CSE 250B: HW1

WEIWEI LI
PID: A53107958

1. The prototype selection is the process that identifies the most significant subset of the training set. One way to speed up nearest neighbor classification is to replace the training set by choosing a good subset of "prototypes".The basic ideas for prototype selection are as follows:

K-means approach 1: First we divide the train data into 10 sets by its labels. Then for each set, we group the points into $M/10$ clusters by K-means algorithm. Therefore, we will get $(M/10) * 10$ cluster centers, which forms the subset of prototypes with size M.

K-means approach 2: The idea is similar to K-means approach 1. We divide the train data into 10 sets by its labels. And for each set, we group the points into 5 clusters by K-means algorithm. Then within each cluster, we randomly pick $M/50$ points. Therefore, we will get $(M/50) * 5 * 10$ points, which forms the subset of prototypes with size M.

2. The pseudocode of the proposed algorithm is as follows:

Input: train data set X, subset size M
Output: prototype set P
K-means approach 1
P=[ ]
for i in range $\{10\}$:
  X = train data with label i
  P += M/10 cluster centers of X using K-means algorithm

K-means approach 2
P=[ ]
for i in range $\{10\}$:
  X = train data with label i
  clusters = 5 clusters of X using K-means algorithm
  P += $M/50$ points randomly picked from each cluster of X

Date: October 24, 2016.

3. The experiment results are as shown in Table 1.
Confidence Interval = $(\mu - z * \frac{\sigma}{\sqrt{n}}, \mu - z * \frac{\sigma}{\sqrt{n}})$

Test error = $\frac{\#\,of\,classified\,wrong}{\#\,of\,classified\,correctly + \#of\,classified\,wrong}$

I used the formula above to calculate the 95% confidence interval and the test error. The random selection of the prototypes are done uniformly random, and is repeated 50 times for each M value.

From Table 1, we can see that K-means approach 1 (when $K = M/10$) is doing much better than K-means approach 2 (when $K = 5$). Since we did some randomization steps in K-means approach 2, it would be better if we could do several experiments of K-means approach 2 and get error bars. But I think the result won't change much, because even K-means approach 2 at $M = 10000$ is doing worse than K-means approach 1 at $M = 500$. However, the advantage of K-means approach 2 is the calculation speed. K-means approach 2 is much faster than K-means approach 1, since K is much smaller.

Comparing with random selection, K-means approach 1 improves performance profoundly. However, the improvement of K-means approach 2 is small. When M is small, K-means approach 2 is doing better than random selection. When M becomes lager, it performs not so good and the error rate is close to or higher than that of random selection. Because K is very small in K-means approach 2. We only grouped each train data category into 5 clusters. When M is very large, this method is close to random selection.

M	500	1000	5000	10000
K-means (K=M/10)	0.0492	0.0407	0.0314	0.0298
K-means (K=5)	0.1454	0.1179	0.0633	0.0525
Random	0.1507 ± 0.0019	0.1142 ± 0.0009	0.0646 ± 0.0005	0.05142 ± 0.0005

TABLE 1. Error rate of the 1-NN classification using M prototypes with different prototype selection scheme.

4. Clearly, K-means approach 1 is an strong improvement over random selection. I will try several K's for K-means approach, greater than 5 and smaller than $M/10$ to see if it would be more accurate than $K = 5$ and faster than $K = M/10$. Also, I will do several experiments for different K's and get error bars, since this algorithm require some randomization steps.