THE PUBG GAME DATASET:
An Investigation of Factors Affecting Team Placement

Team – 442
Gary Guo, Qimeng Luo, Ricky Gui, Yixin Chen
492134, 491803, 492004, 487906
December 16th, 2020

1 INTRODUCTION

PlayerUnknown's Battlegrounds, also referred to as “PUBG” by its fans, is an online multiplayer third-person/ first-person perspective shooting game introduced to the public on December 20th, 2017. This iconoclastic creation of the South Korean gaming company - Bluehole has become viral quickly through streaming platforms such as Twitch and the gaming community. There are many reasons why PUBG established its reputation in such a short period and took its niche in the market. Some argued that the game successfully launched its marketing campaign, while others emphasized its graphic design. Nevertheless, it is undoubtedly that its unique “battleground” game model is what stirred the market.

To elaborate, the setting of the game is on an island-like location with various buildings and land structures throughout the map. The game starts with up to 100 players on a transport aircraft that flies from one side of the map to another. Players are then free to jump at any point as they desire. This game aims to equip oneself with as many weapons and gadgets as possible to protect oneself from others’ attacks. On the other hand, a deadly gas will be released in the “bluezone” after the game starts for a while, where the gas would deal high damage to the players as they stay within the area. Once the game has only one player left, the game ends. With that in mind, the key to success is to pick a landing destination where players are most likely to find a weapon.

Notably, PUBG is only one of the many games in the market built around such a virtual world where players could experience an alternative reality. Other games such as “Grand Theft Auto” and “Cyberpunk 2077” have faced ethical, logical, and technical dilemmas when building their version of their games' reality. Thus, analyzing this multifaceted game model of PUBG would allow us to discover the decisions the game makers made when designing the details of the game and get into their mindset on why they made such decisions. A breakdown of the dilemmas and solutions in this game would further allow future game designers to take hints from their decision-making.

2 DATA ACQUISITION

2.1 DATA DESCRIPTION

The dataset we selected for our following research is called “PUBG Match Deaths and Statistics.” It was uploaded by Kevin Pei on Jan 12th, 2018, on the acclaimed data science community – “Kaggle.” According to Kevin, the original dataset was extracted from the game tracker website called “pubg.op.gg”. The dataset is structured and stored in CSV files. Therefore, this dataset is reasonably relevant, credible, and legal for further analysis considering its released time and source.

At a glance, the dataset has two kinds of data tables, consisting of different combinations of variables and built for different purposes. In total, there are 5 data files for each kind of data table (18.91 GB in total) . Having said that, we are only using one data file from each (3.89 GB in total) to perform our analysis, considering the scope of our research.

2.1.1 Aggregated Match Statistics

The first kind is named “agg_match_states”, which has 13849287 records and 15 columns. Its columns contain aggregated match information accounting for different perspectives of a game, as shown below:

- date: the date-time of the match
- game_size: how many players participated in the match
- match_id: the assigned identification code for the match
- match_mode: the code for the mode of the game
- party_size: how much people there are in the party that a player belongs to
- player_assists: the count for kill assist of a player for its teammates
- player_dbno: “down but not out,” lose all health points but can be healed by team
- player_dist_ride: the total distance that a player traveled in a vehicle in a match
- player_dist_walk: the total distance of walking of a player in a match
- player_dmg: the sum of all damage a player dealt in a match
- player_kills: total number of kills of a player in a game
- player_name: the name (game alias) of the player

- player_survive_time: total time player stood in a game
- team_id: an identification code assigned for the team
- team_placement: the final ranking of a team in a game

2.1.2 Deaths in Match Statistics

The second kind of data table with 13426348 rows and is called “agg_match_stats_final”, which contain information associated with the death of the player, including the following columns/variables:

- killed_by: the way a player was killed (ex. Grenade, punch, M416 …)
- killer_name: the name (game alias) of the killer
- killer_placement: the final placement of the killer (with “1” being the best)
- killer_position_x: the horizontal position of the killer on the map
- killer_position_y: the vertical position of the killer on the map
- map: the type of map (2 kinds: Miramar or Erangel)
- match_id: the identification code assigned for the match
- time: the time in seconds that the victim lived in the game
- victim_name: the name (game alias) of the victim
- victim_placement: the final placement of the victim (with “1” being the best)
- victim_position_x: the horizontal position of the victim on the map
- victim_position_y: the vertical position of the victim on the map

2.1.3 Matches summary

As we summarized in appendix K, we have 31317 unique matches for solo games, 47799 unique matches for duo games and 70749 unique matches for squad games in our data set.

3 PROBLEM STATEMENT

As mentioned, this avant-garde game mode is the main reason we have chosen this game to focus our research. We find it interesting how the timing and placement of weapons on the map and other tools could be critical to one’s victory and final placement in the game.

For instance, if one could get a powerful weapon such as M416, SCAR-L in the early game, one could likely get more kills in the first few minutes and defend oneself with counter-attacks. Also, cars like Mirado, UAZ, and Dacia could help a player get out of the “bluezone” quickly and hit enemies. At the same time, a motorcycle may be agile but expose oneself as a moving target.

As a result, if an experienced player finds a powerful weapon, he/she would likely disrupt the game's momentum. The game designer must then decide on the contribution of different factors to a balanced gameplay.

However, as far as we are concerned, there was no report of, nor have we, players of this game, have experienced any game that seems oddly challenging. That means there is a specific algorithm that ensures all players' gaming experience, whether experienced or not.

For these reasons, we would like to investigate the influence of different factors around the final placement and death, namely:

- What factors influence the final placement of a player/team during a game?
- What further improvement can we make to reach a more balanced gameplay?

4 WHY IS THIS A BIG DATA

By definition, the notion of “big data” is considered as a form of data that is so large in size that it cannot be efficiently processed by our ordinary processing methods or tools in time. There are three “Vs” in big data, which stands for velocity, volume, and variety.

In our case, our data files, when combined, amounts to 4GB in total. This is therefore an example of a big data file with large volume. In the same vein, analyzing our data timely could effectively help the game operation manager to adjust certain aspects of the game, and consequently reflects the velocity component of big data. Finally, this specific dataset consists of a large amount of data and we are not able to efficiently analyze it with the traditional tools such as Excel, MySQL, Python, or R. Again, it proves that we are dealing with a big data problem.

5 METHODOLOGY & RESULTS

5.1 METHODOLOGY

To process the data, we are considering the following steps. First, we create two separate tables using specific code with Impala in Hue and two of our data files, referred to **Appendix A** and **B.** Then we upload our data online to Hue and load our two data files using the code shown in **Appendix C** and **Appendix D** respectively. We will proceed to use Impala to write SQL queries to help us obtain the information we need for further analysis.

We primarily review the column names and the respective description, to decide what variables we will need.

"Team_placement" as an essential index in our analysis is used to represent the ranking of a team. A higher ranking is represented by a lower value of team placement. In this report, we are trying to discuss the factors that could potentially influence the team placement. Also, we notice that there are players whose placement is 0. We will not consider these players because their team placements make no sense in our analysis.

