Determining Pokemon Statistics with MARS

Frank G DeVone

December 17, 2019

Introduction

Non-linear functions allow us to examine non-linear relationships between our variables within the framework of traditional linear regression. We will focus in this analysis on MARS (Multivariate Adaptive Regression Splines), which produces splines in a step-wise matter. To test this method we observe the Pokémon dataset compiled on Kaggle [2]. Pokémon is a video game series that's been ongoing since 1993, and contains pre-constructed and artificial statistics on each of its 801 imaginary creatures. This dataset full of designed relationships is a fun and useful, and wewill use it to demonstrate the MARS method.

MARS Background

Multivariate Adaptive Regression Splines functionally work in a similar manner to traditional splines. Note figure 1 [1]; we form our splines around knots at point t. We can represent this in the basis functions:

(1)

$$(x - t)_+ = \begin{cases} x - t & \text{if x>t} \\ 0 & \text{otherwise} \end{cases}$$

(2)

$$(t - x)_+ = \begin{cases} t - x & \text{if x<t} \\ 0 & \text{otherwise} \end{cases}$$

The key feature of MARS is that in addition to acting as a traditional covariate step-wise it also tests different values of t for possible splines. It accomplishes this through two passes of the data:

The Forward Pass: This part is similar to our traditional forward step-wise regression, only in addition to new and existing terms it also considers possible basis functions or a product of two. We begin with just

the intercept and we then add terms in an attempt to minimize the residual sum-of-squares. Note that this pass alone creates a large model that is prone to over fitting.

<u>The Backward Pass</u>: After the forward pass is done we begin deleting terms based on the smallest increases in residual squared error when we do so. Where we define λ as the number of predictors in the model, we create a best fitting model for each possible value of λ; from the intercept model to the full number of terms found in the forward pass. Once that process is complete we select the value of λ via a generalized cross-validation criterion:

Figure 1: Knots

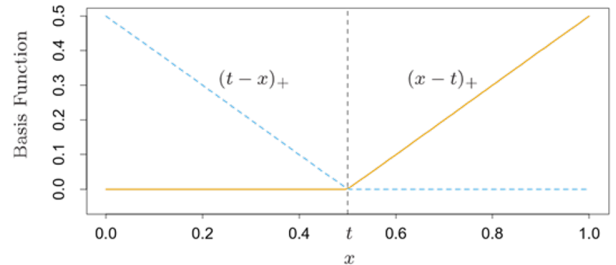$$(3) \ \ GCV(\lambda) = \frac{\sum_{i=1}^{N}(y_i - \widehat{f_\lambda}(x_i))^2}{(1 - \frac{M(\lambda)}{N})^2}$$

Note that "M(λ) is the effective number of parameters in the model: this accounts both for the number of terms in the models, plus the number of parameters used in selecting the optimal positions of the knots."[1] Our final model is the one that minimizes this generalized cross-validation criterion. From our perspective the final model would be similar to something we could produce with a GAM (Generalized Additive Model), though the processes as we described above to get the model are quite different. They are easier to interpret then most non-linear functions of a similar nature, and (critical to our analysis) can handle categorical as well as numerical data. Though MARS is capable of measuring three dimensional relationships as interactions (note product of two terms in the forward pass), for our analysis we will strictly observe two dimensional splines. Early checks into including these interaction splines found that they didn't improve our models much, were hard to interpret, and drastically increased computation time (already a bit of an issue). As such we proceed forward with a MARS analysis that will seek to build a model with our chosen covariates, all possible values t and the spines they'd create, and interactions terms on their own that we will define.

Pokémon Background

A Pokémon is a creature within the fictional world of the video games with extraordinary natural and sometimes supernatural abilities. Our dataset contains data on the eight hundred and one Pokémon species that were know prior to November 2019. A "Generation" of Pokémon refers to the games in the series. The original 151 Pokémon were from the first series of games released in 1993. When the sequels to those games were released, they introduced an additional one hundred Pokémon. This pattern continues up until generation 8 with the amount of Pokémon added varying from game to game. We elect to exclude the most

recent generation, as information on that generation has yet to be reliably compiled. The purpose of the game is both the collection of all available Pokémon, working your way through the story and world with them, and using them in a turn-based battle system against the environment, other players, and computer-controlled opponents. Pokémon have 'stats' that determine their effectiveness in these battles and react with the 'stats' of the opposing Pokémon. Attack deals physical damage and is mitigated by the opposing Pokémon's defense. Special attack deals special damage (think flames and electricity) and is mitigated by the opposing Pokémon's special defense. Damage is dealt to a Pokémon's hp (health points) and Pokémon can no longer participate in battle when hp reaches zero. The final 'stat' is speed, the Pokémon with the highest speed attacks first. These six stats add up to a Pokémon's "Total Stats" which will be the initial focus of our analysis as we attempt to predict it with the Pokémon's other characteristics. Within the games themselves these stats are calculated with the following:

(4) $Stat = [\frac{(2*Base+IV+EV)*level}{100}] * nature + 5$

(5) $Stat_{hp} = [\frac{(2*Base+IV+EV)*level}{100}] + level + 10$

'Base' as listed above are consistent across all Pokémon of the same species. However, IVs, EVs, nature, and level are characteristics the individual Pokémon is "born" with or can be trained by the player. As such our focus will only be on the base stat for a Pokémon species. There are additional characteristics that are consistent across Pokémon species, and we're using these as our potential covariates. The only ones related to real world comparisons are height and weight. Additional characteristics are fictional and will have to be explained. Understanding these characteristics is not critical to our model as we focus mostly on prediction, however their descriptions will be listed below for those interested:

Generation: As previously mentioned we are observing 8 generations of Pokémon. Conveniently each generation happens on a different "region" in the game's fictional world. As such I'm choosing to include it as a categorical variable as an indicator of the Pokémon species' region of origin.

Typing: There are 18 Pokémon types that a Pokémon can have. It is possible for them to have a single type, or two types, but no more then that. Note figure 2 for the listed types and some examples of Pokémon that fit certain paradigms. These interact with the attacks a Pokémon can learn which are also typed. For example, the electric type move thunderbolt will deal two times the damage to a Pokémon with water typing but will only deal half damage to Pokémon with rock typing. In addition, if the Pokémon that used the move is an electric type it will deal one and a half times more damage. When a Pokémon has two types, one is listed first; but from a gameplay prospective there is no difference between the two. Regardless we'll

include the first as "type 1" and the second as "type 2." We also create an additional "type 2" as "none" for Pokémon with only a primary typing.

Legendary Status: Some Pokémon are designated as legendary and are designed to have greater stats then normal. As such we make a binary variable to indicate if a given Pokémon species is legendary.

Evolution: For some Pokémon species; when certain conditions are met, Pokémon can evolve into (usually) strictly better versions of themselves. To measure this phenomenon, we created a numerical variable. A fully evolved Pokémon will be three, a Pokémon that can evolve once will be two, and a Pokémon that can evolve two more times will be one. Like legendaries, there are 'baby' Pokémon that are inherently designed to be weaker and must be evolved in order to reach their potential, these are given a 0.

Base Egg Steps: Within the game you can receive eggs that, given time, will hatch into Pokémon. Time is judged by how many in game steps the player makes, when the threshold is met the egg will hatch. This is consistent across Pokémon species.

Figure 2: Pokémon Types

Base Happiness: Individual Pokémon have happiness levels that increase and decrease dependent on player actions. However, each species has a baseline level of happiness they begin with.

Experience Growth: Recall in Equation 4 and 5 the "level" component. This is a value increased as a Pokémon is used in battle. However, the amount of experience required to increase "level" varies between Pokémon species, typically with stronger Pokémon taking longer to level up.

Capture Rate: There are several in game factors when attempting to capture a 'wild' Pokémon; however each has a default capture rate under the games' baseline conditions. Like experience the capture rate varies between Pokémon species, typically with stronger Pokémon being harder to capture.

