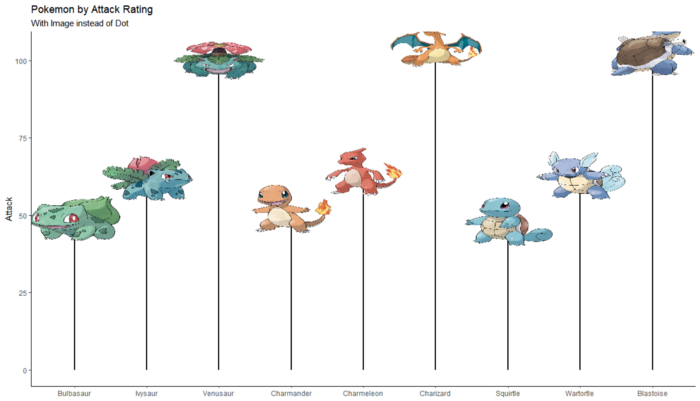
**How To R: Lollipop Charts**

Lollipop charts are an alternative to the bar graph. These can be shown for one or multiple variables.

The lollipop chart

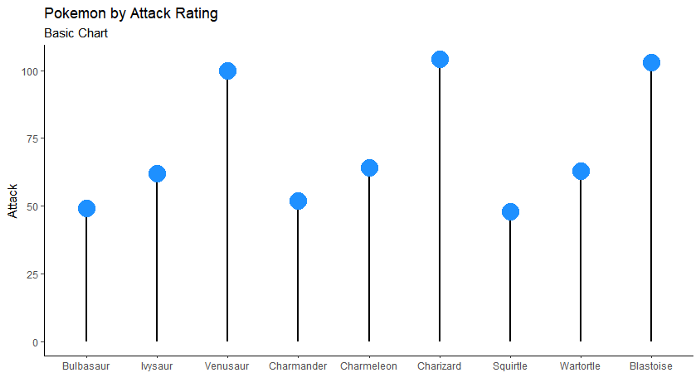


For this article we will be using the [Complete Pokemon Dataset](https://www.kaggle.com/datasets/rounakbanik/pokemon) from Kaggle for the data and the R package ggplot for the chart.

Variations of the lollipop chart shown in this article:

1. Basic
2. Colors by Group
3. Colors by Y-Axis Amount
4. With Labels inside of Dot
5. With Images instead of Dot
6. Comparing Two Variables

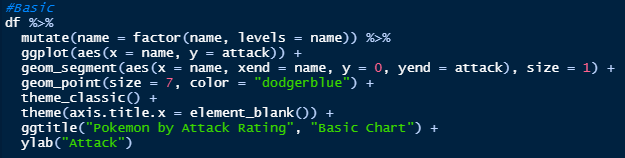
**1. Basic Lollipop Chart**



The basic lollipop chart is the simplest to make. It shows the basic information like a bar graph instead oh having bars it is replaced by a bar and dot.

We can clearly see from this graph that as each of the three starter pokemons (bulbasaur, charmander, and squirtle) evolve their attack increases a lot.

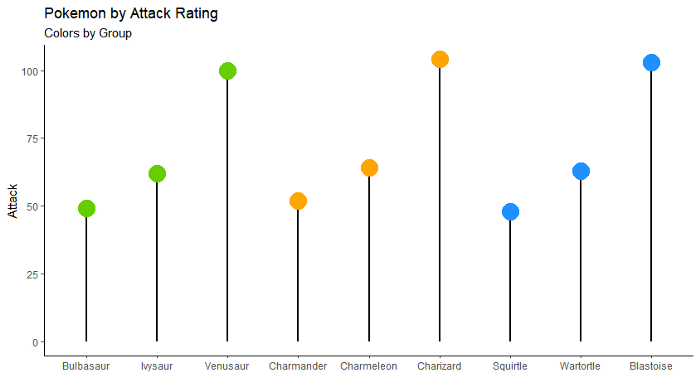
The code used for the graph above.



