NYU | TANDON SCHOOL OF ENGINEERING

Analysis on Catch Rate for Pokémon Go

Zhixia Zhang (zz2445)
Lingfeng Zhao (lz1973)
Chen Cui (cc5824)

Contents

1.Abstract

Pokémon Go players may be interested in the Pokémon catch rate. The topic of our project is using some machine learning techniques to analyze Pokemon properties and predict the catch rates accurately for the huge number of Pokémon in the game.

2.Introduction

Pokémon is media franchise that began as a pair of Role Playing Games that were developed by Game Freak and published by Nintendo, in 1996 and the recently released augmented reality game Pokémon Go!, were caught on in 2016. Pokémon are fictitious animal-like monsters that live in the Pokémon world. Pokémon like fighting with each other, and they usually fight according to their (human) trainers' orders. Because a big number of Pokémon have been introduced throughout these years -seven generations of Pokémon with the order of 100 of Pokémon in each of them and each Pokémon is described with a big number of variables, it is attractive and meaningful to do some statistical analysis.

Our data source comes from a dataset from Kaggle that provides 721 entries of Pokemon and all relevant data about their weaknesses, stats[1], height, weight, etc.. we started off by simply looking at the data, making some graphs and trying to think of what interesting information we could pull from it. we statically analyzed the wide variety of variables used to describe the Pokémon, and there is a chance to to explore the relationships between them. After getting some insights, we tried different methods to predict the catch rate for a specific Pokémon. Finally we compared the performance of different methods for this task.

First we will introduce the variables and instances of the dataset in Section 3. We will explore the variables and their potential dependencies with some data visualizations in Section 4. Once we have studied the variables, we will try to make some predictions of the catch rate using the selected variables in Section 5, and with the summations and conclusions in Section 6.

3.The dataset and its variables

Our dataset is an expanded version of a dataset from kaggle named “721 Pokemon with stats”. The original dataset includes 721 Pokemon, with 13 variables per Pokémon including their number, name, first and second type, and basic stats: HP, Attack, Defense, Special Attack, Special Defense, and Speed that mainly define their ability to fight. To include as many variables as possible, we use this expanded dataset with 23 columns. This allows us to perform an exhaustive analysis.

The data is:

- **Number:** ID for each pokemon.
- **Name:** Name of each pokemon.
- **Type1:** Each pokemon has a type, this determines weakness/ resistance to attacks.
- **Type2**: Some pokemon are dual type and have 2.
- **Total:** sum of all stats that come after this, a general guide to how strong a pokemon is.
- **HP:** hit points, or health, defines how much damage a pokemon can withstand before fainting.
- **Attack:** the base modifier for normal attacks (eg. Scratch, Punch).
- **Defense:** the base damage resistance against normal attacks
- **SP Atk:** special attack, the base modifier for special attacks (e.g. fire blast, bubble beam).
- **SP Def:** the base damage resistance against special attacks.
- **Speed:** determines which pokemon attacks first each round.
- **Generation:** Number of generation.
- **isLegendary:** True if Legendary Pokemon, False if not (more revision on mythical vs legendary needed).

- **Color:** Color of the Pokémon according to the Pokédex. The Pokédex distinguishes between ten colors: Black, Blue, Brown, Green, Grey, Pink, Purple, Red, White, and Yellow.

- **hasGender:** Boolean indicating the Pokémon can be classified as male or female.

- **Pr_Male:** In case the Pokémon has Gender, the probability of its being male. The probability of being female is, of course, 1 minus this value. Like Generation, this variable is numerical and discrete, because although it is the probability of the Pokémon to appear as a female or male in the nature, it can only take 7 values: 0, 0.125, 0.25, 0.5, 0.75, 0.875, and 1.

- **Egg_Group_1:** Categorical value indicating the egg group of the Pokémon. It is related with the race of the Pokémon, and it is a determinant factor in the breeding of the Pokémon. Its 15 possible values are: Amorphous, Bug, Ditto, Dragon, Fairy, Field, Flying, Grass, Human-Like, Mineral, Monster, Undiscovered, Water_1, Water_2, and Water_3.

- **Egg_Group_2:** Similarly to the case of the Pokémon types, Pokémon can belong to two egg groups.

- **hasMegaEvolution:** Boolean indicating whether a Pokémon can mega-evolve or not. Mega-evolving is property that some Pokémon have and allows them to change their appearance, types, and stats during a combat into a much stronger form.

- **Height_m:** Height of the Pokémon according to the Pokédex, measured in meters. It is a numerical continuous variable.

- **Weight_kg:** Weight of the Pokémon according to the Pokédex, measured kilograms. It is also a numerical continuous variable.

- **Catch_Rate:** Numerical variable indicating how easy is to catch a Pokémon when trying to capture it to make it part of your team. It is bounded between 3 and 255. The number of different values it takes is not too high notwithstanding, we can consider it is a continuous variable.

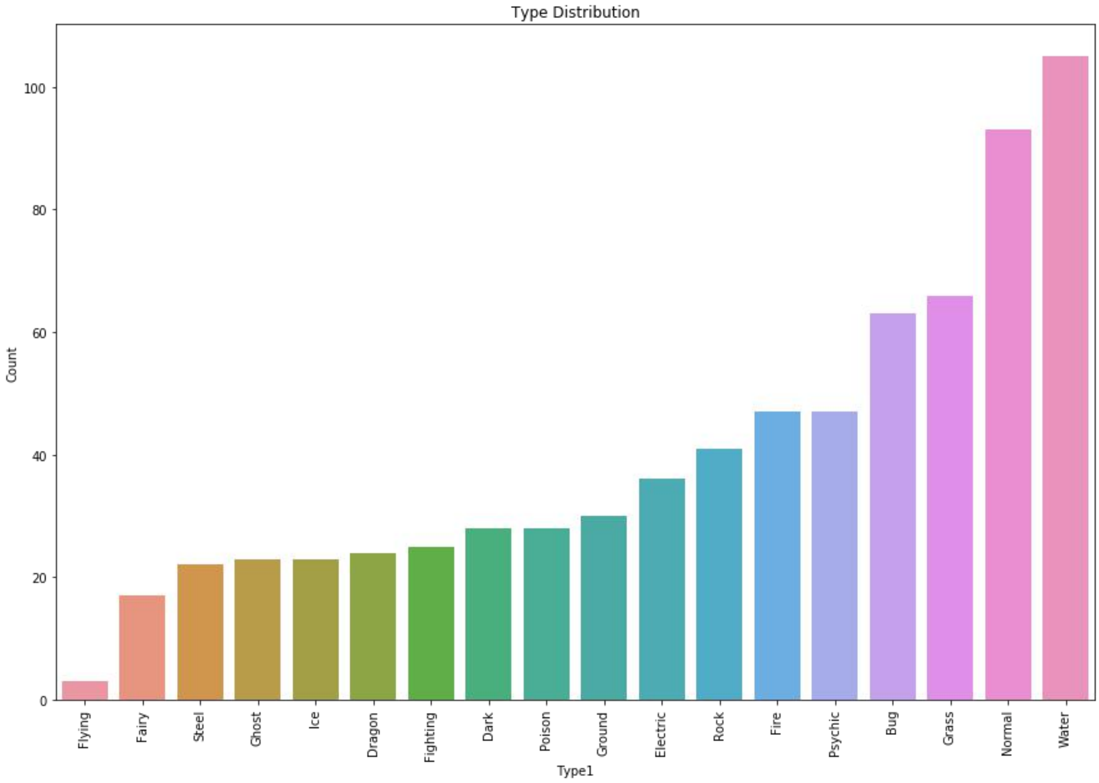
- **Body_Style:** Body style of the Pokémon according to the Pokédex. 14 categories of body style are specified: bipedal_tailed, bipedal_tailless, four_wings, head_arms, head_base, head_legs, head_only, insectoid, multiple_bodies, quadruped, serpentine_body, several_limbs, two_wings, and with_fins.