Data

As previously stated our initial analysis will focus just on base total, the total pool of a Pokémon's stats. We removed Rotom; the unique ghost/electric type Pokémon from our dataset given their ability to switch forms and stats by possessing real world appliances like refrigerators and lawnmowers. Some Pokémon also

have multiple forms, either through special game mechanics or through regional variants, with different stats being present for each. We don't wish to include these, however we include the base form only in our study. The exception to this is mega evolution, which we will discuss at the end of this section.

Note in figure 3 a correlation plot between all of our covariates excluding type, and below a pairs plot (figure 4) of the same with the variables standardized. Note that we have both heavy correlation and apparent linear relationships between several variables. In addition we note that several of our variables are grouped; experience growth and base happiness has 6 unique values, while egg steps has 10 unique values. Despite this they are factors where their
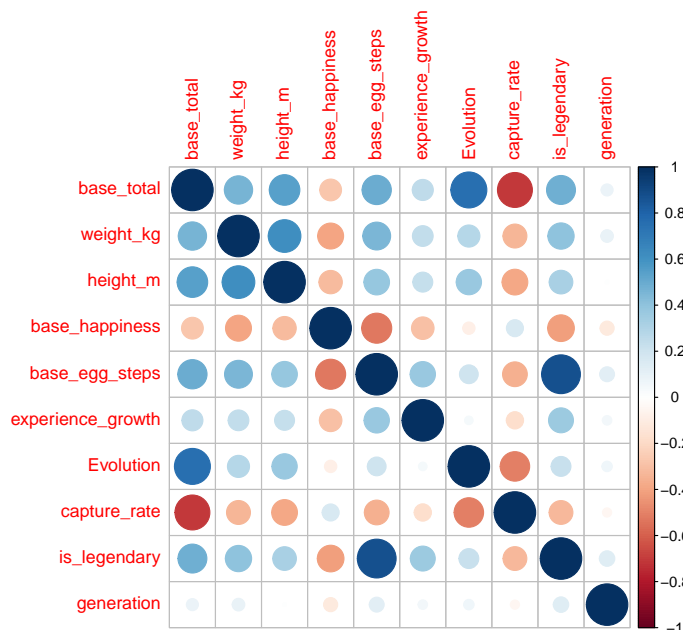

Figure 3: My Flowchart

number has meaning beyond a categorical status, and most appear to be increasing/decreasing as the Pokémon's total stats appear to be increasing. We initially choose to observe interaction in height and weight, egg steps and happiness, as well as egg steps and legendary status. Note that legendary status is rare, making up only 9% of our population. However these are Pokémon essentially designed to be outliers, we hope that this variables helps us balance out this fact. We'll make note of this again later but happiness did not end up being chosen by MARS, so we removed it's interaction term manually. Considerations from how the game world is expected operate also led us to observe the interaction between base egg steps and capture rate. Both are essentially gateways to obtaining a Pokémon, and we'd expect the growth or decline of one to mesh with the other. Additional research also revealed that the correlation between egg steps and legendary status is artificial. Legendaries cannot actually have eggs, and those values are placeholders so commercial hacking devices that force the game to generate legendary eggs don't break the game. In its place we choose to observe the interaction between legendary status and experience growth; a relationship we know should exist as legendaries are purposely made overly difficult to train as a balance on their power.

In figure 5 we view the mean total stats stratified by primary and secondary Pokémon type, with red being the primary typing and the black lines being the secondary typing. There is a notable difference between both the different typing, and in some cases the effect of the type as a secondary or primary typing. This justifies looking at the relationship between type one/two and base stats, as well as the interaction between them. As such our chosen base model is:
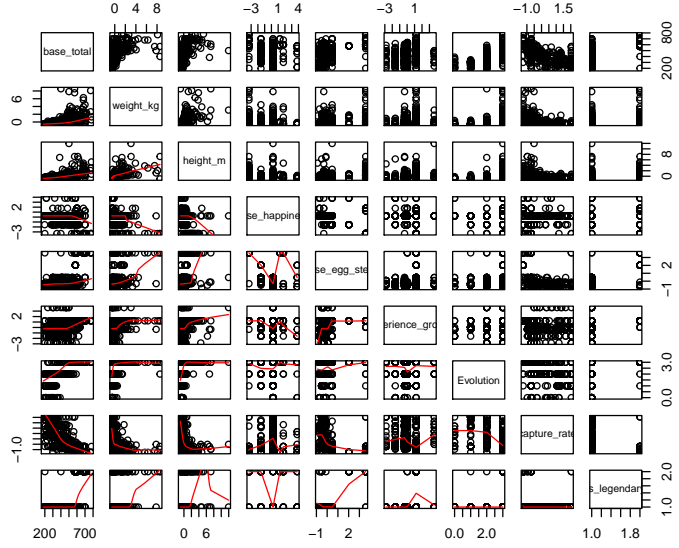

Figure 4: My Flowchart

(5) $basestats\ type1 + type2 + heightm + weightkg + islegendary + experiencegrowth + generation + baseeggsteps + baseeggsteps * capturerate + capturerate + type1 * type2 + heightm * weightkg + islegendary * experiencegrowth$

While observing the data we realized that a handful of Pokémon had higher stats then we'd expect. In Generation 6 a concept called "Mega-evolution" was introduced. It allows for a in battle form and stat increase. I realized that the web-scraping for the dataset (Kaggle sourced) is grabbing these stats instead of the true base stats. These would have to be manually changed, which is outside the scope and time restraints of this project. 56 out of 801 (5.8%) of Pokémon can mega-evolve, and for the purpose of this analysis we assume these are their "correct" stats.
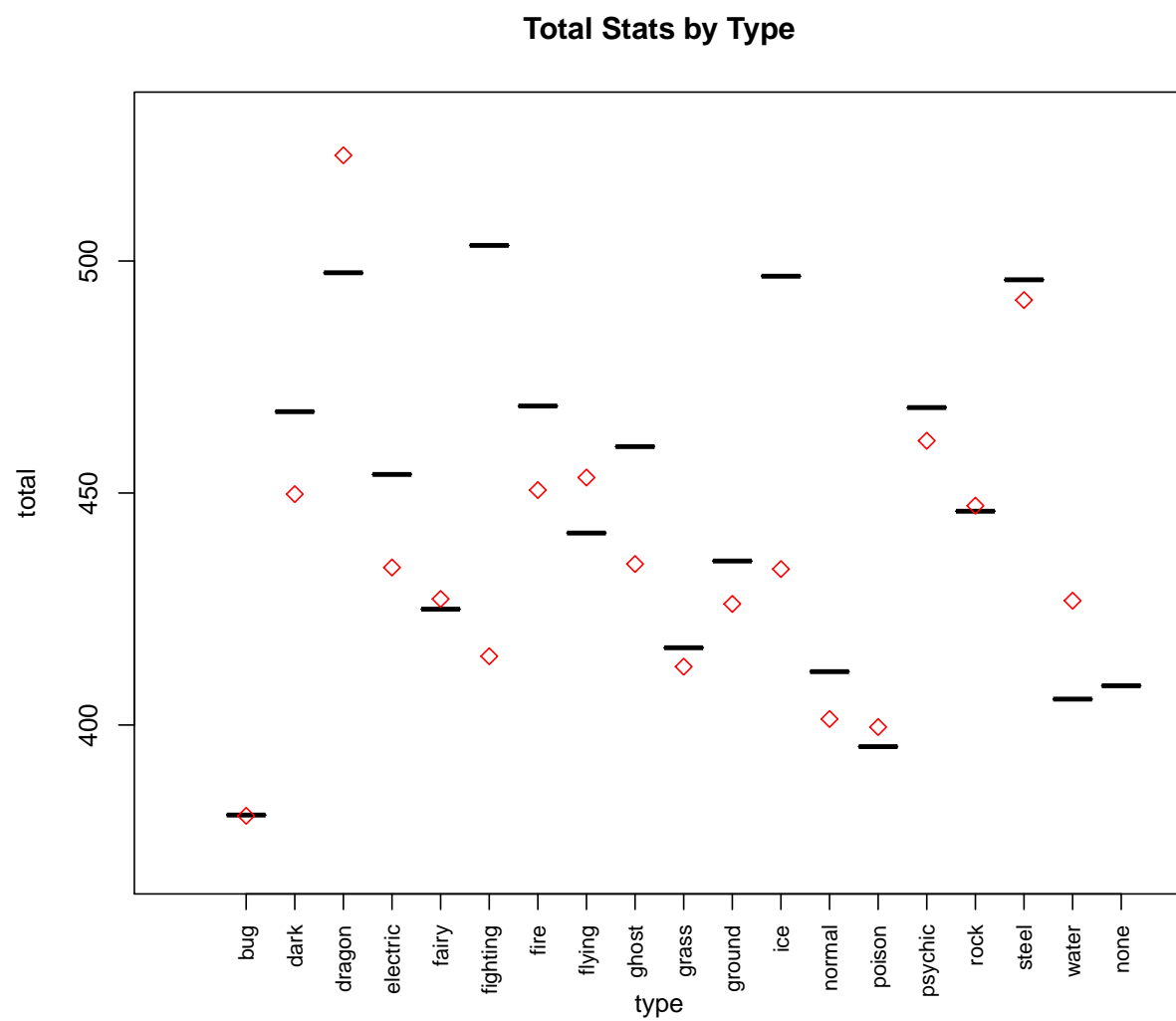
Figure 5: Total Stats by Type