Things to note:

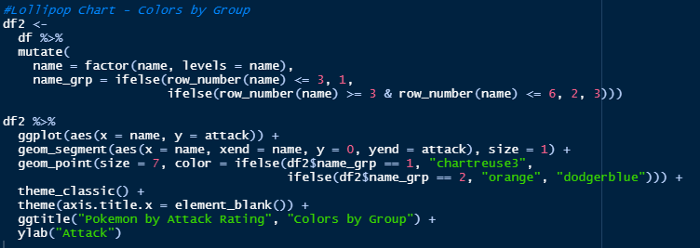
* I had to set the **name** field to a **factor** so that it would not reset the order of the graph to alphabetical (see line 2 of code).
* The **geom\_segment** function relates to the line connecting the dot and the 0 x-intercept. You can change the color and size of these lines as well, as you can see I updated the size and color can be updated in the exact same way.
* The **geom\_point** function is where the dot is defined.

**2. Lollipop Chart with Colors of Dots by Group**



Each starter pokemon’s evolution path was grouped and I added the color of that pokemon to each of the dots for their evolved forms. Bulbasaur is green, charmander is orange, and squirtle in blue. Now it is easier to see each pokemon’s specific path.

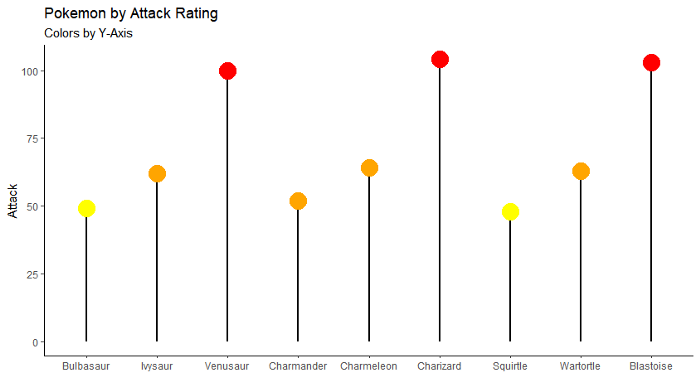
The code used for the graph above.



Things to note:

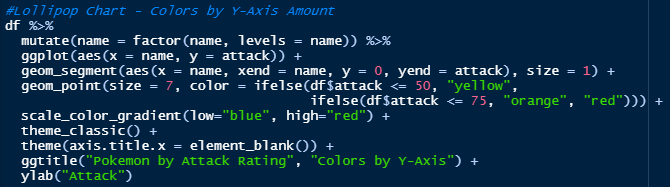
* To set up the groups for the colors I created a new dataframe, df2, and added a new column named **name\_grp**. This column defined groups based on the row id. If it was one of the first 3 rows, tied to bulbasaur, then it was group 1, then group 2 for charmander, and group 3 for squirtle.
* The **color difference** was defined within the **geom\_point** function. In the color equation there is an if statment based on the name\_grp that was created. Setting the color to a green if group 1, orange for group 2, and blue for group 3.

**3. Lollipop Chart with Colors by Y-Axis Amount**



I set the color of this graph based on sections of the y-axis. Setting it to yellow if the attack rating was under 50, orange between 50 and 75, then red above that. This gives us a clearer view of which dots are higher than the other even if they look close to one another, such as bulbasaur vs charmander. These look to be very close to each other, however since charmander is in the orange that means that its attack is above a 50 rating, while bulbasaur is less than 50 since it is yellow.

The code used for the graph above.

