I Wanna Be the Very Best: A Pokemon Go Data Analysis Project

Katie A. Barnhart[1]
[1]*Department of Physics and Astronomy, Texas Tech University, Lubbock, TX 79409-1051, USA*

ABSTRACT

The mobile game Pokemon Go has several aspects to engage players. Via tracking player statistics over time, it was possible to track trends and make predictions for each players. Data was collected, and linear regression was used to obtain average catch per day, battles won per day, and distance traveled per day for a sample of 312 players. The regression performed reasonably well, with RMSE of 537.7, which led to predicted totals that deviated from measured totals by only about 5% in the most severe cases. Then, the scikit-learn K-Means clustering algorithm was used to test the hypothesis that active players fall into four different categories, which is how players tend to sort themselves. The data did not support this hypothesis, and instead showed the optimum number of categories was three, however, outliers in the dataset may have influenced this conclusion. Players also exhibit one of two broad play styles, those that tend to prefer battling and those that tend to prefer collecting. The scikit-learn implementation of the classification algorithms Gaussian Naive Bayes and Quadratic Discriminant Analysis were used to test the viability of sorting players into one of the two categories. Quadratic Discriminant Analysis performed better in this regard, with fewer incorrect classifications on a test dataset.

1. INTRODUCTION

Pokemon has been a cultural phenomenon for more than a quarter of a century, and has given rise to games across numerous generations of gaming consoles and includes many formats, including Nintendo DS games, Switch games, trading card games, and mobile games[1]. Pokemon Go is an augmented reality mobile game where the player can catch creatures called Pokemon, battle raid bosses and other players, complete in-game quests, and more. Most if not all aspects of the game strongly encourage the player to walk around in order to play.

The game has a Friends List feature, and a limited amount of information about other player's stats is available via the Friends List. These include the level, total lifetime catches, total lifetime distance traversed, and total lifetime battles won of each friend. Active players, hereafter called trainers in the Pokemon tradition, informally sort themselves into four categories based on in-game activity level from least to most active. For example, the least active trainers are called "casual" and may only play once in a while, and the most active are called "hardcore" and play many hours every day. The machine learning classification method K-means clustering was used to test this hypothesis based on the three features in the data.

Two of the core mechanics of the game are catching Pokemon and battling, and have been features of the Pokemon franchise since its inception. Trainers venture out into the tall grass, catch Pokemon, train with them to increase their strength, and battle against other trainers and in-game non-player-character adversaries. However, not every trainer will interact with all aspects of the game equally. There are additional statistics that each player can access about themself. If sufficient cooperation were to occur, it might be possible to analyze additional aspects of the game and thereby sort players into more detailed categories. However, the dataset was limited to that information that could be used to sort trainers either into the category of "battler" or "collector." Machine learning discriminative classification algorithms were used to investigate whether it is possible to sort players based on whether they prefer battling or collecting.

In section 2, the data collection and analysis methods are discussed. In section 3, the results of the analysis are presented and limitations are highlighted. In section 4 the concluding remarks can be found. Sections 5 and 6 are the appendices and sources cited.

2. METHODS

The data was collected five times over a period of approximately 50 calendar days. The author collected data by hand from each of the 311 trainers on her friends list as well as from herself. Collected data included trainer level, team, total distance walked, Pokemon caught, and battles won, as well as screen name for each trainer.

A note is germane. Pokemon Go includes a mechanic by which the trainer chooses to join one of three teams extremely early on in their gameplay. The three choices are Team Mystic, Team Valor, or Team Instinct. A trainer can choose to change their team at most once per year. Certain features of the game are inaccessible to trainers who have not chosen a team, but no trainer on the author's friends list had been in that position for at least three years prior to the start of data collection. The choice of which team does not otherwise restrict a trainer from accessing or engaging with any aspect of the game. Any relationships between team choice and other trainer stats were not examined in the course of this work, but could be the subject of a future study.

