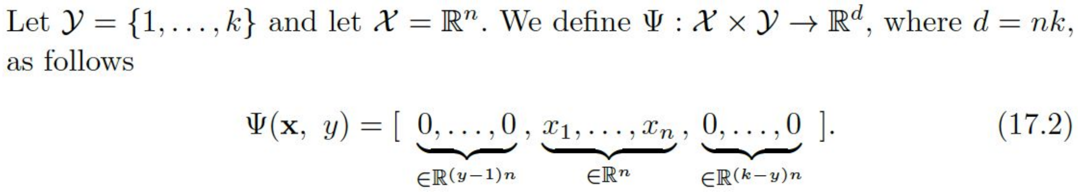
ECS 271 - Assignment 2

Hand Written Digit Prediction

Haitong Li

Question 1)

-Mapping function:



Let $\mathcal{Y} = \{1, \ldots, k\}$ and let $\mathcal{X} = \mathbb{R}^n$. We define $\Psi : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^d$, where $d = nk$, as follows

$$\Psi(\mathbf{x},\ y) = [\ \underbrace{0, \ldots, 0}_{\in\mathbb{R}^{(y-1)n}},\ \underbrace{x_1, \ldots, x_n}_{\in\mathbb{R}^n},\ \underbrace{0, \ldots, 0}_{\in\mathbb{R}^{(k-y)n}}\ ]. \tag{17.2}$$

In this problem, y is the list of ‘Digit’ labels in the train dataset and X is the set of instances in the train set of size 3747*8. In my approach, I calculated phi(x,y) directly with the entire X and y matrices, getting the result of a matrix of phi for each instance and its corresponding label. Then for the right side of the constraint, which is the phi(x,y) calculated with y = {0...9} that are being compared to the phi with correct labels, I calculated phi with the 10 labels and the whole X matrix, and there are total of 10 such phi matrices.

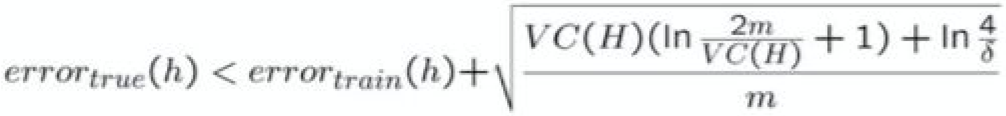
-Loss function:

$\forall y' \in \mathcal{Y} \setminus \{y\}, \quad \langle \mathbf{w}, \Psi(\mathbf{x}, \mathbf{y})\rangle \geq \langle \mathbf{w}, \Psi(\mathbf{x}, \mathbf{y'})\rangle$ + 1 - epsilon (for soft margin)

Question 2)

To estimate the performance, I used the cross validation method which separates the train dataset to a division of 8:2 in which 80% of the set is the training set and the rest 20% is the test set for cross validation.

I first train the SVM with the training set and use the resulting w to predict the labels of the test set. I get an average error rate of 0.3 after several runs.



$$error_{true}(h) < error_{train}(h) + \sqrt{\frac{VC(H)(\ln \frac{2m}{VC(H)} + 1) + \ln \frac{4}{\delta}}{m}}$$

With this equation, I calculated an estimate of the true error which states that error_true < 0.47. It is a reasonable upper bound for the true generalization error since it is greater than the training error within a reasonable range.

Question 4)

Error estimates:

No transfer: 0.10

Hypothesis transfer: 0.10

Instance transfer: 0.11

The transfer learning methods generally perform close to that without transferring. The performances of hypothesis transfer and no transfer are almost the same on this dataset. The fact that the instance transfer method does not increase the accuracy may be because the instances of 1vs9 does not share a high degree of similarity with the 1vs7 instances so the 1vs9 classifier’s support vectors do not help with predicting the 1vs7 problem.

Question 5)

I used the rbf kernel to kernelize the approach. This approach results in an accuracy score of 0.924 which means an error rate of 0.076.