

Project Report Game Analysis

Author : Namita Saxena

Date: September 1, 2014

This is the game user behavior analysis. First I load the data in R and did analysis in three steps :

*Data Quality - Check for redundancy, NA's, duplicates, any other issues related to raw data and tried to remove them.

*Data Manipulation - Subset and sliced the data in different dimensions for analysis.

*Data Analysis - Finding patterns in data and answered the given problems.

The data in given csv file contains 14143 rows and 16 columns. The data is collected from 13 users and has following features/columns :

```
## [1] "clientTimeStamp" "serverTimeStamp" "clientVersion"
## [4] "totalTimePlayed" "userId" "deviceId"
## [7] "activityId" "sessionId" "attempt"
## [10] "sessionOrder" "action" "data"
## [13] "X.detail." "X.value."
"X.detail..x..value."
## [16] "X.ex."
```

Then I check for total NA's in the raw data and further found columns having NA's from summary of data.

```
## Total NA's : 56
```

```
## clientTimeStamp totalTimePlayed userId attempt
## Length:14143 Min. : 0 Min. : 262 Min. : -1.00
## Class :character 1st Qu.: 1029 1st Qu.:6707 1st Qu.: 1.00
## Mode :character Median : 3911 Median :6709 Median : 2.00
## Mean : 7050 Mean :6674 Mean : 2.73
## 3rd Qu.:11353 3rd Qu.:6712 3rd Qu.: 4.00
## Max. :67196 Max. :6721 Max. :45.00
## NA's :11 NA's :45
```

It is clear that the data has 11 NA's in "userId" and 45 NA's in "attempt" columns. "attempt" column also has some negative values. Last four columns looks optional as they have fewer entries and the purpose of putting them in data is also not clear from labels. First I fix the problem of NA's in "userId". As it is clearly visible from

```
##   totalTimePlayed userId  
deviceId  
## 1           0    6706 6706_0E08FB43-A854-44BF-B9BD-  
A5A12B00D383  
## 2           0    6706 6706_0E08FB43-A854-44BF-B9BD-  
A5A12B00D383
```

“userId” is first part of the “deviceId” so I replaced them with correct userId after parsing and splitting “deviceId”.

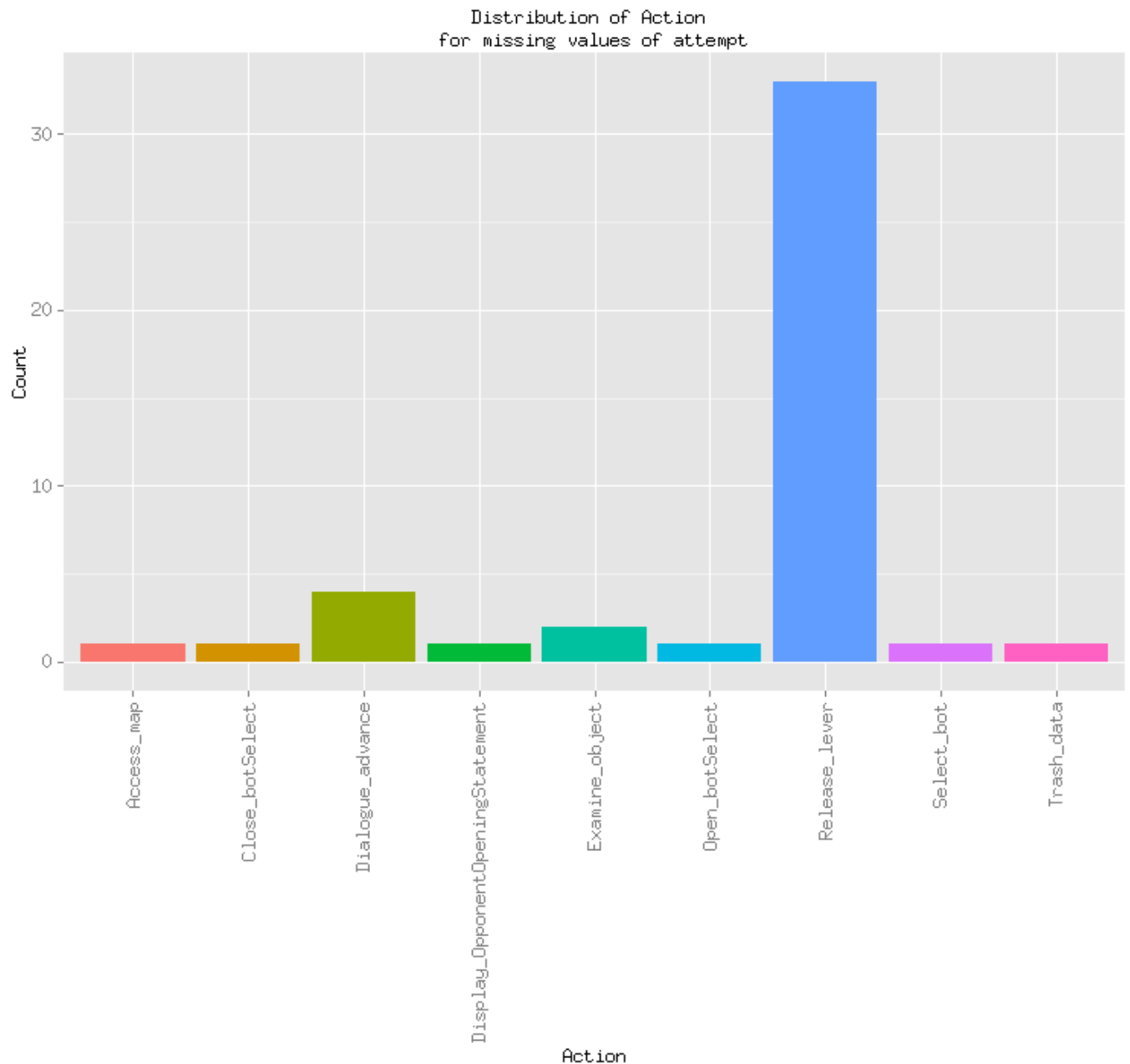
```
# finding index of NA's of userId and replacing them with correct  
values.  
nas_in_userId <- which(is.na(new_data$userId))  
for (i in nas_in_userId) {  
  new_data[i, 5] <- (strsplit(new_data[i, 6], "_")[1])  
}
```

The data also has duplicate values.

```
## Number of duplicate rows : 1
```

I found that rows 2954 and 2955 are identical. I used unique command to remove duplicate rows.

For NA's and negative values in attempt column. First I analyzed data for missing or NA values of attempt.

