Problem Set 4

Problem 1. Consider agglomerative clustering on n points on a line. Explain how you could avoid n^3 distance computations. How many distance computations does your scheme use?

Problem 2. Consider a set of documents that we wish to group into k clusters. Among all possible k-clusterings of these documents, does agglomerative clustering produce the highest Value clusters? (Refer to slide 29 of lecture 8 for the definition of "Value" of a clustering)

Problem 3. Suppose you have n points in d dimensions with each point labeled either red or green. How large does n need to be (as a function of d) in order to create an example with the red and green points not linearly separable? (Assume that the set of n points are *general*, i.e., every subset of d points is linearly independent)

Problem 4. This problem explores assigning labels to clusters. Assume we have generated clusters hierarchically, and we are currently assigning labels to each of 3 subclusters within a computer-related cluster. Given below are some components of the centroid vector of each subcluster (columns represent centroid vectors, rows represent the term axes). Generate a labelling for each centroid by first reweighting each of its components by multiplying the term weight by the Inverse Cluster Frequency of the term. In other words, for each term dimension, multiply the term weight by $log\frac{N}{n}$, where N is the number of subclusters in the current parent cluster, and n is the number of subclusters in which the term has been assigned a nonzero term weight. Then choose the two terms with the highest weight in each centroid to be the label for the cluster corresponding to that centroid. What are the labels? What would the labelling have been if we hadn't reweigted with ICF?

	$CentroidA$	$CentroidB$	$CentroidC$
$computer$	4	3	3
$disk$	3	0	0
$spreadsheet$	0	1	2
$java$	0	3	1
API	0	4	0
CPU	2	2	1
$drives$	2	0	0