We analyzed matches in different party sizes independently. As a matter of fact, the SOLO matches have around 90-100 individual players so as 90-100 teams, DUO matches have 40-50 teams of two and SQUAD matches have 22-25 four-man squads. For example, in a squad match, the team that was eliminated at the beginning of the game will rank at #25. Whereas, in a solo game, player get eliminated at first will rank at #100. Hence, rankings in different modes should be evaluated independently.

After looking at the columns, we selected three columns to conduct our analysis.

1. player_kills in agg_match_stats_final.csv

   After we looked at the column, we found players with extremely high kills. We consider them as outliers and will not count them when we do analysis. We would like to see the average ranking of players with kills from 0 to 16 in solo, duo, and squad mode. We expect that there will be a positive relationship.

2. player_dist_ride in agg_match_stats_final.csv

   Since players who have higher rank travel longer than players who have lower rank, we are not interested in the average distance traveled in vehicle. On the

other hand, we would like to see if players will end up with a higher ranking if they find a vehicle. To ensure that our analysis is less biased, we only consider players who survive for at least 5 minutes because 5 minutes is the time players need to gather weapons and equipment they need after landing.

3. killed_by in agg_match_stats_final.csv
   We want to see if choosing a particular weapon will lead to a higher ranking. However, these two tables don't have a unique code to join them. Therefore, we will create a subtable of distinct match ids for each party size. Then, we will join the new match id tables with team_442_PUBG (table where killed_by in) to obtain total kills for each weapon and average placement of players who used this weapon in each mode.
   For weapons, we only consider auto rifles (M416, SCAR-L, M16A4, AKM) and sniper rifles (Kar98k, SKS, Mini 14) since these weapons play a key role in late games. We don't consider guns from airdrops (AUG, Groza, AWM, MK14) because airdrops are rare, and players could get killed from chasing airdrops.

5.2 RESULTS

5.2.1 Analysis on Kills

First, we are hoping to find out the relationship between the number of kills and the final placement of a player. As shown in code in **Appendix E**. From this analysis, we have visualized the result as shown below. According to our visualizations, Solo, Duo and Squad matches all agree with the following conclusion: the higher amount of kills that a player has the higher the final ranking he will be placed at. If a solo game player gets an average of 30 kills, he/she will probably have a ranking of 1.8 in average. If a duo match team gets an average of 30 kills, they will probably have an average ranking of 1.455. Respectively, if a squad match team gets an average of 30 kills, they will probably have an average ranking of 1.086. In graphs below, the three plots show correlation between kills and placement in different party sizes, and the red line indicates the ranking is going upwards as kills increase.

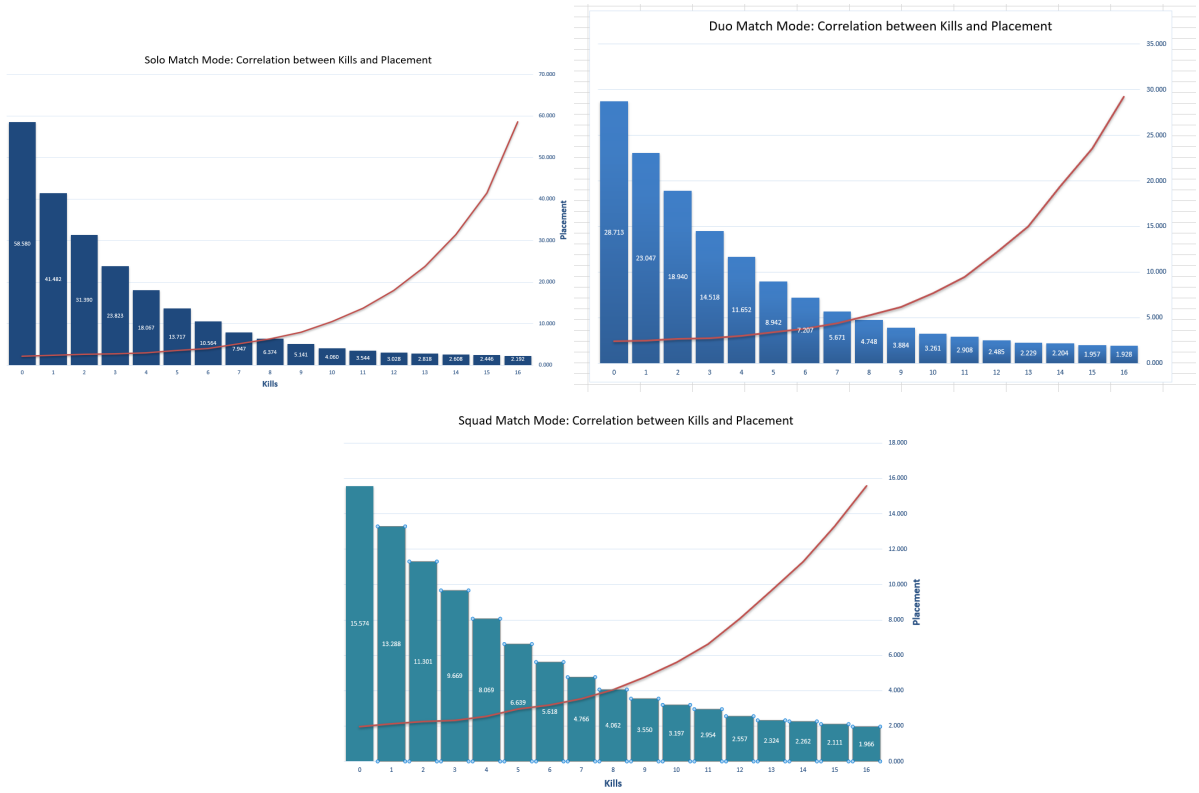


Figure: Correlation Between Kills and Placement (**Appendix F**)

5.2.2 Analysis on Vehicles

Second, we will investigate between players with rides and the final ranking. Using codes shown in Appendix G, we have no matter the party size, one can achieve a higher rank with a vehicle. For instance, if a solo player finds a vehicle, then his/her final placement would likely rise from approximately 41.73 to 23.71, which shows that solo players are the ones most influenced by vehicles as duo player and squad's placement only see changes from 24.82 to 13.58 and 15.42 to 8.61 respectively.