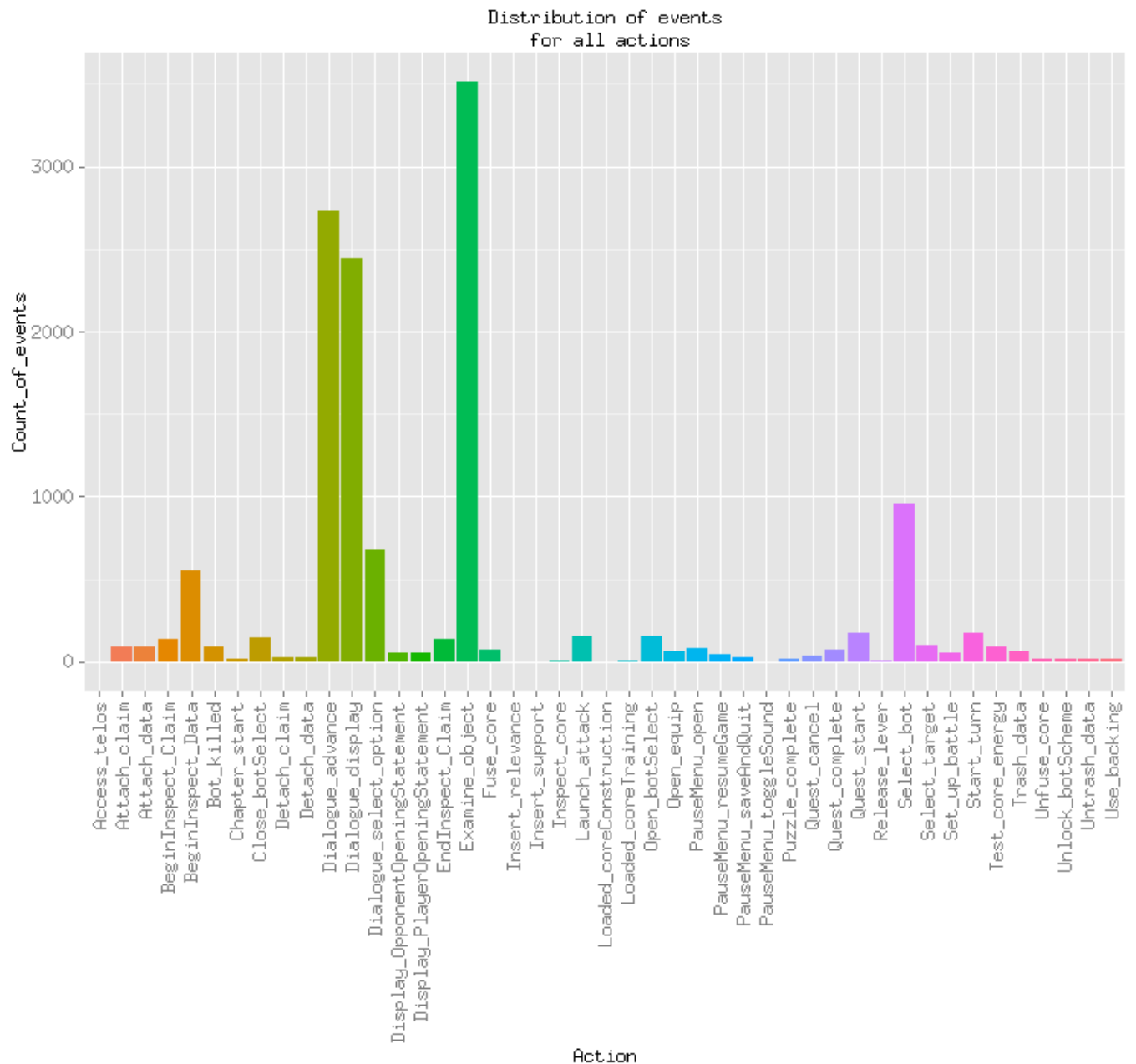


It is clear from the above graph that most missing values for attempt are in action category- "Release_lever" and their values in "data" column are also missing. I stored data for negative attempts in sepearate dataframe for further analysis.

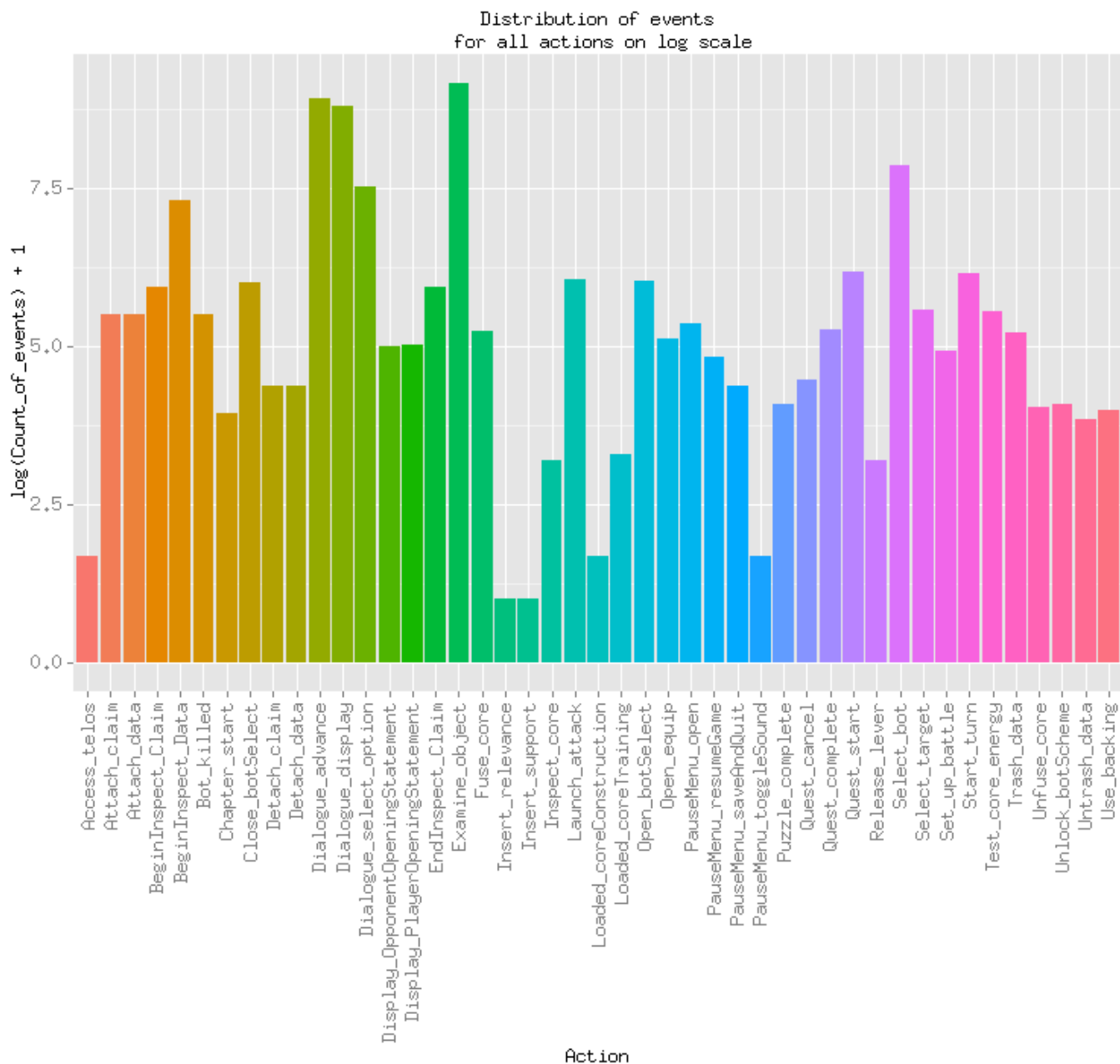
Problem 1: Using one or more statistical tool, preferably coded in some language like R. Extract the number of times students have performed an action (eg: 'fused core').

