Pokemon Final Project

Alexa Fahrer and Nick Maroulis

Introduction and Data

As students who grew up playing Pokemon games, including Pokemon Go, _____, we are interested in examining the factors that contribute to catching Pokemon. After searching for data about Pokemon in relation to catch rate, we came across a data set from Kaggle called "The Complete Pokemon Dataset," with includes information about more than 800 Pokemon from all seven generations (https://www.kaggle.com/datasets/rounakbanik/pokemon?resource=download). The data set has 801 observations– each a unique Pokemon– and 41 variables that are a mix of quantitative and qualitative. For this project, we decided to focus on the predictors name, Pokedex number, generation, capture rate, percentage male, the type(s) of the Pokemon, height, weight, the number of steps per egg cycle (base egg steps), experience growth (XP), base happiness, hp, attack, defense, special attack, special defense, speed, and whether the Pokemon is legendary or not.

When attempting to catch a Pokemon, there are many variables that go into if the Pokemon will be caught or not. These variables include type of Poke ball used, level of the wild Pokemon, etc. All of these factors, in addition to RNG (random number generation), are factored into a specific formula to determine if a Pokemon is caught on any given attempt. Details about the formula can be found here: https://bulbapedia.bulbagarden.net/wiki/Catch_rate. Additionally, every Pokemon has its own "catch rate," which is weighted heavily in the formula. Pokemon with a higher catch rate are easier to catch. For example, Pidgey, which is a weak, unevolved Pokemon, has a catch rate of 255, which is tied for the highest, meaning that Pidgey is very easy to catch. However, Mewtwo, a strong, legendary Pokemon, has a catch rate of 3, which is tied for the lowest, meaning that Mewtwo is extremely difficult to catch. Please note that we use "catch rate" and "capture rate" interchangeably.

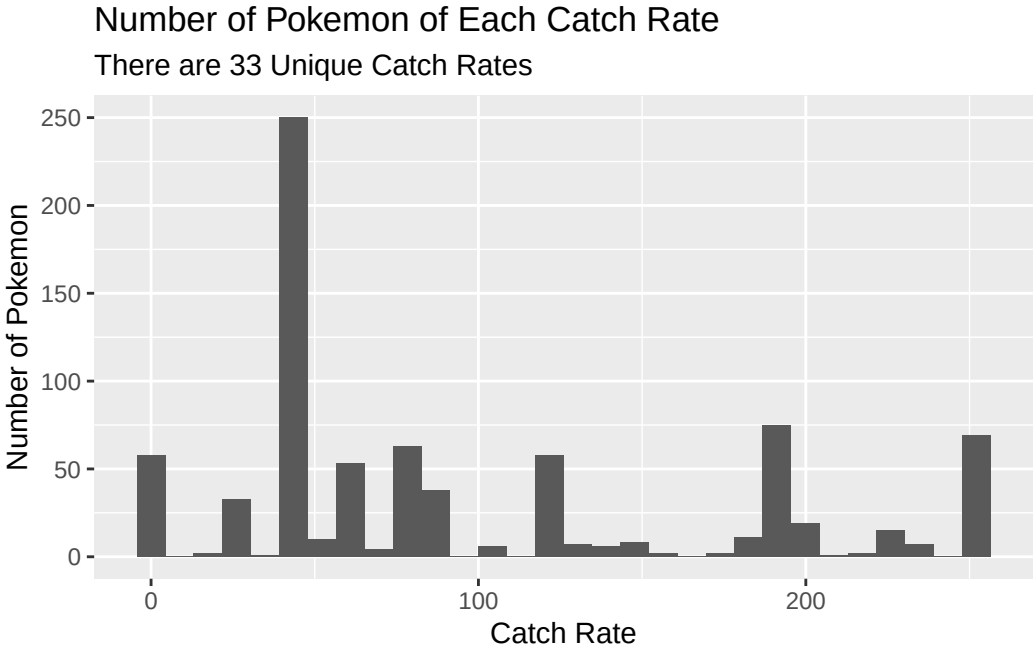
In this project, we are examining the following research question: What characteristics of a Pokemon influence catch rate, and can we develop a model that uses these characteristics to predict the catch rate of a Pokemon? Our main outcome of interest is the variable "capture_rate." Our hypotheses are as follows: larger Pokemon (in terms of height and weight) will have lower capture rates than smaller Pokemon. Pokemon with higher attack and defense stats will have lower capture rates than Pokemon with lower attack and defense stats. Finally, legendary Pokemon will have lower capture rates than non-legendary Pokemon. In

the following project, we will make visualizations to explore the data, consider different types of regression models, use variable selection techniques, consider missing data, and select a final model to examine the characteristics of Pokemon in the data set that are statistically significant in influencing capture rate.