Pokemon Go has several types of battles. The type of battle (raid, gym, or PVP) was not available, as the lifetime battles won statistic available via the friends list does not include information about the type of battle. The data was collected manually and then loaded into a jupyter notebook, where the analysis took place.

The dataset had only three useful features, hence no dimensionality reduction techniques were necessary prior to further analysis. Pre-processing of the data was further simplified due to the lack of measurement error associated with the data, and hence the lack of a need to quantify it. The Scikit-learn method LinearRegression[2] is a type of supervised machine learning in which labeled data are used to train an algorithm to predict, for example, future values, by minimizing least squares. Using the numbers from the first four data collections, linear regression was used to predict a value for the fifth collection day. A value was predicted for Pokemon caught, battles won, and distance traveled for each player, for a total of 936 predictions. These predicted values were then compared to the measured values via the Scikit-learn root_mean_squared_error method[3].

In order to proceed with the activity level classification, inactive trainers were removed from the dataset. Since catching a Pokemon is the easiest and least time consuming task in the game, inactive trainers were defined as those who's catch total remained unchanged during the course of data collection. Once the data had been filtered, 203 active trainers remained in the dataset. The number of catches per day, battles won per day, and distance travelled per day output by the linear regression method above were used going forward in order to examine trainer classification.

K-Means clustering is a way to sort data into two or more categories via unsupervised machine learning, and works by minimizing the squared difference between each member of a category and the mean of that category[4]. The method is particularly well suited to data that clusters in spheres. Since the number of categories was being investigated and the data was unlabeled in this regard, Scikit-learn's KMeans function was used. Since K-means clustering requires normalized data[4], the StandardScaler method available from Scikit-learn was used to ensure proper normalization before proceeding.

K-Means was iterated over 2 to 11 clusters, and then the silhouette score method and the elbow method were used to determine the optimal number of of clusters, ie categories. The elbow method calculates the inertia for each number of clusters, which is the sum of the distances to the nearest cluster squared. There is a noticeable change in slope of the plot where the optimal number of clusters is located[5]. The Silhouette score is a measure of how well each data point fits into its assigned cluster, and ranges from negative one indicating a possible misclassification to positive one indicating a close fit. This number is highest for the optimal number of clusters.

In order to classify trainers into either the collector category or the battler category, the discriminative classification methods Gaussian Naive Bayes (GNB) classification and Quadratic Discriminant Analysis (QDA) classification via Scikit-learn were used. The data were labeled for this analysis; GNB and QDA are supervised algorithms. Based on experience with the game and it's players, a ratio of less than 15 catches per battle was used to label a battler, and more than that was a collector.

The filtered data was split into a training set consisting of 80% of the trainers and a testing set consisting of 20% of the trainers using the train_test_split method. It was ensured that the relative number of battlers vs collectors was the same for the test set and the training set. Then a GNB model and a QDA model were trained, and confusion matrices were computed for each model. The confusion matrix details the number of True Negatives (battlers correctly classified as battlers), False Negatives (collectors incorrectly identified as battlers), False Positives (battlers incorrectly identified as collectors), and True Positives (collectors correctly identified as collectors). Since Battlers are rarer than collectors, a low False Positive score is desirable.

3. RESULTS and DISCUSSION

The linear regression yielded a root mean squared error (RMSE) of 537.7 for the number of lifetime catches. While that may seem alarming, the measure was increased by a few large outliers that occurred for trainers with a reasonably high activity level. Lifetime catch totals tend to be quite significant, with the majority of active trainers falling roughly between 50,000 and 500,000 lifetime catches. The largest individual error, which was 4445, occurred for a trainer with a lifetime catch total of 161,265 catches. This represents an error of only 2.75%. In order to rigorously confirm the reasonableness of fit, a calculation of such percentages could be done for each trainer and statistic. However it is quite easy, based on prior experience, to start up Pokemon Go and accumulate over 100 catches in less than an hour, so the linear regression provided a more than adequate fit for lifetime catches without needing to compute such percents. The present study lacks exceedingly stringent requirements for fits.