Analysis: The list of actions is :

##	Action	Count_of_events
## 1	Access_telos	2
## 2	Attach_claim	91
## 3	Attach_data	90
## 4	BeginInspect_Claim	138
## 5	BeginInspect_Data	554
## 6	Bot_killed	91
## 7	Chapter_start	19
## 8	Close_botSelect	149
## 9	Detach_claim	29
## 10	Detach_data	29
## 11	Dialogue_advance	2732
## 12	Dialogue_display	2443
## 13	Dialogue_select_option	678
## 14	Display_OpponentOpeningStatement	55
## 15	Display_PlayerOpeningStatement	56
## 16	EndInspect_Claim	139
## 17	Examine_object	3515
## 18	Fuse_core	70
## 19	Insert_relevance	1
## 20	Insert_support	1
## 21	Inspect_core	9
## 22	Launch_attack	158
## 23	Loaded_coreConstruction	2
## 24	Loaded_coreTraining	10
## 25	Open_botSelect	153
## 26	Open_equip	61
## 27	PauseMenu_open	79
## 28	PauseMenu_resumeGame	46
## 29	PauseMenu_saveAndQuit	29
## 30	PauseMenu_toggleSound	2
## 31	Puzzle_complete	22
## 32	Quest_cancel	32
## 33	Quest_complete	71
## 34	Quest_start	178
## 35	Release_lever	9
## 36	Select_bot	957
## 37	Select_target	97
## 38	Set_up_battle	51
## 39	Start_turn	173
## 40	Test_core_energy	95
## 41	Trash_data	67
## 42	Unfuse_core	21
## 43	Unlock_botScheme	22
## 44	Untrash_data	17
## 45	Use_backing	20



As some actions were not visible in above graph, though they represent actual numbers. There is lot of variation in attempt column as the minimum value is 1 and maximum reaches to 45. I made the bar graph again using log



Code for extracting number of times students have performed an action i.e. "Fuse core"

```
count <- pbl_table[(pbl_table$Action == "Fuse_core"), 2]
cat("Number of attempts for Fused core :", count)
```

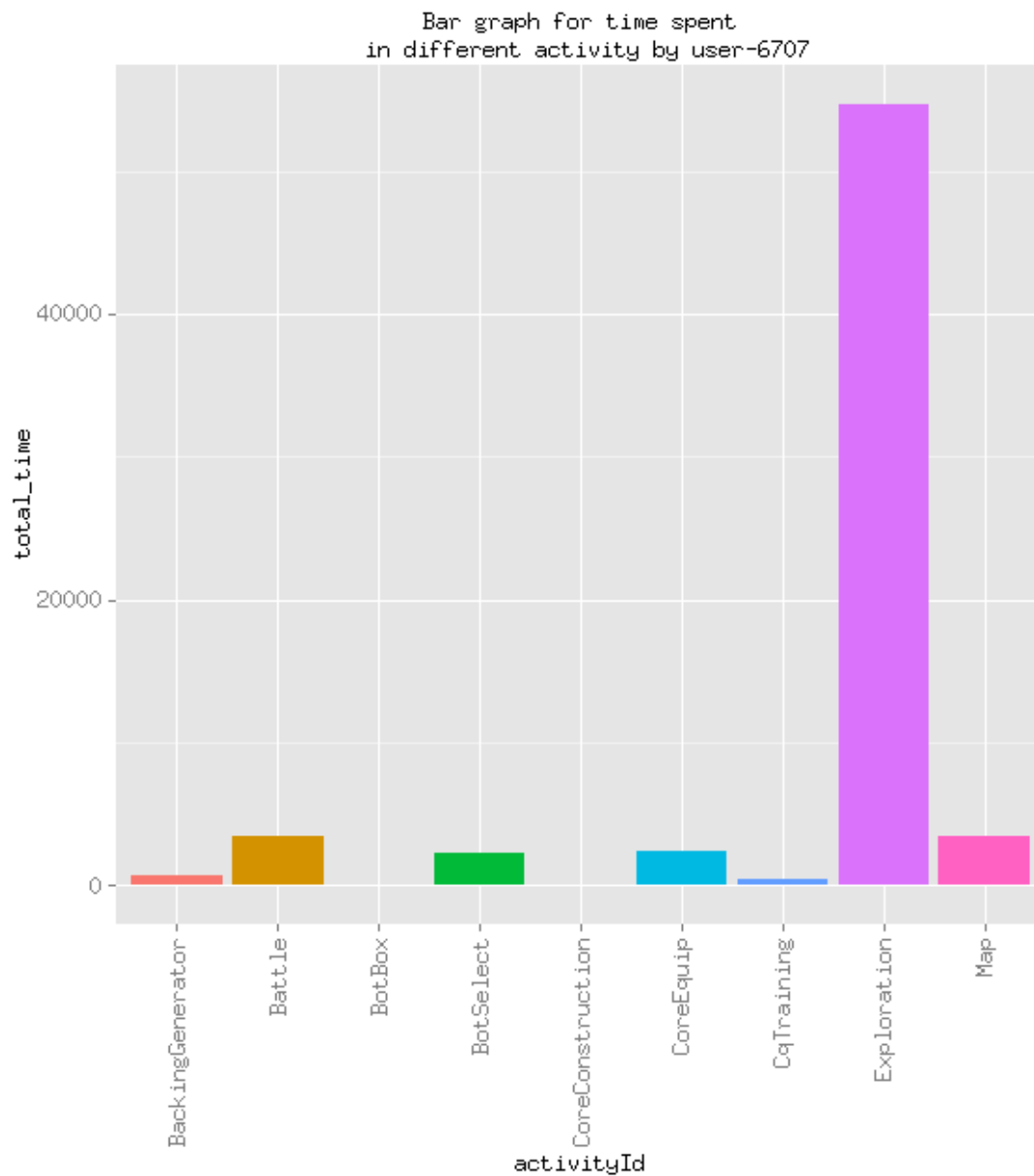
```
## Number of attempts for Fused core : 70
```

Problem 2 & 3. Determine some patterns on session open & close. Provide the algorithm to do so and any/all code used to demonstration marking open/close of session.

#####Analysis : For the analysis of session I analysed the data in virtual and real time. The virtual time analysis is based on “clienttimestamp” and real time time analysis is based on “totaltimeplayed”.

