IST707 - Applied Machine Learning

Finding Legendary Pokemons by Machine Learning

By
Jianghui Li - jli159@syr.edu
Chengsheng Ma - cma110@syr.edu

Abstract

This report contains descriptive statistics and machine learning of a pokémon dataset. Pokémon is a series of combat-based video games developed by Game Freak and published by Nintendo. The dataset contains information on pokémon stats, types, and names, etc. The goal of this project is to use relatively simple algorithms to enable machines to predict legendary pokémon since these pokémon species are rare and have great combat powers. After finishing supervised machine learning, the results were satisfying because the highest accuracy is above 96%.

Table of Contents

Introduction

Background information:
The dataset that was used for this project was from a Pokémon RPG game published by Nintendo in 1996. Since Pokémon video games are ability-based, and legendary pokémon normally have the best abilities; hence, we are interested in finding those legendary pokémon and how all pokémon's abilities, types, and stats, etc. are correlated or related to whether they are legendaries or not. Thus, analyzing a wide variety of variables used to describe the pokémon is essential, and there are chances for us to find relationships between them, and also to cluster the pokémon according to some criteria. In the rest of the report, we will explore pokémon and their corresponding variables that appear in the RPGs.

Why are we doing this?
There are countless examples of people doing machine learning on animals and plants. People can classify animals and plants according to their traits and use their traits to predict what they are. What if we try something virtual using virtual monsters like pokémon? Pokemon are virtual animals or monsters that have their traits. Perhaps we can let machines learn to predict and classify different pokemon.

The dataset and variables:
The Pokémon dataset we used contains 1032 rows and 44 columns. The dataset was retrieved from kaggle.com. The rows and columns tell us that there are 1032 unique Pokémons with 44 different variables containing their names, stats, types, height, weight, etc. Since there are too many variables in the dataset, we only explored some of the interesting variables. For instance, our focus is to find legendary pokémon, so we can use different machine learning methods to predict legendary pokémon from the other variables in the dataset.

Objective
In this project, we will be focusing on building machine learning algorithms to find and classify legendary pokémon. First, we need to know that most Pokémon games are battle-based. Although some people might be fond of collecting pokemon that look cute, most people would still like to get pokémon with very combat capabilities. There are some pokémon in Pokémon games that are legendary grade. These pokémon usually have high stats and are hard to catch. Hence, many people are interested in collecting these legendary pokémon since most of them have overwhelmingly strong combat powers.

General descriptive analysis of the variables

There are some visualizations about different attributes of all pokémon, and we separate legendary pokémon and the rest of pokémon using two different colors. The light blue color represents legendary pokémon, and the red color represents all the non-legendaries.

