# The Data

## The Size

Our data set will contain 31000 tweets containing links to the Nike Plus website of which about ***83%*** are links to usable workouts.

## The Gathering

To get our data, we had to parse the HTML file for the javascript tags. Within the body of those tags were our target json variables. To get this data, we wrote our own parse to go through the javascript code and return the json data. Each page had 6 target variables we were looking to obtain and use as the data associated with that workout.

We found that we were able to get at least some data for 83% of the links we found. The other 17% were lost due to python not being able to handle the data for whatever reason or the link leading to an advertisement rather than an actual workout.

## The Cleaning

The cleaning was much more challenging than we expected. Different workouts had a wide range of data and not every workout had every data point. The json blobs we obtained were nested up to 8 keys deep and were not guaranteed to have any of the same keys as any of the other workouts. Even the values of the keys differed. For example, some workouts would have a temperature of “5” while others might have it listed as “5 C”. In addition, if a numerical value was missing, such as distance run, it could be a blank entry or a 0. There were no standard rules or formats we could rely on.

Thus, our cleaning was us parsing through the data we wanted, attempted to parse it and if it was missing, attempt to fill it in by making inferences from other data points. An example of this would be if duration was missing, we would check to see if the GPS was collecting data and if so, how often it was collecting data. If it was collection data, we would multiply the number of entries by the interval between each data point to infer a duration. In one case, there were 67 entries with an interval of 10 seconds in between, so we inferred this as 670 seconds.

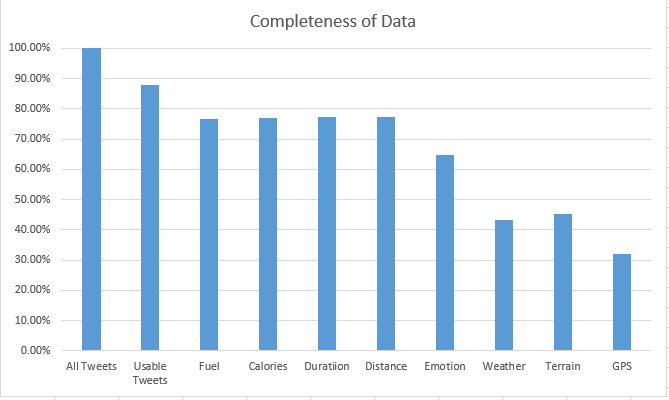
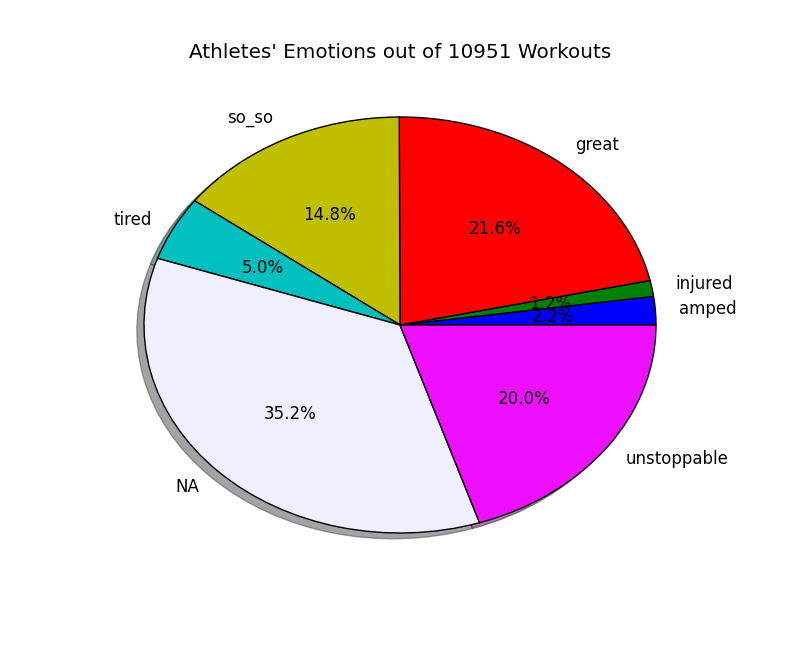
Cleaning the data will be an ongoing process as we narrow down what we want to do. We will have to clean each entry type in its own unique way. In general, we are making a user object and filling in the data members to flatten the data we got from the HTML, so instead of being nested 8 deep, it is just a simpler list of key data points we want to collect.

# Summary and Descriptive Statistics of Data

We found that our data was missing a lot more than we originally thought. For any given data point, such as just temperate or just mood or just speed, we only had, on average, a 40% chance of actually finding that data in a workout. For example, with emotions: 35.2% of the users had no recorded emotion. Of the recorded emotions, there were 6 unique emotions including tired, so\_so, great, injured, amped, and unstoppable. In the below figures you can find a pie chart for terrain as another example.