Methodology

Type of Model

We decided a Linear Regression Model is the best model to predict the catch rate of a Pokemon.



In this data set, the catch rate of Pokemon is an integer from 3 to 255. However, there are only 33 unique catch rates among the 801 Pokemon. For example, 50 Pokemon have a catch rate of 60 and a whopping 250 Pokemon have a catch rate of 45. Based off of this information, it may make sense to use a multinomial regression model. Since the data are ordered, an ordinal regression model would be another good option. This would lead to a model that predicts the catch rate of a Pokemon relative to the 33 unique catch rates. However, this is a lot of outcomes for the response variable. We could simplify this ordinal regression model with some number of different "bins." For example, bin 0 could be all Pokemon with a catch rate of 0-50, bin 1 is all Pokemon with a catch rate of 50-100, etc.

We decided against using the ordinal regression model for multiple reasons. First, 33 different outcomes for the response variable is a lot. It creates an unnecessarily complicated model. This

problem can be fixed by grouping the catch rates into different "bins." However, this is not optimal, because we don't have a strategy for how to create the "bins." How many "bins" should we have? Should they all be of equal length? Without the proper tools, this method does not make sense for our purposes. However, overall, we decided against the ordinal regression model because even though many Pokemon share a catch rate with many others, there are still Pokemon that have a unique catch rate, meaning the current catch rates of Pokemon are not the only possible catch rates. There could still be a new Pokemon introduced that has a fully unique catch rate. If we were tasked with predicting the catch rate of a new Pokemon with certain attributes, our model would no longer make sense if that Pokemon's catch rate was fully unique. We would be looking at the probability that this Pokemon has a certain catch rate, but the Pokemon wouldn't have any of the listed catch rates.

A logistic regression model also does not make sense in the context of our data because our main outcome of interest, capture rate, is a numerical predictor, and logistic regression requires the response to be categorical and binary. This leaves us with linear regression, which best fits the data we are using since our response variable is numeric. Linear regression with multiple predictors will allow us to examine the relationship between our predictors of interest and outcome while holding other variables constant. Therefore, we decided that a linear regression model would best suit our research question.

Missing Data

Many species of Pokemon do not have a gender, leading to many NA values for the *percentage_male* variable. However, there is a disproportionate amount of legendary Pokemon that do not have a gender. This is because in a Pokemon game, legendaries are unique and special; there is only one of each legendary Pokemon in the "world." Thus, in our data set, 63/70 of legendary Pokemon have an NA for *percentage_male*. This data is MAR (missing at random) since legendary status is an observed variable in the data set. Because these Pokemon are legendary, many of them have extremely low catch rates. Specifically, 53/70 legendary Pokemon have a catch rate of 3, the lowest possible catch rate. This means that our model will most likely not be able to accurately predict legendary Pokemon's catch rate, and there will be worse representation among Pokemon with lower catch rates. The best and easiest solution to this problem is to eliminate the *percentage_male* predictor from the model.

Next, we noticed that while reading the .csv file, R automatically translated the *capture_rate* variable to characters. Upon inspection, Pokemon number 774, Minior, had this listed as its catch rate: "30 (Meteorite)255 (Core)". During battle, Minior has two forms: meteor form and core form. Minior has the "shields down" ability, which means that when Minior is at half health or less, Minior changes from the Meteor form to the core form. However, Minior's catch rate also changes when it changes forms, from 30 (meteor form) to 255 (core form). Due to Minior's ability having a direct relation to the catch rate, we chose to exclude Minior as an observation. We decided to update Minior's catch rate to be NA, thus excluding this observation from the model.

Many Pokemon only have one type, and thus do not have a second type. This is not missing data, this is just the classification of the Pokemon.

Variable Selection

After acknowledging missing data, we must next narrow down the variables in the data set to only the important predictors for our model.

For classification and readability purposes, we chose to keep the variables *name*, *pokedex_number*, and *generation*. The response variable is *capture_rate*, so we must of course keep this variable. Other important variables that may play a role in capture rate are *type1*, *type2*, *height_m*, *weight_kg*, *base_egg_steps*, *experience_growth*, *base_happiness*, *hp*, *attack*, *defense*, *sp_attack*, *sp_defense*, *speed*, and *is_legendary*.

We chose to manually eliminate some variables from the data set based on our knowledge of Pokemon. First, the *japanese_name* is just another version of *name*. The *against_?* variables are dependent on only typing (*type1* and *type2*), and it will be more useful to just look at the typing of a Pokemon rather than how effective certain moves are against it. There are hundreds of different abilities a Pokemon can have, and very little overlap of abilities between Pokemon, so we do not need the *abilities* variable. The classification of a Pokemon is almost unique for every Pokemon (there is very little overlap), and it mainly just groups Pokemon by their evolution line, yielding *classification* unwanted. Finally, the *percentage_male* variable is excluded due to missing data, as dicussed above.