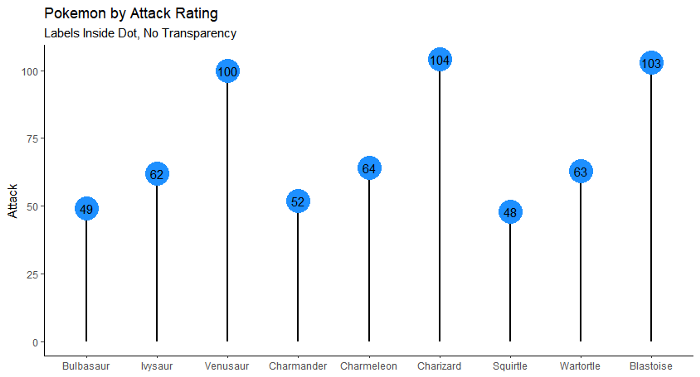


Things to note:

* The color of the dots were set in the geom\_point function, utilizing an if statement on df$attack this time.

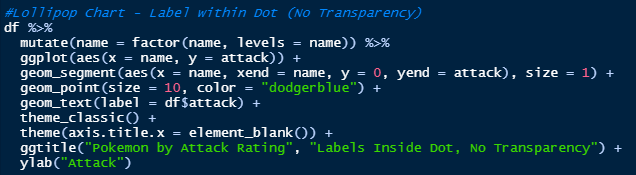
**4. Lollipop Chart with Labels inside of Dot**

**No Transparency**



In this version of the chart, there is no guessing about what the attack rating is as they are easy to see right there in each dot. Now I know that charmander has a 52 attack rating, while bulbasaur has a 49, and squirtle is last with a 48.

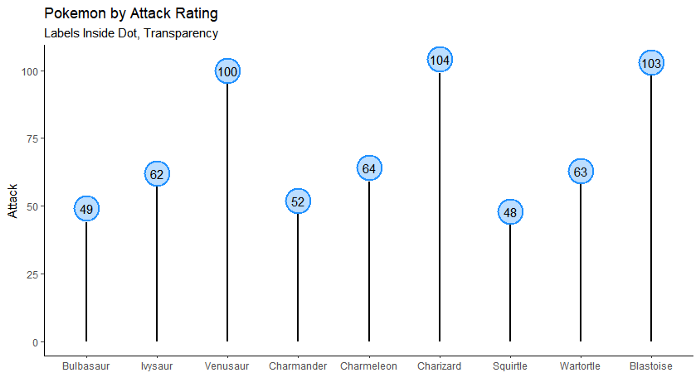
The code used for the graph above.



Things to note:

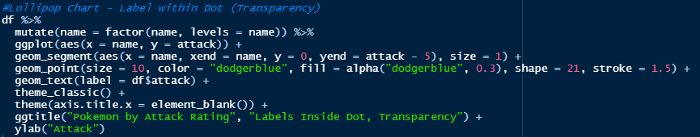
* To add labels into each dot a new line was added, the geom\_text function. Here I set the label to be equal to the attack rating so that is what would be showing.
* To make sure that the label fit inside the dot I increased the size of the point to 10.

**Transparency**



For this one, there is some transparency inside the dot, with a clear border around it. This will add readability to each of the labels within the dots.

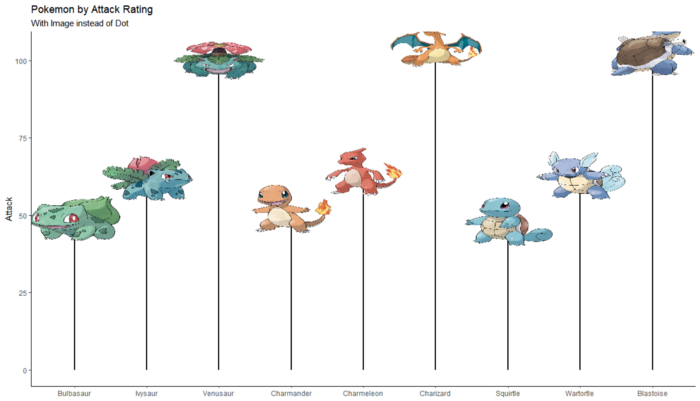
The code used for the graph above.