4.Descriptive Analysis and Data Visualizations

In this section, we will try to gain an insight of the distributions of the different variables as well as some relationships between them. Numerical and categorical variables will be treated in different manners.

4.1 Univariate analysis

Figure 4.1.1 shows the histograms of the primary and secondary types. Figure 4.1.2 shows the first and second egg groups of the Pokémon. The most common primary types are Water, Normal, and Grass, while the most common secondary type is Flying, because most of the times a Pokémon is able to fly the other type is considered first. We can also see that more or less the half of the Pokémon do not have any secondary type.



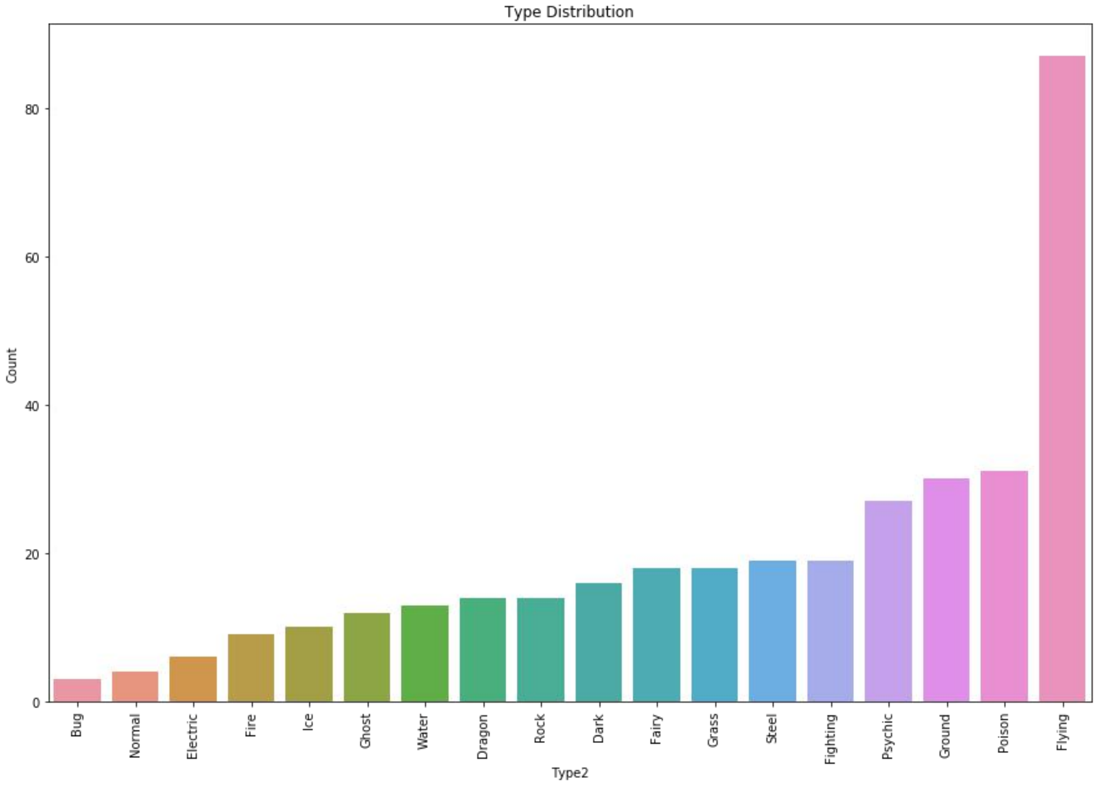


Figure 4.1.1 Histograms of the primary (up) and secondary (down) types

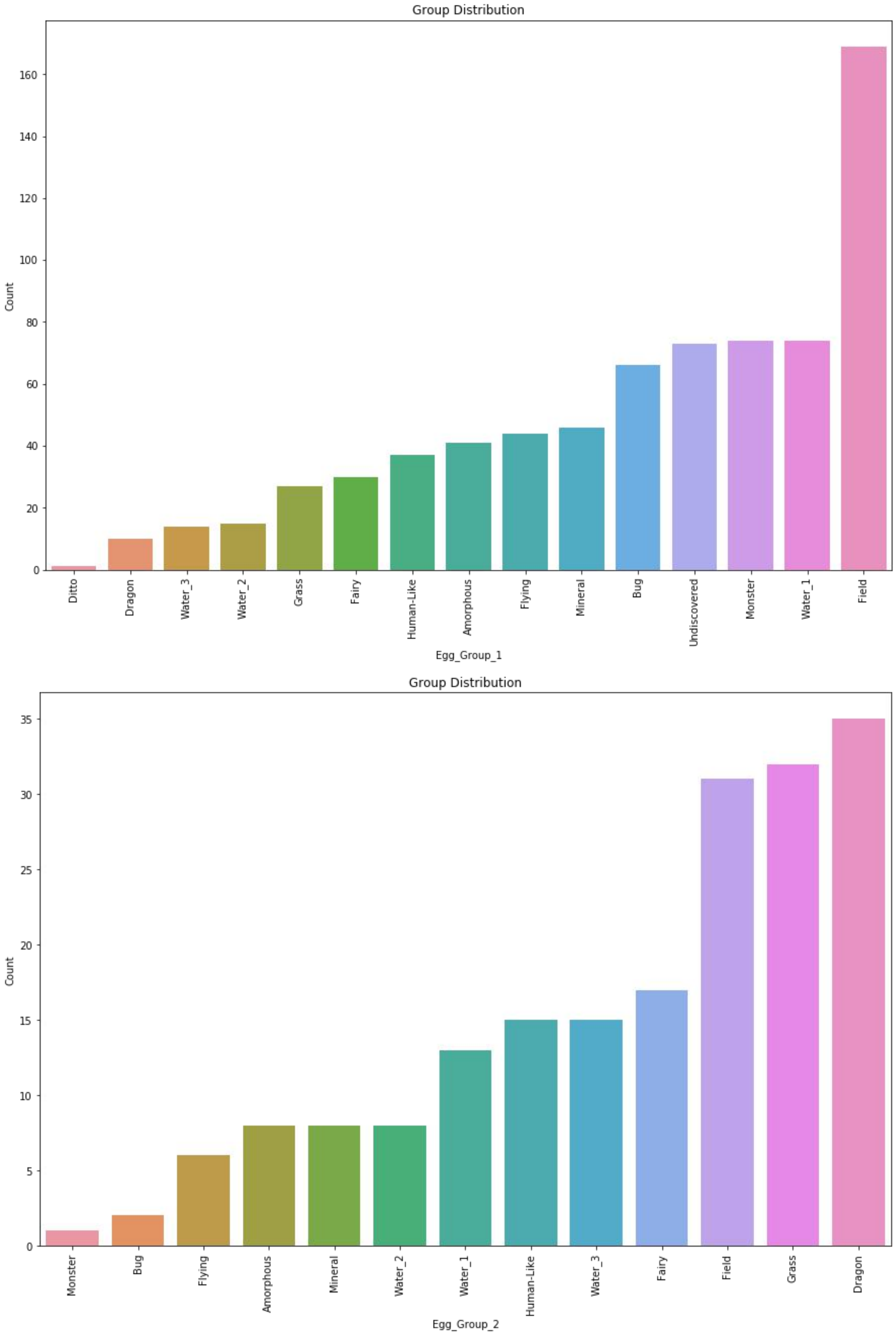


Figure 4.1.2 Histograms of the first (up) and second (down) egg groups

We can look for the more and less common egg groups in the histograms shown in Figure 4.1.2. We can observe that the most common egg groups (that can somehow be understood like races) are Field, followed by Water_1, Monster, Undiscovered, and Bug. Pokémon from the Field egg group tend to be terrestrial creatures. Pokémon from the three water egg groups live in or around the water, and Monster group Pokémon are usually among the most powerful. Undiscovered egg group is characterized by its members' inability to breed. Most of the Pokémon in the group are baby Pokémon, or legendary Pokémon.