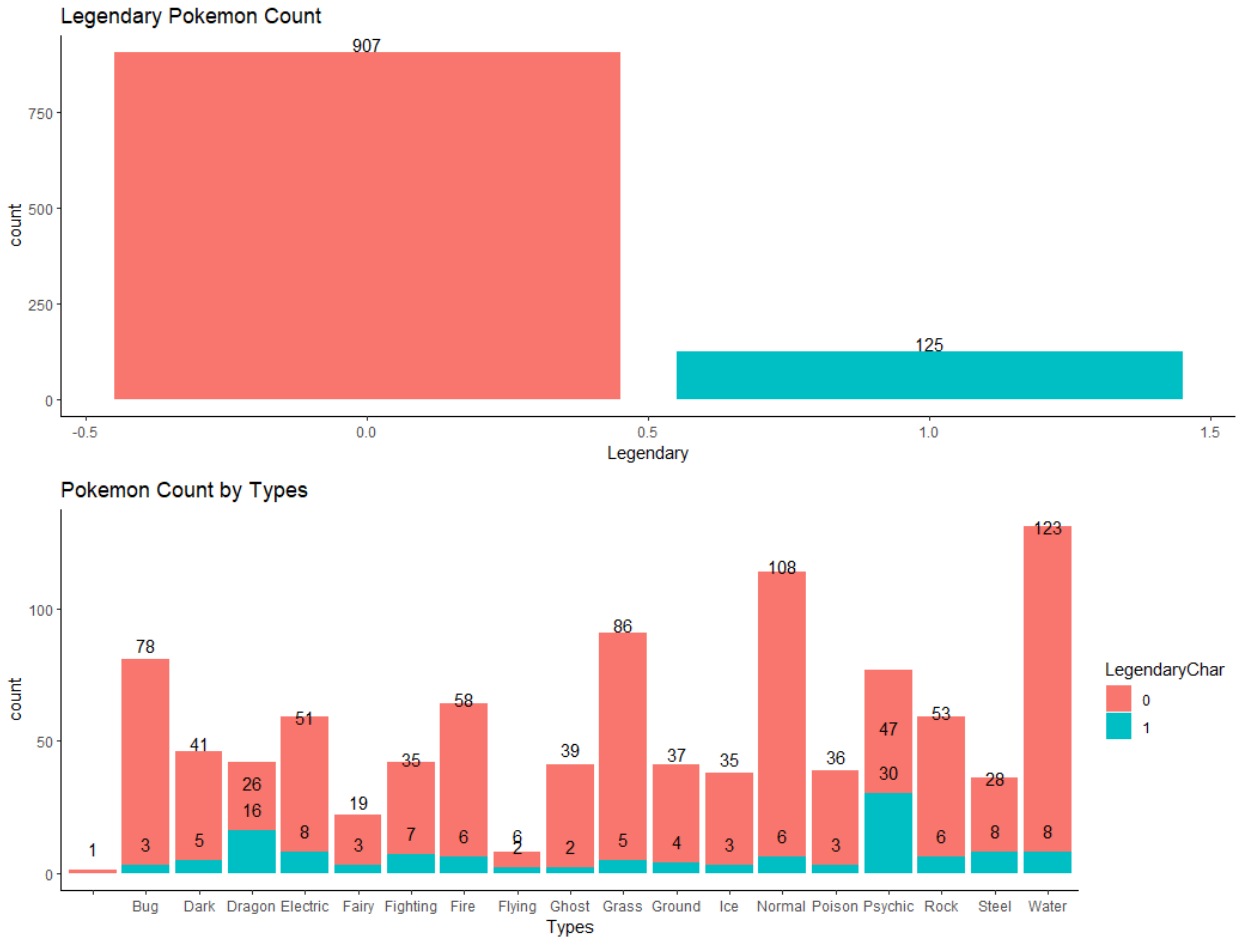

Figure 01 - The plot at the top is pokemon count by legendary and non-legendary, and the plot at the bottom is pokemon count by types

In figure 01, the very first plot shows there are 125 legendary pokémon in total.
The second plot shows pokemon count by their types. We can see that there are legendary pokémon across all types, and the dragon and psychic types have a great portion of legendary pokémon.

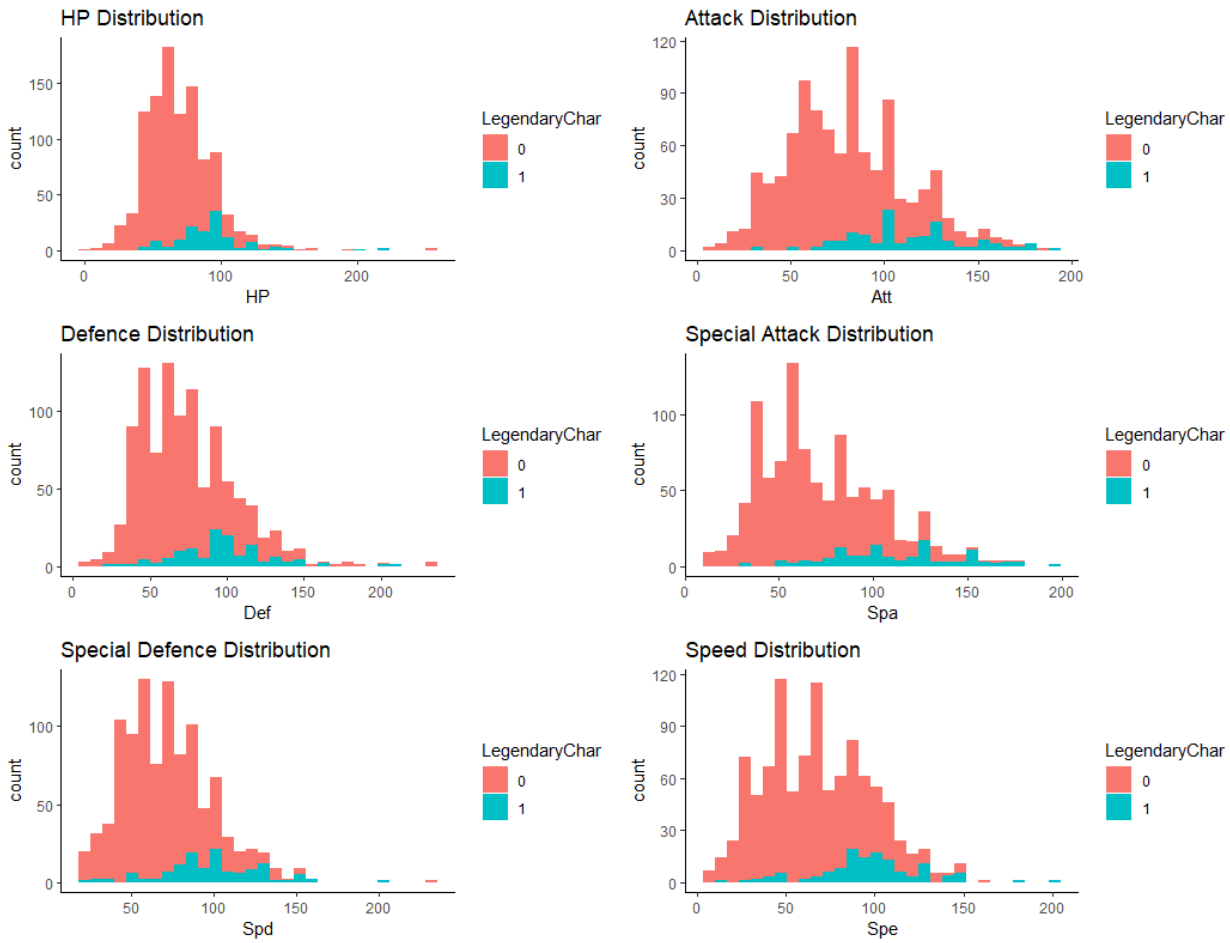

Figure 02 - Six histograms of pokemons' individual stats

These are plots showing pokémon's individual stats distribution. It seems like legendary pokémon's average individual stats are stronger than that of non-legendaries. However, legendary pokémon's stats do not look overwhelmingly strong by looking at all the individual stats. It is very possible that a legendary pokemon is low in HP but very strong on attack and other stats.

