ISM 6136

DATA MINING

PREDICTIVE ANALYSIS OF WIN PLACE POSITION PERCENTAGE PLAYER IN PUBG

MEMBERS:
PRATIK UTPAT
RUSHIKESH MAHESHWARI
SAMEERA PRASAD
SANJIVANI GADKARI

INTRODUCTION

Players Unknown Battleground (PUBG) is multiplayer online war-strategy game players start from their base camp. The accessible safe territory of the map diminishes in size after some time, guiding surviving players into more tightly regions to drive experiences. The last player or team standing wins the round.

Players can enter the match solo, or with a little team of up to four individuals. In either case, the last individual or team left alive wins the match. The game starts with releasing the players from the flight with the help of a parachute where the player/team land on the designated region. The area where the match is to be held tends to reduce accordingly towards a random location and if any player is outside the zone, gets eliminated for not being in the "safe" zone in that period of time.

Over the span of the match, random regions of the map are highlighted in red and bombed, representing a danger to players who stay here. The players are cautioned a couple of minutes before these occasions, giving them an opportunity to migrate to security.

OBJECTIVE

The objective of this analysis is to predict how likely a player is going to win the game. The response variable i.e. 'win_place_percentage' is a number ranging between 0 to 1, denotes the percentage chances of player surviving till the end of the game where 1 defines player has 100% chance to win the game (survives till the end).

We downloaded the data from Kaggle.com. It consisted of 25 predictor variables like kills, boosts, distance traveled by a player walking, riding a vehicle and swimming, match Duration etc..These variables were used to predict the winning percentage of a player. We used **R** for data cleaning and detailed descriptive analysis and used **SAS Enterprise Mine**r to build various predictive models.

DATA CLEANING

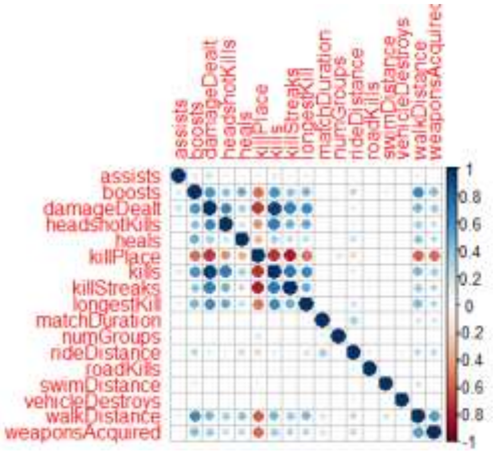
The data had more than 2 million observations.This contained data of players in 3 types of matches: Solo, Duo, and Quadro. We scoped down our analysis just for the 'Solo' category with close to 7 lac. Observations. We removed columns such as assists, DBNOs (knocked out), revives which do not carry any significance in 'Solo' game type. With the remaining data, we performed descriptive analysis using R. We found out certain columns has high correlation and used dimensionality reduction methods by simply processing some columns to have more qualitative information for analysis.

