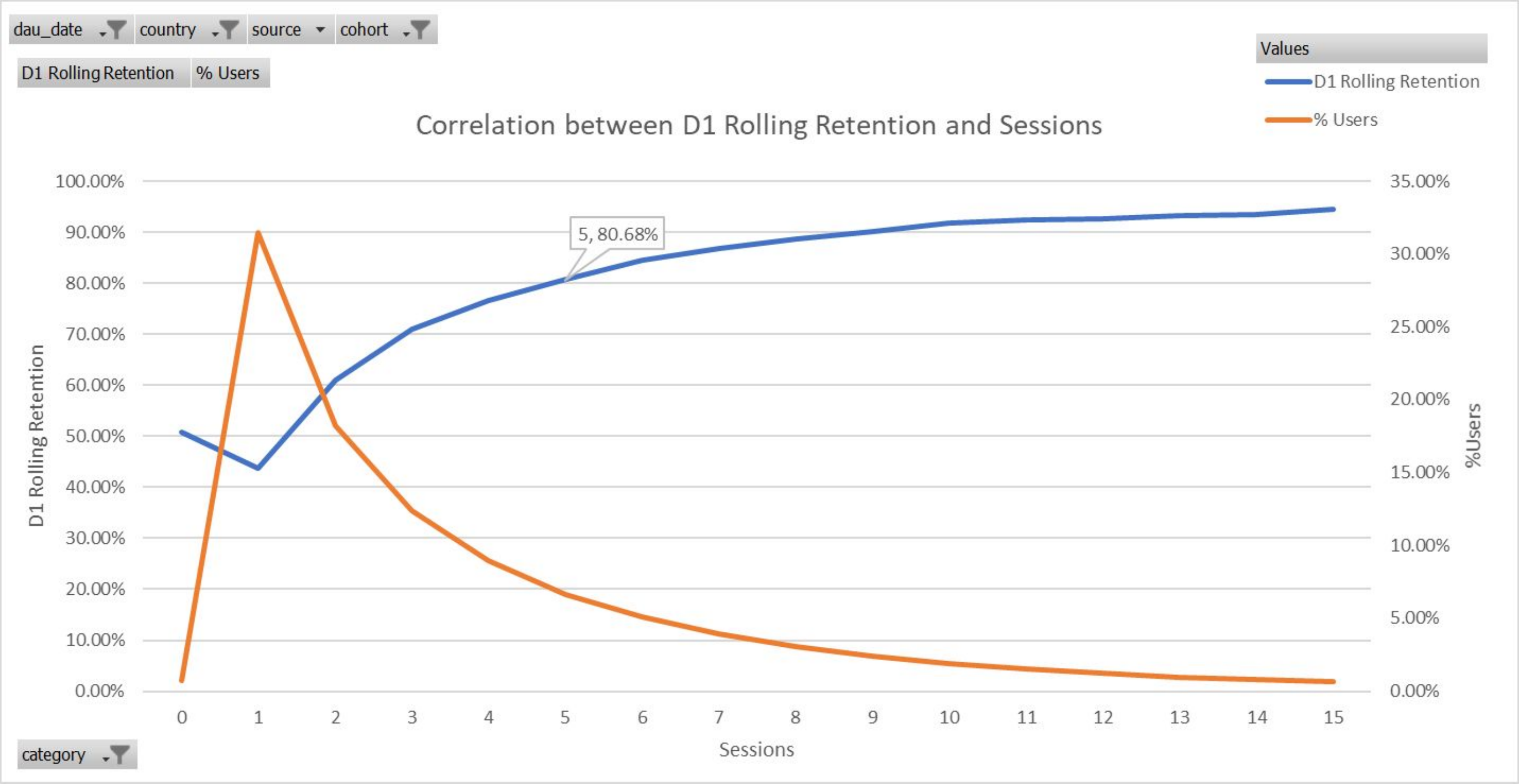
Correlation between D1 Rolling Retention and Moves Played



Continued (Contains D1 RR of 0 moves played)



Continued (Contains D1 RR of 0 sessions)


dau_date
country
source
cohort
D1 Rolling Retention
% Users
Values
D1 Rolling Retention
% Users
Correlation between D1 Rolling Retention and Sessions
D1 Rolling Retention
%Users
Sessions
category