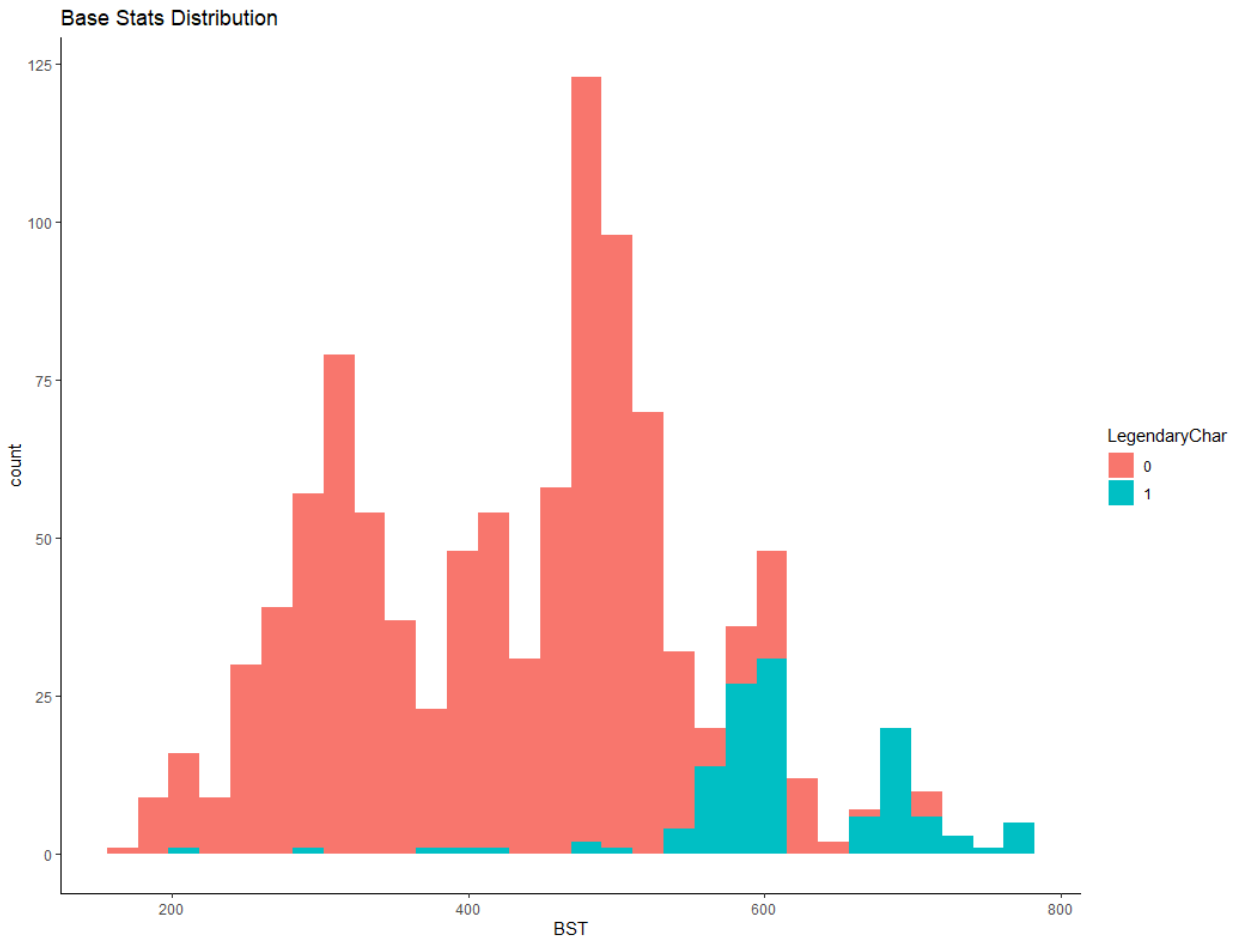

Figure 03 - Histogram of pokemons' base stats

When adding all six stats together, we get their base stats shown in figure 03. By looking at the base stats distribution. Unlike the individual stats histograms, the average legendary pokémon's stats do look strong compared to that of the rest of pokémon.