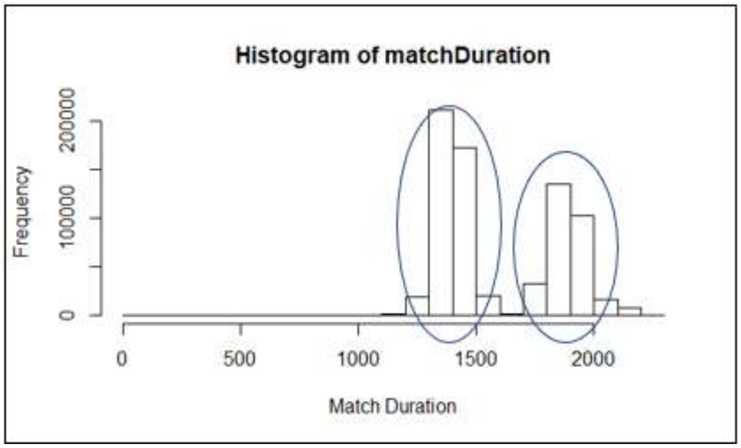
EXPLORATORY DATA ANALYSIS AND FEATURE ENGINEERING

With exploratory data analysis, we reduced the 25 attributes to 12 attributes to predict the WinPlacePerc. It transformed the raw data to the improved accuracy of the data. With the help of R programming, we reduced two parameters. We combined 'ride distance' and 'walk distance' covered by a player. The parameter swim distance was dropped because only 0.5% of the observations had values for this. We normalized ride distance in terms of walk distance of the player by taking the average of the ratios of (ride distance)/(walk distance) and created a new 'distance' variable which explains the both. On similar lines we combined attributes 'number of road kills' and 'number of headshot kills' into efficient kills i.e. 'Eff_kills'. Efficient kills are the percentage of players killed by a particular player with minimum resources. Hence we took a ratio of the sum of ‘Headshot kills’ and ‘Road Kills’ to total ‘kills’. This gave us more information about a player’s efficiency while killing which is again a new attribute which was not present in data. Thus we gained the following attributes.

1. The total distance of the player with the help of different modes of transportation.
2. The Efficient kills total number of kills the player killed the component with the help of different weapons.

Since we had planned to run a regression analysis, we checked the correlation of all the variables with respect to each other. Below were the results:





1(a) CorrPlot before feature reduction

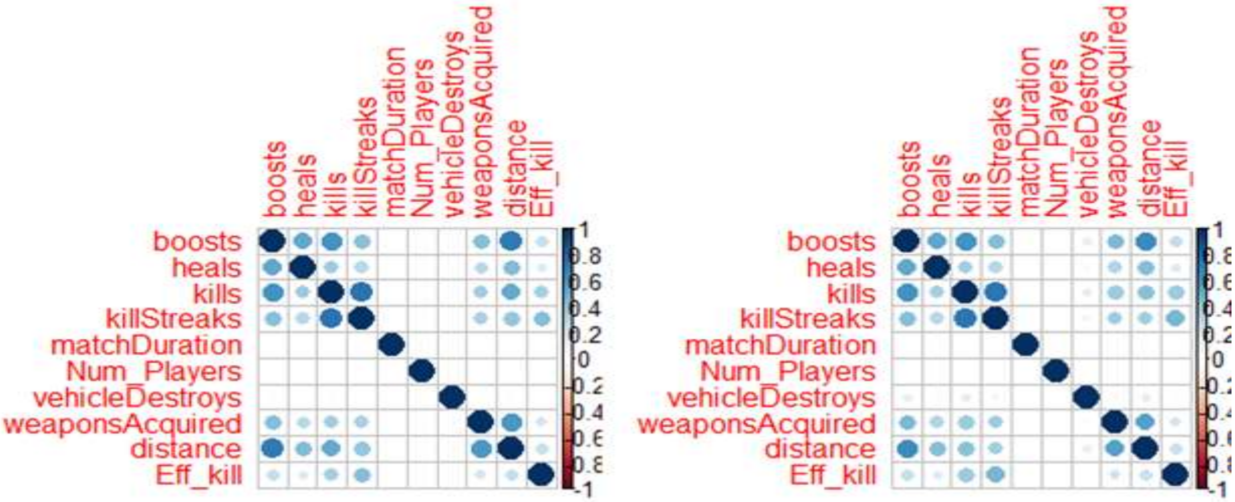
We dropped variables which had a high correlation with each other.

