**Discussion of Linear Scoring (SVM) Methods**

Here we give a brief introduction to Linear Scoring (SVM) Methods. In the simplest case, with *m* original features and regularization for non-separable groups (see the discussion below), the SVM classification and the parametric (normal distribution-based) linear discriminant approach will be similar. However, the support vector machine approach can be made very general by mapping the original features into higher dimensional feature spaces. Polynomial and Gaussian kernel functions lead to general nonlinear, but also computationally very efficient, classifiers.

The basic idea behind SVM is to find a linear classifier using a *m*-dimensional feature vector *x* in a binary classification problem with two group labels *y* (-1, 1). The linear classifier  selects group 1 if  and group -1 if . Information from the training data consisting of *d* documents  is used to determine the weight vector *w* and the intercept *b* of the separating hyperplane. In the separable situation (where there is no overlap between the data from the two groups) the idea behind SVM is quite simple and intuitive. It tries to find the separating hyperplane such that the perpendicular (and equal) distance between the hyperplane and the two closest points (one from each of the two groups) is maximized. In other words, the coefficients in the linear classifier maximize the margin width of the separating hyperplane, leading to the furthest separation of the two groups. The term “support vector” comes from the fact that the classification pivots on (that is, is supported by) just the two points (vectors) from opposite sides of the hyperplane. The coefficients for this optimal margin classifier can be found by solving a quadratic optimization problem with linear inequality constraints. It is even simpler to solve this optimization problem by considering the Lagrange dual optimization version. It turns out that the coefficients  in the solution are obtained by finding the maximum

 , (1)

subject to the linear constraints  for  and . The term  represents the vector inner product of a pair of feature vectors. Having found the optimal solution, the optimal value for the intercept is given by.

There is a reason why the Lagrange dual formulation of the optimization is so convenient for the calculations. It is unlikely that the feature vectors are separable as one can expect some overlap of the two groups. One has certainly a better chance of separation if one considers “extended” feature vectors with components that are nonlinear functions of the original features, such as polynomials or cross-products. Such extended feature vectors can be defined implicitly through special kernel functions such as the polynomial kernel, . For such special kernels it can be shown that. For example, with *d* = 3 features and polynomial degree *g* = 2, . The extended feature vector is now of dimension 6 (we count the same cross-products only once), and of higher dimension than the starting feature vector which was 3. This simple example illustrates that such kernel functions allow for greater flexibility of the classification. On the other hand, this added flexibility is achieved with little extra computational effort as the inner product of the extended feature vectors in equation (1), , can be replaced with ; and that is certainly easy to calculate. Other, and even more general kernels can be used such as the generalized polynomial kernel, , and the Gaussian kernel, .

Even after extending the feature vector, it may not be possible to achieve complete separation of the two groups with no overlap between the two data clouds in now certainly high dimensions. However, the algorithm can be made to work for linearly non-separable data sets and also in a way that reduces the method’s sensitivity to outliers. This is achieved by introducing a penalty function into the optimization. It penalizes a case that lies on the “wrong” side of the hyperplane and it does so in proportion to its perpendicular distance to the hyperplane. It turns out that the only change to the optimization in equation (1) is a change in the constraint, which becomes instead of ; *C* is the additional cost parameter.

Several excellent packages for a SVM analysis are available. Package e1071 in the R Statistical Software makes use of LIBSVM, an extensive library for Support Vector Machines. RtextTools, a machine learning package for automatic text classification, includes SVM techniques as well.

**References**

Boser BE, Guyon IM, Vapnik VN. A training algorithm for optimal margin classifiers. Proceedings of the Fifth Annual Workshop on Computational Learning Theory 1992; 144.

Christianini N, Shawe-Taylor J. An Introduction to Support Vector Machines and other Kernel-based Learning Methods. Cambridge, UK: Cambridge University Press; 2000.