Thus, we can get an idea of the ratio of the legendary non-legendary Pokémon ratio, and also gain some insights about the most common colors, most common body shapes etc. by plotting the histogram of other boolean and categorical variables. We will not include all these figures in our report and just focus on the catch rate analysis.

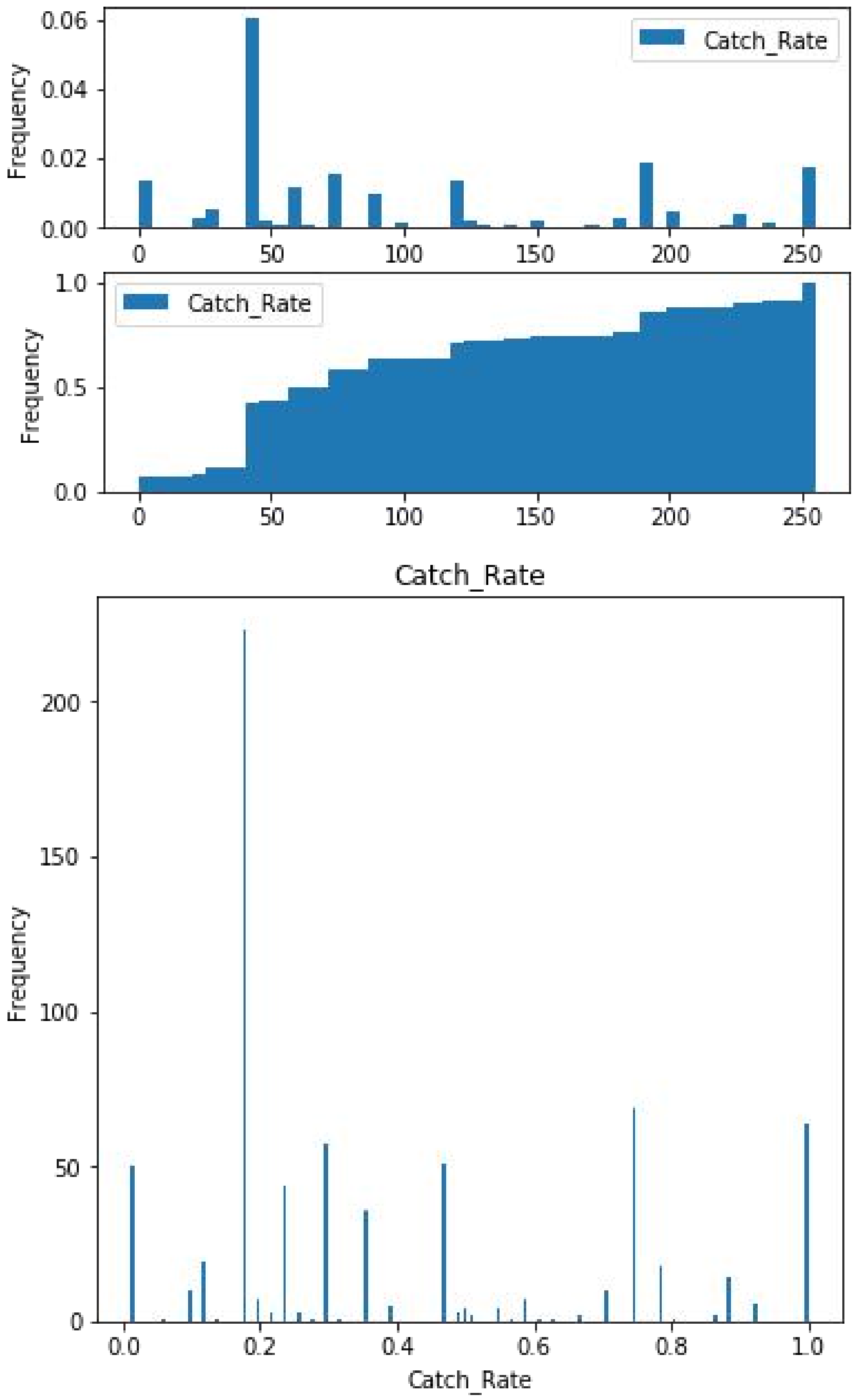


Figure 4.1.3 The distribution of the catch rate

We also took a look on the distribution of the catch rate to get an rough insight about which rates are the most common. We can see from the normalized version of this plot (down) that most Pokémon have a catch rate of 18%.

4.2 Relations and dependencies between variables

Now we will go deeper in the dataset and explore the possible relations between the variables. First we can try to see which of the numerical variables are highly correlated. Figure 4.2.1 shows the correlations between any two numerical variables.