We also observed the distribution of match_duration. It gave us significant insight into the different map sizes which is not part of the current data set. Hence we included a new column ‘mapsize’ to analyze the WinPlacePerc for small(Sanhock) and large maps(Erangel)

Sanhok is a 4x4km arena Erangel which is 8x8km(*matchduration < 1600 seconds).*

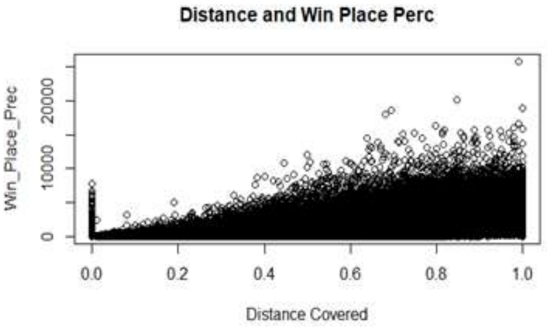
Different features would be required to predict our target WinPlacePerc of players depending upon the map. Hence, we have split the data set into two parts according to the map types.

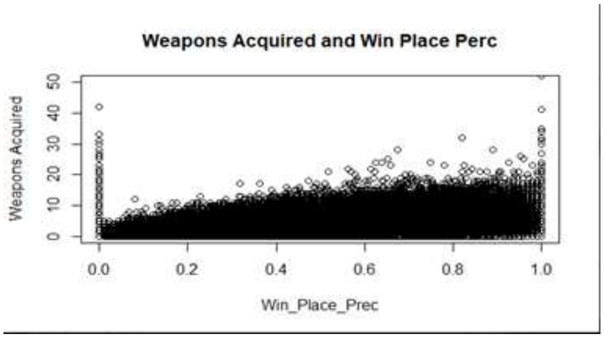
.

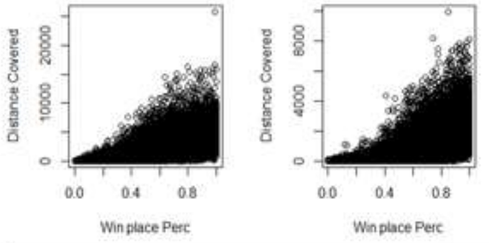


1(b)CorrPlot after feature reduction (Small map correction vs large map correlation)

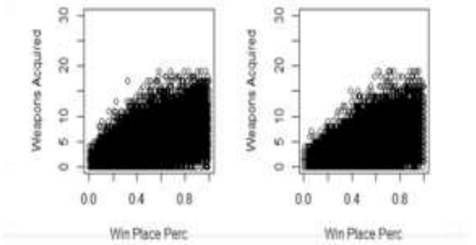
We can see in the figures that the relationship between distance covered, weapons acquired and WinPlacPerc is gradually increasing.







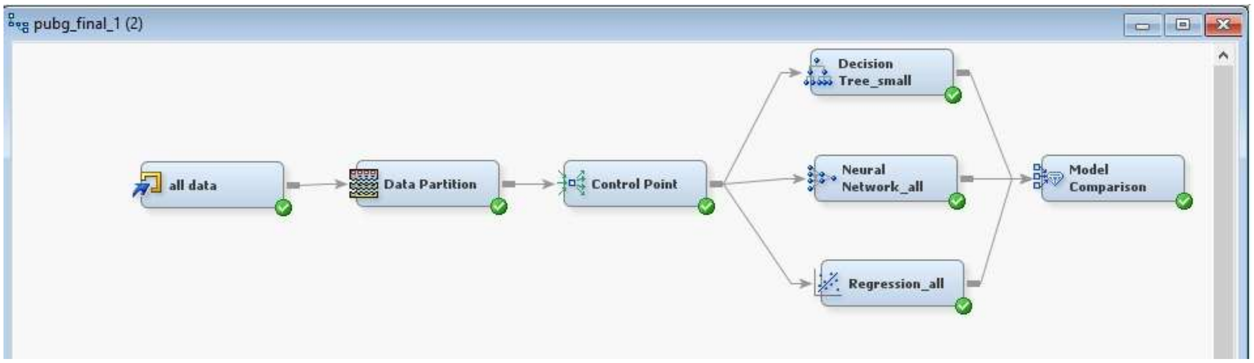
This depicts the win place prediction with respect to the distance covered by the player for small and large map