'Total Stats' Analysis

We proceed forward with equation 5 as our base model. For our MARS analysis we use the earth [4] package. This allows us to perform not only perform our step-wise analysis, it also allows us to perform cross validation within the MARS process. There are very strong subsets in our data, as such we choose to implement ten-fold cross validation, which we perform fifty times. We report a R^2 of .8620 with a standard deviation of 2.01 across our folds and cross validations. Table 1 lists our coefficients, while figure 6 shows the splines graphically. It's worth noting that a traditional stepwise produces a R^2 of .8135, lower but still good. However It relies heavily on type over the other variables, likely meaning it is more overfit, where as splines allow us to make similar inferences with the continuous variables.

Table 1: Table 1

	base_total
(Intercept)	453.183912
h(Evolution-2)	44.509092
h(capture_rate–0.641389)	-43.246467
h(2.71591-height_m)	-29.923972
h(base_egg_steps–0.12094)	-14.854871
h(-0.12094-base_egg_steps)	-63.309470
h(experience_growth-1.21573)	-12.320847
h(1.21573-experience_growth)	-14.999033
type1dragon:type2flying	126.638550
type1fairy	59.311369
h(-0.0889986-weight_kg)	-63.047054
is_legendary1	210.559350
h(capture_rate:base_egg_steps–2.54373)	8.163230
h(-2.54373-capture_rate:base_egg_steps)	40.451879
is_legendary1:experience_growth	-102.276379
type1rock:type2dark	166.366905
type1fire:type2ground	109.755237
generation2	-16.335411
type1steel:type2ghost	84.158279

	base_total
type2steel	26.642377
type1grass:type2poison	33.320044
type1rock:type2flying	54.549863
h(height_m:weight_kg-0.0836128)	-1.562577
h(capture_rate-0.274437)	32.565468
h(Evolution-1)	51.664960
type1fighting:type2none	-24.805810

We have heavy outliers in height and weight, so what we're observing is that for the bulk of our height and weight values we have a normal regression. But this is completely flattened at a certain point to control for the outliers. As previously stated happiness was not included, initially the interaction term was however we manually removed it. There is a small effect if a Pokémon is from generation two, however in general they appear to have no effect. Legendary status has an extraordinary effect, which we expect, while most typing does not. Fairy typing is the newest type to be introduced, so that may explain it's positive effect on base stats. A primary typing of fighting with a secondary typing of none is the only typing effect that is negative, while steel is the only secondary typing that is positive. The interactions, such as type1steel:type2ghost are pointing to one or only a handful of Pokémon. They are acting more as controls on these unique Pokémon, but could also bee seen as a symptom of over-fitting.

Evolution acts as expected, though we see no distinction between baby Pokémon and those at the lowest stage of a three part evolution. The remaining variables vary and have smaller effects the we might have expected. Capture rate makes sense as a harder to obtain Pokémon should have higher stats. Experience growth does initial increase as we'd think it would, but appears to decrease slightly towards our maximum values. Base egg steps paradoxically does not appear to act as we'd expect, decreasing beyond the standardized average. Upon further thought Pokémon that evolve have the same base egg steps as their lowest evolution (when a evolved Pokémon has a egg it is of their lowest evolution), so we are likely observing this effect.
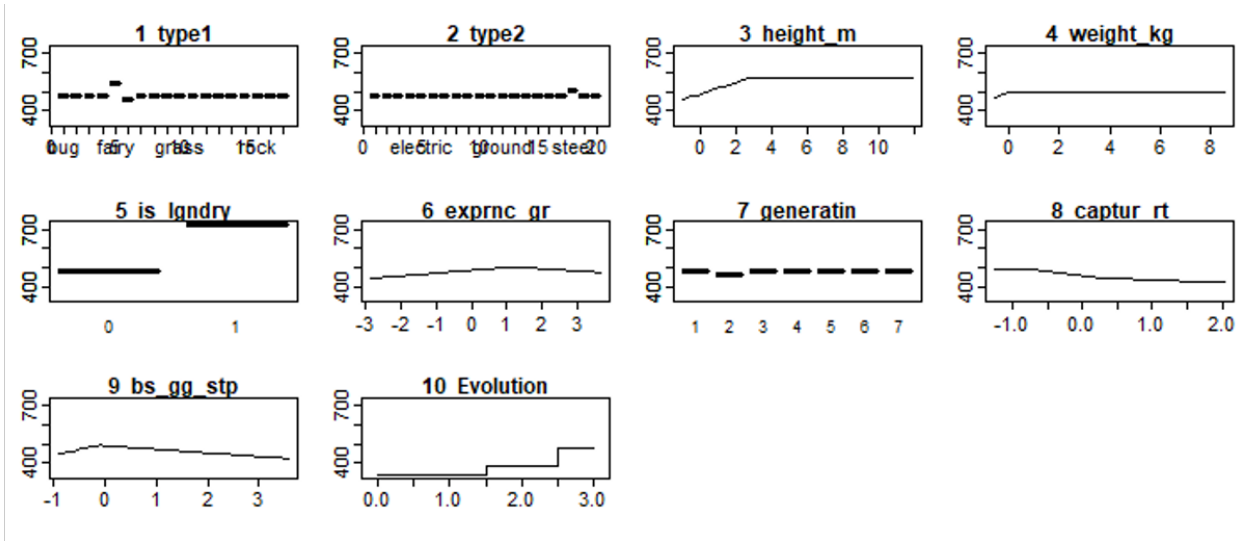
Figure 6: Base Stats Splines

Our model diagnostics are also quite sound. The QQ plot has heavy tails, which we expect given the nature of the outlier filled dataset, however the center appears quite normal. The Residuals vs. fitted appear to qualify our assumption of homosecedasticy by being mostly balanced around a single horizontal line at zero. As fits the general pattern their are outliers regardless.
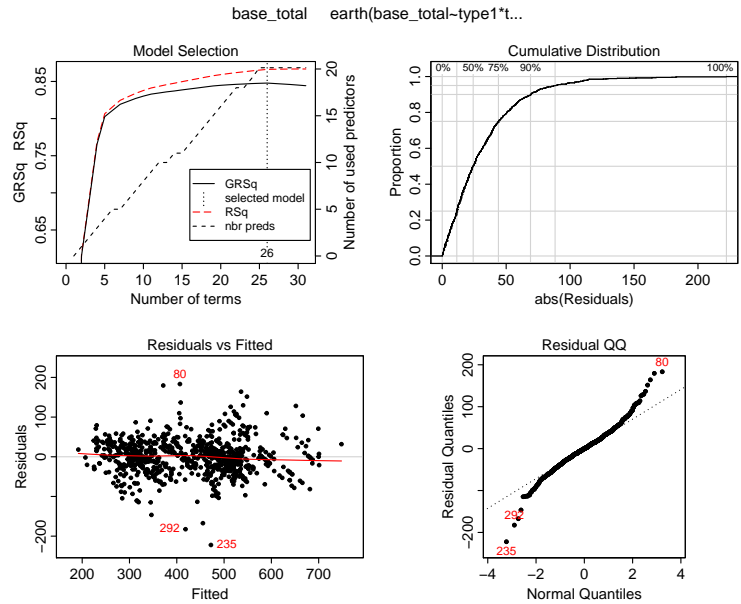
In figure 8 We have sorted all 800 Pokémon by their predicted base total. The black