The linear regression yielded an RMSE of 35.3 for lifetime battle wins and 79.0km for lifetime distance travelled. Those two lifetime totals tend to be smaller than the catch total, but still range in the many thousands, again pointing to a reasonable fit. Using a linear regression instead of merely estimating the three statistics per day by means of a simple slope-intercept model ensured a more accurate fit.

The K-Means classification pointed toward three categories as the optimal number (see figs. 1 and 2), so the initial hypothesis of four categories was not supported by the data. This might be because there was not much evidence to support separating the two medium intensity categories from each other, and instead the data pointed toward low, medium, and high levels of activity. There is one cluster that comprised relatively few trainers, which was surprising. If a data set involving more trainers was available, performing K-Means on such an increased data set may elucidate this aspect. Additionally, the data collected was not representative of the global active trainer population, so may not be adequate to estimate global trainer categories.

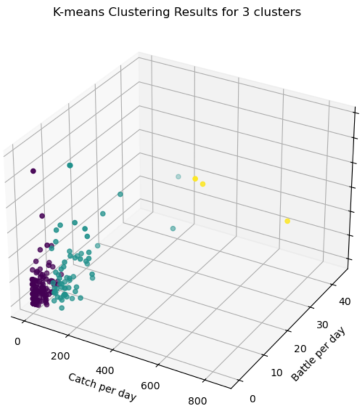

Fig. 1. A plot of the results of the KMeans clustering for the optimal k=3

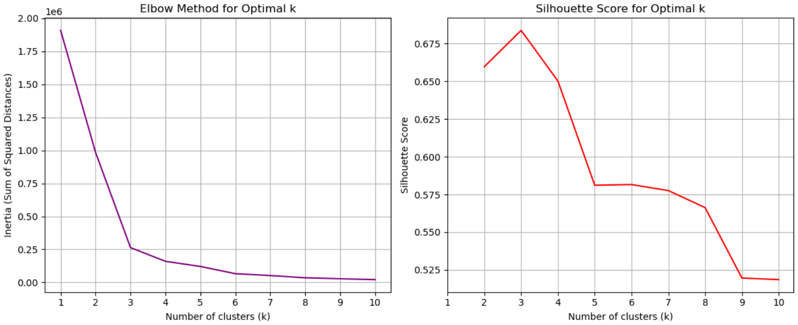

Fig. 2. Elbow Plot and Silhouette Score Plot. The marked decrease in slope at k=3 in the elbow plot is readily seen, and the maximum of the silhouette scores is in agreement with k=3

The maximum silhouette score was less that 0.70 and occurred for three clusters. This could indicate that the data may not have clustered extremely well, and that some other means of classification may be more suited to these data. K-Means classification is dependent on the data clustering spherically, so certain distributions are entirely unsuited to it. While initial visualizations of this data set did not point toward that condition, time constraints prevented further exploration of this topic. But, other methods could be explored in the future in another project.

Concerning discriminative classification, both of the methods tested performed well, with GNB misclassifying four of the 41 test points and QDA misclassifying only one (See Table 1). GNB had a False Positive rate of three while QDA had a False Positive rate of one. QDA was the superior method of classification (see Fig. 3). In future work, it would be advisable to calculate ROC curves or rather completeness versus efficiency curves for the two methods for further comparison, as this could help compare the methods given that one classification was more common than the other, and highlight potential weaknesses. In addition, other methods of discriminative classification such as decision trees, Gaussian mixture, and k-nearest-neighbor classification algorithms could also be used and compared with the initial two that were tested in this work.

GNB		QDA	
1	3	3	1
1	36	0	37

Table 1. Confusion matrices for both GNB and QDA detailing correct classifications as well as errors.

