Convex and Nonsmooth Optimization
HW8: ADMM

Michael Overton

Spring 2020

Implement the ADMM method for the LASSO problem on p. 43 of the ADMM paper by Boyd et al. See p. 32 for the definition of the soft thresholding operator S. My annotated version of the relevant parts of the Boyd paper is here. First test the method on a small problem and check whether you get the right answer using CVX. Then fix A to be a randomly generated large but very sparse matrix via `A=sprandn(M,N,density);` (sparse-random-normal distribution) with M=100000, N=10000, density=2/M. Take a look at its sparsity pattern by typing `spy(A),shg` (note that `nz`, at the bottom, means number of nonzeros, which you can also display directly with `nnz(A)`). Set `b=randn(M,1);` and fix ρ (`rho`) to some positive value. Then, before the ADMM iteration starts, compute the *sparse* Cholesky factorization LL^T (BV p. 669) of $A^TA + \rho I$ via `B = A'*A + rho*speye(N); L=chol(B,'lower');` (here, `speye` means sparse identity; do *not* compute the dense identity matrix `eye(N)`). Take a look at L's sparsity with `spy(L),shg`. Then, inside the ADMM iteration you can solve the relevant systems with forward and back substitution (BV, Algorithm C.2, p. 670), using the backslash operator `\`. For both the small problem and the large problem:

- Experiment with λ : does larger λ result in solutions x which are more sparse, as it should?
- Experiment with ρ : what effect does this have on the method?

Choose the stopping criteria so you get the residuals as small as you can within a reasonable running time (the definition of "reasonable" is entirely up to you!) Submit your MATLAB files, *including comments for full credit*, and relevant log plots showing the behavior of the method, along with your discussion.