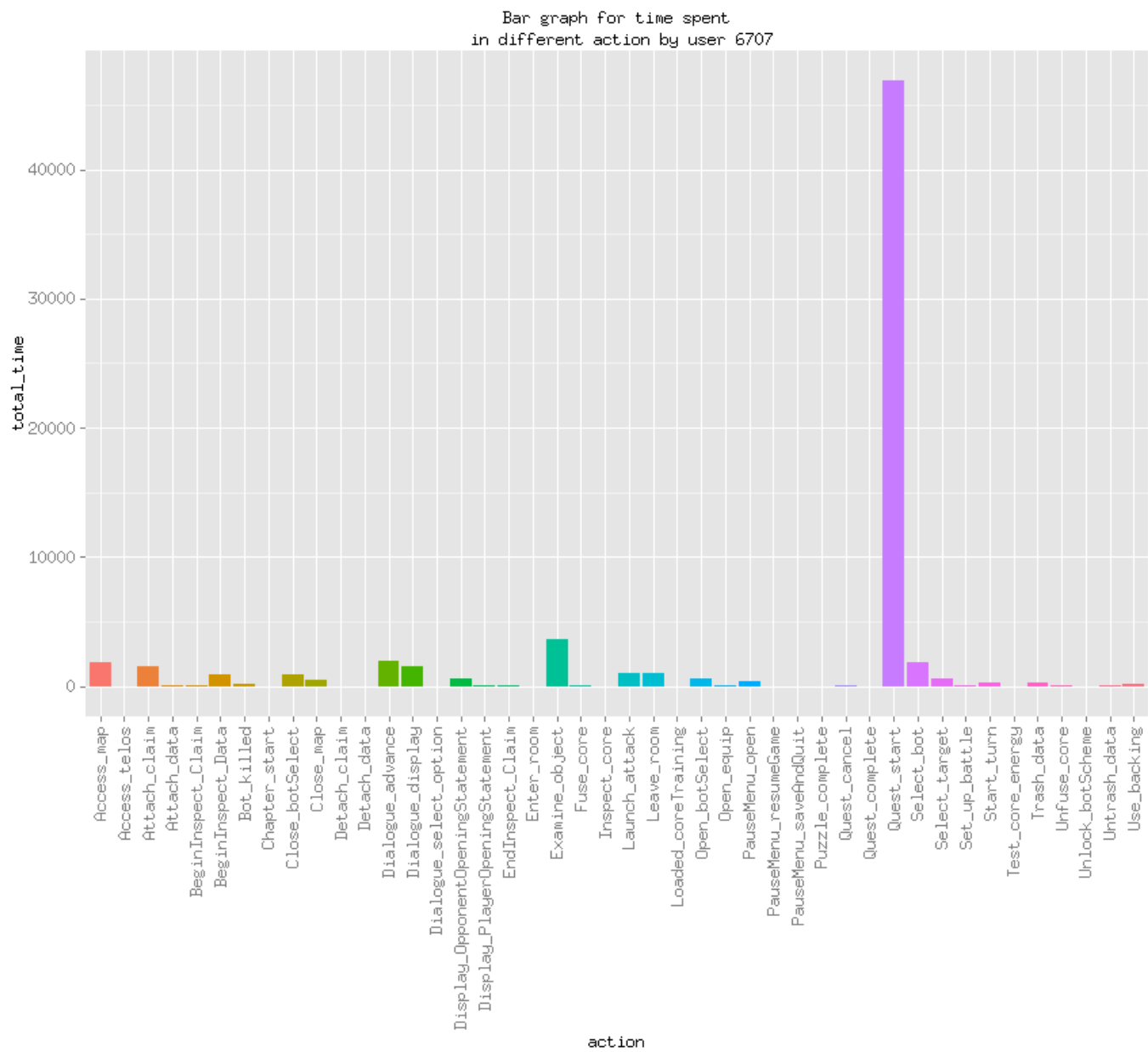
Graphs in real time

##	activityId	total_time
## 1	BackingGenerator	695
## 2	Battle	3362
## 3	BotBox	3
## 4	BotSelect	2203
## 5	CoreConstruction	82
## 6	CoreEquip	2374
## 7	CqTraining	400
## 8	Exploration	54722
## 9	Map	3378