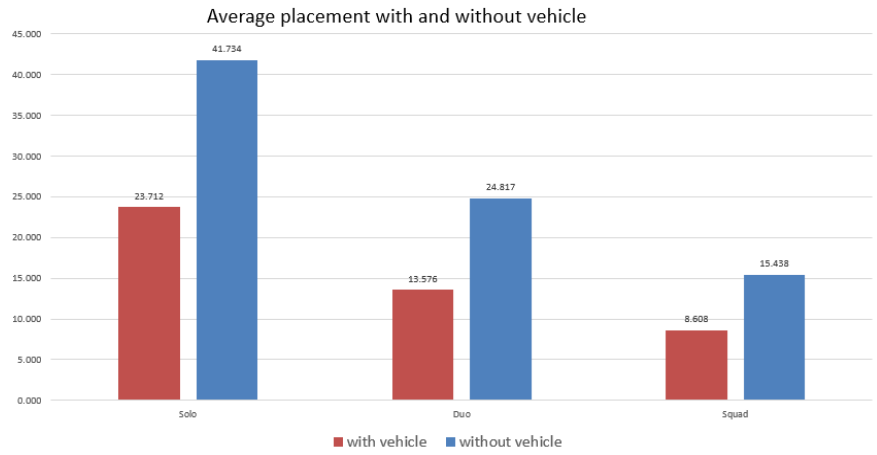
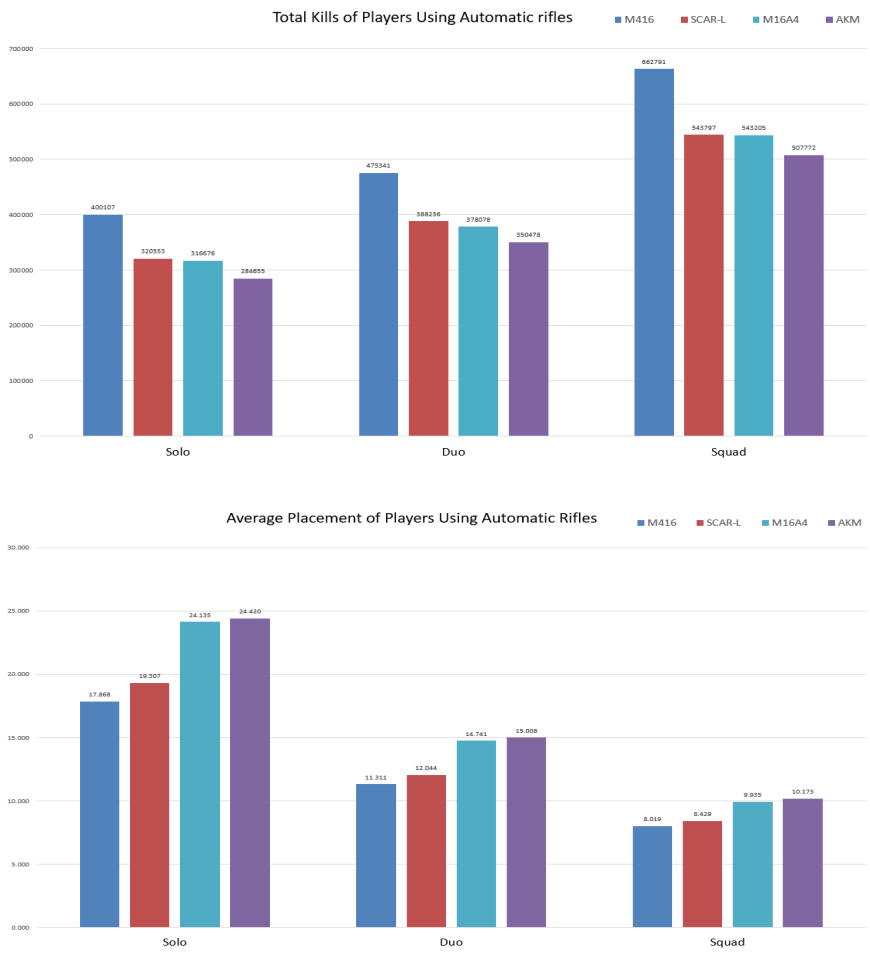


Figure: Average Placement with and without Vehicle (**Appendix H**)

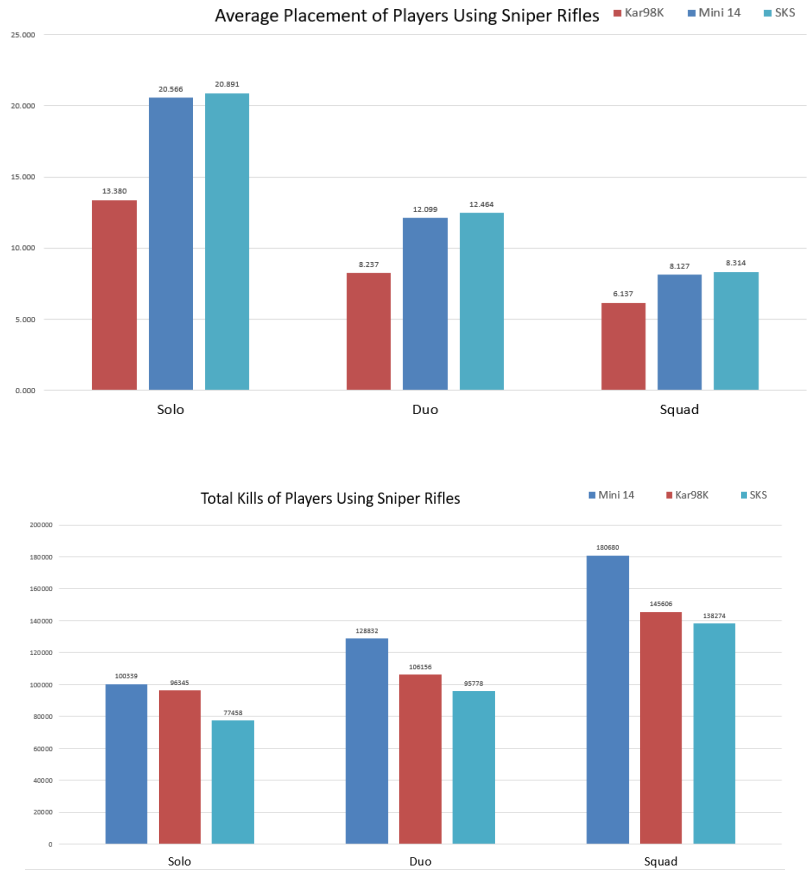
5.2.3 Analysis on Weapons Equipped

We firstly start with both types of rifles, sniper rifles (SR) and assault rifles (AR). According to **Appendix I and J**, we found that in all three types of team, M416 has always shown the most preeminent performance among all ARs, and Kar98k dominated the SRs. M416 has 400,107 total kills in solo matches, 475,341 total kills in duo matches, 662,791 total kills in squad matches. Players in solo games with M416 equipped have an average placement of 17.87, teams in duo games with M416 have an average rank of 11.51 and teams in squad games have a ranking of 8.02. Therefore, in assault rifle groups, players or teams with M416 equipped in a match will theoretically have more kills than those using SCAR-L, M16A4 and AKM. These three weapons have similar probabilities, then total kills are reasonable in the AR group. Consequently, more kills will lead to higher placement.



Graphs of Total Kills and Final Placement by Sniper Rifles (**Appendix J**)

Mini-14 has respectively 100339, 96345, 77458 total kills in solo, duo and squad matches. Players equipped with Kar98 have an average placement of 13.38 in solo games, an average placement of 8.24 in duo games and 6.14 in squad games. In the sniper rifle group, players equipped with Mini-14 in a match will theoretically have more kills than others with SKS, Kar98k, but considering the amount difference of Kar98k and mini-14 refreshed on the map. Total skills of a weapon is not strong enough for predicting. Then, the average placement could be relatively strong in this case.



