CCIT Report #557 September 2005

Support Vector Machine via Sequential Subspace Optimization

Guy Narkiss guyn@siglab.technion.ac.il
Michael Zibulevsky mzib@ee.technion.ac.il
Department of Electrical Engineering
Technion - Israel Institute of Technology
Haifa 32000, Israel

Abstract

We present an optimization engine for large scale pattern recognition using Support Vector Machine (SVM). Our treatment is based on conversion of soft-margin SVM constrained optimization problem to an unconstrained form, and solving it using newly developed Sequential Subspace Optimization (SESOP) method. SESOP is a general tool for large-scale smooth unconstrained optimization. At each iteration the method minimizes the objective function over a subspace spanned by the current gradient and by directions of few previous steps and gradients. Following an approach of A. Nemirovski, we also include into the search subspace the direction from the starting point to the current point, and a weighted sum of all previous gradients: this provides the worst case optimality of the method. The subspace optimization can be performed extremely fast in the cases when the objective function is a combination of expensive linear mappings with computationally cheap non-linear functions, like in the unconstrained SVM problem. Presented numerical results demonstrate high efficiency of the method.

Keywords: Large-scale optimization, pattern recognition, Support Vector Machine, conjugate gradients, subspace optimization

1. Introduction

The problem of large-scale binary data classification arises in many applications, like recognition of text, hand-written characters, images, medical diagnostics, etc. Quite often, the number of features or examples is very large, say $10^4 - 10^7$ and more, and there is a need for algorithms, for which storage requirement and computational cost per iteration grow not more than linearly in those parameters. One way to treat such problems with SVM (Vapnik, 1998) is to convert a constrained SVM problem into an unconstrained one and solve it with an optimization method having non-expensive iteration cost and storage. An appropriate optimization algorithm of this type is the conjugate gradient (CG) method (Hestenes and Stiefel, 1952; Gill et al., 1981; Shewchuk, 1994). It is known that CG worst case convergence rate for quadratic problems is $O(k^{-2})$ (in terms of objective function calculations), where k is the iteration count. This rate of convergence is independent of the problem size and is optimal, *i.e.* it coincides with the complexity of convex smooth unconstrained optimization (see *e.g.* Nemirovski (1994)). However the standard extensions of CG to nonlinear functions by Fletcher-Reeves and Polak-Ribière (see *e.g.* Shewchuk (1994)) are no longer worst-case optimal.