

Coders Inc Design Decisions

Point

Point Representation

- Points are stored as HashMaps
 - Allows us to coordinate like values among the coordinates by using keys
 - We felt that there was a lower of chance of mishandling the data by using keys, rather than maintaining a specific ordering in a structure such as a list

DimensionalSpace

kNN Algorithm

- Once a close neighbour is found, it is transformed into a tuple containing its value and its distance
 - This makes it easier to store and sort the close neighbours and allows
 - Allows easier comparisons between points
 - Maintains the distance to avoid recalculation
 - Will allow an easier transition to weighted averages for multiple neighbours down the line

Standardization

- To standardize the data input, we converted the points to a normal distribution using the mean values of the coordinates and their standard deviations
 - We felt that this was good starting place as it was easy to implement
 - We thought that it was the best way to standardize the data as the possibility for extremely different scales was a concern for us

Running Summation and Averaging

- While the points are added to the space, the total sum of each coordinate is continually updated
 - We felt that this was the most efficient way to calculate the sum as opposed to going through the entire dataset afterwards
- Having the total sum for each coordinate also made it very easy to find the average of the corresponding coordinate

Inner Tuple Class

- The tuple class for the kNN algorithm was made a private inner class within DimensionalSpace as the kNN algorithm is the only time the tuple class is used