**Problem 1 Disucssion:**

MrJob utilizes generators and parallelization to minimize computational time and memory usage. Thus, a discussion of the more general generators advantages is easier to discuss since MrJob disk access pattern is dependent upon them. Since generators only call upon the specific data it needs each time it is needed it greatly reduces the memory usage required for the operation compared to a list. Therefore, when each datapoint only needs to be access once it highly efficient, both from a computational time aspect and memory aspect. However, since a k-means implementation of MrJob would need to be called repeatedly each datapoint has to be generated as many times as the program is called. This process could hence take longer time then if the entire list were already stored in the memory. Therefore, the scenarios when using a MRJob implementation for k-means would be advisable would depend upon the memory capacity available, the size of the file and how many times MrJob must be called upon to get a complete k-means result for the data.

**Problem 2 Speed Up Plot:**