Frequency Distribution Project  
Robert Schwarz  
Link to GitHub containing the project:   
  
Thank you for the opportunity to apply for this position. Below I have detailed my thinking and approach to solve the provided problem.  
  
When initially approaching the project, I considered employing a graph coloring algorithm as a potential solution. However, upon further analysis, it became apparent that this approach would yield suboptimal results.

Subsequently, I explored two alternative methods to address the problem. The first involved utilizing a tree-based graph search to thoroughly navigate the solution space and obtain an optimal solution. The second approach entailed formulating the problem as an Integer Linear Programming (ILP) problem. Unfortunately, due to time constraints, I was unable to delve deeply into the latter approach and opted to pursue the former.

For the tree-based approach, I implemented a recursive algorithm to construct a tree structure, systematically traversing through the towers and assigning frequencies. Employing a Depth-First Search (DFS) strategy with backtracking proved efficient, given the vast solution space. At each node, I evaluated its value by computing the distance between each newly assigned tower's frequency and the value of the parent node. Additionally, I maintained a record of the best solution encountered thus far and pruned any branches where the potential value exceeded this best solution, enhancing efficiency.

Although this approach ensures an optimal solution, the time required to explore the entire solution space is extensive. Regrettably, my laptop lacks the necessary resources to provide a solution within the stipulated timeframe. This Computation speed may be improved by parallelization, however I had insufficient time to explore this option.

As a pragmatic alternative, I developed a greedy approach. By establishing a graph and connecting adjacent towers based on a predefined distance threshold, I executed a greedy algorithm on the given towers to achieve somewhat satisfactory results as displayed in the graphic below.