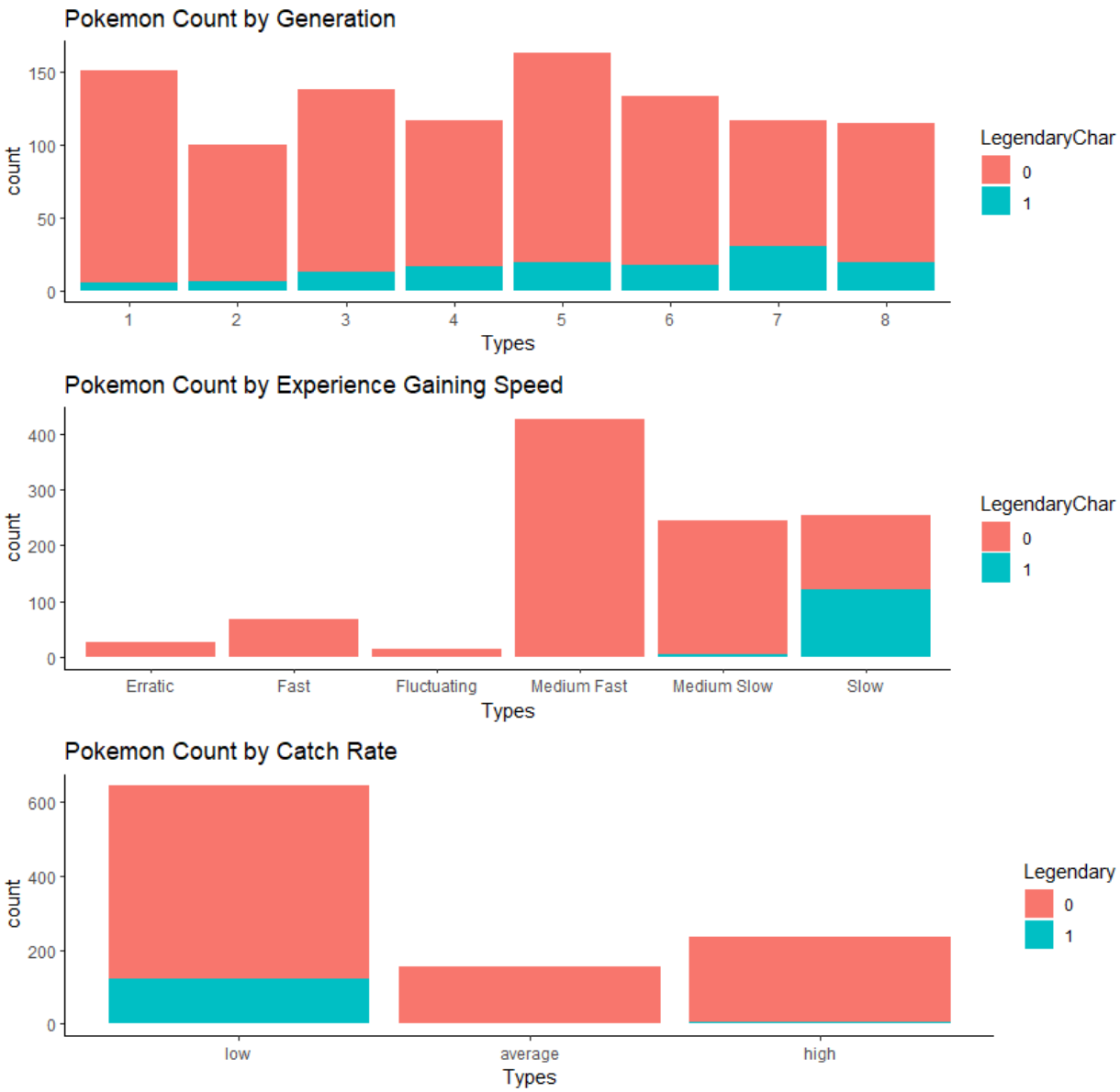

Figure 04 - From top to bottom, the first plot is the pokemon count by their generations, then the plot in middle is a plot of pokemon count by their experience gaining speed, and the last plot is a plot of pokemon count by their catch rate.

In figure 04, even though every generation has a similar amount of total different pokemon species, it seems like newer generations tend to have more legendary pokémon.
A vast majority of legendary pokémon tend to have slow experience gaining speed and low catch rate.