Figure 7: Model Diagnostics

line consists of these points, with 95% confidence bands around it. The colored dots are the true values for those Pokémon, blue if within the confidence bands and red otherwise. Only 4% of our Pokémon predictions were outside out confidence bands! Combine this with the good R^2 value and we are confident in our models ability to predict the base stats of most Pokémon. The outliers also tend to be Pokémon like ditto (transforms

into other Pokémon) or Magikarp (intentionally weak with a powerful evolution), who have special abilities outside their stats to give them advantages. A version of this figure with the outlier Pokémon visualized can be found in our final section labeled "Visual Predictions Per Variable and Additional Graphs."
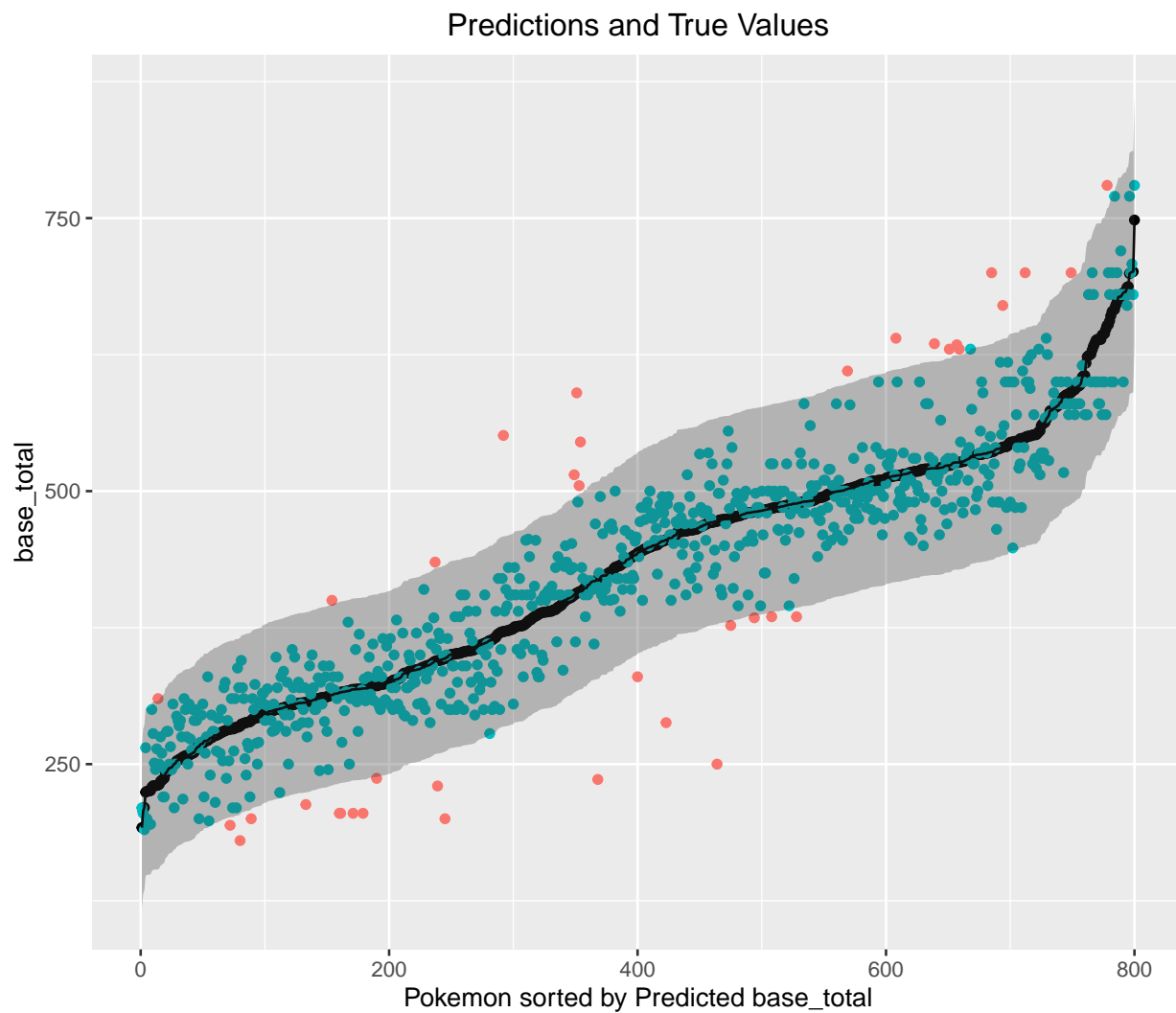

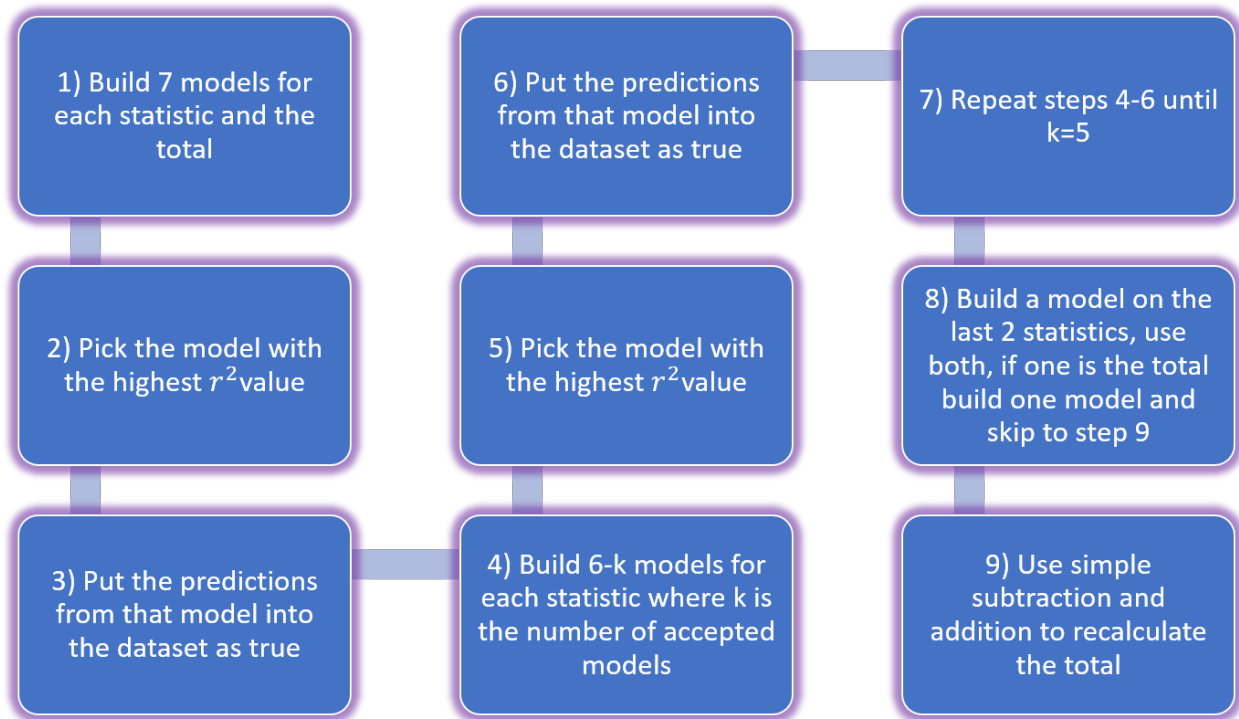Figure 8: Full Base Results

Full Analysis

1) Build 7 models for each statistic and the total	6) Put the predictions from that model into the dataset as true	7) Repeat steps 4-6 until k=5
2) Pick the model with the highest r^2 value	5) Pick the model with the highest r^2 value	8) Build a model on the last 2 statistics, use both, if one is the total build one model and skip to step 9
3) Put the predictions from that model into the dataset as true	4) Build 6-k models for each statistic where k is the number of accepted models	9) Use simple subtraction and addition to recalculate the total

Figure 9: Proposed Process

We will build off the results of our analysis in the last section in order to build a full model for a fictional scenario. Let's say, within the world of Pokémon we have all information on generations one through six. We then discover generation seven and promptly set to categorizing them. We will treat generations one through six as our training data and generation 7 as our test data in this scenario. Using MARS we will attempt to predict the stats of these "new" Pokémon as accurately as possible. Figure nine walks through the pace of this process. In addition note that each time a stat is confirmed, it will be included in the next model as a possible predictor. For example if defense is chosen in step 2, then in step 5 it will be included in our model if MARS selects it. Once the models are build we will use the predict function to build an analysis as in figure 8 to observe the accuracy of our predictions. Pokémon stats must be at least one. If any stat is found to be less then one it will be manually changed along with the total to compensate.

