MATH 689 / CSI 873

Final Exam. December 11, 2019.

Hand written digits recognition problem using the support vector machine.

Implement your Support Vector Machine for the hand written digits recognition in MATLAB or Python using available subroutines for solving a quadratic optimization problem. You can use any other programming language as long as it allows using a subroutine for solving a quadratic constrained optimization problems.

1) Use the provided discretized hand written digits data sets (both training and testing, scaled between 0 and 1, fractional values are fine).
2) Formulate the dual soft margin SVM in MATLAB by specifying all the required matrices and vectors.
3) Train the dual soft-margin SVM (the one that incorporates a non-separable case) to classify 2s vs. 5s only. Select 500 training data points (250 for 2s, 250 for 5s). Use the dual radial basis function machine $\gamma = 0.05$. Use C = 100 as the penalty parameter. Increase if necessary.
4) Calculate the error/accuracy for testing examples.
5) Reduce the original number of pixels (784) uniformly by 50%, 75%, 90% and 95%. Calculate the testing accuracy for all the cases. Describe the observations.
6) Apply the SVD decomposition to the training data to prepare a lower quality data. Reduce the original 784 dimension by 50%, 75%, 90% and 95%. Calculate the testing accuracy for all the cases. Describe the observations.
7) Reduce the original number of number of training examples (500) by 50%, 75%, 90% and 95%. Calculate the testing accuracy for all the cases. Describe the observations.
8) Train the dual radial basis function machine $\gamma = 0.05$ to classify even vs. odd numbers. Select 1000 training data points (100 for each digit), use all 784 pixels. Calculate the error for testing examples.
9) Run 10 SVMs to train to detect a particular digit (e.g. 2) against the rest digits (e.g. 0, 1, 3, 4, 5, 6, 7, 8, 9). In the training use the value y = +1 for a particular digit and y = -1 for the rest of them. Obtain 10 different separating hyperplanes $h_0,\ldots,h_9$ that separate each 0,1,…,9 from the rest digits.
10) While testing the digits you may find out that a particular digit may be classified not uniquely. For example, some tested digit can be on the positive side of h_3, h_5 and h_8, meaning that this digit can be classified as 3, 5 or 8. Alternatively, you may find out that the tested digit is on negative side for all hyperplanes. To resolve the problem classify this digit as the one that corresponds to the hyperplane with the maximum classification number
$$\sum_{i=1}^{l} y_i \alpha_i^* K(x_i, x) - b$$
11) Calculate the error rate. Compare the results with those obtained for the artificial neural network, Bayes naïve classifier, and k-nearest neighbor algorithms.
12) Document your experiments, prepare the report, submit it, and have a great holiday season!