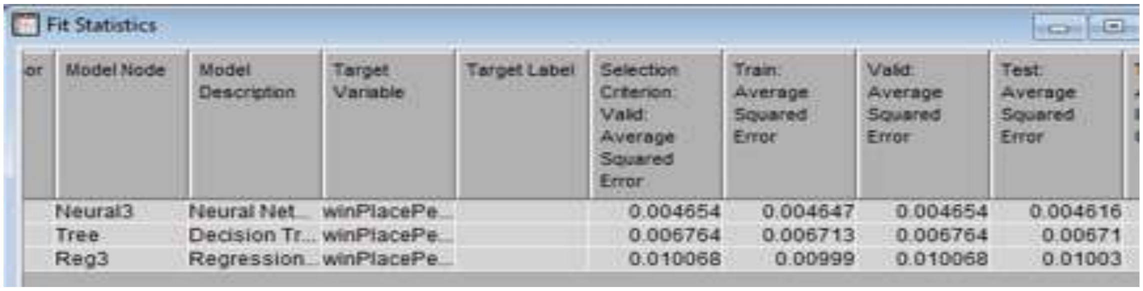
This depicts the win place prediction with respect to the weapons acquired covered by the player for small and large map

DATA MODELLING:

Post data preprocessing, we used different data mining algorithms such as Neural Networks, Regression, and Decision Tree to model the cleaned data. The we used ‘Averaged Squared Error’ as a decision making parameter for a better model in this case.



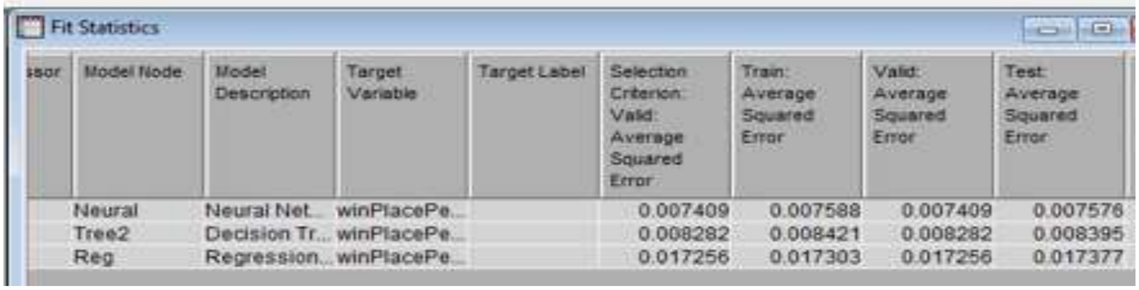
MODEL INTERPRETATION:



Fit Statistics

or	Model Node	Model Description	Target Variable	Target Label	Selection Criterion: Valid: Average Squared Error	Train: Average Squared Error	Valid: Average Squared Error	Test: Average Squared Error
	Neural3	Neural Net...	winPlacePe...		0.004654	0.004647	0.004654	0.004616
	Tree	Decision Tr...	winPlacePe...		0.006764	0.006713	0.006764	0.00671
	Reg3	Regression...	winPlacePe...		0.010068	0.00999	0.010068	0.01003

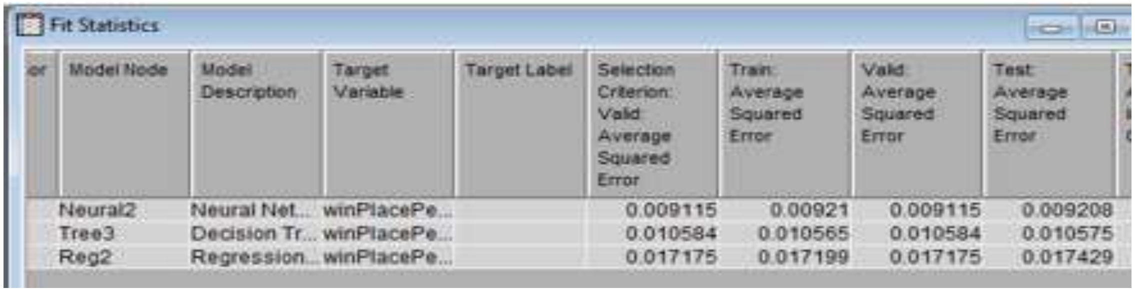
Model 1 : All variable data. Neural Netwrok > Decision Tree > Regression