In addition, we found that a lot of our numerical data, such as speed, tended to fit a bell shaped curve. For example, minutes per mile was a bell curve centered at around 9 to 9.5 minutes per mile.

We found that our data was more complete than we thought it would be. We found that as long as we limit what points we cluster on, there is a strong enough signal to cluster on. Below is a graph depicting what percentage of tweets had what workout data and an example breakdown of one of the features (emotion in this case) We will have to be careful with what we cluster with though because our signal strength will have an upper bound of the weakest feature we choose. For example, if we choose GPS as a feature, we automatically limit ourselves to a little over 30% of the total data.

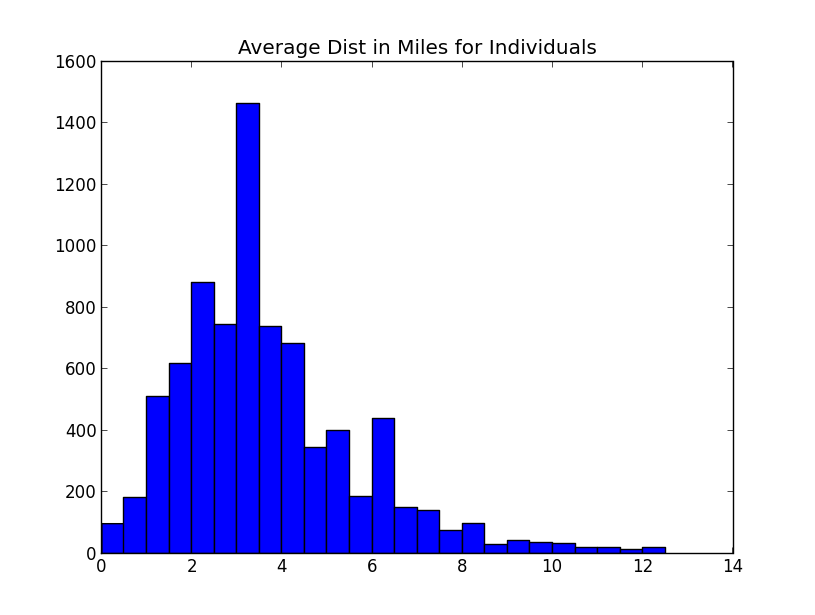
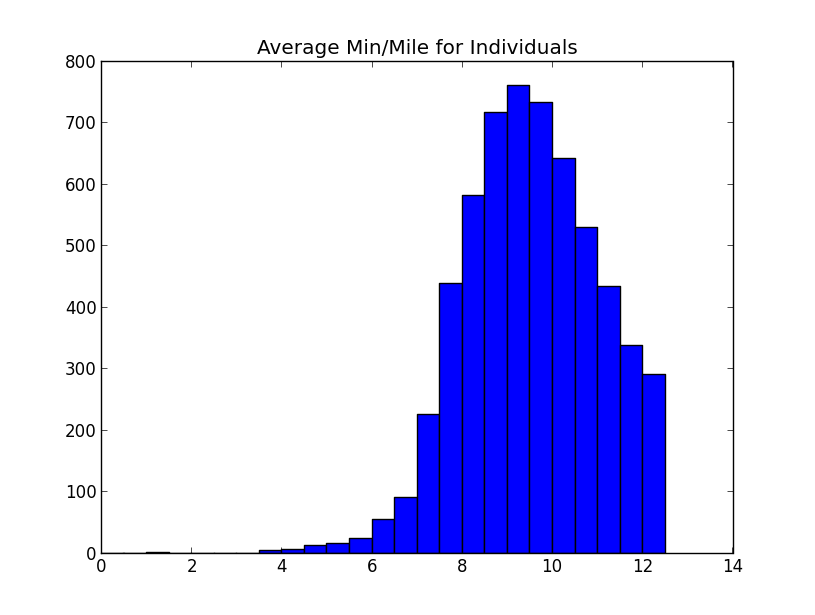
 

# Insights and Next Steps

One of our primary insights was that a lot of our data fit a bell curve and much of what we synthesized, such as speed from duration and distance, also fit a bell curve. Some of these curves allowed us to see interesting spikes. For example, in our distance graph, we saw a spike at the 5k running distance. Our other primary insight was that we do indeed have enough data to cluster on at least 7 features. Using 7 features will give us around 10-11 thousand points to cluster on. If we limit our features to ones with an even stronger signal, we can get above 20 thousand, though at that point we would only be clustering on 4-5 features instead.

Our next steps include the following:

1. Explore presenting the data from the perspective of the runner vs the business man
2. Dive deeper into the bell shaped curve and see how we might be able to use that
   1. If we go with the runner perspective, we can compare them to other runners and say they are better than x% of runners
   2. If we go with the businessman perspective, we can look at confidence intervals and hypothesis testing

Sdfsdf