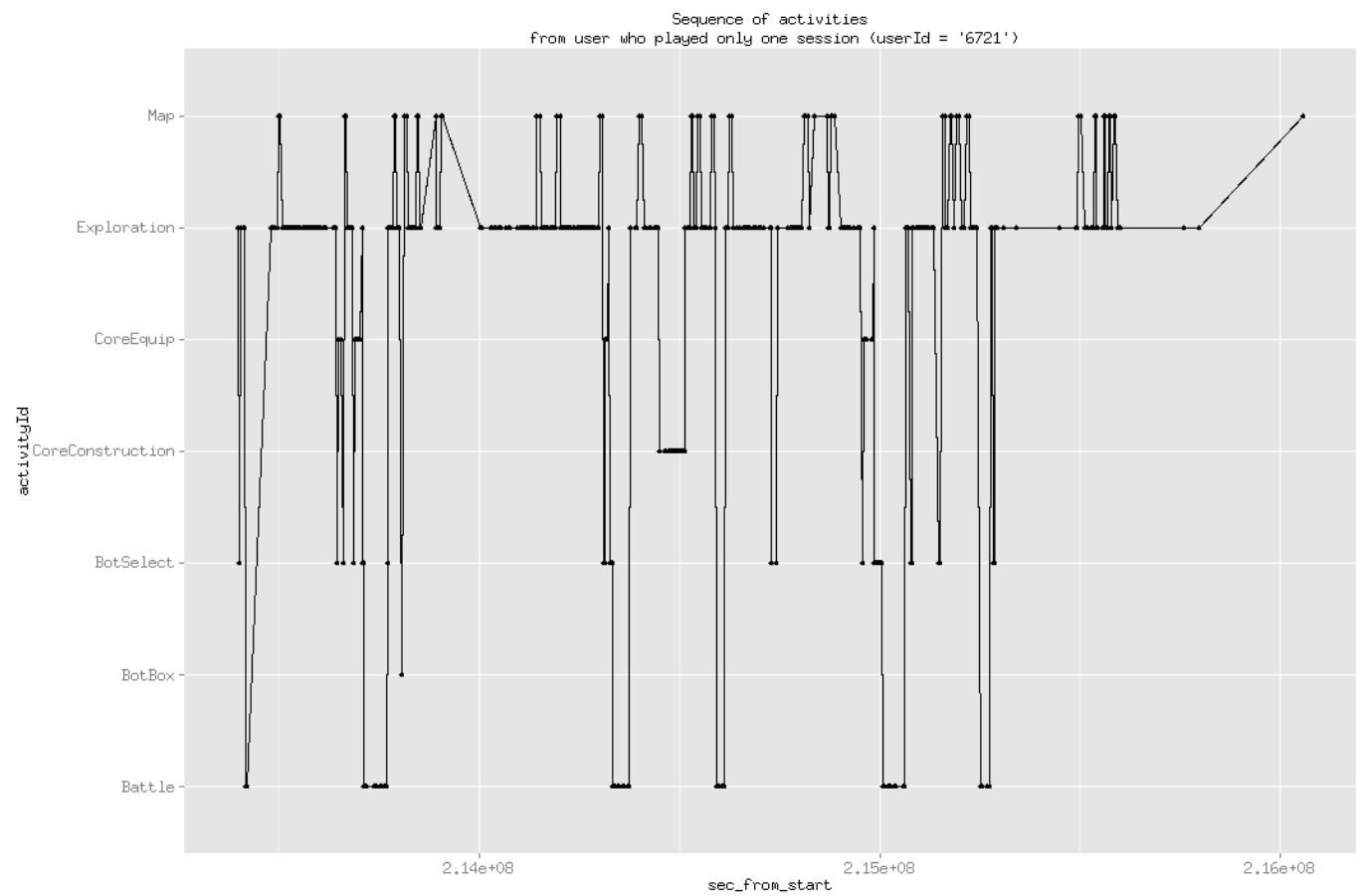
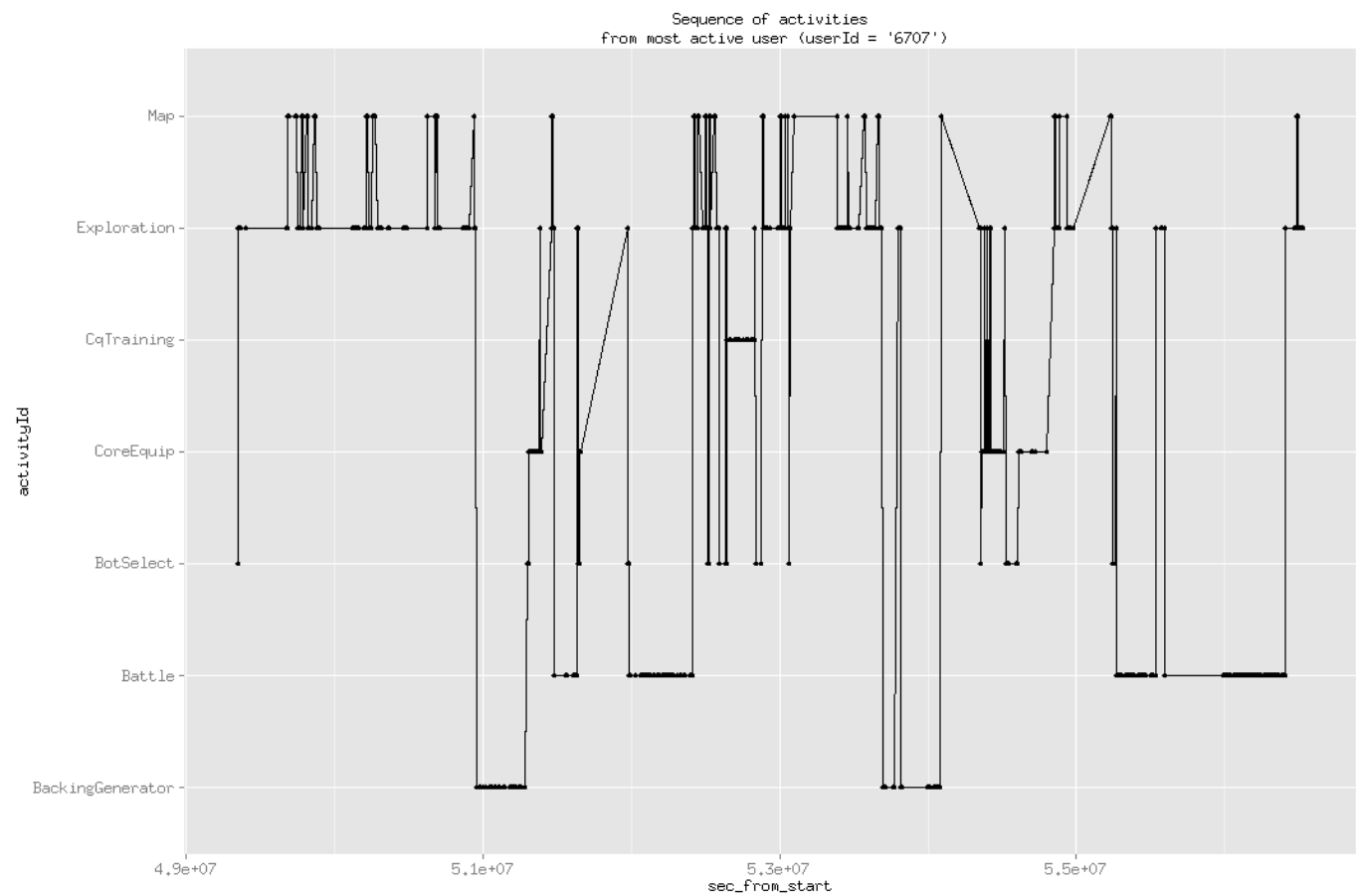
From the above graph it is clear that user-6707 mostly spent his time in Exploration activity and he also tried "Backing generator" which other users did not try.

##	action	total_time
## 1	Access_map	1834
## 2	Access_telos	14
## 3	Attach_claim	1534
## 4	Attach_data	37
## 5	BeginInspect_Claim	114
## 6	BeginInspect_Data	961
## 7	Bot_killed	138
## 8	Chapter_start	3
## 9	Close_botSelect	897
## 10	Close_map	520
## 11	Detach_claim	11
## 12	Detach_data	6
## 13	Dialogue_advance	1912
## 14	Dialogue_display	1506
## 15	Dialogue_select_option	2
## 16	Display_OpponentOpeningStatement	556
## 17	Display_PlayerOpeningStatement	35
## 18	EndInspect_Claim	74
## 19	Enter_room	0
## 20	Examine_object	3591
## 21	Fuse_core	40
## 22	Inspect_core	3
## 23	Launch_attack	1005
## 24	Leave_room	1024
## 25	Loaded_coreTraining	4
## 26	Open_botSelect	620
## 27	Open_equip	30
## 28	PauseMenu_open	425
## 29	PauseMenu_resumeGame	0
## 30	PauseMenu_saveAndQuit	0
## 31	Puzzle_complete	9
## 32	Quest_cancel	25
## 33	Quest_complete	0
## 34	Quest_start	46877
## 35	Select_bot	1850
## 36	Select_target	597
## 37	Set_up_battle	50
## 38	Start_turn	326
## 39	Test_core_energy	11
## 40	Trash_data	295
## 41	Unfuse_core	39
## 42	Unlock_botScheme	0
## 43	Untrash_data	78
## 44	Use_backing	166



In the above graph missing bars represent that the user did not perform those actions. This analysis can be extended for other users also.