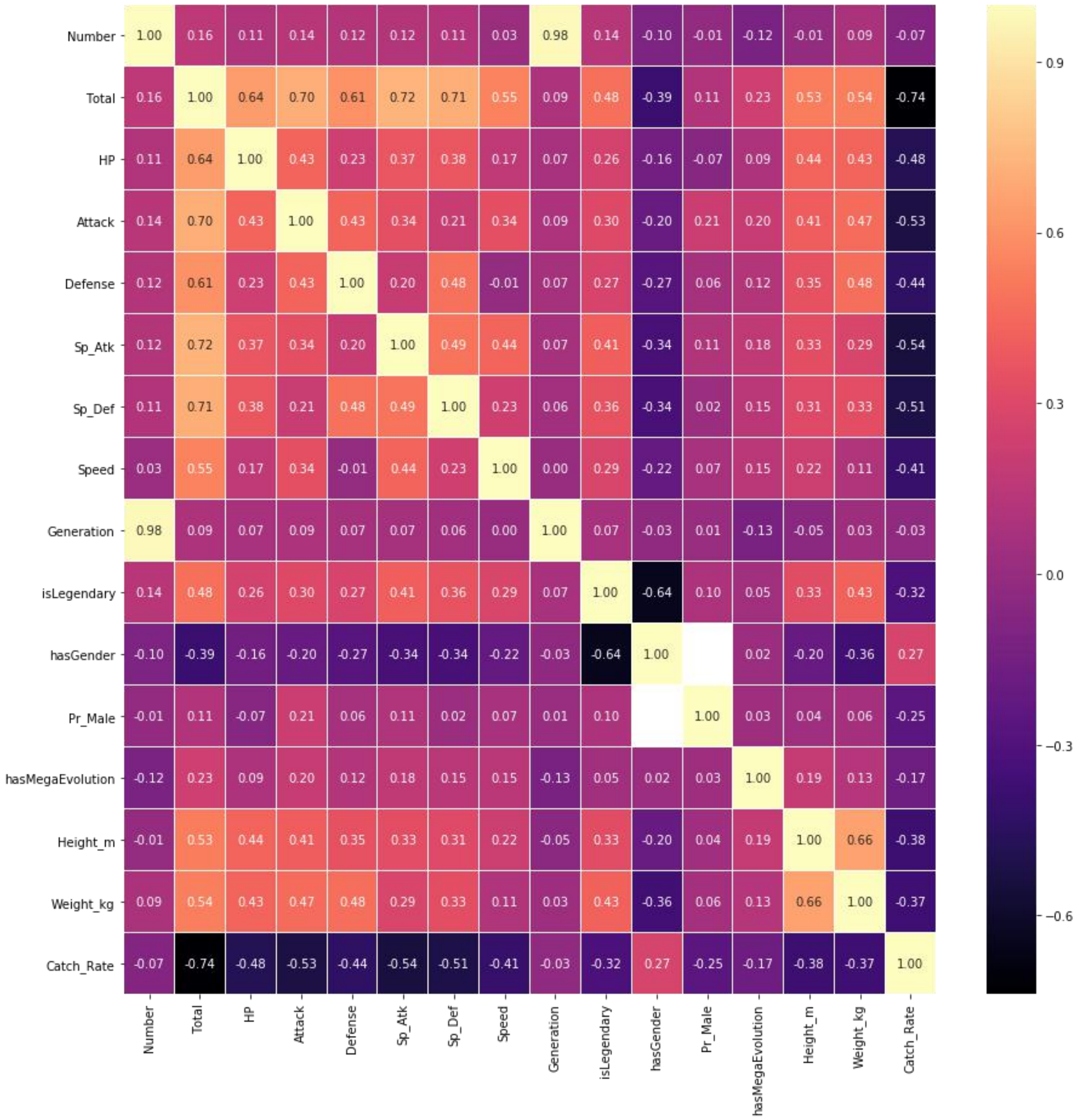


Figure 4.2.1 The correlations between any two numerical variables

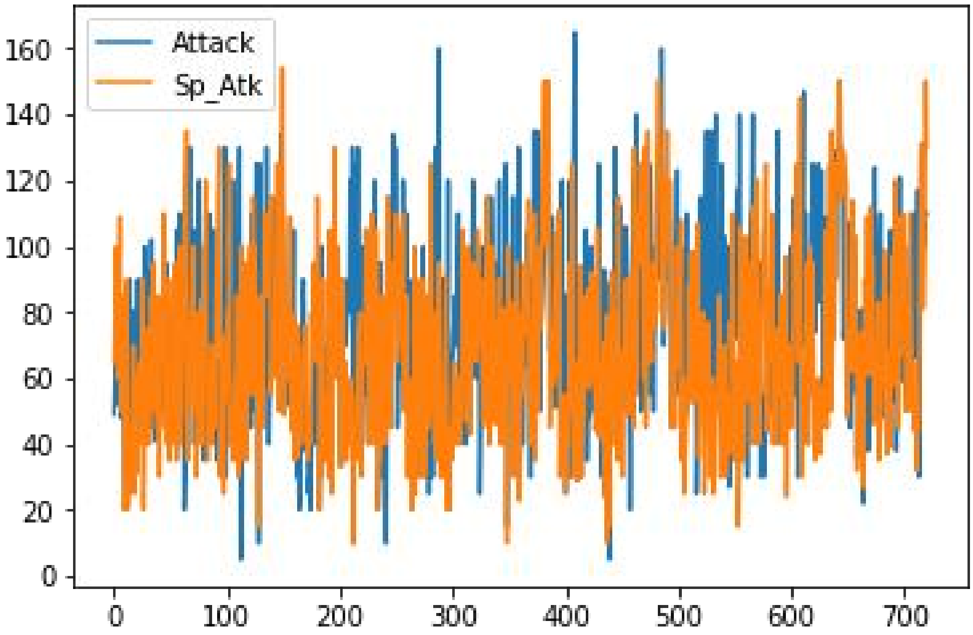


Figure 4.2.2 The similarity between *Attack* and *Sp_Atk*

We can see from this plot that *Total, HP, Attack, Defense, Sp_Atk, Sp_Def and Speed* are more correlated with *catch rate* and other variables has fewer contribution. Now we know which variables to keep an eye on.

Also, we can discover the strong relations seem to be *Attack* with *HP*, *Defense* with *Attack* and *Sp_Def, Sp_Def* with *Sp_Atk* and Defense, and *Speed* with the two attack stats. This again be visualized by Figure 4.2.2. Then, we can notice that the *Weight_kg* and the *Height_m* are very related. Moreover, on average the heavier and the taller a Pokémon is, the higher its battle stats will be, with the exception of the speed. It makes sense, because usually (in the real) the bigger an animal is, the more powerful it is, but usually its mobility will also be reduced. The next dependency we can find that, in general, the more powerful a Pokémon is, it will also be harder to catch it, i.e the bigger the combat stats, the *Weight_kg*, and the *Height_m*, the lower the *Catch_rate*. This is a very general tendency in the world of the video-games, and probably is also applicable in many other areas, big rewards require bigger efforts. Finally, we can say that there seem to be any relation between the Generation and any other variable.