Fit Statistics

sor	Model Node	Model Description	Target Variable	Target Label	Selection Criterion: Valid: Average Squared Error	Train: Average Squared Error	Valid: Average Squared Error	Test: Average Squared Error
	Neural	Neural Net...	winPlacePe...		0.007409	0.007588	0.007409	0.007576
	Tree2	Decision Tr...	winPlacePe...		0.008282	0.008421	0.008282	0.008395
	Reg	Regression...	winPlacePe...		0.017256	0.017303	0.017256	0.017377

Model 2 : Small Map data Neural Netwrok > Decision Tree > Regression



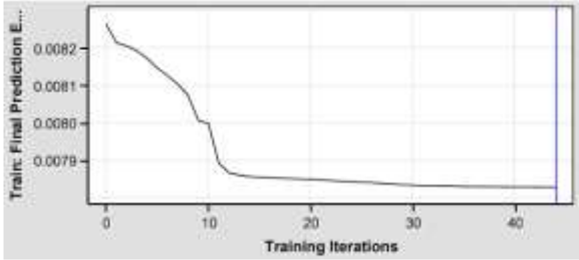
Fit Statistics

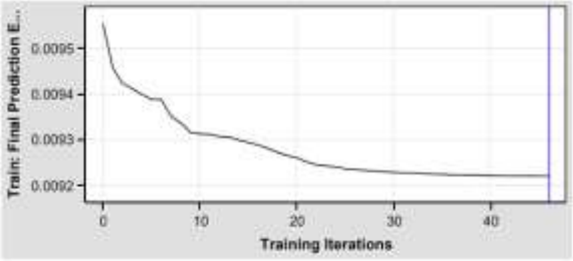
or	Model Node	Model Description	Target Variable	Target Label	Selection Criterion: Valid: Average Squared Error	Train: Average Squared Error	Valid: Average Squared Error	Test: Average Squared Error
	Neural2	Neural Net...	winPlacePe...		0.009115	0.00921	0.009115	0.009208
	Tree3	Decision Tr...	winPlacePe...		0.010584	0.010565	0.010584	0.010575
	Reg2	Regression...	winPlacePe...		0.017175	0.017199	0.017175	0.017429

Model 3 : Large Map data Neural Netwrok > Decision Tree > Regression

1.Neural Networks:

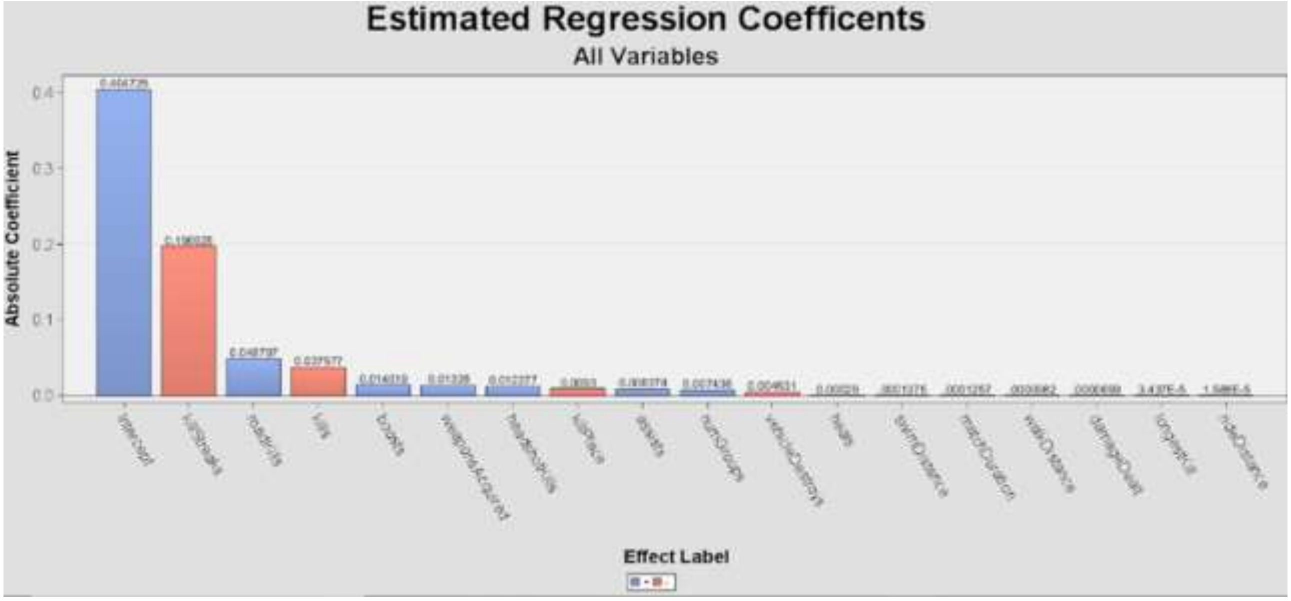
Below is the final prediction error graph. As observed that with neural networks algorithm the prediction error of the model has increased pre and post segregating the data. Although it seems like errors have increased in the case where the data is seperated as large and small map data, the results in the case of the large map data are more realistic, this can be supported with the coefficient values and worth obtained from linear regression model