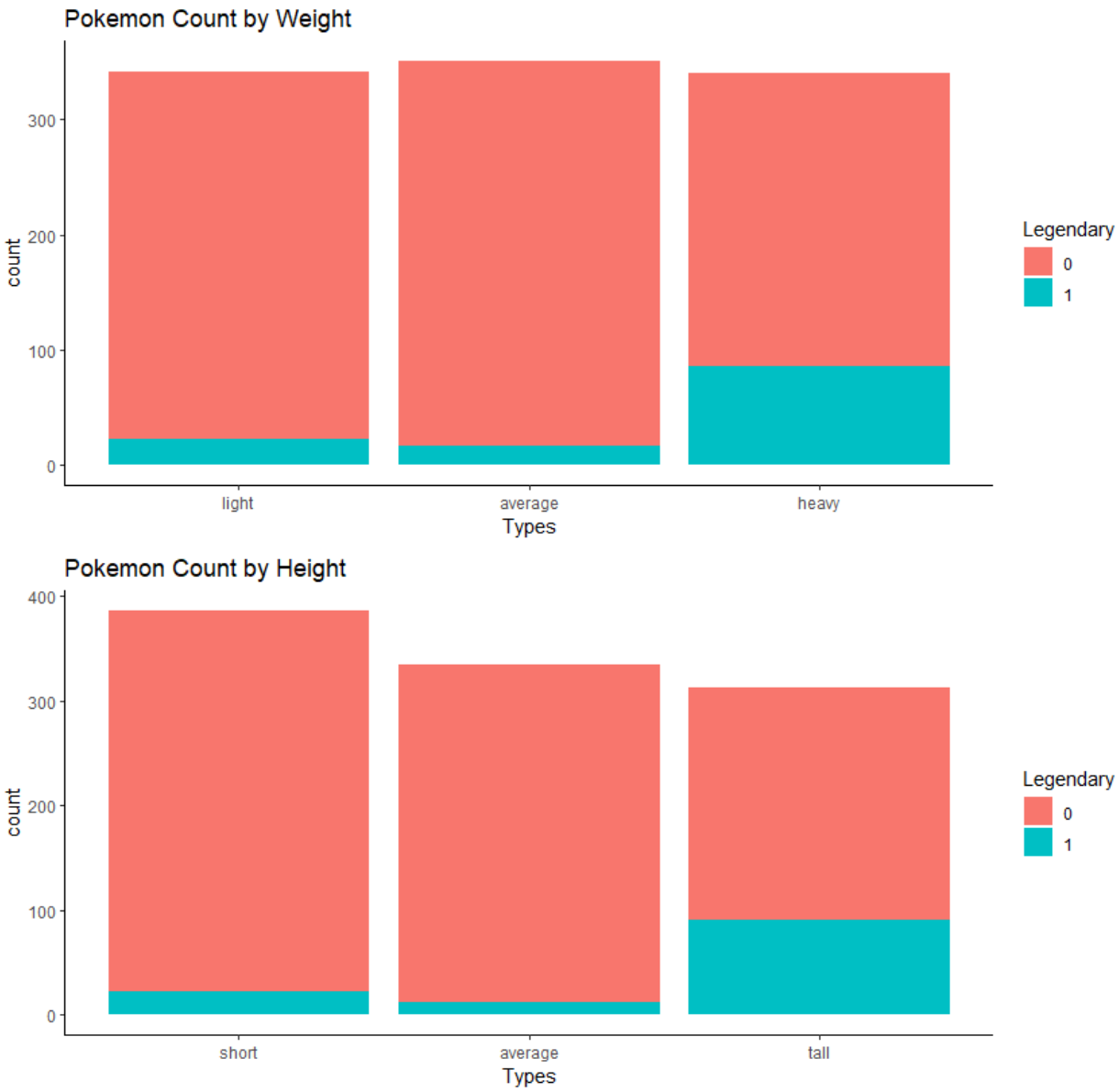

Figure 05 - The plot at the top is pokemon count by weight, and the plot at the bottom is pokemon count by height

What we can get from figure 05 is that legendary pokémon can have different weights and heights, but we can see a good amount of them are tall and heavy.

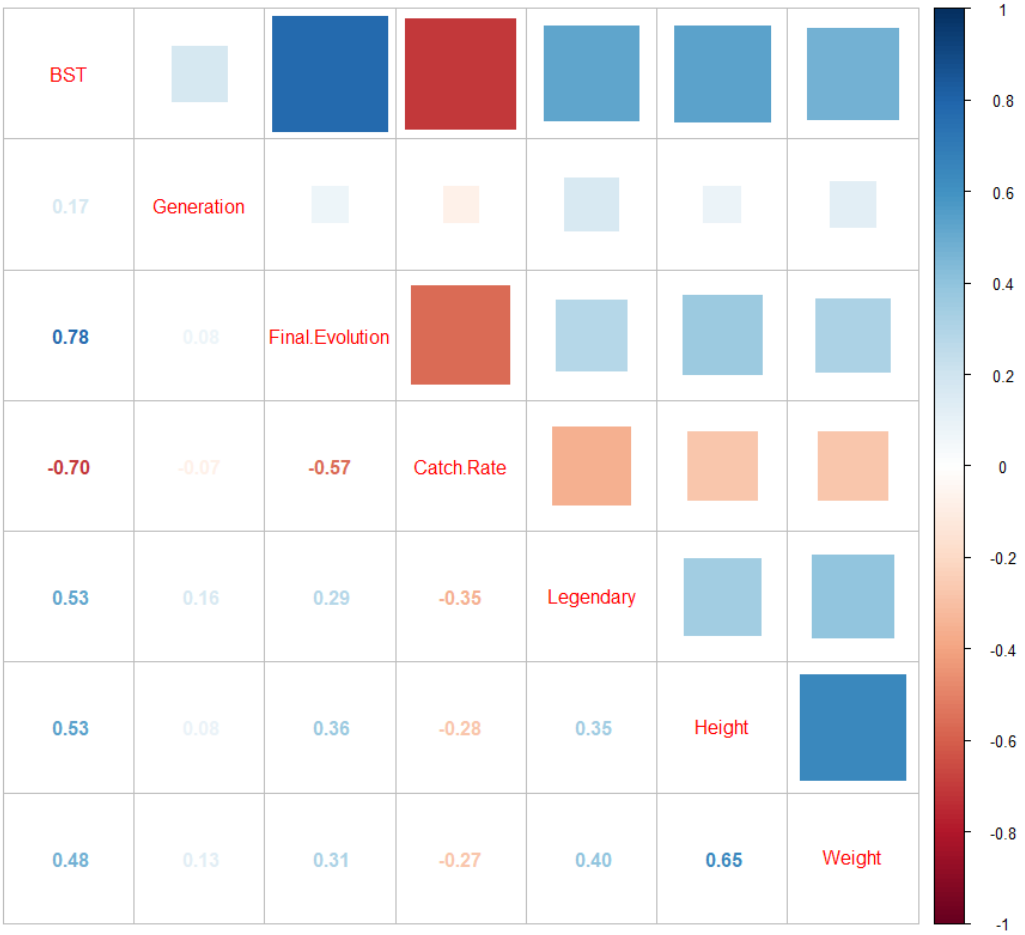

Figure 06 - This is a correlation plot of the variables we are interested in; blue color represents positive correlation, and brown color represents negative correlation

Figure 06 is a correlation plot of some of the pokemon attributions we are interested in. For legendary vs other variables, it is easy to tell that BST (base stats) has the strongest positive correlation with legendary. Therefore, pokémon's base stats will be used to predict the legendaries.