Things to note:

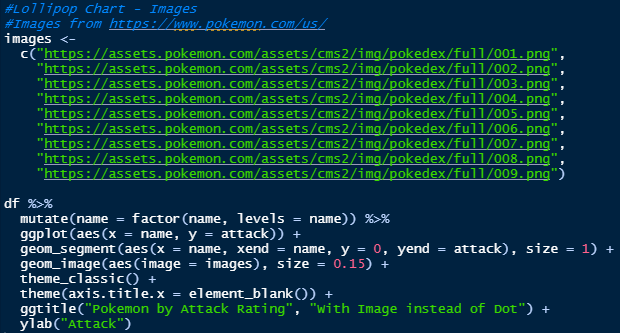
* To add transparency a few things need to be added to the geom\_point function. First you need to add a fill equation and set this inside of alpha, and this will cause the color to be more transparent. Then, the shape needs to be 21. And finally, the stroke is the size of the border.
* The segment goes to the actual attack rating and the dot utilizes this exact same point for its center. Because of this, the line goes all the way to the center of the dot and can be seen with transparency. To get around this you need to look at the size of the dot and divide it by two then remove this from the yend equation in geom\_segment. For example, here the dot has a size of 10, half of that is 5. So in geom\_segment the yend = attack — 5.

**5. Lollipop Chart with Image instead of Dot**



Here we can see the pokemon themselves at the point of their attack rating. This helps to easily visualize where each pokemon is.

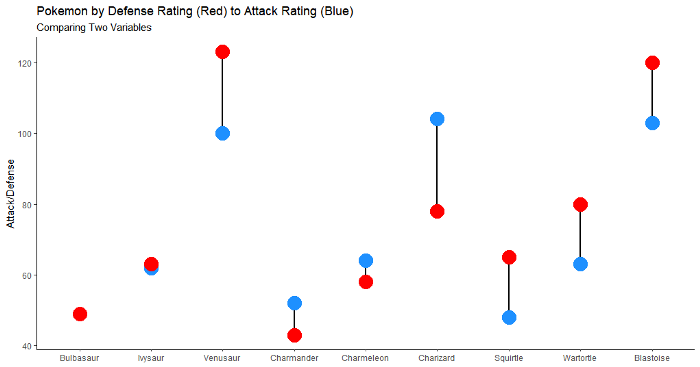
The code used for the graph above.



Things to note:

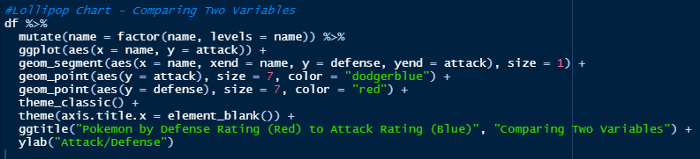
* I got the images for each pokemon from the [pokemon.com](https://www.pokemon.com/us/) site. I then created an images vector with the url for each of these images, in the same order as they appear in the dataset.
* To add the images to the lollipop chart you need to install the ggimages package in R. Then use the geom\_image function instead of the geom\_point function and voila!

**6. Lollipop Chart — Comparing Two Variables**



In this chart, we can clearly see where the attack and defense ratings are at for each pokemon, and how they compare to each other. For example, squirtle and each of its evolved forms have a defense rating much higher than its attack rating. Conversly, charmander has a greater attack than defense in each of its evolved forms. Depending on what you care more about, you may select squirle or charmander to start with.

The code used for the graph above.



Things to note:

* For this to work you need to make sure the segment, in the y and yend each reference one of the variables, instead of 0. For this one, I placed y = defense and yend = attack.
* There needs to be two geom\_point functions, each one referencing its own variable through aes. Here I have one with aes(y = attack), and the other with aes(y = defense).

**Recap**

I hope you all enjoyed this article and learned how to create some of the various versions of lollipop charts.