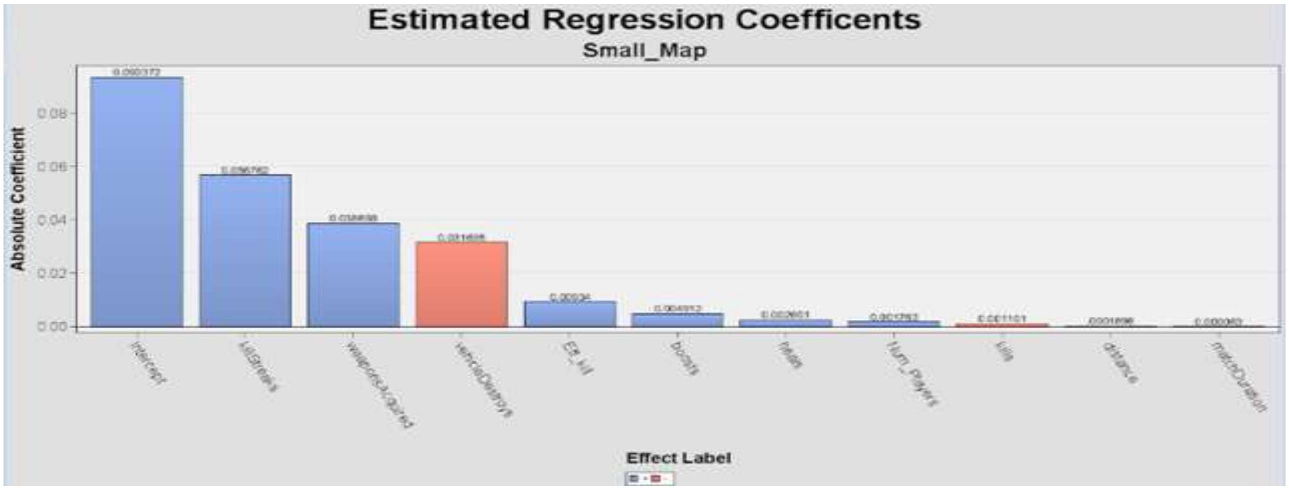


Model 1: Final error preidction : All Variable data

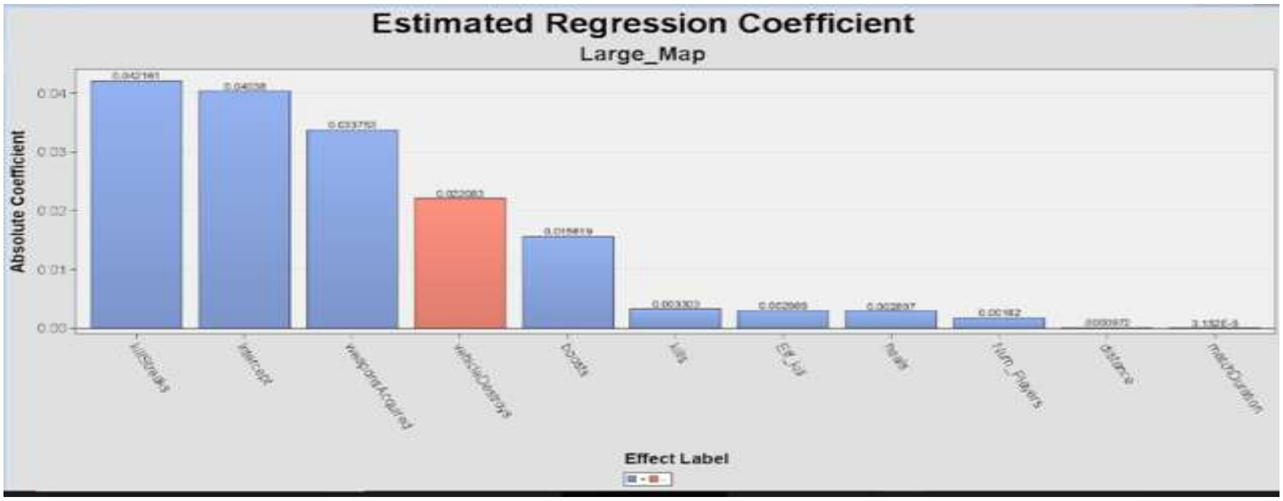
Model 2: Final Error prediction :Large Map data



Here the variables like ‘kills’ and ‘KillStreak’ has a negative effect on the winning rank which isn’t sound logical as more kills can eliminate the competition for you and you can win.



However, when separate the data in small and large map, the analysis makes sense. Here the kill coefficient is negative but the intensity is reduced. This can be supported by the reason that since 100 players are in 4x4 km map size These are the coefficients we found for Small Map which has a high amount of Vehicle Destroy. However in the case of a large map, the kill and kill streak coefficients are both positive. With this, we can infer that the strategy for both the map is different and hence this analysis is more realistic.



INFERENCE:

With the analysis we performed and results which we got, we have come up with few recommendations for both the payers who play PUBG as well as to developers of the game to make it more interesting and challenging.

So for the players, based on our analysis, it seems like the number of kills has a negative impact on players winning chances in small size map. However, in the large map, kill has got a positive effect on players ranking. In both the game set, having acquired more weapons, boost material and heals increases players winning chances. Also, the parameter which we defined which is efficient kills is also having a positive impact on final rankings. With this, we can infer that the strategy for smaller map and larger map in different for the players.

On the business side, the developers can utilize this analysis to cluster the players with their respective win rank prediction and put them under one rank category. So this change will make sure that every player has a fair chance of winning in their own category and can be promoted to a higher rank on consistent performance. Also, the developers can create a real-time rank prediction in the game as the player progresses in the match.

This analysis can be extended to due and Quadro game types so that we can find more interesting insights and strategies for the game.

FUTURE SCOPE:

In this case, we have focused on the solo player matches i.e. we have not considered duo(2 player teams) and Quadro matches(4 player teams).

In the future, we can extend the analysis for these 2 modes where we can find the chances of winning for a team.
This will help the company build their prediction on chances of winning on basis of the data obtained.
Players get habitual to the map on which they play. This helps the player to track down the safe houses, and weapons to kill the opposite competent. If the game developers introduce different maps, this will change the level of strategy of the players. This will again create a newer scenario for the multi-players in the game and the chances of prediction of winning may change.