Supervised Learning

1. Decision Tree:

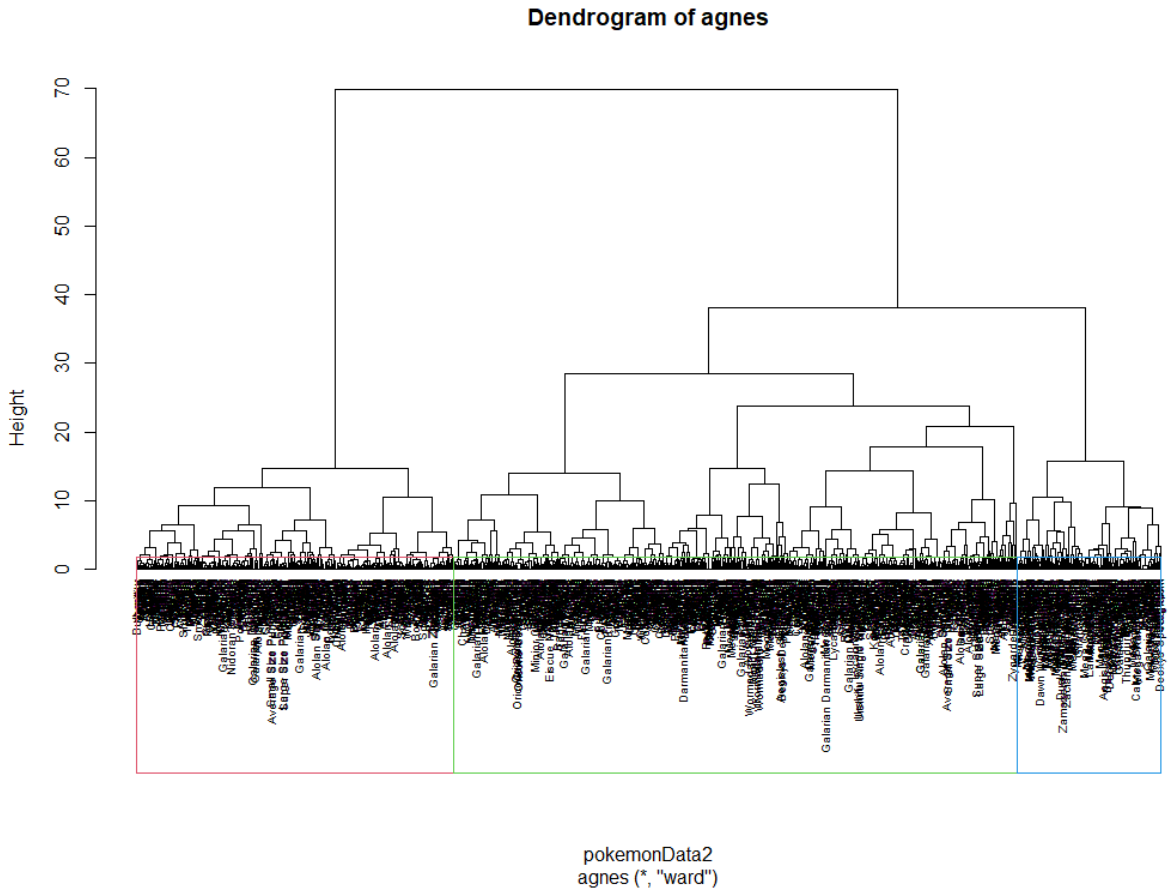

Figure 07 - Ward method decision tree

From Figure 07, it is very hard to get detailed information about pokémon. However, this Figure 07 can help to find the K numbers for the Clustering algorithm. The optimal K-means value for clustering can be from 3 to 5. This research will use other methods to make sure the optimal K-means later.

2. Support Vector Machine:

Setting seed to 123 before running any of these models listed below. K-fold cross-validations were used and optimal k should be 3 to 5. Thus, 3 and 5 fold cross-validation were used in all the models.

Svm Linear Kernel CV 3 fold

```
> confusionMatrix(svm.m1)
Cross-Validated (3 fold) Confusion Matrix
(entries are percentual average cell counts across resamples)
          Reference
Prediction    0    1
         0 84.5  3.2
         1  3.5  8.9
 Accuracy (average) : 0.9335
```

Svm Linear Kernel CV 5 fold, repeated 10 times

```
Cross-Validated (5 fold, repeated 10 times) Confusion Matrix
(entries are percentual average cell counts across resamples)
          Reference
Prediction    0    1
         0 84.9  4.6
         1  3.1  7.4
 Accuracy (average) : 0.923
```

Svm Radial Kernel CV 3 fold

```
> confusionMatrix(svm.m1)
Cross-Validated (3 fold) Confusion Matrix
(entries are percentual average cell counts across resamples)
          Reference
Prediction    0    1
         0 85.3  4.6
         1  2.6  7.5
 Accuracy (average) : 0.928
```

Svm Radial Kernel CV 5 fold, repeated 10 times

```
Cross-Validated (5 fold, repeated 10 times) Confusion Matrix
(entries are percentual average cell counts across resamples)
          Reference
Prediction    0    1
         0 85.1  4.4
```

1 2.9 7.7
Accuracy (average) : 0.9271

Svm Polynomial Kernel CV 3 fold

Cross-Validated (3 fold) Confusion Matrix
(entries are percentual average cell counts across resamples)

Prediction	Reference 0	Reference 1
0	85.5	4.3
1	2.5	7.8

Accuracy (average) : 0.9321

Svm Polynomial Kernel CV 5 fold, repeated 10 times

Cross-Validated (5 fold, repeated 10 times) Confusion Matrix
(entries are percentual average cell counts across resamples)

Prediction	Reference 0	Reference 1
0	84.9	3.9
1	3.0	8.2

Accuracy (average) : 0.9309

3. KNN:

KNN CV 3 fold

Cross-Validated (3 fold) Confusion Matrix
(entries are percentual average cell counts across resamples)

Prediction	Reference 0	Reference 1
0	85.5	2.4
1	2.5	9.7

Accuracy (average) : 0.9515

KNN CV 5 fold, repeated 10 times

Cross-Validated (5 fold, repeated 10 times) Confusion Matrix
(entries are percentual average cell counts across resamples)