We now run variable selection models to determine which of the following variables are best/important for predicting catch rate: *type1*, *type2*, *height_m*, *weight_kg*, *base_egg_steps*, *experience_growth*, *base_happiness*, *hp*, *attack*, *defense*, *sp_attack*, *sp_defense*, *speed*, and *is_legendary*.

First, we tried _____ We decided not to use forward selection and backward elimination methods because they often don't work well with highly correlated variables, and we suspect that some of our predictors are correlated.

```
  (Intercept) height_m weight_kg base_egg_steps experience_growth
1        TRUE    FALSE     FALSE          FALSE             FALSE
2        TRUE    FALSE     FALSE          FALSE             FALSE
3        TRUE    FALSE     FALSE          FALSE             FALSE
4        TRUE    FALSE     FALSE          FALSE             FALSE
5        TRUE    FALSE     FALSE          FALSE             FALSE
  base_happiness    hp attack defense sp_attack sp_defense speed is_legendary
1          FALSE FALSE  FALSE   FALSE     FALSE       TRUE FALSE        FALSE
2          FALSE FALSE   TRUE   FALSE     FALSE       TRUE FALSE        FALSE
3          FALSE  TRUE  FALSE    TRUE     FALSE      FALSE  TRUE        FALSE
4          FALSE  TRUE  FALSE    TRUE      TRUE      FALSE  TRUE        FALSE
```

```
5          FALSE  TRUE  FALSE    TRUE      TRUE       TRUE  TRUE        FALSE
```

DO VARIABLE SELECTION HERE

Interaction Terms?

Start with visualizing the data Hypothesis testing Interaction terms Transformations of variables? Variable selection Linear / logistic / ordinal / mulitinomial regression Missing data Mixed models

Prediction

Results

```
Call:
lm(formula = capture_rate ~ height_m + weight_kg + base_egg_steps +
    experience_growth + base_happiness + hp + attack + defense +
    sp_attack + sp_defense + speed + is_legendary, data = pokemon)

Residuals:
     Min       1Q   Median       3Q      Max
-161.066  -29.445   -4.779   34.315  217.509

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)        3.088e+02  1.879e+01  16.439  < 2e-16 ***
height_m           1.088e-02  2.505e+00   0.004   0.9965
weight_kg         -4.075e-03  2.538e-02  -0.161   0.8725
base_egg_steps    -1.099e-03  6.640e-04  -1.655   0.0983 .
experience_growth  1.514e-06  1.318e-05   0.115   0.9086
base_happiness    -1.496e-01  1.241e-01  -1.205   0.2284
hp                -5.702e-01  9.337e-02  -6.107 1.62e-09 ***
attack            -3.522e-01  8.130e-02  -4.331 1.68e-05 ***
defense           -4.762e-01  8.805e-02  -5.409 8.49e-08 ***
sp_attack         -4.036e-01  7.939e-02  -5.083 4.67e-07 ***
sp_defense        -4.730e-01  9.590e-02  -4.932 9.99e-07 ***
speed             -4.913e-01  8.204e-02  -5.988 3.26e-09 ***
is_legendary       2.317e+01  1.427e+01   1.624   0.1048
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 53.76 on 767 degrees of freedom
  (21 observations deleted due to missingness)
Multiple R-squared:  0.5058,    Adjusted R-squared:  0.4981
F-statistic: 65.43 on 12 and 767 DF,  p-value: < 2.2e-16



Call:
lm(formula = capture_rate ~ height_m * weight_kg + hp + attack +
    defense + sp_attack + sp_defense + speed + is_legendary,
    data = pokemon)

Residuals:
     Min       1Q   Median       3Q      Max
-160.838  -27.807   -4.873   31.614  216.618

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)        294.322211   8.492803  34.655  < 2e-16 ***
height_m            -3.505595   3.349198  -1.047    0.296
weight_kg           -0.034734   0.031999  -1.085    0.278
hp                  -0.590229   0.090686  -6.509 1.37e-10 ***
attack              -0.327013   0.081582  -4.008 6.71e-05 ***
defense             -0.457644   0.088715  -5.159 3.17e-07 ***
sp_attack           -0.398175   0.079142  -5.031 6.07e-07 ***
sp_defense          -0.455465   0.096253  -4.732 2.65e-06 ***
speed               -0.482409   0.082247  -5.865 6.65e-09 ***
is_legendary         4.785690   8.128106   0.589    0.556
height_m:weight_kg   0.012698   0.007846   1.618    0.106
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 53.71 on 769 degrees of freedom
  (21 observations deleted due to missingness)
Multiple R-squared:  0.5055,    Adjusted R-squared:  0.499
F-statistic:  78.6 on 10 and 769 DF,  p-value: < 2.2e-16
```