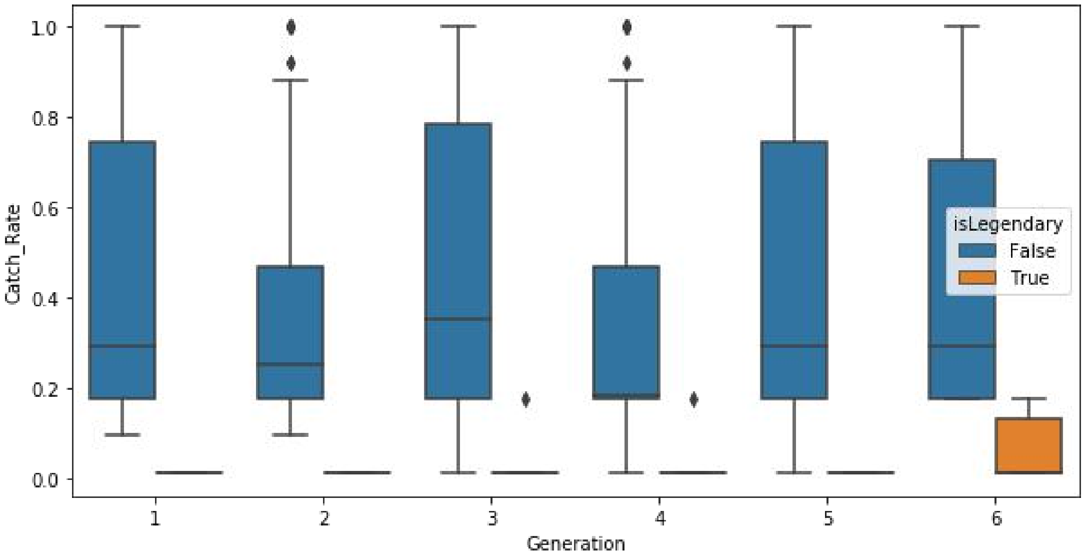


Figure 4.2.3 The box plot of catch rate distributions for different generations

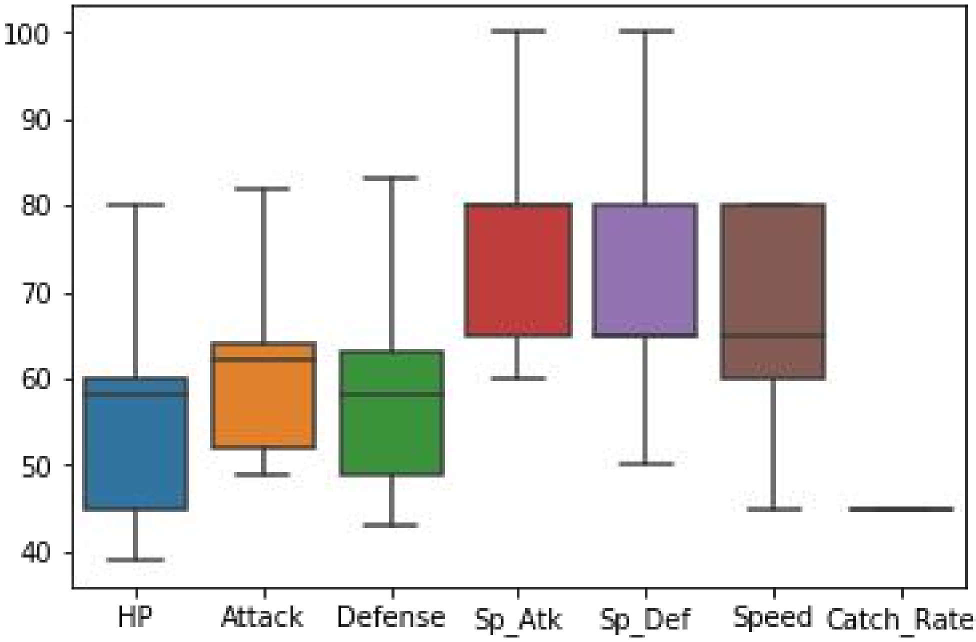


Figure 4.2.4 The box plot for some mentioned variables

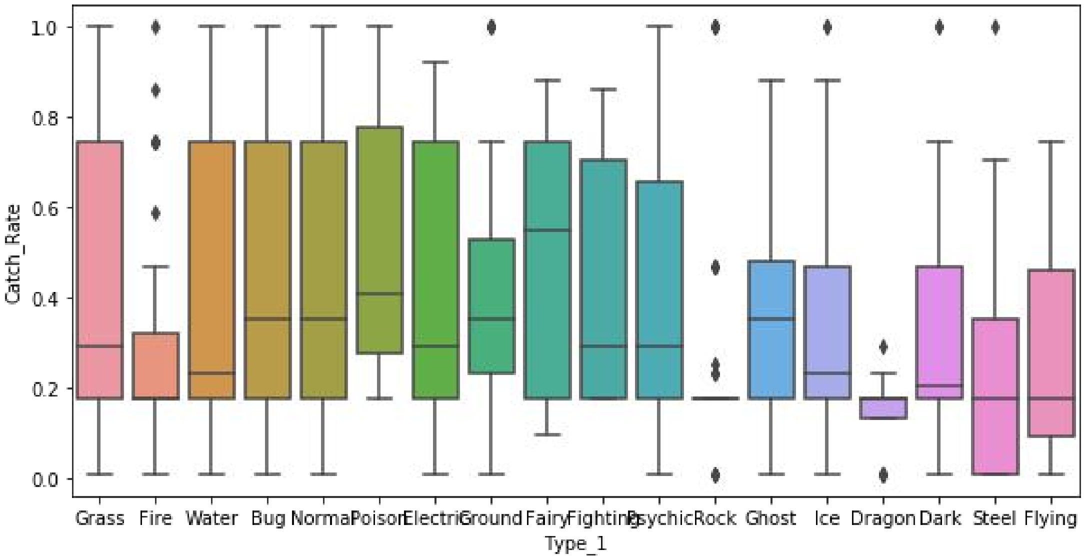


Figure 4.2.5 The catch rate distributions for Type_1 variables

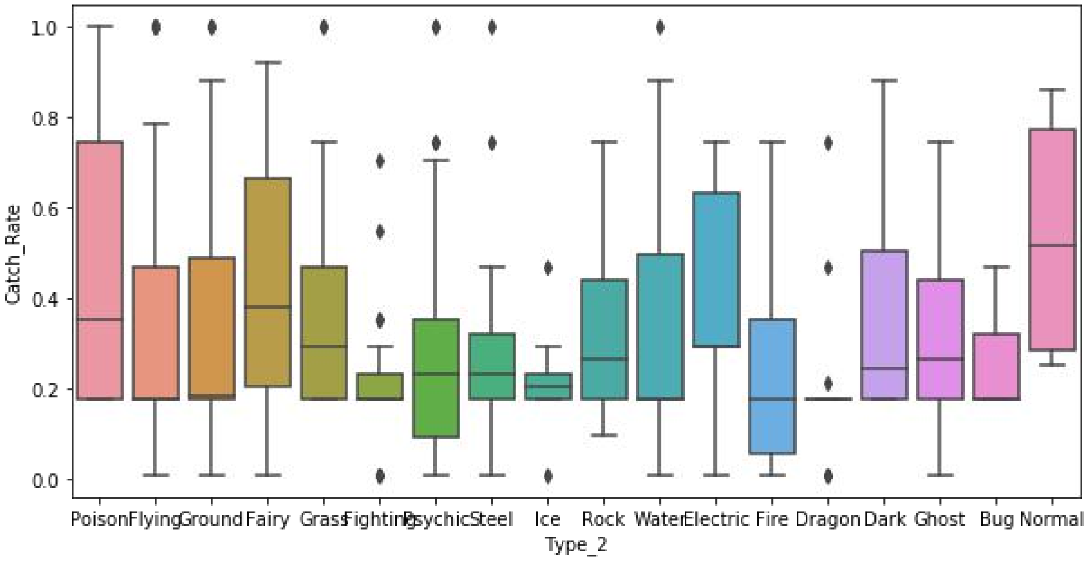


Figure 4.2.6 The catch rate distributions for Type_2 variables

We did some box plots to gain some general insights about the distribution for different variables. From Figure 4.2.3, we can see that different generations have overlapped catch rates which again confirmed that there is no obvious relation between generation and catch_rate. Figure 4.2.4 shows that the mentioned variables all have some centered distribution, for instance, *HP* is centered under the value between 45 and 60 and *Attack* has most values falling in the interval of [53,64]. Figure 4.2.5 and 4.2.6 show the catch rate distributions for *Type_1* and *Type_2* variables.

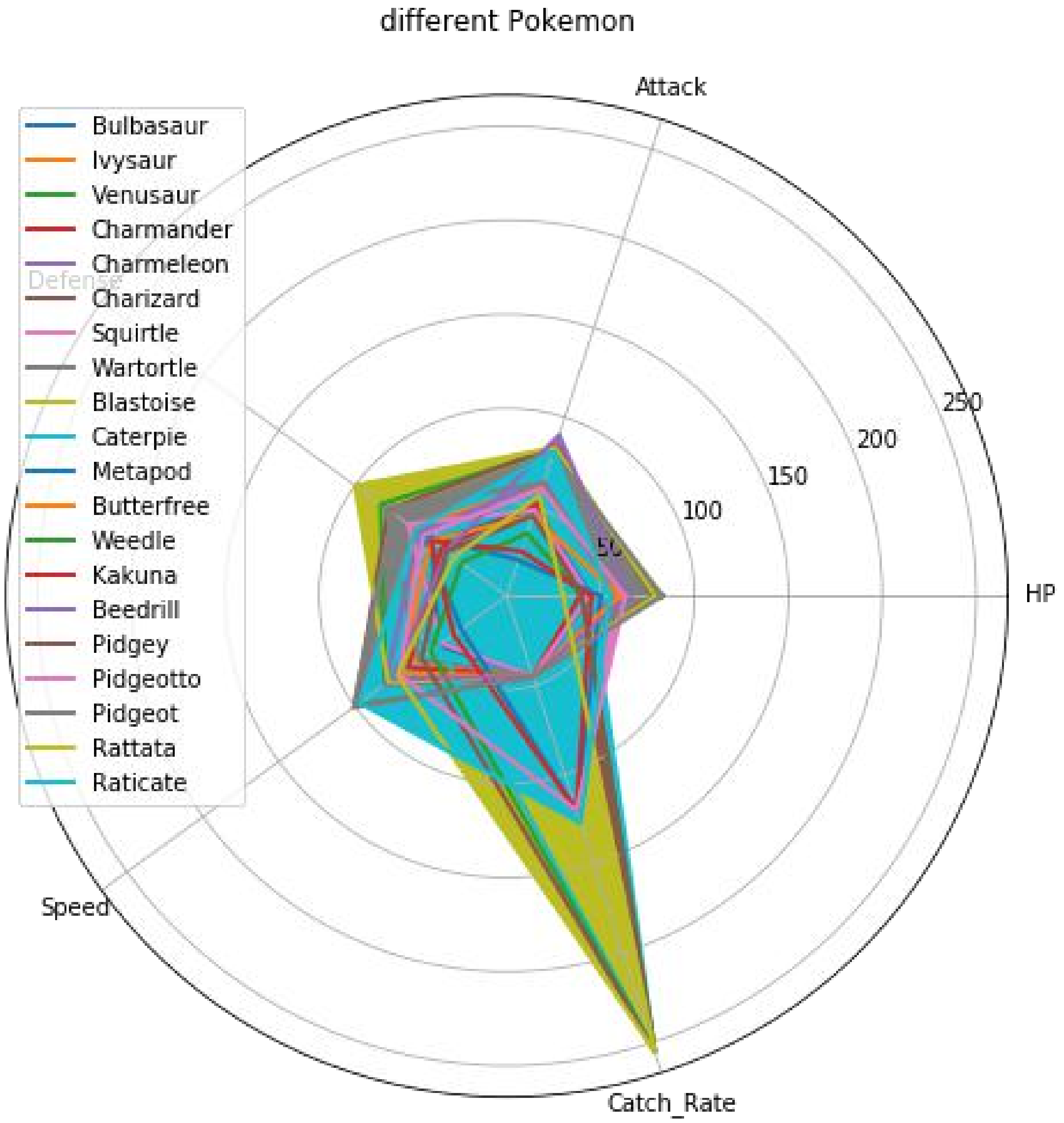


Figure 4.2.7 The radar plot of *Catch_Rate, Speed, Defence, Attack* and *HP* for some randonly picked Pokémon.