Graphs of Total Kills and Final Placement by Sniper Rifles (**Appendix J**)

6 Conclusion

In conclusion, we noticed that the amount of kills, usage of vehicle and weapon choice are three predominant factors that contribute to the final placement during the matches of PlayerUnknow's Battlegrounds. For players who are seeking to improve their match placement, we would recommend primarily picking the M416 as the main weapon, find a vehicle on the map and practice more to get more kills. Moreover, our analysis also suggests that the M416 is much more popular than the others, so we would recommend officially nerf the overall performance of M416 from its recoil and damage to balance the gaming experience.

7 Bibliography

Pei, Kevin. "PUBG Match Deaths and Statistics." Kaggle, 12 Jan. 2018, www.kaggle.com/skihikingkevin/pubg-match-deaths.

8 APPENDICES

Appendix A : Uploading Data for Aggregated Match Information

```
CREATE TABLE team_442_PUBG_match (play_date STRING,
game_size int,
match_id STRING,
match_mode STRING,
party_size int,
player_assists int,
player_dbno int,
player_dist_ride FLOAT,
player_dist_walk FLOAT,
player_dmg int,
player_kills int ,
player_name STRING,
player_survive_time FLOAT,
team_id int,
team_placement int)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
STORED AS TEXTFILE TBLPROPERTIES("skip.header.line.count"="1");
```

```
Success.
```

Appendix B : Uploading Data for Deaths in Match Statistics

```

CREATE TABLE team_442_PUBG (play_date int,
game_size int,
match_id STRING,
match_mode STRING,
party_size int,
player_assists int,
player_dbno int,
player_dist_ride FLOAT,
player_dist_walk FLOAT,
player_dmg int,
player_kills int ,
player_name STRING,
player_survive_time FLOAT,
team_id int,
team_placement int)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
STORED AS TEXTFILE TBLPROPERTIES("skip.header.line.count"="1");
```

Appendix C : Data Table for Aggregated Match Information

```
load data inpath '/user/l.qimeng/agg_match_stats_0.csv'
into table team_442_pubg_match

```

Appendix D : Data Table for Deaths in Match Statistics

```
load data inpath '/user/l.qimeng/kill_match_stats_final_0.csv'
into table team_442_pubg
```

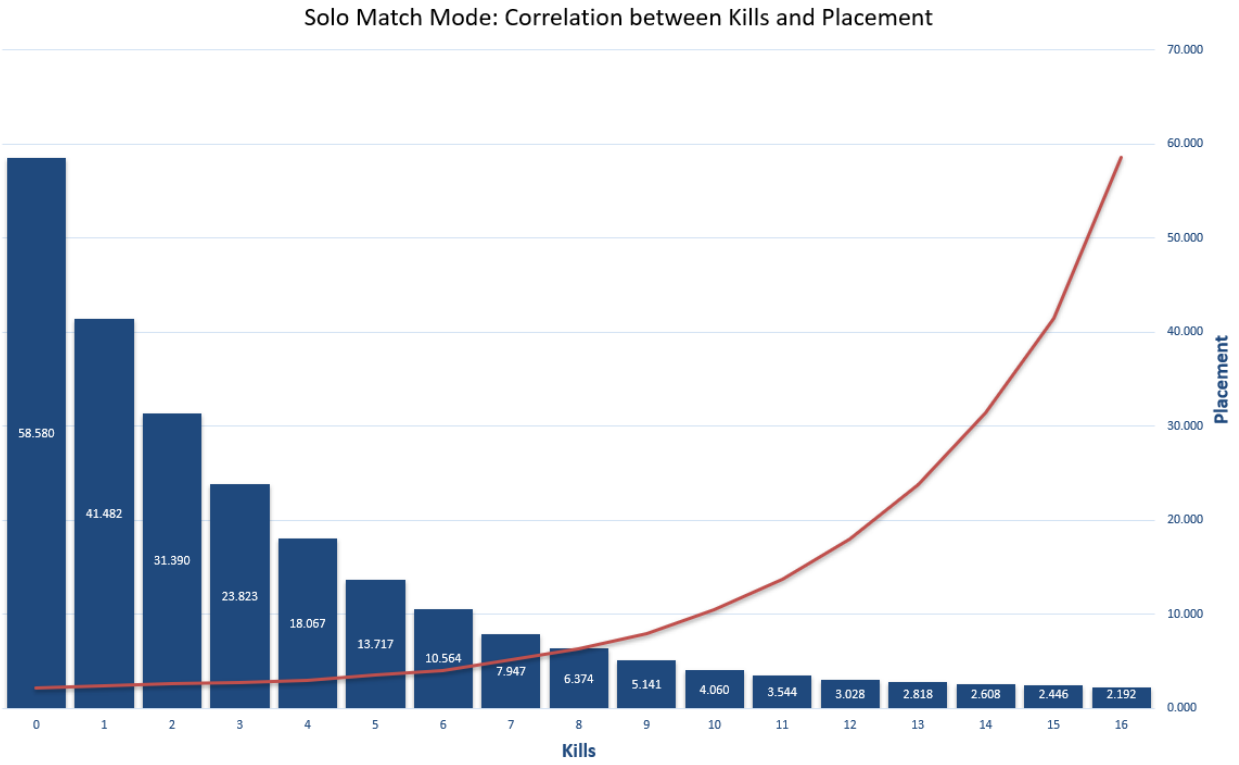
Appendix E: Codes for Correlation of Kills Count and Final Placement (party size 1, 2, 4)