Annexure:

1. Data Dictionary:

Variables	Definition
Id	ID to identify match. There are no matches that are in both the training and testing set.
Boosts	Number of boost items collected
Heals	Number of healing items collected.

Kills	Number of a total of kills by the player
killStreaks	Max number of enemy players killed in a short amount of time.
MatchDuration	Duration of match in seconds.
Num_Play	Number of players in the match
Vehicle Destroy	Vehicles destroyed during the game
Weapons Acquired	Number of weapons picked up
distance	Total distance traveled by the player.
Eff_kill	Efficient kills are kills with minimum resources.
Mapsize	Size of the map according to the game.

Analysis Result:

Large Map

```
Assessment Score Distribution

Data Role=VALIDATION Target Variable=winPlacePerc Target Label=' '

  Range for        Mean       Mean       Number of       Model

  Predicted        Target    Predicted   Observations     Score

1.088 -  1.161     0.91601     1.11440        10        1.12476
```

1.015 - 1.088	0.91011	1.04221	87	1.05144
0.941 - 1.015	0.95422	0.95667	7672	0.97813
0.868 - 0.941	0.90369	0.90253	40135	0.90482
0.795 - 0.868	0.83398	0.83292	42105	0.83151
0.722 - 0.795	0.75774	0.75953	33371	0.75819
0.648 - 0.722	0.68362	0.68544	27647	0.68488
0.575 - 0.648	0.61311	0.61161	26375	0.61157
0.502 - 0.575	0.54286	0.53777	27685	0.53826
0.428 - 0.502	0.47369	0.46437	30656	0.46495
0.355 - 0.428	0.39997	0.39111	33432	0.39163
0.282 - 0.355	0.31689	0.31777	35988	0.31832
0.208 - 0.282	0.23359	0.24609	34960	0.24501
0.135 - 0.208	0.16776	0.17263	29568	0.17170
0.062 - 0.135	0.09137	0.09786	28889	0.09838
-0.012 - 0.062	0.02661	0.02661	29353	0.02507
-0.085 - -0.012	0.00306	-0.02314	3251	-0.04824
-0.158 - -0.085	0.18213	-0.10974	23	-0.12155
-0.232 - -0.158	0.21370	-0.19493	11	-0.19487
-0.305 - -0.232	0.39548	-0.28068	4	-0.26818

Small Map

Data Role=VALIDATE Target Variable=winPlacePerc Target Label=' '

Range for	Mean	Mean	Number of	Model
Predicted	Target	Predicted	Observations	Score
0.907 - 0.953	0.93036	0.93092	4886	0.92977
0.860 - 0.907	0.87874	0.88427	3608	0.88339
0.814 - 0.860	0.83693	0.83488	3964	0.83701

0.767 -	0.814	0.79155	0.79239	4008	0.79062
0.721 -	0.767	0.74925	0.74417	3050	0.74424
0.675 -	0.721	0.70637	0.69725	3262	0.69786
0.628 -	0.675	0.65408	0.65130	3497	0.65148
0.582 -	0.628	0.59834	0.60574	3407	0.60509
0.536 -	0.582	0.55015	0.55976	3141	0.55871
0.489 -	0.536	0.49893	0.51277	2840	0.51233
0.443 -	0.489	0.45629	0.46615	2771	0.46595
0.396 -	0.443	0.41901	0.41971	2841	0.41956
0.350 -	0.396	0.37956	0.37346	2943	0.37318
0.304 -	0.350	0.33739	0.32636	2980	0.32680
0.257 -	0.304	0.29334	0.28018	3294	0.28042
0.211 -	0.257	0.24613	0.23344	3481	0.23403
0.164 -	0.211	0.19021	0.18693	4020	0.18765
0.118 -	0.164	0.12890	0.14099	4756	0.14127
0.072 -	0.118	0.07404	0.09597	4854	0.09489
0.025 -	0.072	0.07342	0.04915	2398	0.04851