The radar plot is generated through some features that are highly related to catch rate. We randomly picked some Pokémon to see the corresponding status.

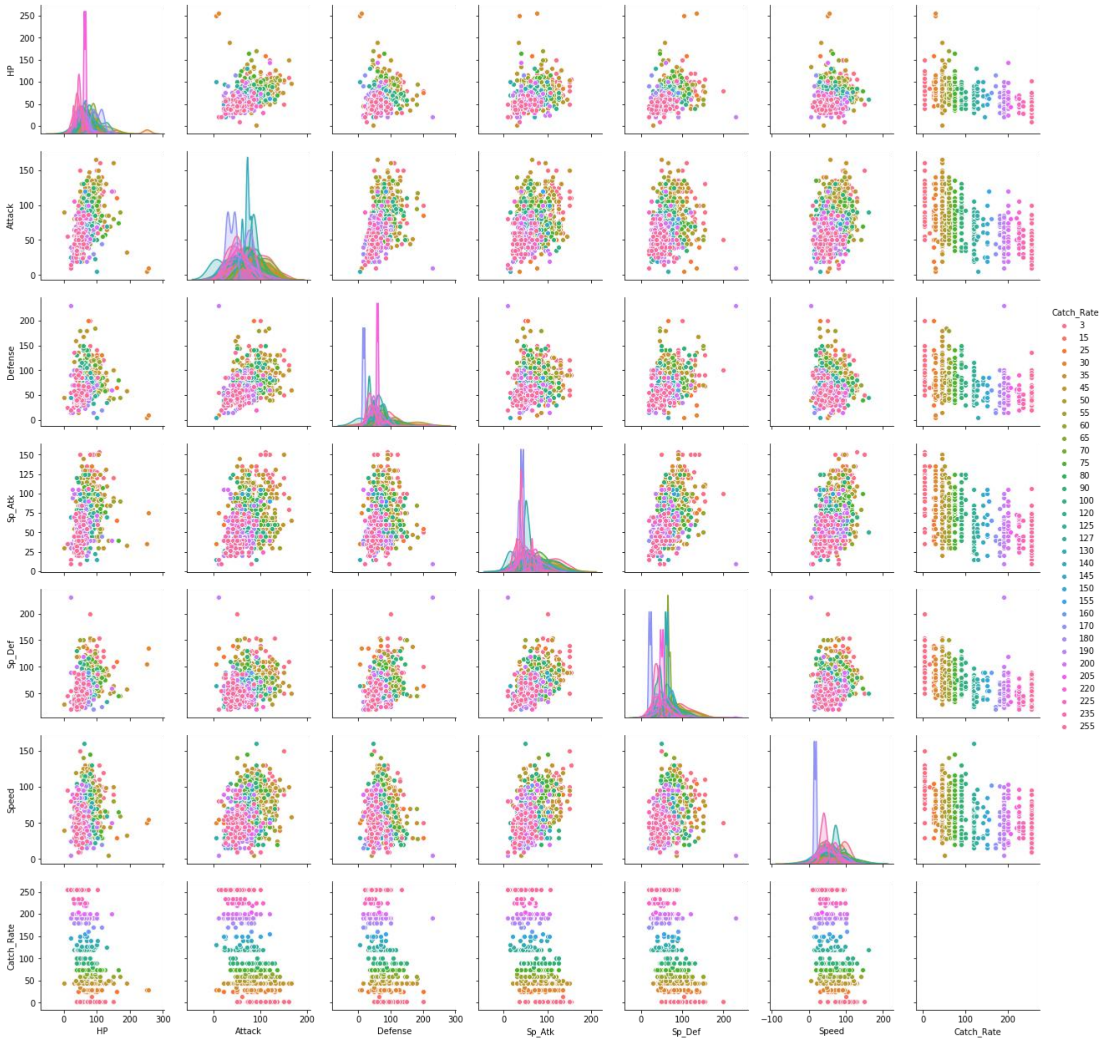


Figure 4.2.8 Pair plot of *Catch_Rate, Speed, Defence, Attack,Sp_Atk, Sp_Def* and *HP*

Finally, this pair plot confirmed that the listed variables (Speed, Sp_Def, Sp_Atk, Defence, Attack and HP) are strongly correlated with Catch_Rate.

5.Predictive modeling

In this section, we will conduct two types of predictive models on this dataset to predict the Catch_Rate. First, we will try to predict the Catch_Rate with some linear models. Then we will implement the Neural Network technique to obtain a higher accuracy.

We start off by simply importing many packages needed for data analysis in Python, such as Pandas and NumPy, then reading in the datasets we will be using. We started by cleaning the data set to get rid of columns we did not need at the time to make it easier to work with the data.

Then we use 'dataframe.info()' to get some good summary statistics on all the data to better understand the dataset.

Then we did data preprocessing. We converted text labels to one hot codes to enable the categorical variables to be fed into our models. Also, we replace some missing values that were originally just 'NaN' with '0'. This is why we could analyze height and weight since it would not work with a 'NaN' value.

Then we implemented several models demonstrated below to get a best result.

5.1 Linear Regression

We first tried to predict Catch_Rate with basic linear regression. Figure 5.1.1 shows the coefficients corresponding to different features. The test accuracy of this model is 0.965.