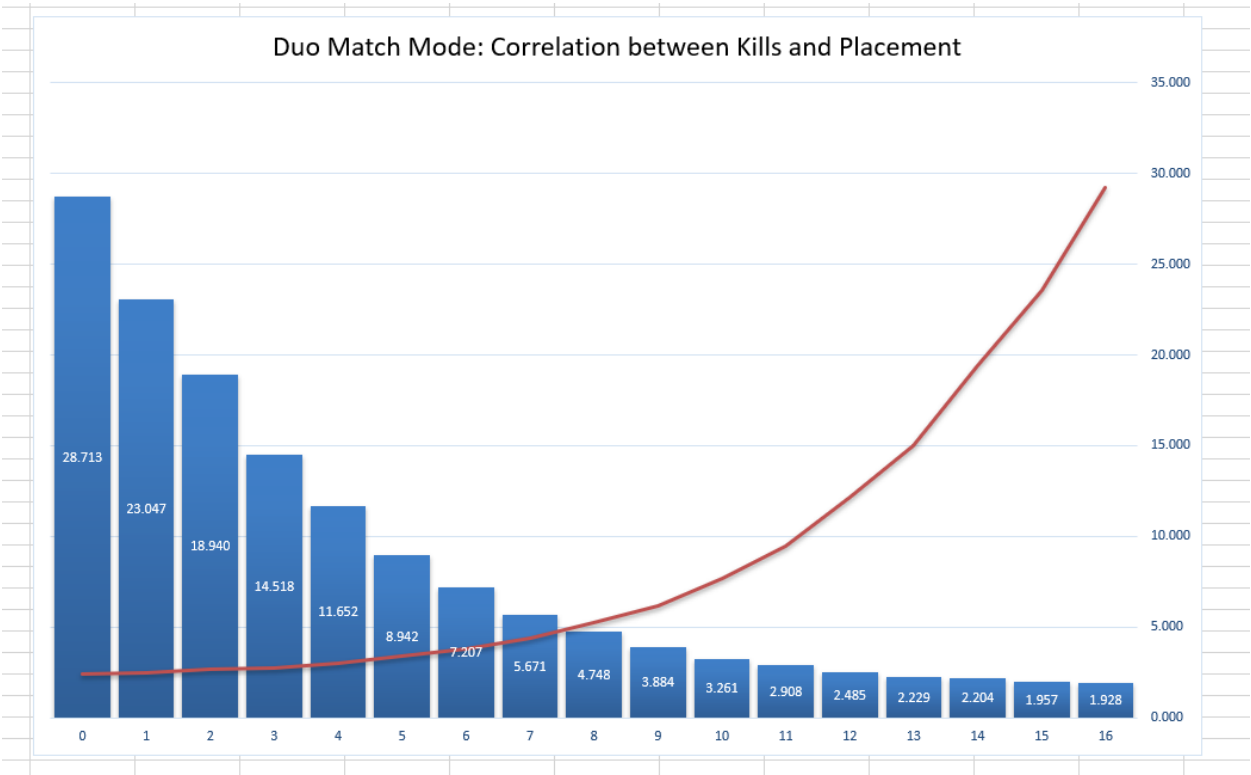
```
SELECT team_placement, avg(kills.totalkills)
FROM (SELECT match_id, team_placement, sum(player_kills) as totalkills
FROM team_442_pubg_match
where party_size = 1
group by match_id, team_placement
ORDER BY team_placement) as kills
GROUP BY team_placement
ORDER BY team_placement
LIMIT 30

```

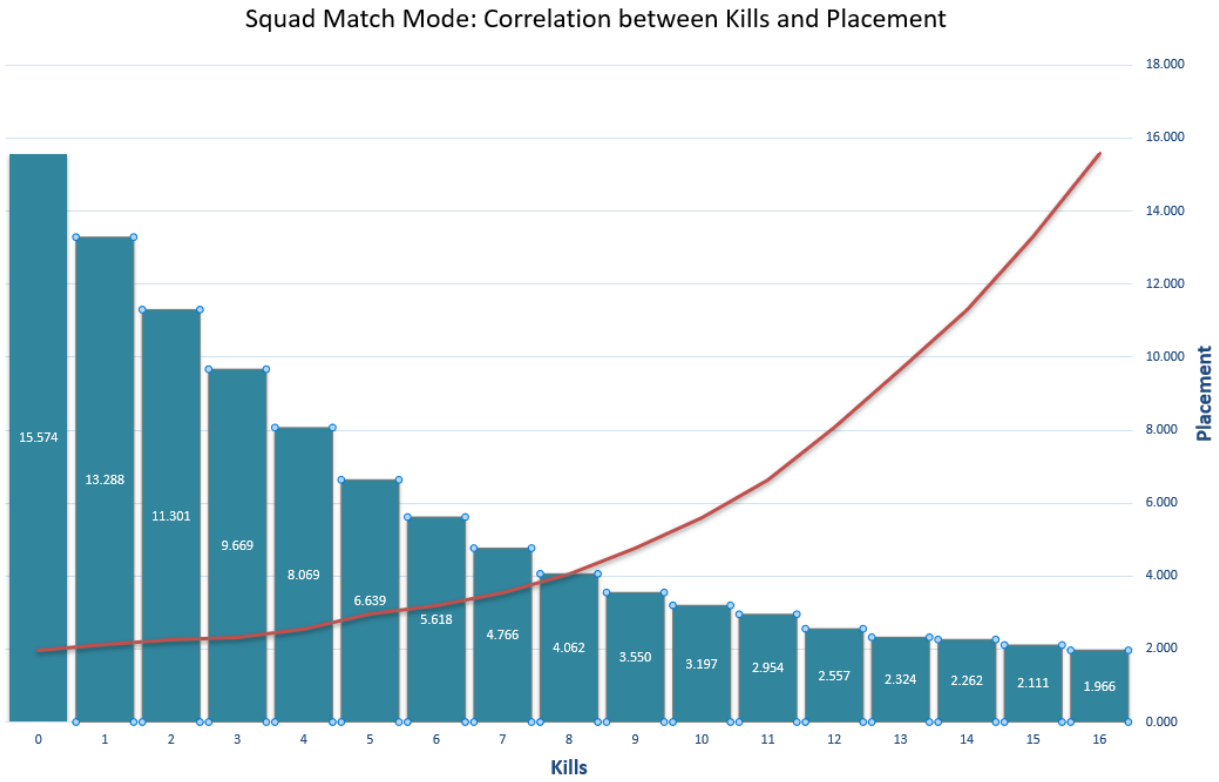
```
SELECT team_placement, avg(kills.totalkills)
FROM (SELECT match_id, team_placement, sum(player_kills) as totalkills
FROM team_442_pubg_match
where party_size = 2
group by match_id, team_placement
ORDER BY team_placement) as kills
GROUP BY team_placement
ORDER BY team_placement
LIMIT 30

```

```
SELECT team_placement, avg(kills.totalkills)
FROM (SELECT match_id, team_placement, sum(player_kills) as totalkills
FROM team_442_pubg_match
where party_size = 4 and team_placement != 0
group by match_id, team_placement
ORDER BY team_placement) as kills
GROUP BY team_placement
ORDER BY team_placement
LIMIT 30
```

Appendix F: Graphs of Correlation of Kills Count and Final Placement (party size 1,2,4)