Prediction	Reference 0	Reference 1
0	85.4	2.8
1	2.5	9.2

Accuracy (average) : 0.9463

4. Random Forest:

Random Forest CV 3 fold

Cross-Validated (3 fold) Confusion Matrix

(entries are percentual average cell counts across resamples)

Prediction	Reference 0	1
0	84.9	1.9
1	2.8	10.4

Accuracy (average) : 0.9529

Random Forest CV 5 fold, repeated 10 times (Highest Accuracy)

Cross-Validated (5 fold, repeated 10 times) Confusion Matrix

(entries are percentual average cell counts across resamples)

Prediction	Reference 0	1
0	85.8	1.8
1	2.1	10.3

Accuracy (average) : 0.9611

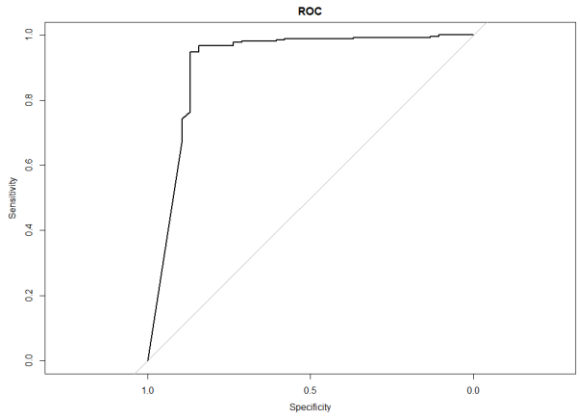

Figure 08 - ROC plot

No matter whether the condition of CV 3 fold or the condition of CV 5 fold repeated 10 times, the random forest algorithm had the best accuracy than other models. As a result, the best model for this investigation is the random forest algorithm.

This is the roc plot of random forest cross-validated 5 fold, repeated 10 times model, the line almost looks like a right angle and far away from the random classifiers, which means our test was very accurate.

Unsupervised Learning

Clustering:

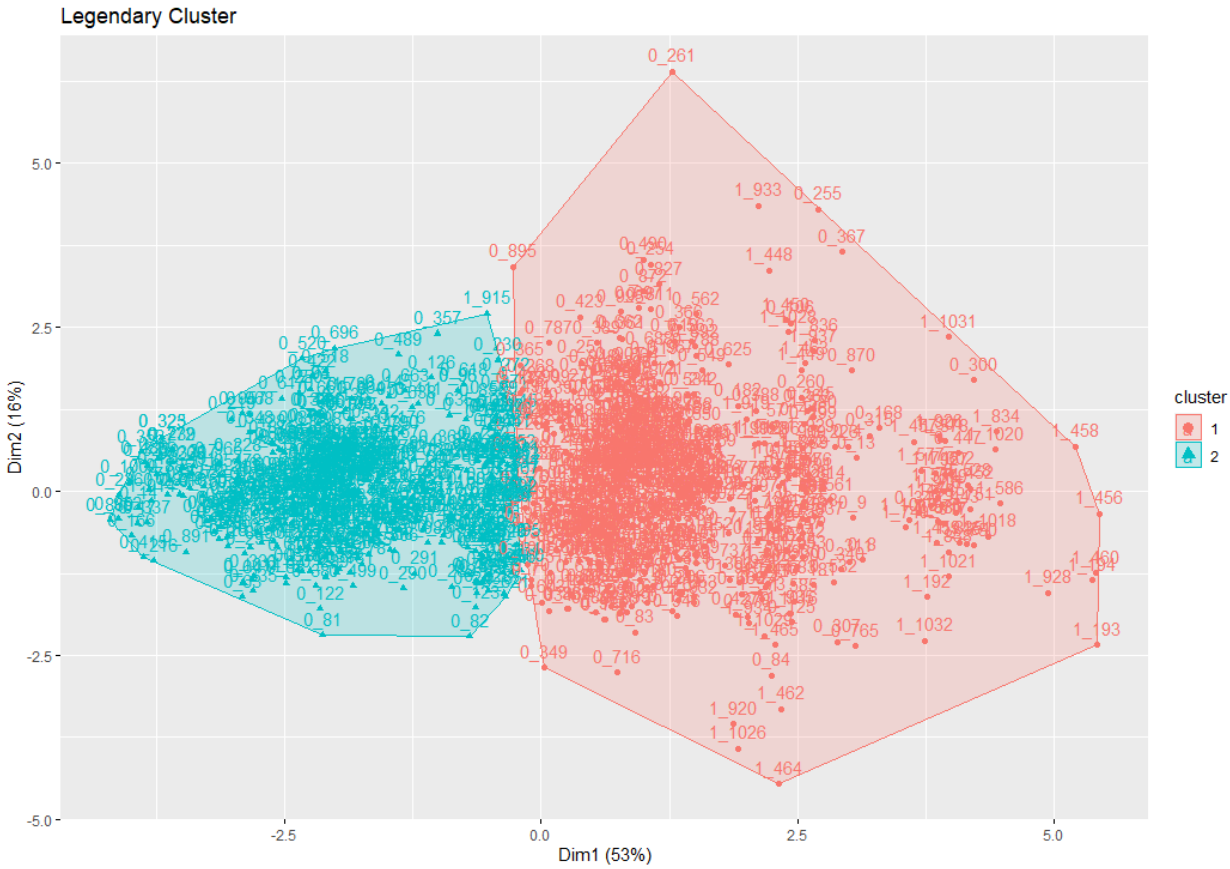

Figure 09 - Ward method two groups clustering

Our investigation labels the Pokemon id with 0_iiii and 1_iii to present whether the pokémon is legendary or not. Most of the left blue clustering groups are non-legendaries. Most of the pokémon in the red clustering group are legendaries. However, there are still a lot of non-legendary pokémon in the red clustering, such as 0_261 and 0_255. As a result, adding one more K for the clustering algorithm may give more detailed information.