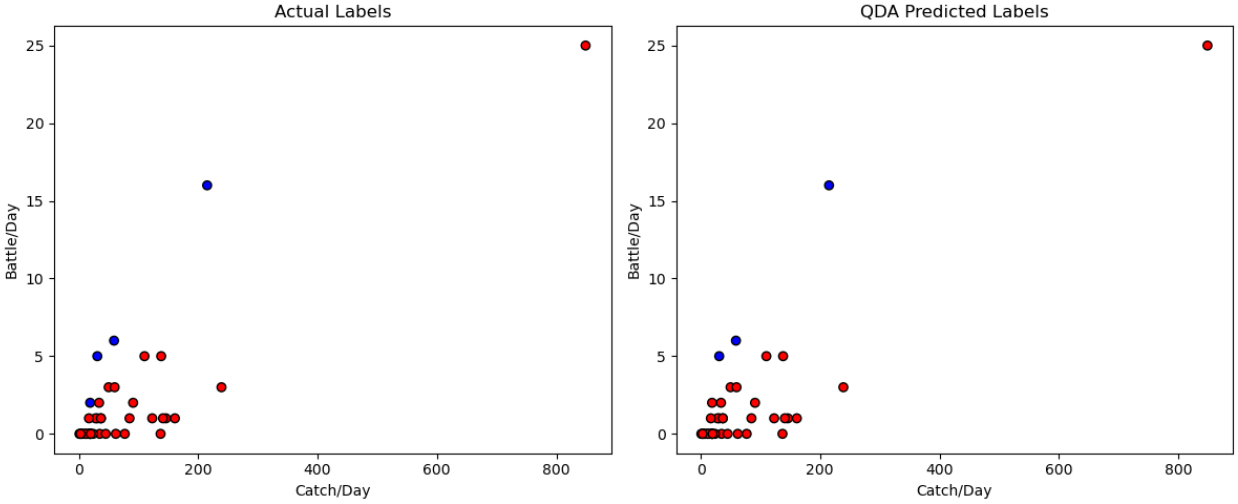

Fig. 3: Plots of the test dataset. Battlers are in blue and Collectors are in red. The left plot shows the true labels, and the right shows the labels that QDA predicted, with fewer errors than GNB.

Additionally a logistic regression could be tested as a means to find an optimized way to label battlers versus collectors, and then classification studies could be repeated. Again, a larger data set would be beneficial in this pursuit.

Some sections of the code were modified or generated by AI, specifically by ChatGPT. No sections of code that were generated or modified by ChatGPT were implemented before they were thoroughly vetted by the author.

4. CONCLUSION

Linear regression produced good fits, and predicted future activity well for most trainers with a few outliers with poorer but acceptable fits. The output was able to be used in classification algorithms going forward. K-Means was used to test the hypothesis that trainers fall into one or four categories based on activity level, and the data instead supported three categories supported by an Elbow curve and a silhouette score test. The data was not perfectly suited to K-means; there could be a more efficient classification algorithm that has yet to be explored. GNB and QDA discriminatory classifications were used to test the sorting of trainers into either the battler category or the collector category, and QDA was superior for this dataset as indicated by fewer type I and type II errors indicated in the confusion matrix. Further exploration of such classification algorithms would be an excellent topic for future work.

5. APPENDIX

In this appendix, a link to the github page the Jupyter notebook used in this project can be found. The notebook: https://github.com/katbarnh/ASTR_3300_S2025/blob/main/coursework/student_folders/katie_barnhart/Final_project/AstroStats_Project_FINAL.ipynb

6. REFERENCES

1. "Pokémon Video Games." Pokémon Video Games | Pokemon.Com, The Pokemon Company, www.pokemon.com/us/pokemon-video-games/all-pokemon-games. Accessed 9 May 2025.
2. Pol, Nihan. "Regression: I." Astr 3300 Spring 2025.

3. "3.4. Metrics and Scoring: quantifying the quality of predictions." scikit-learn developers, scikit-learn, https://scikit-learn.org/stable/modules/model_evaluation.html. Accessed 9 May 2025.
4. Pol, Nihan. "Density Estimation & Clustering." Astr 3300 Spring 2025.
5. "Elbow Method for optimal value of k in KMeans." geeksforgeeks.org, https://www.geeksforgeeks.org/elbow-method-for-optimal-value-of-k-in-kmeans/. 2 April 2025, Accessed 4 May 2025.