Table 2: Process Results

stats	Pass 1	Pass 2	Pass 3	Pass 4	Pass 5	Pass 6	PassKept	PercentError
Attack	0.6504125	0.7292287	0.7308185	0.7481108	NA	NA	4	7.50
Base Stats	0.8718046	NA	NA	NA	NA	NA	1	16.25
Defense	0.6592608	0.7081765	0.7493421	NA	NA	NA	3	16.25
HP	0.6430970	0.6880077	0.6802731	0.7129220	0.7356434	0.7500085	6	8.75
Sp Attack	0.6757858	0.7779986	NA	NA	NA	NA	2	16.25
Sp Defense	0.6599160	0.7135845	0.6976845	0.6951320	0.7887812	NA	5	21.25
Speed	0.4698838	0.5622602	0.5715792	0.6710497	0.6913079	0.7819206	6	7.50

In table 2 we outline going through our proposed process and the results we obtained. "PassKept" refers to the point in our process when we accepted a model, and percent error is the amount of Pokémon whose true stat is within our confidence bands. This is graphically represented in the section labeled "Visual Predictions Per Variable and Additional Graphs" after our conclusion. Initially our process was different, in our first attempt we excluded base stats until the end. However as can be seen in the pass 1 column without base stats out predictive power is very low on our other models. Once it is incorporated into our models in pass 2 there is a increase across the board in our R^2 values. In our second attempt we did not recalculate our base stats at the end, instead whichever model was not chosen