Graphs in virtual time

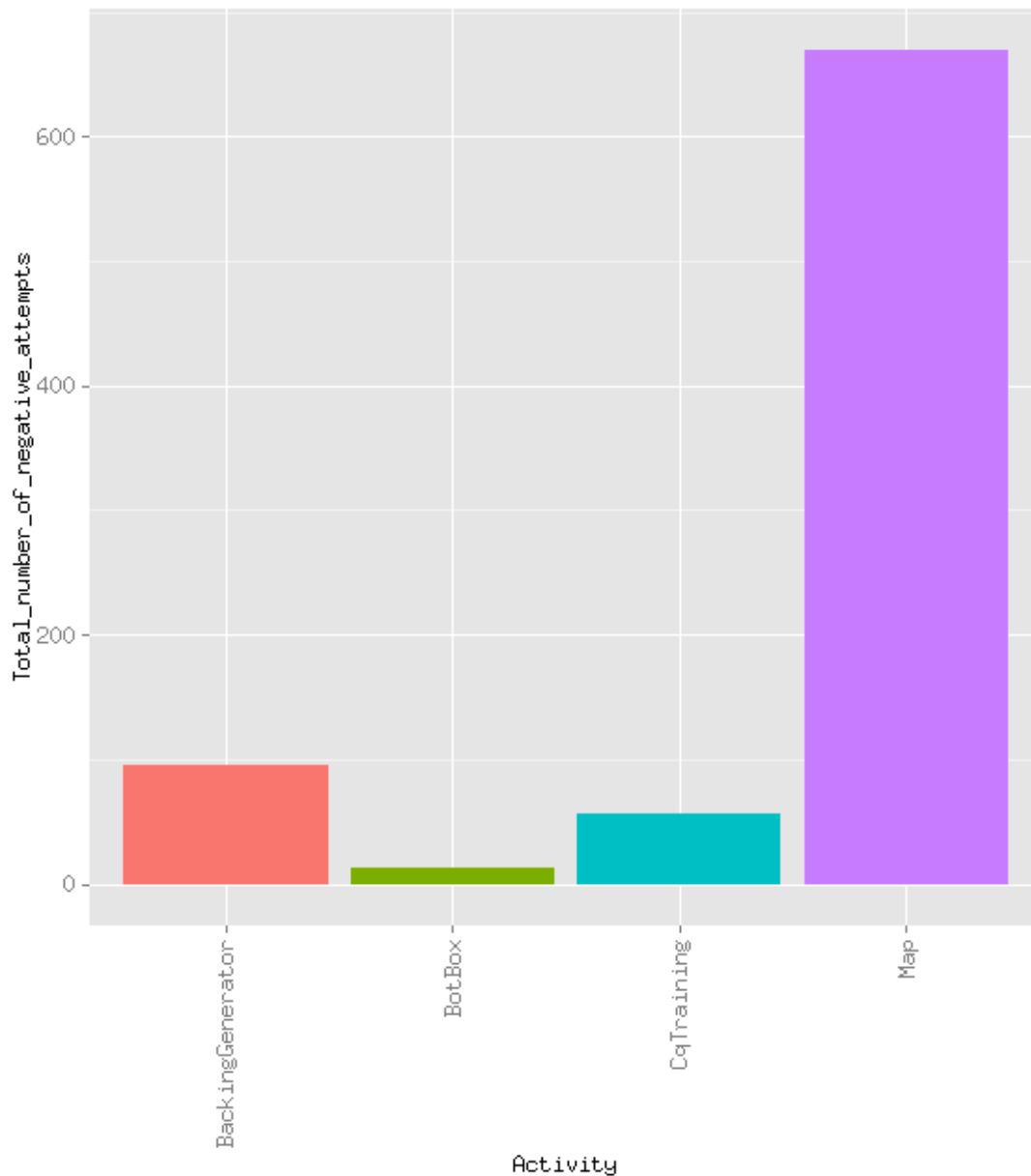


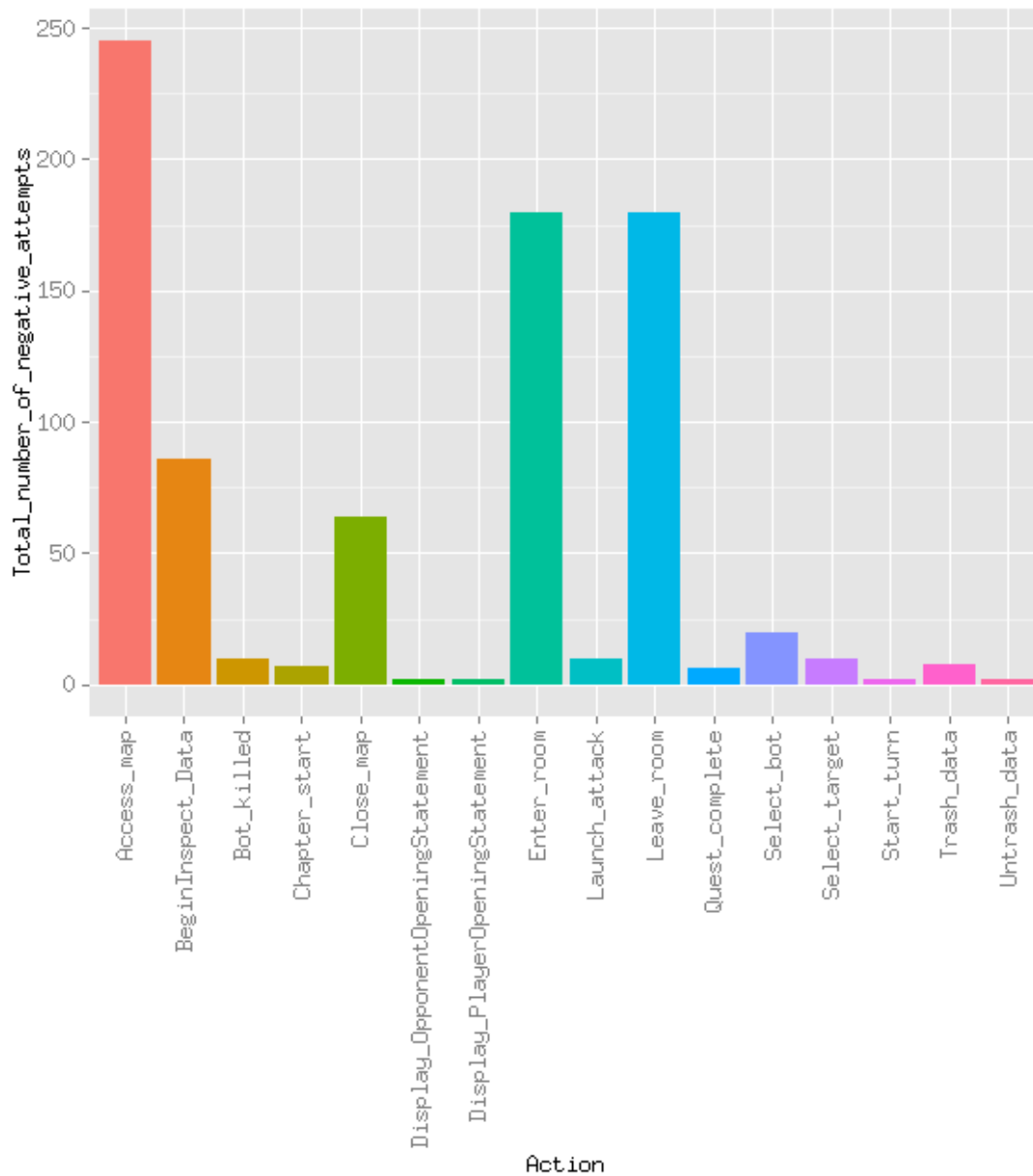
From graph it is clear that user “6707” is very active and he tried every game except botbox. From graph it is also clear that “Exploration” is most popular among all users of data. This is how I found pattern in sessions using time and sessionId column.