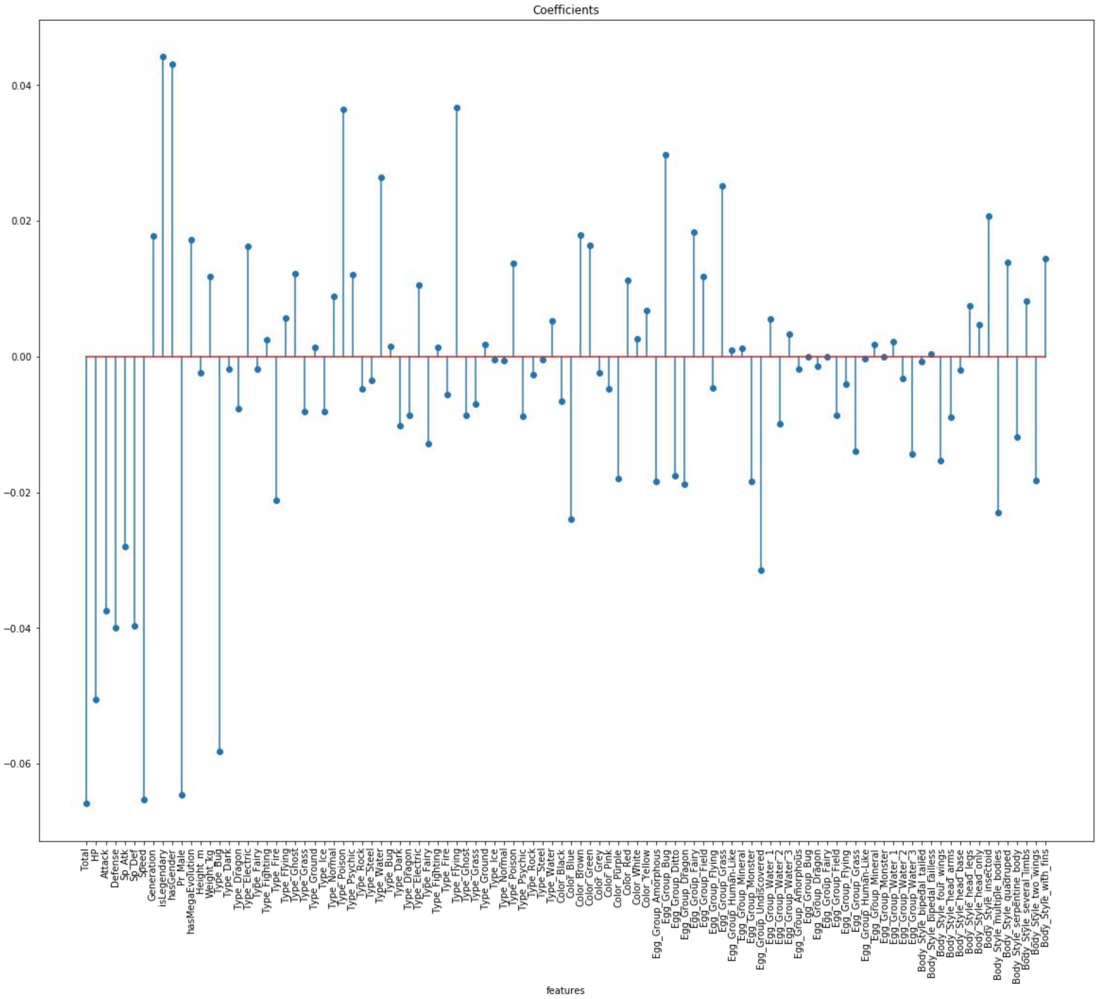


Figure 5.1.1 The coefficients corresponding to basic linear model

Then we tried linear regression with LASSO regularization with different values of alpha and picked the optimal alpha for this model. Figure 5.1.2 shows the coefficients corresponding to this model (Linear regression with LASSO) and the test score now is 0.964.

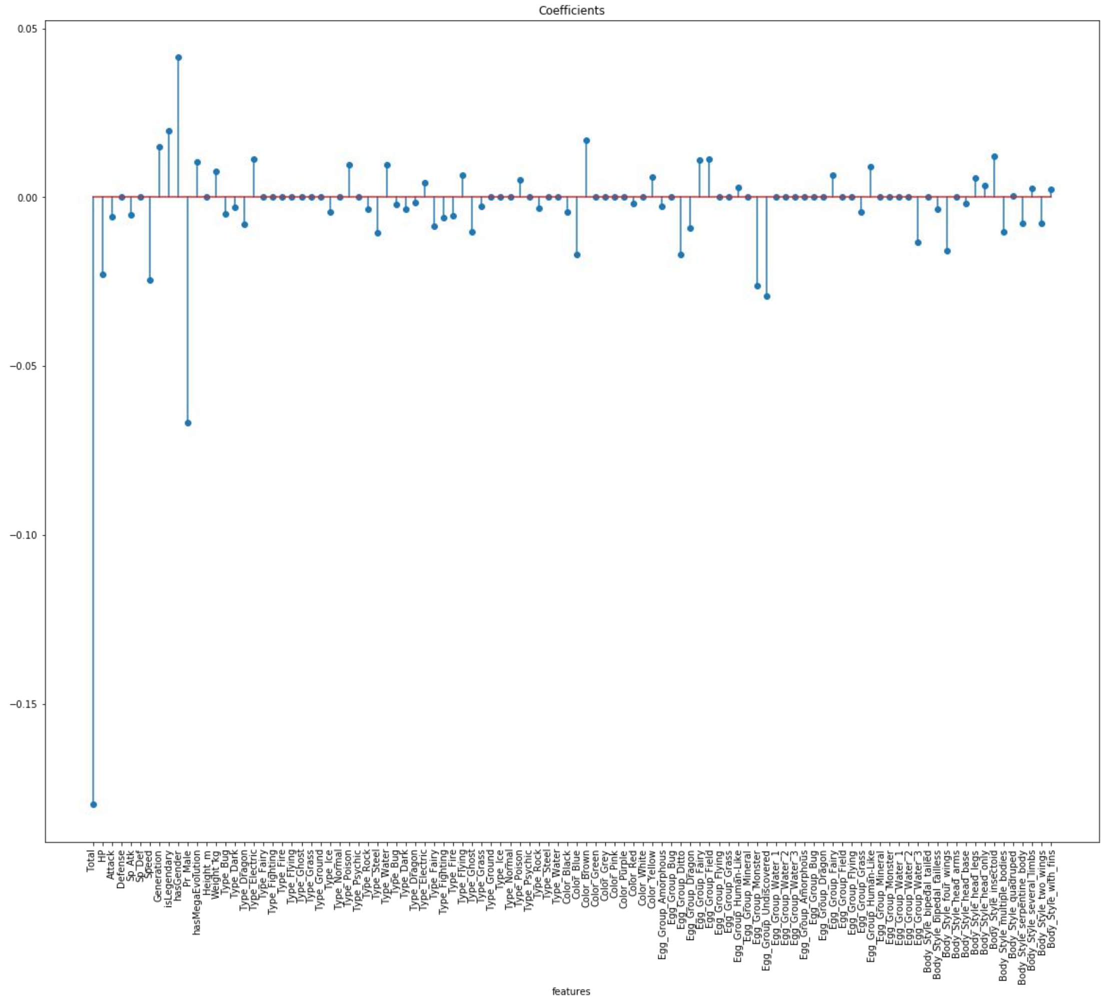


Figure 5.1.2 The coefficients corresponding to linear model with LASSO

We also tried linear regression with Ridge regularization with different values of alpha and picked the optimal alpha for this model. Figure 5.1.3 shows the coefficients corresponding to this model (Linear regression with Ridge) and the test score now is 0.965.