in pass 6 was calculated though subtraction of all other stats from base stat. Unfortunately this variable ended up having unacceptable percent error," as high as 90%. So while we now calculate base stats first, we recalculate it at the end. It has a higher percent error this way (it was previously about 6%), but it is much lower then the approximately 90% error we were getting on our calculated statistic before.

Conclusion

Ultimately we were able to make a good model to predict a Pokémon's base stats, and a acceptable method to predict all of their stats. Ideally we'd have the data on the full new generation, released in November of 2019, however that data is not yet available from our source. The web-scraping of this sort of data is beyond the scope and timeline of this project, as would fixing the mega-evolution problem previously mentioned. Regardless we have presented the MARS method, and found success in it. If a research suspects the inclusion of splines may aid their research, MARS (and the EARTH function) provide a simple and interpretable way

to do so.

Visual Predictions Per Varaible and Additional Graphs

Note the results of our first analysis with the outlier Pokemon visualized:
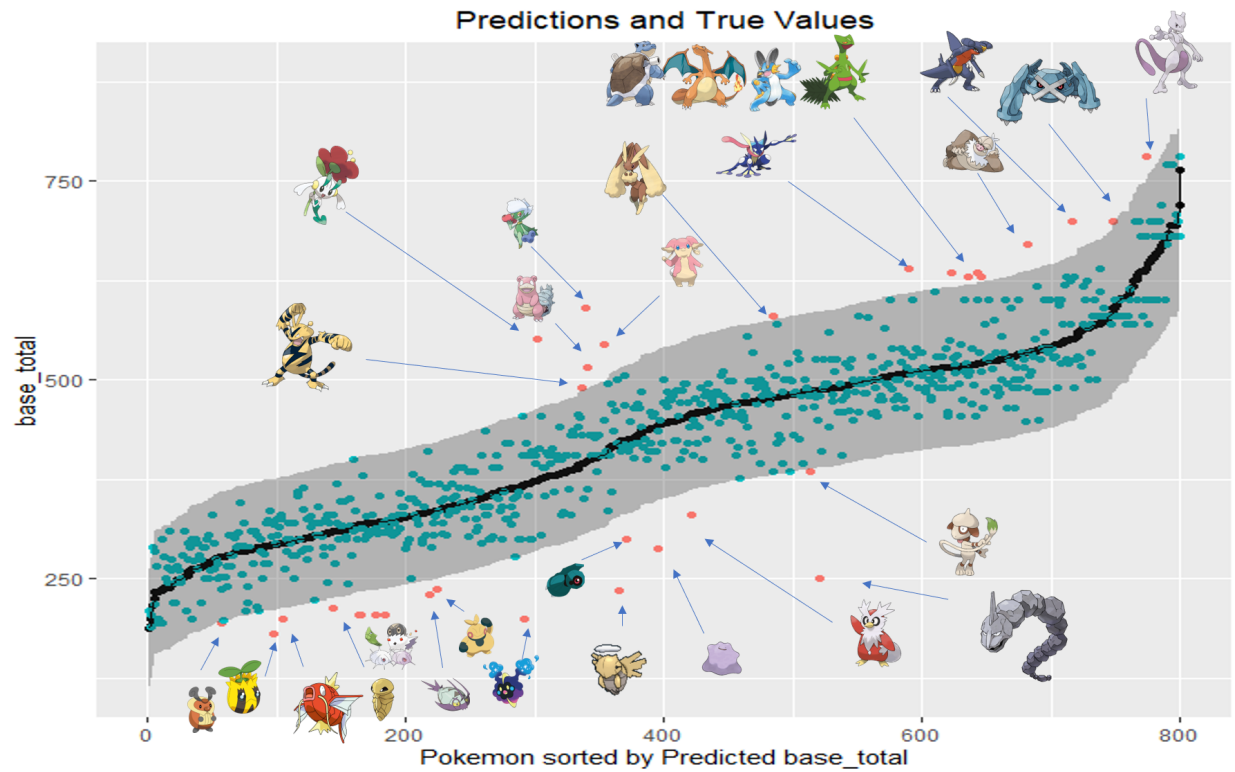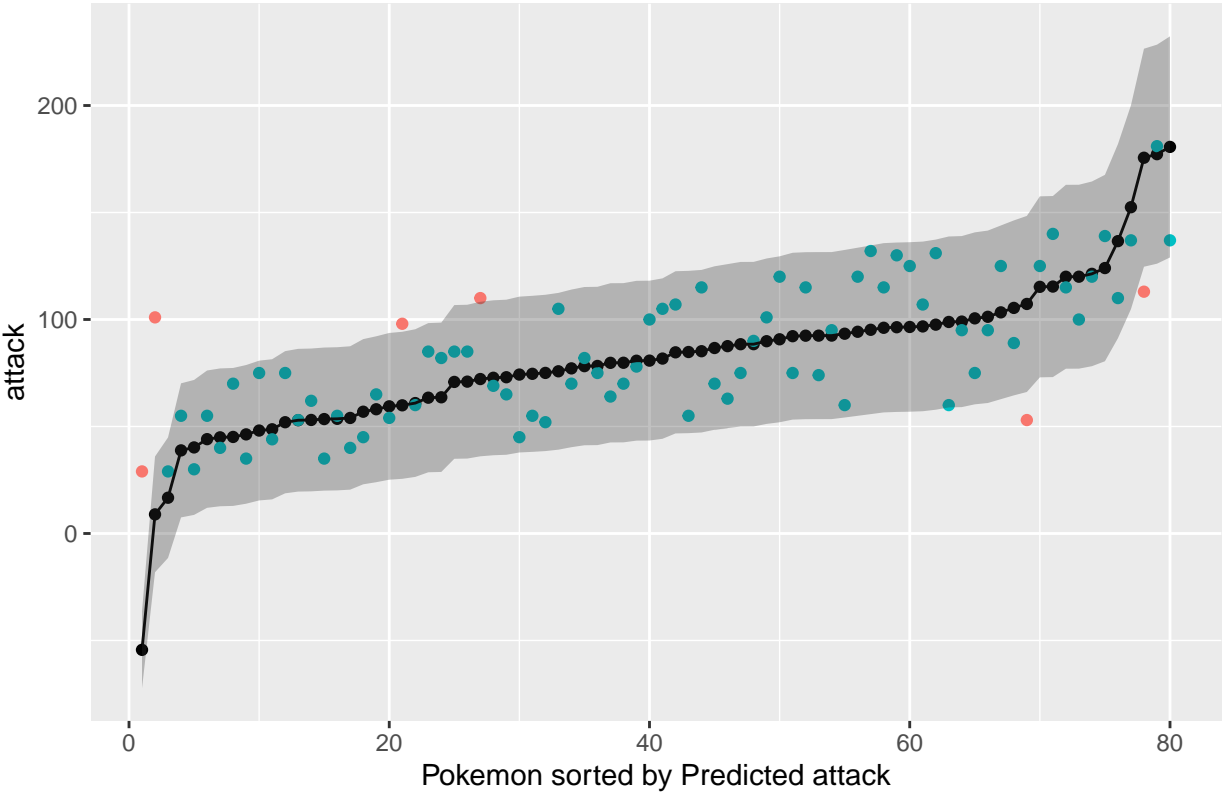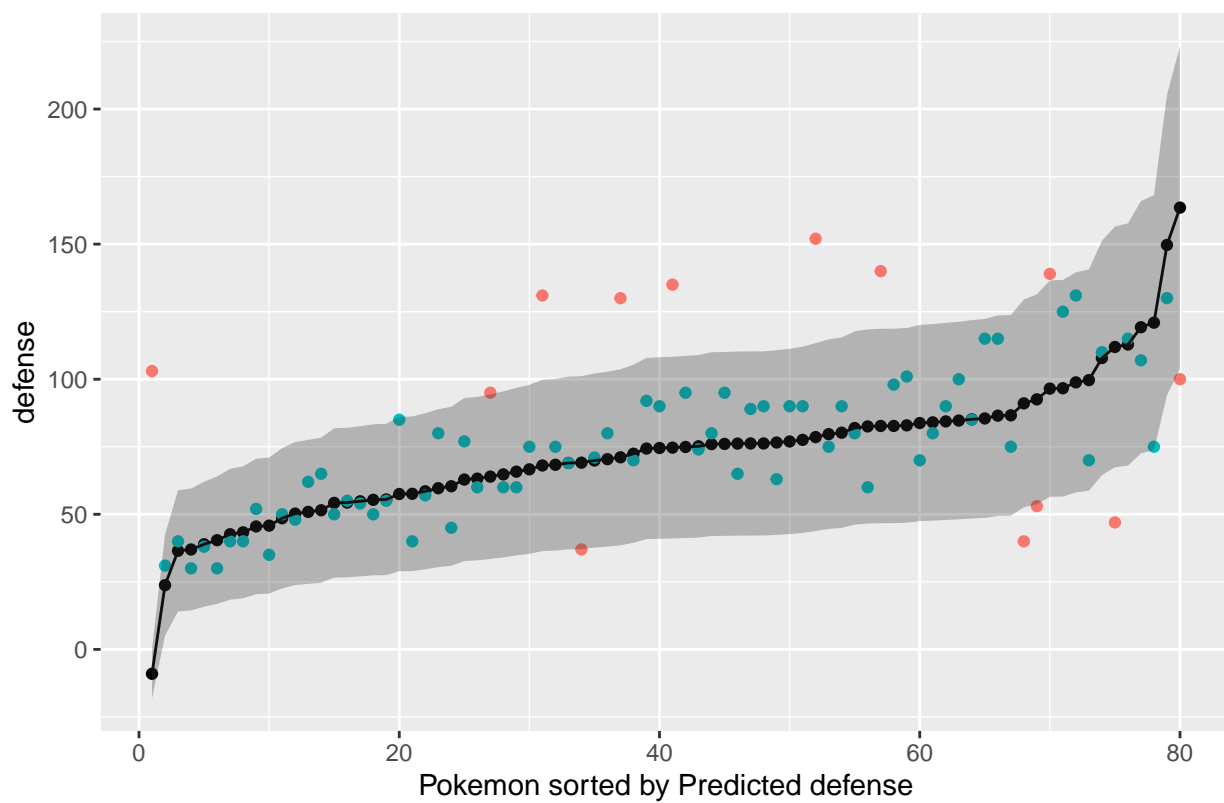