Problem 4. Find one or more other interesting pattern in the data.

Analysis part 1

In this section I first analysed data for negative “attempt”. “attempt” column has only -1, negative value for 834 rows. It is interesting to know that maximum negative values occur for map activity and access_map, enter_room, leave_room actions.





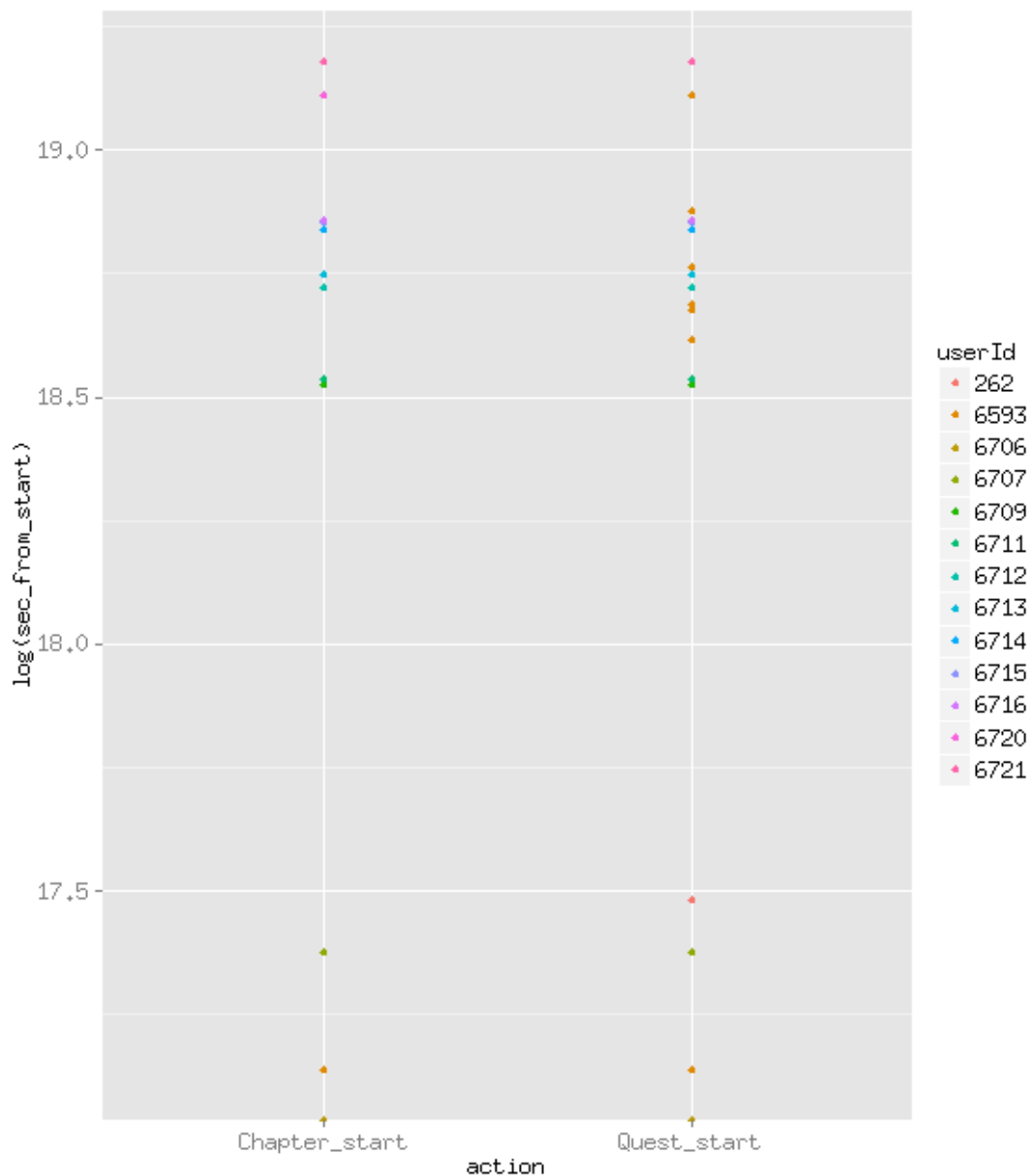
Analysis part 2

In this part I analysed data for “totalTimePlayed” having 0 values. All those values occurred only for Exploration activity and for chapter_start and quest_start actions.

##	userId	totalTimePlayed	activityId
## 1	6706	0	Exploration
## 2	6706	0	Exploration
## 1496	6593	0	Exploration
## 1497	6593	0	Exploration
## 1597	6707	0	Exploration
## 1598	6707	0	Exploration
## 2624	262	0	Exploration
## 5853	6709	0	Exploration
## 5854	6709	0	Exploration

