



**Data Description**

We chose to do our project on Pokemon and to try and determine the “best” Pokemon type. We went to [Kaggle](https://www.kaggle.com/mariotormo/complete-pokemon-dataset-updated-090420) and found the complete set of Pokemon from generation 1 through 9. Generations are groupings of Pokemon games that separate Pokemon based on the new pokemon that are introduced into the game. When looking through the data, the first thing that we wanted to look at was the total number of Pokemon for each type. A type is a property of Pokemon that determines the type of moves a Pokemon will learn and what it is strong and weak against. As of the 8th generation, there are 18 different types. Each Pokemon has a primary type which is what we used to group pokemon together. Next, we looked at the Pokemon’s total points, which is a combination of all of its stats. Stats are numerical values tied to 6 different attributes useful in battle (HP, Attack, Special Attack, Defense, Special Defense, and Speed). The last point variable that we looked at was the catch rate. In the Pokemon games, catch rate is determined on a scale from 0-255 all under that same standard of trying to catch a pokemon with full HP with a Pokeball. Looking at the scale, we’re able to calculate the actual catch percentage through our data filtering.

Because the dataset was so extensive, we needed to do a bit of filtering before having the data up to date. In a few of the most recent generations, new variations of Pokemon were created as “Alolan” “Galarian” and “Mega” pokemon. Because we decided we only wanted to use the original forms of the pokemon, we filtered out any Pokemon with those in their name. Additionally, there are a lot of event Pokemon that are slightly different and don’t add anything to the actual Pokemon. Because of this, we then went through the whole dataset and filtered out any Pokemon that had alternate forms that didn’t have any extra types. If a pokemon had different forms with different types, we kept those in for the total count of each type. The other filtering we had to do besides taking out Pokemon we weren’t going to use was converting the catch rate into an actual percentage. We needed to divide every number that was listed in the catch rate by 765 in order to get an accurate percent scale. The dividing number is 765 because with a catch rate of 255, it was listed that the probability of catching it with a Pokeball at full health was 33.3%.

**Design Rationale**

The bar chart of Pokemon by type uses rectangles as marks and uses hue, aligned vertical length, and horizontal position as channels. Each rectangle represents a different Pokemon type. The vertical length of each rectangle corresponds linearly with the number of Pokeman of said type. Below the x axis, each rectangle is labeled by its type. Pokemon type is also displayed by coloring each rectangle according to its Pokemon type color. This is especially useful for people who are familiar with Pokemon.

The scatter plot displaying Pokemon Catch Rate vs Total Points uses circles as marks. These represent individual Pokemon. The plot uses aligned x and y position as channels. Originally, we colored each circle according to its type. This would allow users to visualize which Pokemon types are generally more or less catchable and which have more or less total points. However, the high density of circles on the graph made the colors blend together and it was overall very cluttered. It also did not add to the main idea of the visualization, which is that Pokemon catch rate is correlated with total points. Thus, we decided to color all of the circles steel blue and we added the regression line that we calculated to show the aforementioned relationship.

**The Story**

With our data we sought to prove that is the “best” Pokemon type across all the games. We first wanted to see how many Pokemon were in each type which corresponds to our first chart. Next, we were curious if there was a relationship between the catch rate of Pokemon and their overall total stat points. Our second graph was used to see if there was a negative correlation between the two to determine that lower catch rates corresponded to higher total stat points. The second chart has the linear regression line of best fit to show the negative correlation between the two. With this conclusion we wanted to see if there was a clear “best” type by having the highest average total stat points and the lowest average catch rate. When graphing our results, I was surprised to see that Dragon was by far the highest total points average but was tied with a catch rate against flying Pokemon. Additionally, it was surprising to see how the majority of the other types were clustered around each other with Steel, Fairy, Flying, and Dragon being the most noticeable outliers in the data. With the conclusions proven before, we wanted to try and visualize the “best” Pokemon type in all of the games by their total stat points and their catch rate averaged across all Pokemon with their primary type. By being tied for the lowest catch rate and having the highest total points, we’re able to conclude that dragon types are the “best” type in Pokemon. Despite only having a total of 30 Pokemon as primarily dragon types it shows that each dragon type is a very powerful Pokemon.

**Member Contributions**

Max: I first went through the whole dataset and looked at every Pokemon and determined which ones needed to be put into the filter and what key words could be used to filter out groups at a time. I then created the filtering system for the catch rate to convert that into a percentage. I created the “total” function that was used to make sure that our filtering was correct and that we had the correct number of pokemon in each generation. I created the dictionary for the types chart and then created both the types chart and the catch rate to total points chart that was used to prove our point about the “best” type by catch rate and total points. I created the variables and equations used to find the intercept and slope of our lines for the second chart. I worked on the data description aspect of the write up.

Leo: I worked on the Pokemon Types chart, the Catch Rate vs Total Points chart, and the Design Rationale. For the first chart, I wrote code to create a dictionary with the Pokemon types in ascending order based on frequency. I also added frequency counts at the top of each bar. Finally, I refined our data join. We were originally adding a bar for each pokemon, not each pokemon type. For the second chart, I added the linear regression line and made visual changes to the chart to make our overarching point more salient. For both charts, I added axis labels. For the Design Rationale, I described our thought processes behind making the two charts that I worked on.

Sam: After discussing the dataset filtering and preprocessing with Max, I developed our first preliminary visualization: a scatter plot of all the Pokemon’s attack and defense stats. This was a useful sanity check to ensure that we could load the data properly and by swapping in different variables, we could confirm that there were no outliers or duplicate types in our dataset. As we moved forward, we developed designs for our visualizations, and chose what plots we wanted to build to support our argument. We came to the conclusion that the attack/defense statistics would likely introduce too much complexity to include in a static display, and determined that a “total points” statistic would be more useful. I used the total points field that we constructed to adapt the initial scatter plot to show the relationship between catch rate and the total points of the pokemon. We colored this plot by type, so that any groupings of types would become visible, but we found that there was no apparent clustering, and that the colors were very difficult to distinguish in such a densely populated chart. To address this concern, I designed an additional scatter plot which showed only the averages of catch rate and total points for each type, and colored and labeled those points. Because this chart is less dense, it’s much easier to process for the audience. This visualization is where I spent most of my time, since I took charge of the design, formatting and implementation throughout its development.