Discussion

Appendix

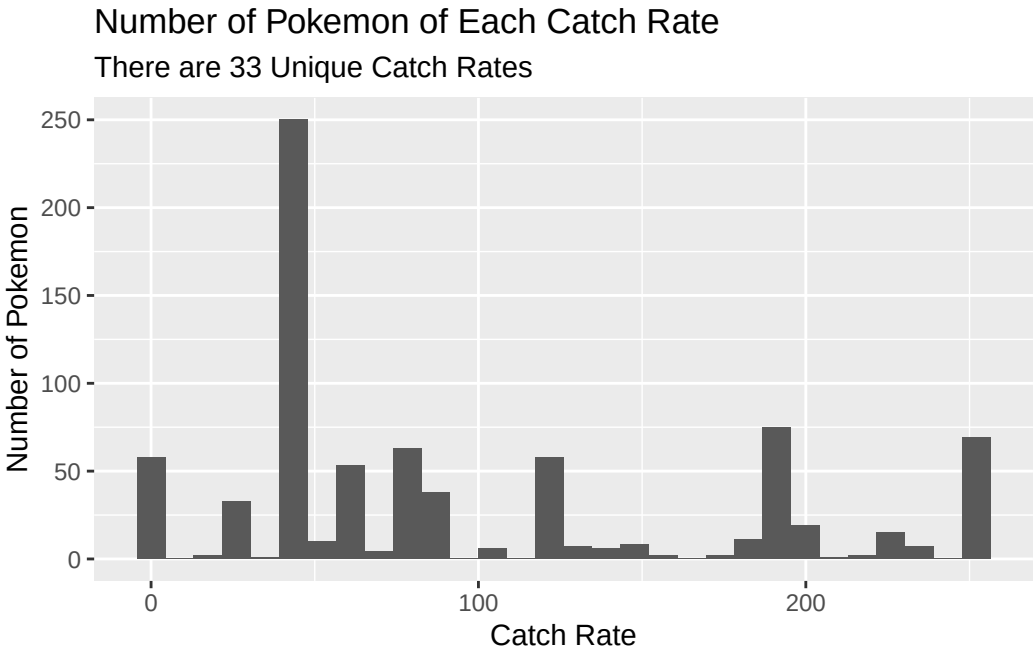
Here is a partial look at the Pokemon data set, including the transformations we completed (manually exluding some variables, changing *capture_rate* to be an integer).

```
        name pokedex_number generation capture_rate percentage_male type1
1  Bulbasaur              1          1           45            88.1 grass
2    Ivysaur              2          1           45            88.1 grass
3   Venusaur              3          1           45            88.1 grass
4 Charmander              4          1           45            88.1  fire
5 Charmeleon              5          1           45            88.1  fire
   type2 height_m weight_kg base_egg_steps experience_growth base_happiness hp
1 poison      0.7       6.9           5120           1059860             70 45
2 poison      1.0      13.0           5120           1059860             70 60
3 poison      2.0     100.0           5120           1059860             70 80
4             0.6       8.5           5120           1059860             70 39
5             1.1      19.0           5120           1059860             70 58
  attack defense sp_attack sp_defense speed is_legendary
1     49      49        65         65    45            0
2     62      63        80         80    60            0
3    100     123       122        120    80            0
4     52      43        60         50    65            0
5     64      58        80         65    80            0
```

```
   capture_rate   n
1             3  58
2            15   2
3            25  13
4            30  20
5            35   1
6            45 250
7            50   7
8            55   3
9            60  50
10           65   3
11           70   4
12           75  61
13           80   2
14           90  38
15          100   6
```

```
16          120  55
17          125   3
18          127   5
19          130   2
20          140   6
21          145   1
22          150   7
23          155   1
24          160   1
25          170   2
26          180  11
27          190  75
28          200  19
29          205   1
30          220   2
31          225  15
32          235   7
33          255  69
34           NA   1
```

```
    name pokedex_number generation capture_rate percentage_male type1  type2
1 Minior            774          7           NA              NA  rock flying
  height_m weight_kg base_egg_steps experience_growth base_happiness hp attack
1      0.3        40           6400           1059860             70 60    100
  defense sp_attack sp_defense speed is_legendary
1      60       100         60   120            0
```


Number of Pokemon of Each Catch Rate
There are 33 Unique Catch Rates
Number of Pokemon
250
200
150
100
50
0
0
100
200
Catch Rate