##	6323	6711	0	Exploration	
##	6324	6711	0	Exploration	
##	8382	6593	0	Exploration	
##	9891	6593	0	Exploration	
##	9942	6593	0	Exploration	
##	10109	6712	0	Exploration	
##	10110	6712	0	Exploration	
##	10780	6713	0	Exploration	
##	10781	6713	0	Exploration	
##	11404	6593	0	Exploration	
##	12091	6714	0	Exploration	
##	12092	6714	0	Exploration	
##	12408	6715	0	Exploration	
##	12409	6715	0	Exploration	
##	12512	6716	0	Exploration	
##	12513	6716	0	Exploration	
##	12701	6593	0	Exploration	
##	12702	6593	0	Exploration	
##	12707	6720	0	Exploration	
##	12708	6720	0	Exploration	
##	12750	6593	0	Exploration	
##	12752	262	0	Exploration	
##	12757	6593	0	Exploration	
##	12765	6721	0	Exploration	
##	12766	6721	0	Exploration	
##			sessionId	attempt	sessionOrder
##	1	13b78a70-25ed-11e4-adc4-b1ca37c95b39	1		2
##	2	13b78a70-25ed-11e4-adc4-b1ca37c95b39	1		3
##	1496	a9d88280-262d-11e4-9600-69986cc0e309	1		2
##	1497	a9d88280-262d-11e4-9600-69986cc0e309	1		3
##	1597	0939b710-263f-11e4-9600-69986cc0e309	1		2
##	1598	0939b710-263f-11e4-9600-69986cc0e309	1		3
##	2624	457f29e0-2648-11e4-9600-69986cc0e309	1		2
##	5853	d373fa30-26ef-11e4-b64c-23ecc8f0c749	1		2
##	5854	d373fa30-26ef-11e4-b64c-23ecc8f0c749	1		3
##	6323	ef771f20-26f2-11e4-b64c-23ecc8f0c749	1		2
##	6324	ef771f20-26f2-11e4-b64c-23ecc8f0c749	1		3
##	8382	d8e79ef0-2707-11e4-b64c-23ecc8f0c749	1		2
##	9891	253da4a0-2719-11e4-b64c-23ecc8f0c749	1		2
##	9942	5500fb20-271d-11e4-b64c-23ecc8f0c749	1		2
##	10109	13e39fd0-2727-11e4-b64c-23ecc8f0c749	1		2
##	10110	13e39fd0-2727-11e4-b64c-23ecc8f0c749	1		3
##	10780	3a5c52c0-2730-11e4-b64c-23ecc8f0c749	1		2
##	10781	3a5c52c0-2730-11e4-b64c-23ecc8f0c749	1		3
##	11404	19ed75a0-2735-11e4-b64c-23ecc8f0c749	1		2
##	12091	be177c20-274e-11e4-b64c-23ecc8f0c749	1		2
##	12092	be177c20-274e-11e4-b64c-23ecc8f0c749	1		3
##	12408	6630fc60-2754-11e4-b64c-23ecc8f0c749	1		2
##	12409	6630fc60-2754-11e4-b64c-23ecc8f0c749	1		3
##	12512	fbef7d30-2754-11e4-b64c-23ecc8f0c749	1		2
##	12513	fbef7d30-2754-11e4-b64c-23ecc8f0c749	1		3

## 12701	60fbb2a0-275c-11e4-b64c-23ecc8f0c749	1	2
## 12702	60fbb2a0-275c-11e4-b64c-23ecc8f0c749	1	2
## 12707	ad89ed70-27bc-11e4-b888-3f4840ca9c86	1	2
## 12708	ad89ed70-27bc-11e4-b888-3f4840ca9c86	1	3
## 12750	dc92d4f0-27bd-11e4-b888-3f4840ca9c86	1	2
## 12752	708c5c20-27dd-11e4-9845-175fcf1a3e72	1	2
## 12757	830eabf0-27dd-11e4-9845-175fcf1a3e72	1	1
## 12765	ee76b450-27dd-11e4-9845-175fcf1a3e72	1	2
## 12766	ee76b450-27dd-11e4-9845-175fcf1a3e72	1	3
##	action sec_from_start		
## 1	Quest_start	0	
## 2	Chapter_start	0	
## 1496	Quest_start	27744000	
## 1497	Chapter_start	27744000	
## 1597	Quest_start	35201000	
## 1598	Chapter_start	35201000	
## 2624	Quest_start	39161000	
## 5853	Quest_start	111131000	
## 5854	Chapter_start	111131000	
## 6323	Quest_start	112284000	
## 6324	Chapter_start	112284000	
## 8382	Quest_start	121379000	
## 9891	Quest_start	128879000	
## 9942	Quest_start	130677000	
## 10109	Quest_start	134858000	
## 10110	Chapter_start	134858000	
## 10780	Quest_start	138796000	
## 10781	Chapter_start	138796000	
## 11404	Quest_start	140876000	
## 12091	Quest_start	151769000	
## 12092	Chapter_start	151769000	
## 12408	Quest_start	154320000	
## 12409	Chapter_start	154320000	
## 12512	Quest_start	154572000	
## 12513	Chapter_start	154572000	
## 12701	Quest_start	157587000	
## 12702	Quest_start	157587000	
## 12707	Quest_start	199105000	
## 12708	Chapter_start	199105000	
## 12750	Quest_start	199611000	
## 12752	Quest_start	213177000	
## 12757	Quest_start	213224000	
## 12765	Quest_start	213398000	
## 12766	Chapter_start	213398000	



Analysis part 3

There are six devices. Same device is used by multiple users, this can be infer from following table:

```
## [1] "0E08FB43-A854-44BF-B9BD-A5A12B00D383"
## [2] "DFD58325-26E1-4EE1-942A-A813560340C1"
## [3] "75BF90B6-EBBF-4B97-9D7A-A6B934E770E6"
## [4] "24CED89F-F6D5-4B31-9442-CD29DCD8465F"
## [5] "585A3601-E316-4296-A693-FF4B2192AA72"
## [6] "4C0AEE21-D51A-4030-90DC-BBCA081742D2"
```

##	userId	only_device_id
## 1	6706	0E08FB43-A854-44BF-B9BD-A5A12B00D383
## 1496	6593	DFD58325-26E1-4EE1-942A-A813560340C1
## 1597	6707	75BF90B6-EBBF-4B97-9D7A-A6B934E770E6
## 2626	262	DFD58325-26E1-4EE1-942A-A813560340C1
## 5864	6709	0E08FB43-A854-44BF-B9BD-A5A12B00D383
## 6334	6711	24CED89F-F6D5-4B31-9442-CD29DCD8465F
## 10146	6712	75BF90B6-EBBF-4B97-9D7A-A6B934E770E6
## 10818	6713	0E08FB43-A854-44BF-B9BD-A5A12B00D383
## 12133	6714	585A3601-E316-4296-A693-FF4B2192AA72
## 12450	6715	75BF90B6-EBBF-4B97-9D7A-A6B934E770E6
## 12555	6716	4C0AEE21-D51A-4030-90DC-BBCA081742D2
## 12750	6720	0E08FB43-A854-44BF-B9BD-A5A12B00D383
## 12808	6721	DFD58325-26E1-4EE1-942A-A813560340C1

Problem 5. What would you change in the telemetry data?

At times, server and client are not synchronized well which cause repeated messages. Local time from user device and location data can help us understand when user behavior and given that the purpose of these games to enhance learning, these are important factors. Location data for individual users may be used to locate and analyse less used or popular activities.

Problem 6. What are some compression methods for this data?

1. DeviceId has userId embedded into it so userId field is redundant.
2. There are repeated rows with server time different.
3. Last four columns appear to be key value pair which are already present in "data" column. Although it is possible that implementation requires those columns to simplify processing
4. Some of the columns can be encoded.

Future work:

This is exploratory analysis and lot more can be done to analyse relations between activity, action with respect to time and user after melting and reshaping the data. The data can be analysed more to predict which activity user like, based on past data or can group users with similar playing behavior. This data can be analysed for those users who spent less time and can make changes to attract those. Shinny can be used to make the analysis more user friendly(for data analysis purpose). I wanted to explore "data" column more but I couldn't do that due to lack of time.