Relevance Vector Machine

Artem Los, Clment Morin, Kevin Dallatorre, Rmi Lacombe

I. Introduction

For supervised learning, support vector machine (SVM) is a state-of-the art way of performing classification and regression. The idea behind SVM is to *minimize the training set error* of a target function (define) while simultaneously maximizing the margin between the two classes [1].

However, SVM's have four problems that can be solved using the Relevance Vector Machines (RVM). First, SVM's do not give us a probabilistic answer, which is useful when we want to apply the notion of uncertainty in the classifier (i.e. similar to error bars). An SVM classification will produce a hard binary decision whereas the SVM regression will produce a point estimate. Secondly, the required number of kernel functions will grow steeply with the size of the training set. Thirdly, a constant C (aka error/margin trade off) has to be estimated. Finally, the kernel functions require Mercer's condition to be satisfied. [1]

The RVM solves all the problems mentioned above. The idea is to associate a prior to each weight (governed by a hyper-parameter for each weight), which are estimated from the data. As a result, fewer kernel functions are required than for SVMs, the error is in general smaller for RVMs than SVMs and the number of relevance vectors needed (similar to support vectors in SVMs) is smaller. On the contrary, it takes longer time to train an RVM than an SVM. [1]

II. Method

III. Result

IV. Discussion

V. References

1 Original article (fix ref).