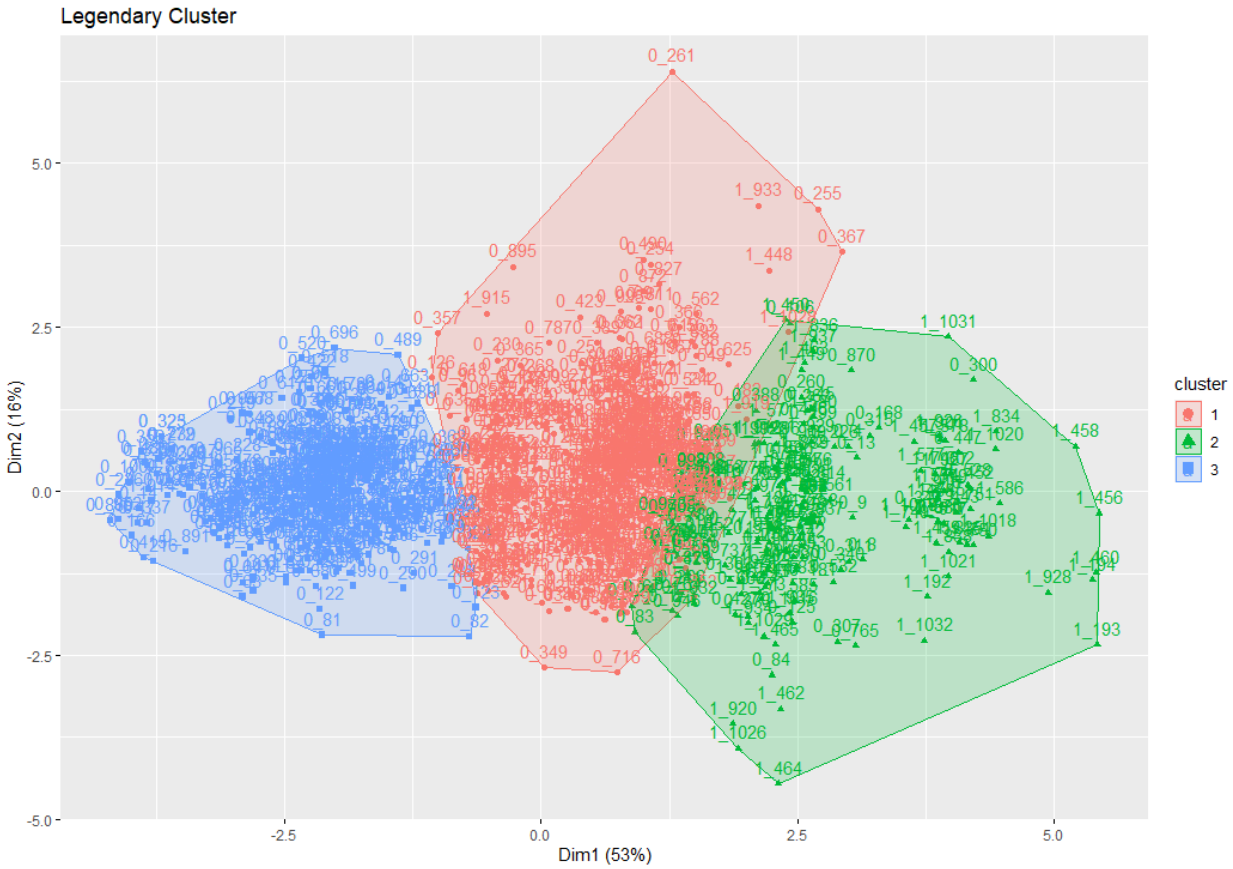

Figure 10 - Ward method three groups clustering

The above Figure 10 can point out more detailed information that most legendary pokémon have high stats than normal pokémon. The green clustering in Figure 10 only contains several non-legendaries. This makes sense since legendaries are rare and really hard to catch. And also, most legendaries have overwhelming stats than other non-legendaries. Players who do not encounter legendaries still can foster some non-legendaries instead since there are several of them in the green clustering.

Conclusion

In this investigation, more than one thousand unique pokemon from 8 different generations only contain a little more than one hundred legendaries. Our target is to find those legendary pokémon; thus, discovering the correlation between legendary pokémon and other pokemons' attributes is important. After creating some bar plots and histograms, noticing that legendary pokémon have correlations with their stats, generations, catch rates, etc. Amongst all the correlations, pokemons' based stats have the strongest correlation with whether the pokemon is legendary or not. Then, the decision tree, clusterings, SVM, KNN, and the Random Forest models were used for this investigation based on pokemons' stats. The decision tree roughly tells us the k-value can be 3 to 5. In the clusterings, we put pokémon into two and three groups, and legendary pokémon tend to belong to the groups that have higher overall stats, so the clusterings help to figure out legendaries have better stats than the normal pokemon. After using three-fold and five-fold cross-validation and repeated 10 times, the random forest model had the best accuracy for this investigation. We may consider using the random forest model for predicting the legendaries.

Data source: https://www.kaggle.com/akbarjaffery/pokmon-pokedex-gens-18