Figure 10: Proposed Process

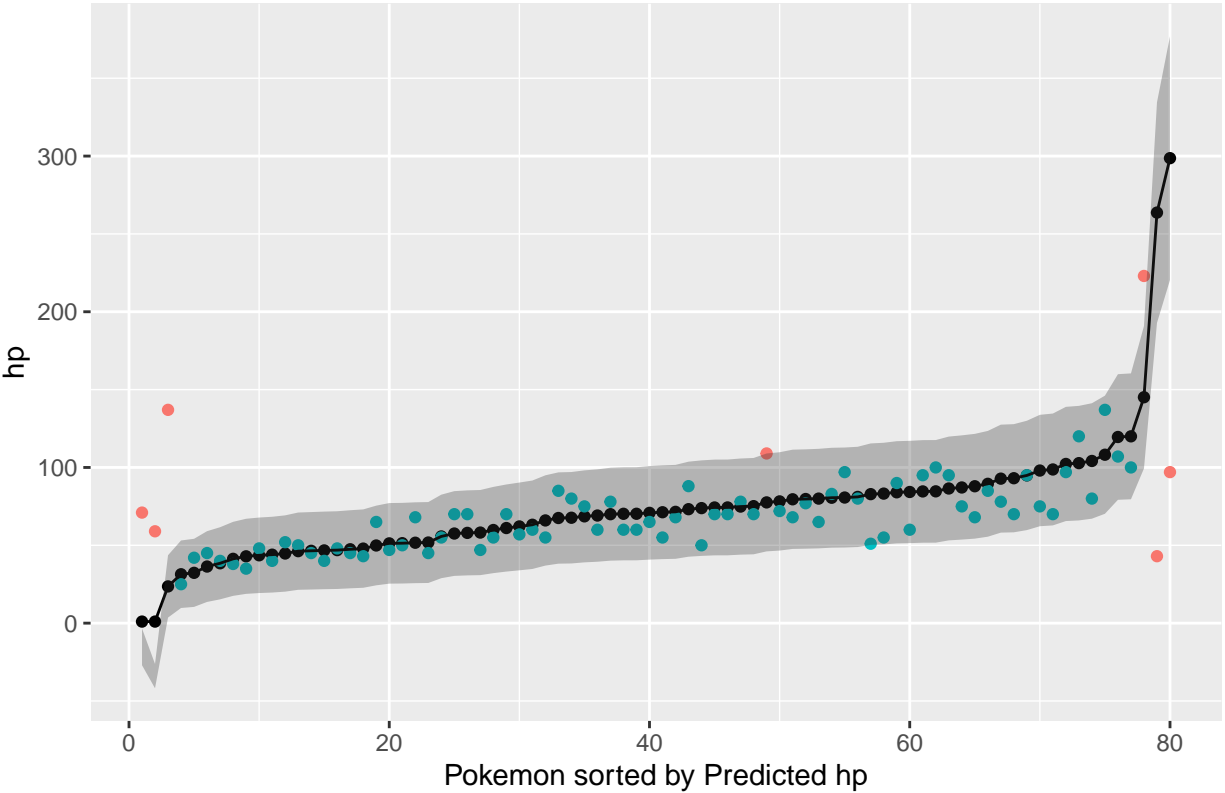Given the scope of our full analysis we list the graphs showcasing our predictions here in their own separate section.
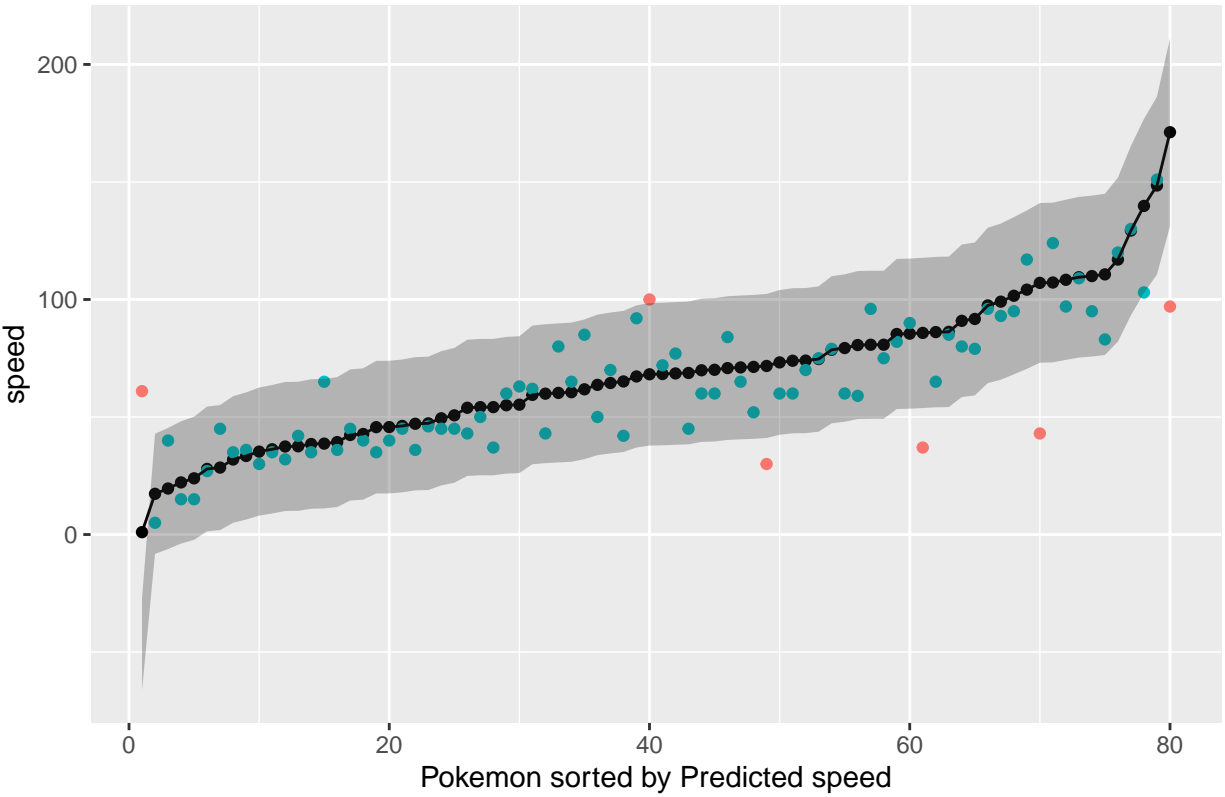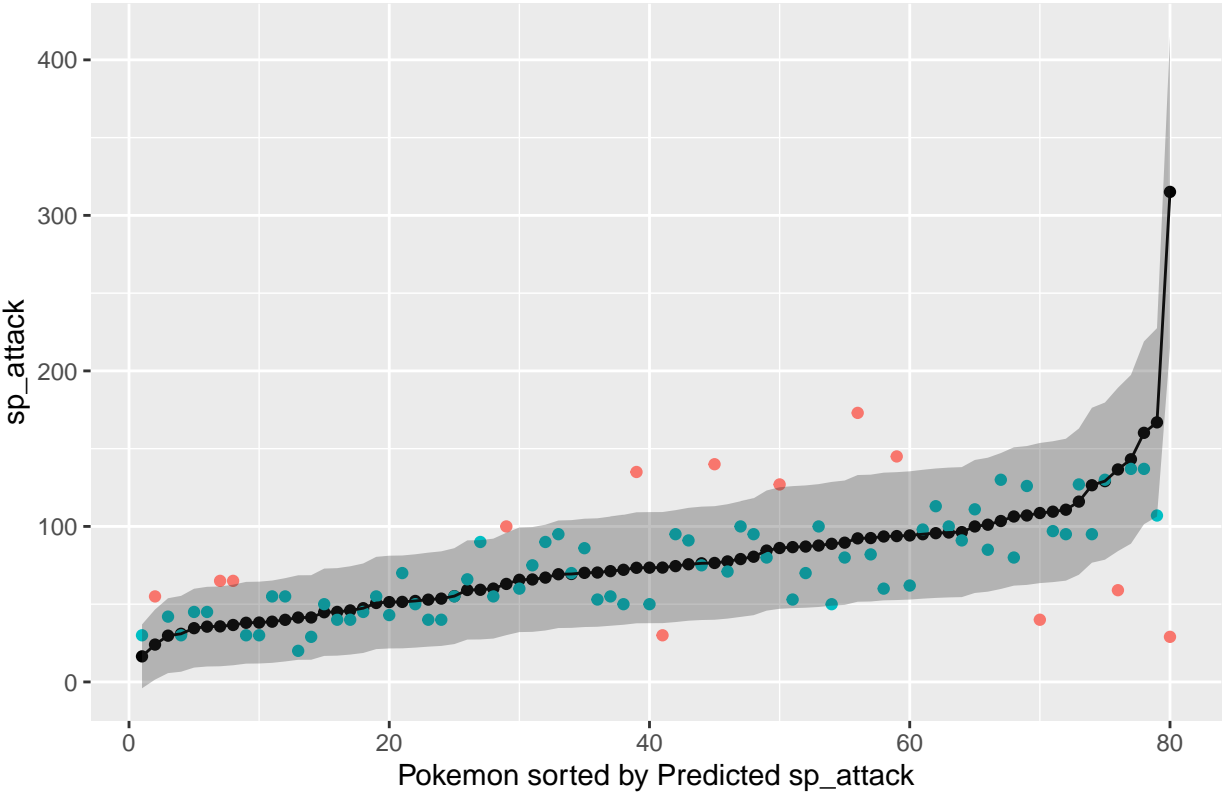
Predictions and True Values

