Fast Feature Selection Algorithm for Data Classification

Daniel J. Garcia, Dr. Lawrence Hall, Dr. Dmitry Goldgof, Kurt Kramer
Department of Computer Science and Engineering
University of South Florida, College of Engineering

REU 2006
Poster No.
CSE.3

Introduction

Feature selection methods are used to find the set of features that yield the best classification accuracy for a given data set. This results in better training and classification time for a support vector machine, in addition to better classification accuracy. Feature selection, however, is a time consuming process unfit for real time applications.

Methodology

Our fast algorithm is composed of two stages:

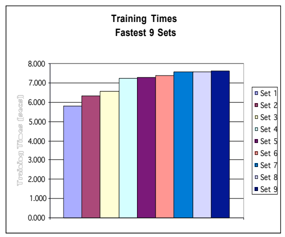
• First Stage: we generate a number of feature sets of fixed size and then train the learning algorithm using these sets. The sets of features which yield the lowest training time will then be used for the second stage of our algorithm. It is important to emphasize that the features in these sets are randomly selected out of the pool of all features and thus these sets are generated in a very short amount of time. In addition, they are completely independent from each other.

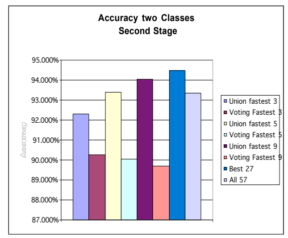
• Second Stage: using the fastest sets from the first stage we either apply a voting process to find how well they collectively describe the data or we create a new set by taking the union of the features found in the original sets into a new set. We then train the learning algorithm with this new set to determine how well this new set describes the data. The number of feature sets selected for the second stage of our method can vary from 2 to the number of sets generated during the first stage of the process.

Results

We present results obtained from experiments applying our fast algorithms to problems with different number of classes. Our results are compared with results acquired using an "optimal" set of features obtained from a different feature selection algorithm.

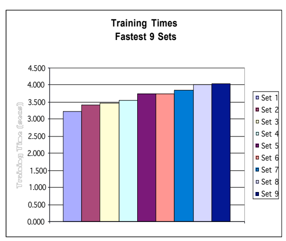
Two Classes Experiment

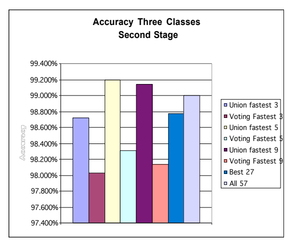




Average training time for the 200 random sets is 11.62 seconds. Difference in accuracy between the "optimal" set and the union of the fastest 5 sets is 1.125%, and the difference between the "optimal" set and the union of the fastest 9 sets is 0.458%

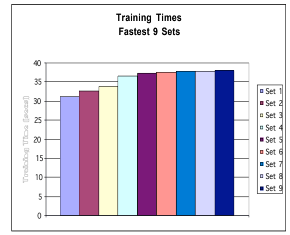
Three Classes Experiment

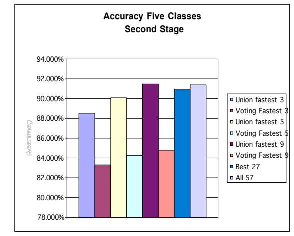




Average training time for the 200 random sets is 7.60 seconds. The union of the fastest 5 sets yields a better accuracy by 0.416% over the "optimal" set, and the union of the fastest 9 sets yields a better accuracy by 0.361% over the "optimal" feature set.

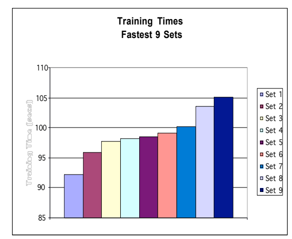
Five Classes Experiment

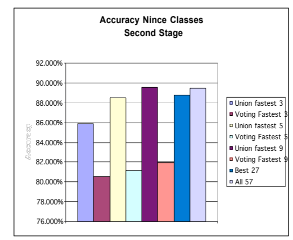




Average training time for the 200 random sets is 56.47 seconds. The difference in accuracy between the "optimal" set and the union of the fastest 5 sets is 0.933%, and the accuracy of the fastest 9 sets when compared with the "optimal" set is actually superior by 0.467%.

Nine Classes Experiment





Average training time for the 200 random sets is 163.53 seconds. The difference in accuracy between the "optimal" set and the union of the fastest 5 sets is 0.204%, and the accuracy of the fastest 9 sets when compared with the "optimal" set is better by 0.852%.

Conclusion

As has been shown, using random feature sets as a feature selection tool provides benefits for learning algorithms. Real time application is one of the greatest benefits, perhaps allowing a limited feature selection algoritm to be run as new data is gathered. The random sets approach is fast, is very accurate in certain situations, and takes great advantage of parallel processing