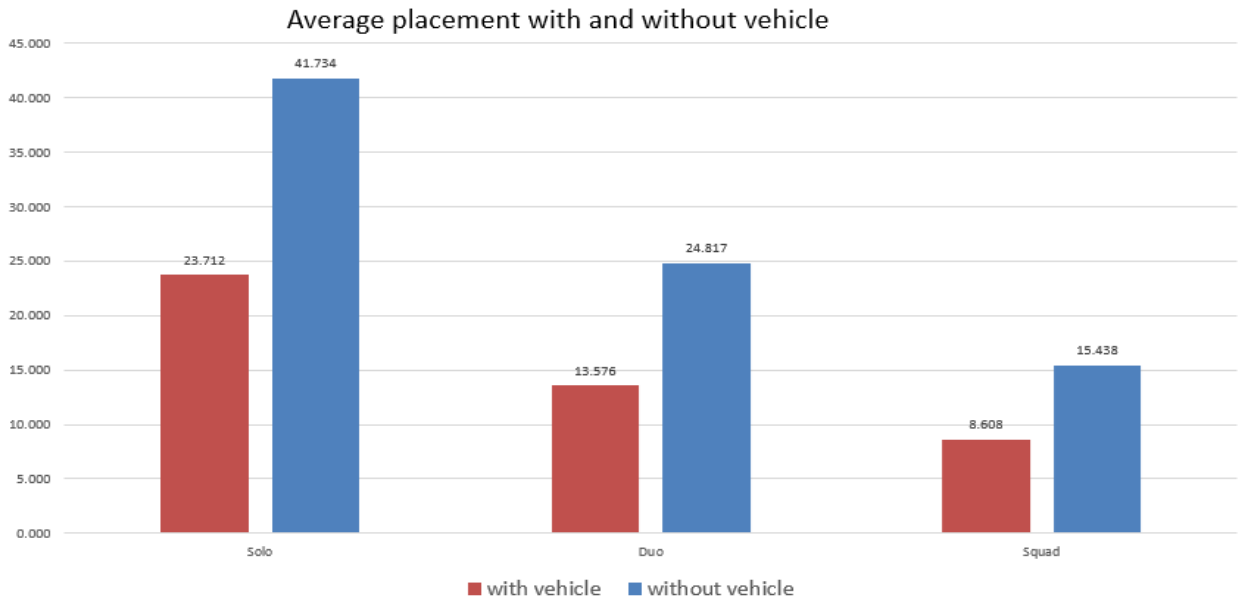

Squad Match Mode: Correlation between Kills and Placement
18.000
16.000
14.000
12.000
10.000
8.000
6.000
4.000
2.000
0.000
Placement
15.574
13.288
11.301
9.669
8.069
6.639
5.618
4.766
4.062
3.550
3.197
2.954
2.557
2.324
2.262
2.111
1.966
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
Kills

Appendix G: Codes for Correlation of Vehicle and Final Placement

```
SELECT withveh.party_size, withoutveh.average_rank_without_veh, withveh.average_rank_with_veh
FROM (SELECT party_size, avg(team_placement) as average_rank_without_veh
FROM team_442_pubg_match
WHERE player_dist_ride = 0 and player_survive_time>=300
GROUP BY party_size) as withoutveh
INNER JOIN
(SELECT party_size, avg(team_placement) as average_rank_with_veh
FROM team_442_pubg_match
WHERE player_dist_ride != 0 and player_survive_time>=300
GROUP BY party_size) as withveh
ON withoutveh.party_size = withveh.party_size
ORDER BY party_size
```

Query History Saved Queries Results (3)

	party_size	average_rank_without_veh	average_rank_with_veh
1	1	41.73356069941861	23.712145384275978
2	2	24.816864740106432	13.57601941866132
3	4	15.438092786599258	8.608481311536305



Appendix I: Codes and Graphs of Total Kills and Final Placement by Automatic Rifles

```
SELECT killed_by as weapon, count(killer_name)
FROM team_442_pubg
INNER JOIN
(SELECT DISTINCT match_id
FROM team_442_pubg_match
WHERE party_size = 1) as party
USING (match_id)
WHERE killed_by='M416' or killed_by='M16A4' or killed_by='SCAR-L' or killed_by='AKM'
GROUP BY killed_by
ORDER BY count(killer_name) DESC
```

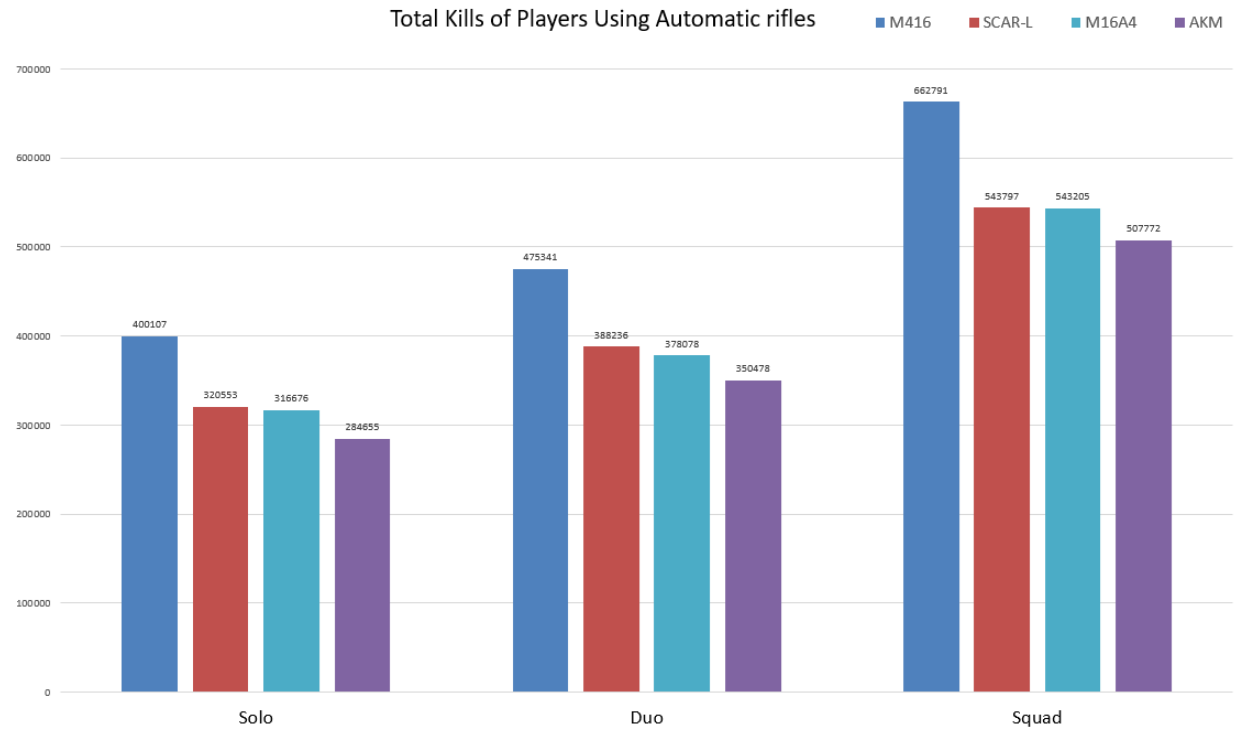
```
SELECT killed_by as weapon, count(killer_name)
FROM team_442_pubg
INNER JOIN
(SELECT DISTINCT match_id
FROM team_442_pubg_match
WHERE party_size = 2) as party
USING (match_id)
WHERE killed_by='M416' or killed_by='M16A4' or killed_by='SCAR-L' or killed_by='AKM'
GROUP BY killed_by
ORDER BY count(killer_name) DESC
```

