Homework 1 - Prototype selection for nearest neighbor

Evan Gravelle October 13, 2016

1. Prototype Selection Methodology

My method for prototype selection from MNIST data utilizes k-means clustering to group the training data into k clusters. Each cluster's mean is then assigned a label, by performing a nearest neighbor search on the original training data. The set of means with associated labels serves as the set of prototypes.

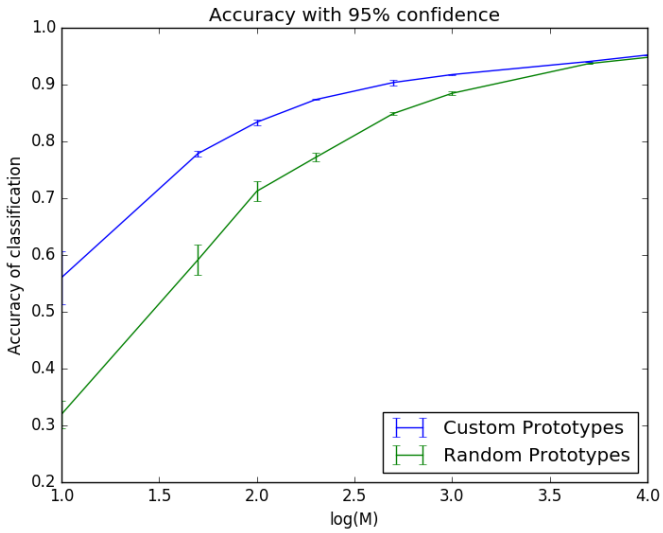
2. Pseudocode

Algorithm 1: Prototype Selection Algorithm

Input : Training set T, number of prototypes m
Output: Prototype set P

Initialize P randomly from T, where $|P| = m$
Assign each $t \in T$ to nearest cluster $p \in P$
Define Q as average distance from point $t \in T$ to assigned cluster $p \in P$
while Q *has no sufficiently converged* **do**
    for $t \in T$ **do**
        Calculate centroid C_p of each cluster
        Assign $P = \{C_p\}$ for all $p \in P$
        Assign t to nearest cluster $p \in P$
        Update Q
    end
end
for $p \in P$ **do**
    Assign label of nearest neighbor of p to P
end
Return P

3. Experimental



Three data points were recorded for each value of m under 5000. Only one data point was recorded when $m \in \{5000, 10000\}$, this is due to the long running time of the algorithm.

4. Critical Evaluation

As seen in Figure 3, this method of prototype selection outperforms random prototype selection for all tested values of m between 10 and 10,000. Still, there are potentially multiple avenues for improvement in prototype selection. k-means clustering does not use the label during cluster formation, so a constrained version could help, where each point in a cluster the same label. The nearest neighbor of the cluster centroid is used to determine the label for the cluster, when perhaps a higher order nearest neighbor assignment would work better, or some sort of function approximation to assign the label.