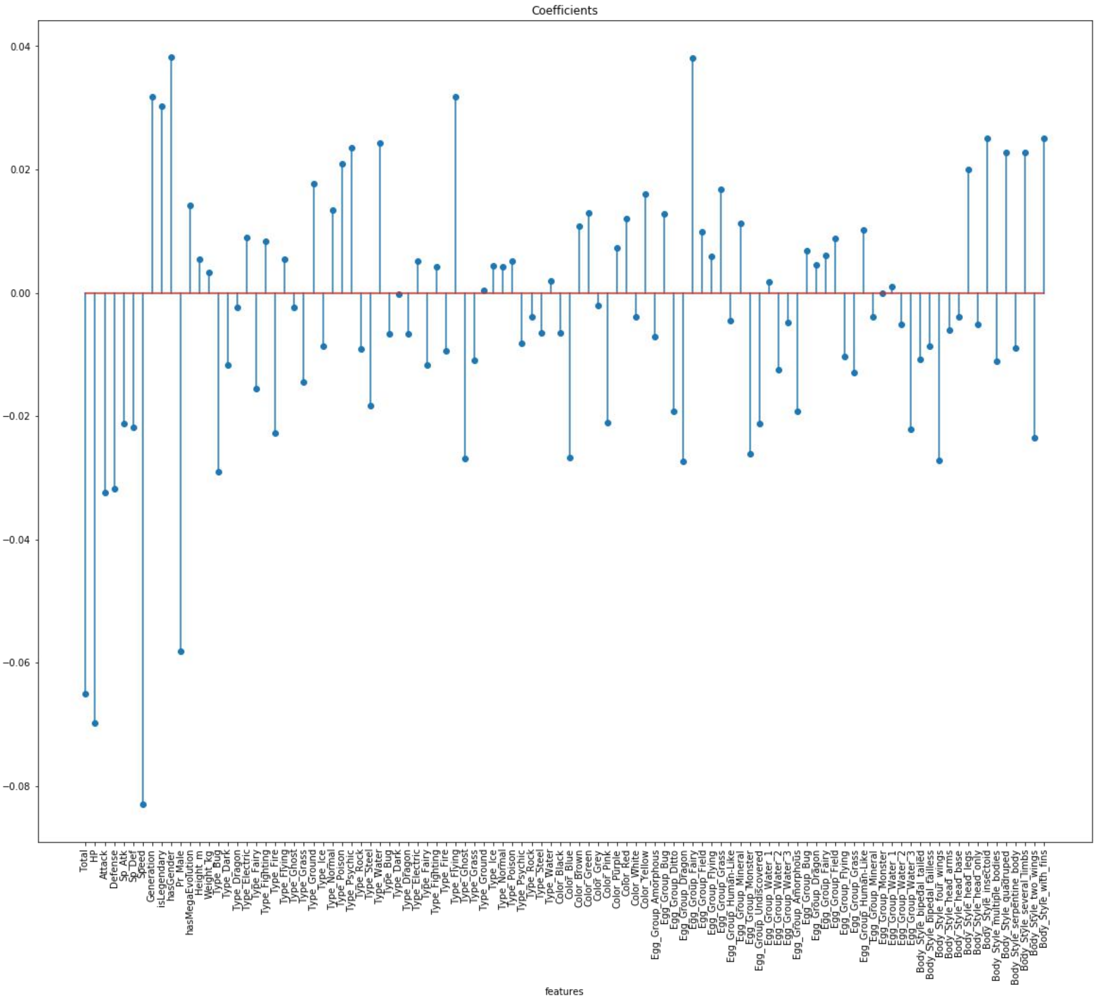


Figure 5.1.3 The coefficients corresponding to linear model with Ridge

We then tried linear regression with Principal Component Analysis. We try to find several new variables that explain more the real variability of the data, with the hope that those new variables will be enough to understand the properties. We loop over different n_components and find that the best test score for this model is 0.954. Figure 5.1.4 shows the train and test score vs. the number of components.

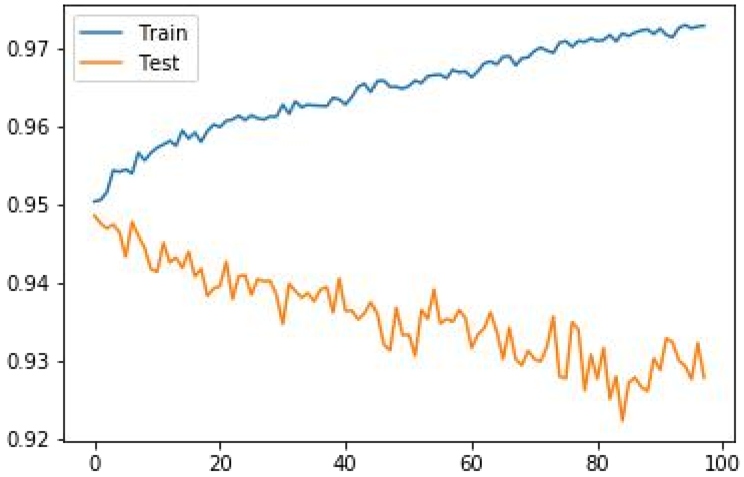


Figure 5.1.4 The train and test score vs. the number of components

5.2 Neural network

We've learned the power of Neural Network in Machine Learning class and are curious about how it performs on this specific application. We trained a simple neural network with the batch size of 50 and 100 epochs, run it several times with GPU to adjust the parameters. Figure 5.2.1 shows the test score vs the epochs. The final test score is around 0.905. We are still working around the parameters to improve its performance.

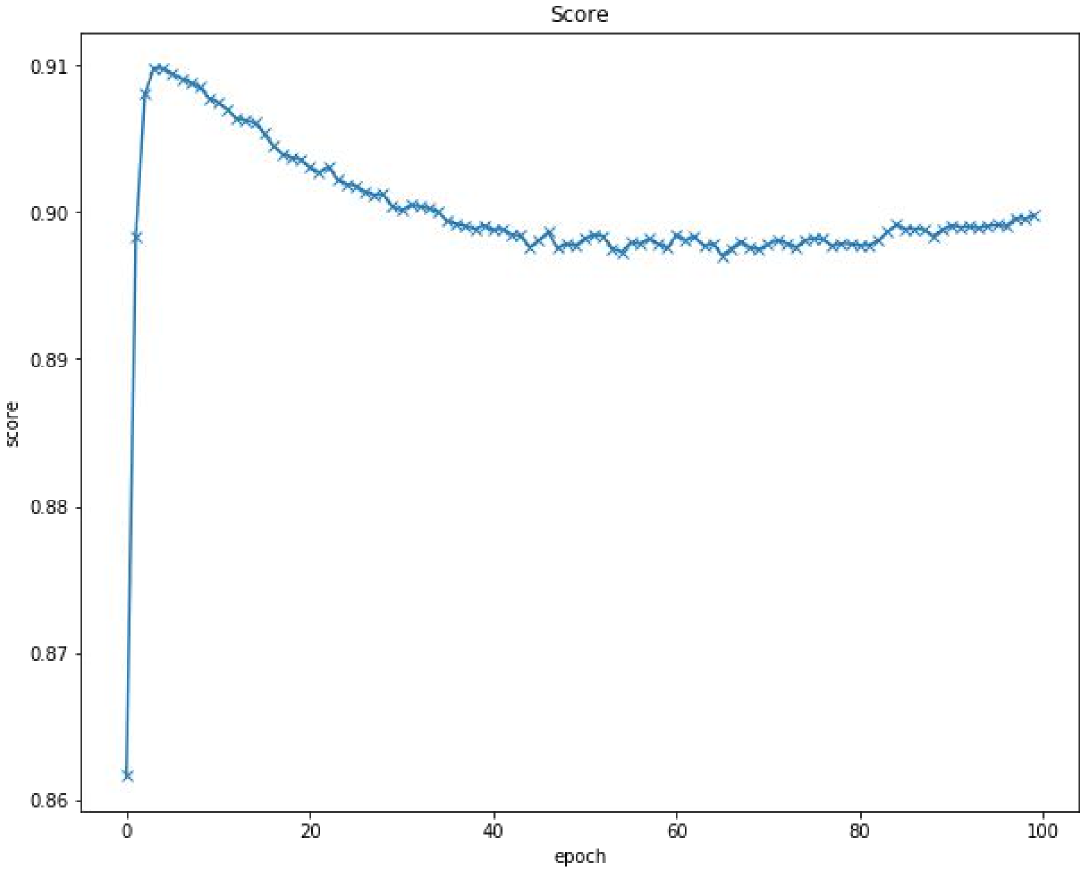


Figure 5.2.1 The test score vs the epochs

6.Summary and Conclusions

In this work, we have carried out univariate analysis for some variables in the dataset. We have analyzed how the numerical variables are distributed. In the case of the categorical variables, we have used histograms with the same end. Then we have tried to find correlations between the numerical variables. The higher the stats of the Pokémon, it will also be bigger, heavier and harder to catch. No big dependencies have been found with the generation a Pokémon was released or its probability of being female or male.

And then we have built a predictive model that we tried to predict the catch rate of a Pokemon based on its numerical variables. In the process of modelling we used different kinds of methods to improve the accuracy.

The analysis of the predicted results confirmed our analysis about the difficulty of capturing a specific Pokemon, which is largely determined by its HP and Total. However, its generation, type and other features have little effects on catch rate. As we all know, in the process of playing games, some players are not only prefer capturing pokemons with high total and HP , but also prefer the cute, small-sized, specific type pokemons.

Through our research, for these players, they don't have to spend lots of game gold coins on purchasing the special Poke balls which raises the catch rate to grab their favorite pokemons. As long as the target pokemon’s HP and total are not very high, they can use the ordinary Poke ball to catch them still with a high catch rate. This will eventually saves a lot of game coins for these users.

On the other hand, with our prediction model and the characteristics handbook of the pokemons, players can predict the catch rate of the new Pokemon that he or she encounters. This helps them to select poke balls correctly to improve the capture rate on catching, which also saves a lot of game resources for them.

References

[1] Alberto Barradas. Pokemon with stats.
https://www.kaggle.com/abcsds/pokemon.

[2] Asier LÃ¸spez Zorrilla. PokÃl'mon for Data Mining and Machine Learning.
https://www.kaggle.com/alopez247/pokemon.

[3] David W Scott. Multivariate density estimation: theory, practice, and visualization, 2015.

[4] S. S. Shapiro and M. B. Wilk. An analysis of variance test for normality (complete samples). Biometrika,