Predictions and True Values

Predictions and True Values
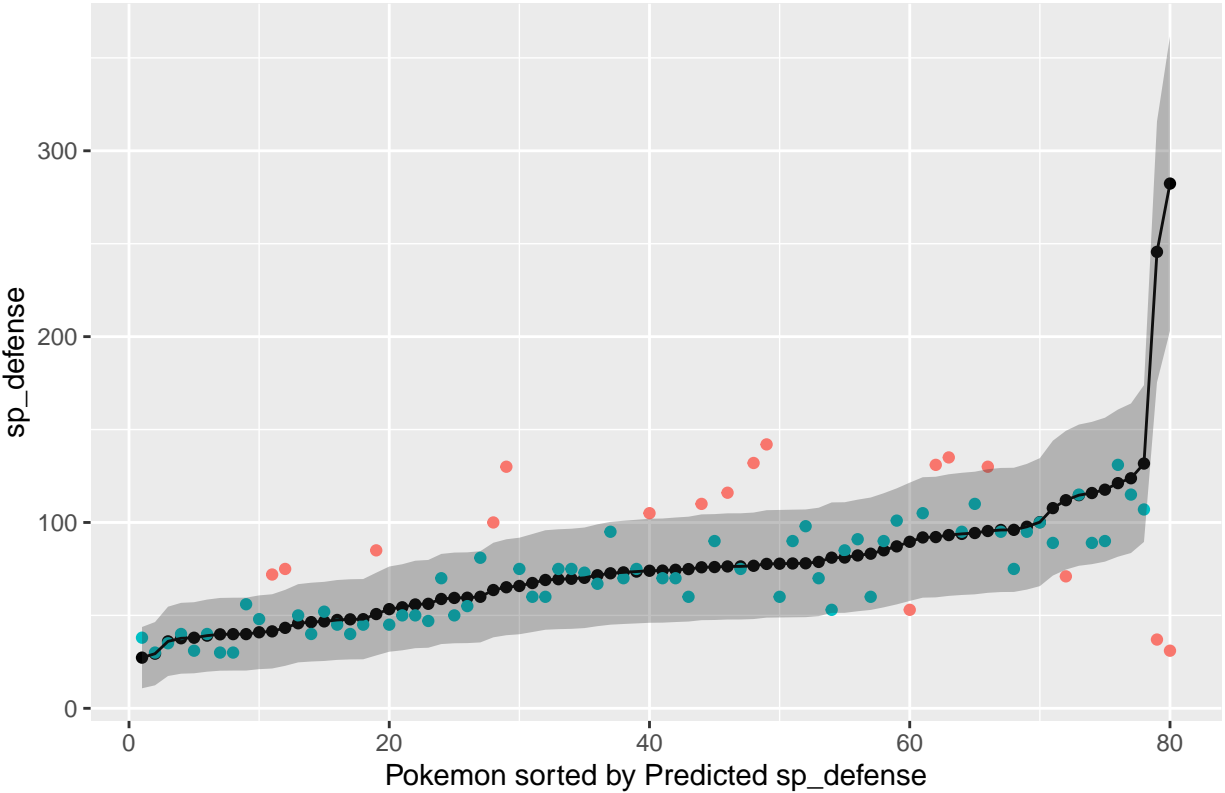
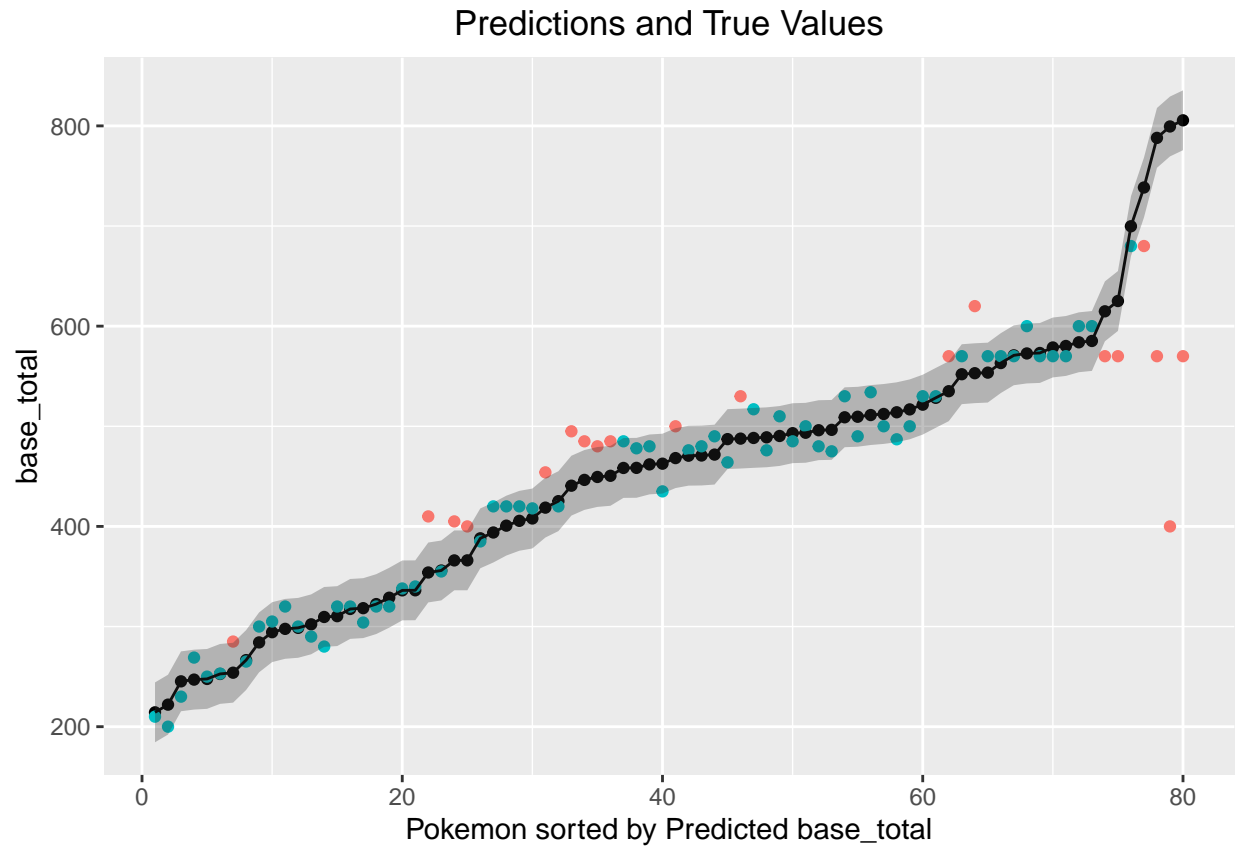Predictions and True Values

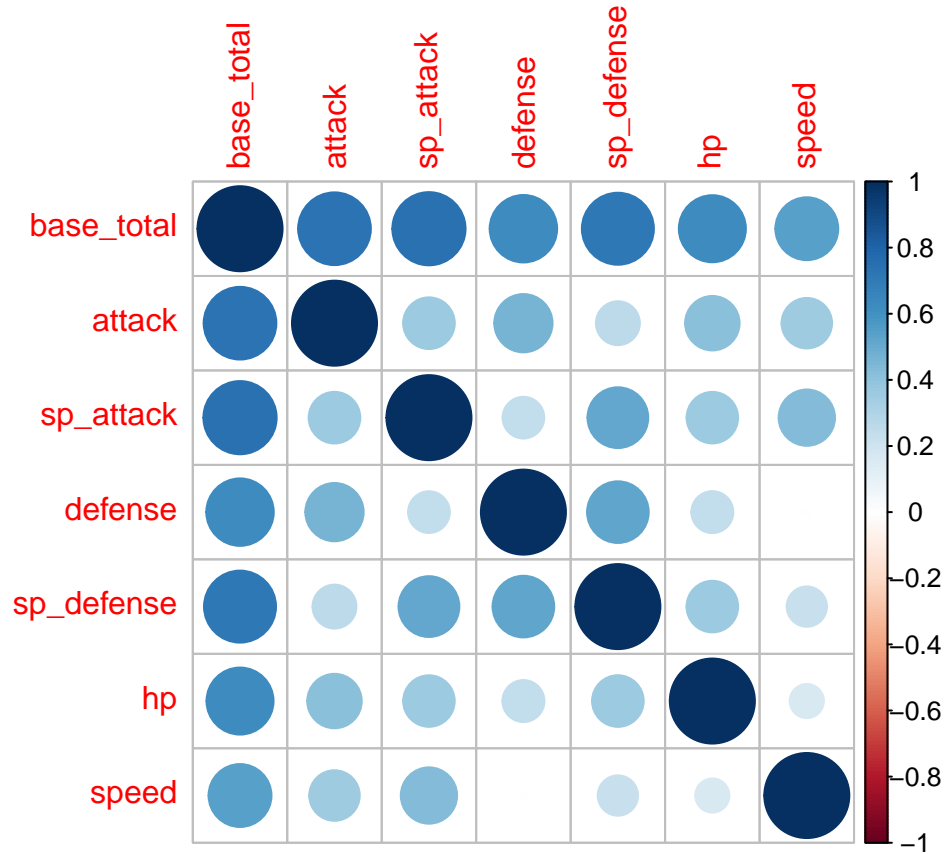Predictions and True Values

Predictions and True Values

Predictions and True Values

Additonally we include the corelation for all 800 Pokémon's base stats:

Refrences

[1] Hastie, Trevor,, Robert Tibshirani, and J. H Friedman. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd ed. New York: Springer, 2009. Print.

[2] Data via: https://www.kaggle.com/rounakbanik/pokemon web scraped from https://www.serebii.net/

[3] © 2019 Pokémon. © 1995–2019 Nintendo/Creatures Inc./GAME FREAK inc. Pokémon, Pokémon character names, Nintendo Switch, Nintendo 3DS, Nintendo DS, Wii, Wii U, and WiiWare are trademarks of Nintendo. The YouTube logo is a trademark of Google Inc. Other trademarks are the property of their respective owners.

[4] https://cran.r-project.org/web/packages/earth/index.html