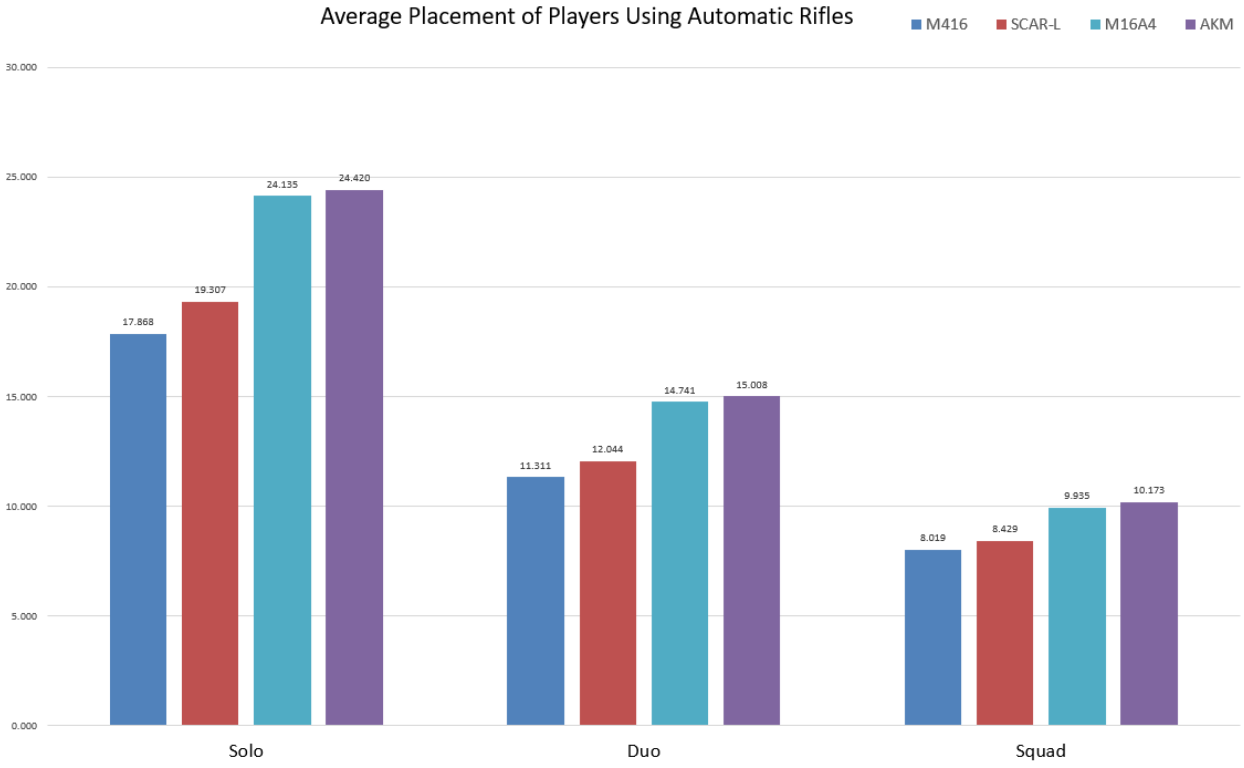
```
SELECT killed_by as weapon, count(killer_name)
FROM team_442_pubg
INNER JOIN
(SELECT DISTINCT match_id
FROM team_442_pubg_match
WHERE party_size = 4) as party
USING (match_id)
WHERE killed_by='M416' or killed_by='M16A4' or killed_by='SCAR-L' or killed_by='AKM'
GROUP BY killed_by
ORDER BY count(killer_name) DESC
```

```
SELECT killed_by as weapon, avg(killer_placement)
FROM team_442_pubg
INNER JOIN
(SELECT DISTINCT match_id
FROM team_442_pubg_match
WHERE party_size = 1) as party
USING (match_id)
WHERE killed_by='M416' or killed_by='M16A4' or killed_by='SCAR-L' or killed_by='AKM'
GROUP BY killed_by
ORDER BY avg(killer_placement)
```

```sql
SELECT killed_by as weapon, avg(killer_placement)
FROM team_442_pubg
INNER JOIN
(SELECT DISTINCT match_id
FROM team_442_pubg_match
WHERE party_size = 2) as party
USING (match_id)
WHERE killed_by='M416' or killed_by='M16A4' or killed_by='SCAR-L' or killed_by='AKM'
GROUP BY killed_by
ORDER BY avg(killer_placement)
```

```sql
SELECT killed_by as weapon, avg(killer_placement)
FROM team_442_pubg
INNER JOIN
(SELECT DISTINCT match_id
FROM team_442_pubg_match
WHERE party_size = 4) as party
USING (match_id)
WHERE killed_by='M416' or killed_by='M16A4' or killed_by='SCAR-L' or killed_by='AKM'
GROUP BY killed_by
ORDER BY avg(killer_placement)
```


Total Kills of Players Using Automatic rifles
M416
SCAR-L
M16A4
AKM
700000
600000
500000
400000
300000
200000
100000
0
400107
320553
316676
284655
475341
388236
378078
350478
662791
543797
543205
507772
Solo
Duo
Squad


Average Placement of Players Using Automatic Rifles
M416
SCAR-L
M16A4
AKM
30.000
25.000
20.000
15.000
10.000
5.000
0.000
17.868
19.307
24.135
24.420
11.311
12.044
14.741
15.008
8.019
8.429
9.935
10.173
Solo
Duo
Squad

Appendix J: Codes and Graphs of Total Kills and Final Placement by Sniper Rifles

```sql
SELECT killed_by as weapon, count(killer_name)
FROM team_442_pubg
INNER JOIN
(SELECT DISTINCT match_id
FROM team_442_pubg_match
WHERE party_size = 1) as party
USING (match_id)
WHERE killed_by='Kar98k' or killed_by='SKS' or killed_by='Mini 14'
GROUP BY killed_by
ORDER BY count(killer_name) DESC
```

