Car evaluation based on support vector machine

We use SVM to solve the car evaluation problem.

1.method

We read john C.Platt’s paper “Sequential Minimal Optimization :“A Fast Algorithm for Training Support Vector Machines”(**you can do refer this paper in our report in the theory part**). And we implemented the SMO algorithm to do the optimization problem.

In this problem, the features in the dataset is string-type, so we first transfer the samples into numbers. For example, the feature ‘buying’ has four value ‘vhigh’, ‘high’, ‘med’, ‘low’, so we transfer each value to 0.4,0.3,0.2,0.1. We have 1728 samples in the dataset, so we take 1200 samples as training set and the rest as testing set.

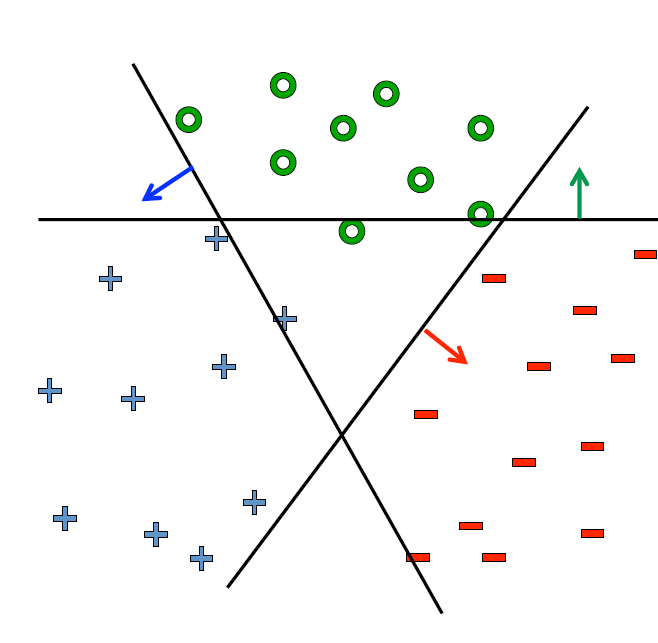
In this problem, we have to do a multi-class classification, therefore we have to approach to do it. And we implemented two methods among them.

We implemented this algorithm in python according to the pseudo code in john C.Platt’s paper.

2.one-versus-all classification

First we do the one-versus-rest classification. Here we have 4 different labels(unacc,acc,good,vgood), so every time we take one of the class as the positive class (we label them 1) and the rest as the negative class (we label them -1) and we train a SVM. Finally, we get 4 SVM. Given a test sample, we have to predict which class it belongs to. Using 4 set of weights, we predict the sample as:

y=arg max .x+. K=1,2,3,4



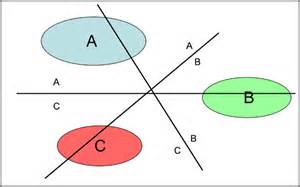
We use Gaussian kernel to do the training, but the result is not so satisfying . The accuracy is 61.1 %. Since in this problem, the data is not linearly-separable. And we use libsvm to do the classification, the accuracy is only 61%.

3.one-versus-one classification

Second we do the one-versus-one classification. Every time we take 2 out of 4 class to train a SVM, then we would have 6 SVM, when we do the prediction. We use every SVM to classify the test sample into the positive class or the negative class.

Finally, we have a vote for each class. Then we can predict the test sample as the class correspond to the maximal vote.

Since the linear kernel is not effective, we use Gaussian kernel instead. And the result is improved.



When training each SVM, the classification on the training sample can be different:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Class | Unacc-acc | Unacc-good | Unacc-vgood | Acc-good | Acc-vgood | Good-vgood |
| accuracy | 53.2% | 99% | 98.4% | 99.1% | 95.4% | 92.7% |

However, when we do predicting on the testing set. The accuracy is 85.2%. Therefore, the one-versus-one approach is more effective.

4.problem

The most important problem in this approach is that the data here is not linearly- separable. And most of them are overlapping each other. And we can find a better kernel to solve the problem.