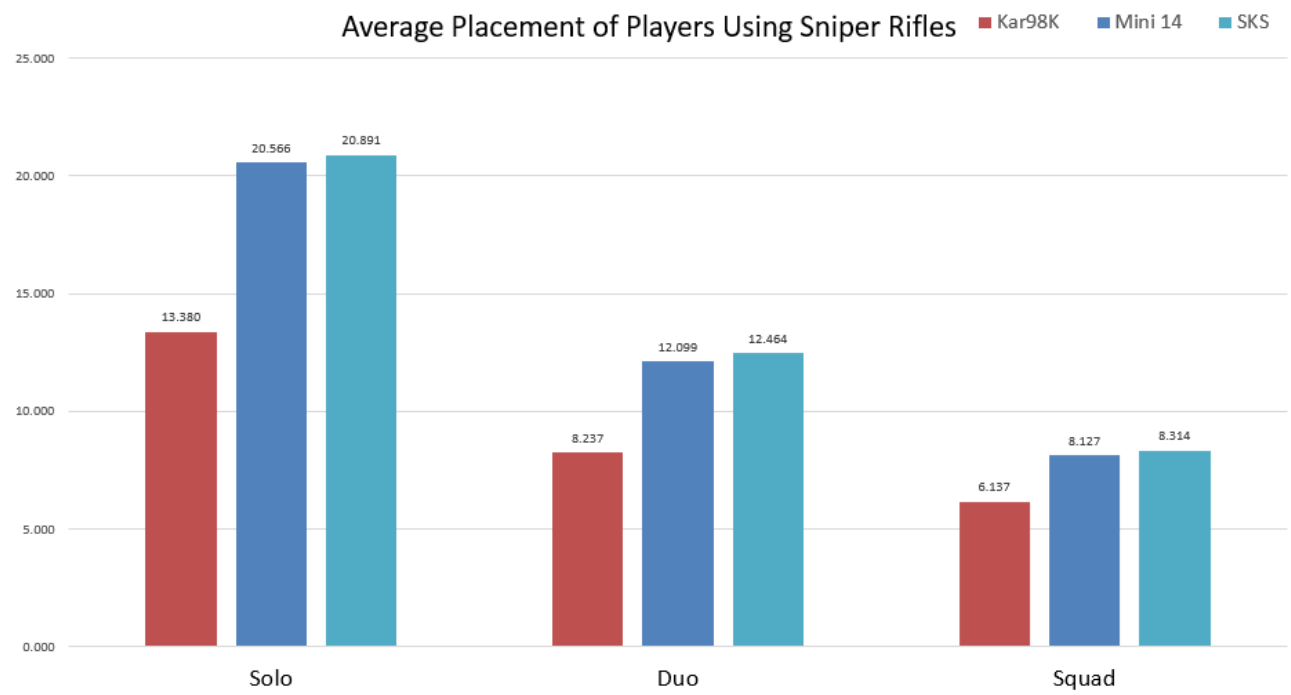
```sql
SELECT killed_by as weapon, count(killer_name)
FROM team_442_pubg
INNER JOIN
(SELECT DISTINCT match_id
FROM team_442_pubg_match
WHERE party_size = 2) as party
USING (match_id)
WHERE killed_by='Kar98k' or killed_by='SKS' or killed_by='Mini 14'
GROUP BY killed_by
ORDER BY count(killer_name) DESC
```

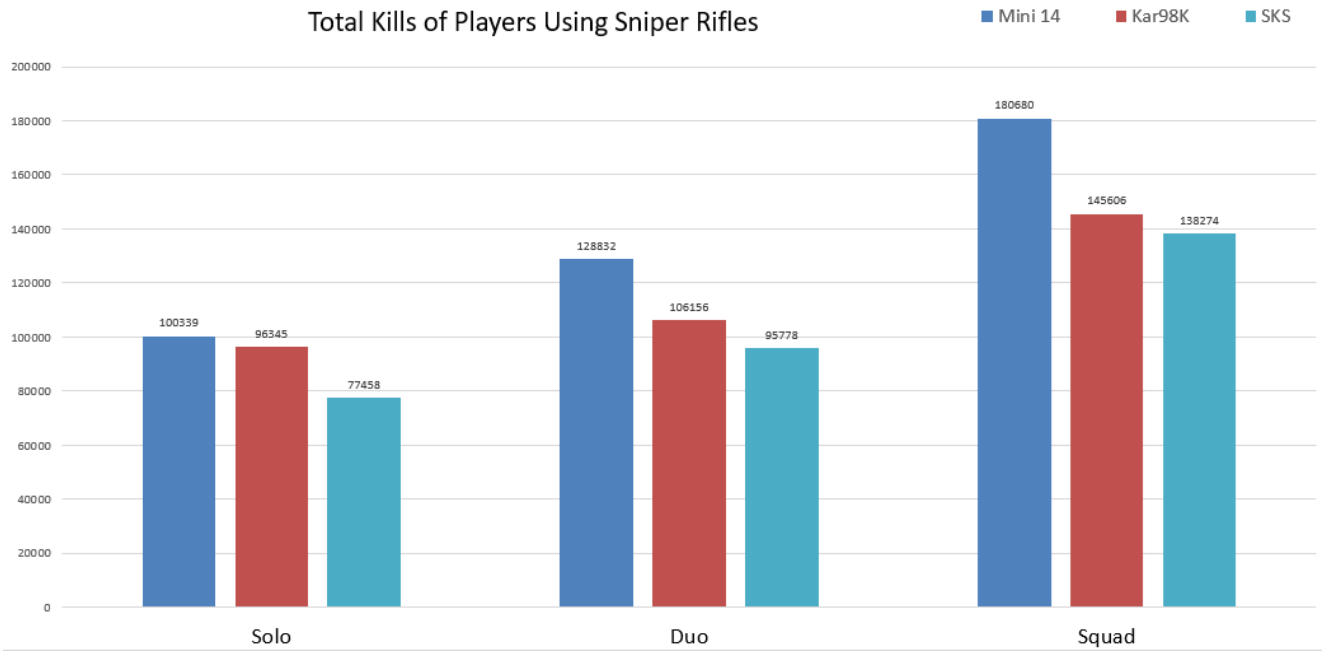
```sql
SELECT killed_by as weapon, count(killer_name)
FROM team_442_pubg
INNER JOIN
(SELECT DISTINCT match_id
FROM team_442_pubg_match
WHERE party_size = 4) as party
USING (match_id)
WHERE killed_by='Kar98k' or killed_by='SKS' or killed_by='Mini 14'
GROUP BY killed_by
ORDER BY count(killer_name) DESC
```

```sql
SELECT killed_by as weapon, avg(killer_placement)
FROM team_442_pubg
INNER JOIN
(SELECT DISTINCT match_id
FROM team_442_pubg_match
WHERE party_size = 1) as party
USING (match_id)
WHERE killed_by='Kar98k' or killed_by='SKS' or killed_by='Mini 14'
GROUP BY killed_by
ORDER BY avg(killer_placement)
```

```sql
SELECT killed_by as weapon, avg(killer_placement)
FROM team_442_pubg
INNER JOIN
(SELECT DISTINCT match_id
FROM team_442_pubg_match
WHERE party_size = 2) as party
USING (match_id)
WHERE killed_by='Kar98k' or killed_by='SKS' or killed_by='Mini 14'
GROUP BY killed_by
ORDER BY avg(killer_placement)
```

```sql
SELECT killed_by as weapon, avg(killer_placement)
FROM team_442_pubg
INNER JOIN
(SELECT DISTINCT match_id
FROM team_442_pubg_match
WHERE party_size = 4) as party
USING (match_id)
WHERE killed_by='Kar98k' or killed_by='SKS' or killed_by='Mini 14'
GROUP BY killed_by
ORDER BY avg(killer_placement)
```


Average Placement of Players Using Sniper Rifles
Kar98K
Mini 14
SKS
25.000
20.000
15.000
10.000
5.000
0.000
13.380
20.566
20.891
8.237
12.099
12.464
6.137
8.127
8.314
Solo
Duo
Squad


Total Kills of Players Using Sniper Rifles
Mini 14
Kar98K
SKS
200000
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
100339
96345
77458
128832
106156
95778
180680
145606
138274
Solo
Duo
Squad