Homework 6: Conjugate Gradients

CS 205A: Mathematical Methods for Robotics, Vision, and Graphics (Fall 2013),
Stanford University

Due Monday, November 18, at midnight

This problem set is a bit shorter to help those students who have not used Matlab before. If you do not have Matlab on your personal computer, it is installed on the cluster machines in Gates.

Problem 1 (50 points). *A* graph *is a data structure $G = (V, E)$ consisting of n vertices in a set $V = \{1, \ldots, n\}$ and a set of edges $E \subseteq V \times V$. A common problem is* graph layout, *where we choose positions of the vertices in V on the plane $\mathbb{R}^2$ respecting the connectivity of G. For this problem we will assume $(i, i) \notin E$ for all $i \in V$.*

(a) *Take $\vec{v}_1, \ldots, \vec{v}_n \in \mathbb{R}^2$ to be the positions of the vertices in V; these are the unknowns in graph layout. The Dirichlet energy of a layout is*

$$E(\vec{v}_1, \ldots, \vec{v}_n) = \sum_{(i,j)\in E} \|\vec{v}_i - \vec{v}_j\|_2^2.$$

Suppose an artist specifies positions of vertices in a nonempty subset $V_0 \subseteq V$. We will label these positions as $\vec{v}_k^0$ for $k \in V_0$. Derive two $(n - |V_0|) \times (n - |V_0|)$ linear systems of equations satisfied by the x and y components of the unknown $\vec{v}_i$'s solving the following minimization problem:

$$\begin{aligned} &\text{minimize } E(\vec{v}_1, \ldots, \vec{v}_n) \\ &\text{such that } \vec{v}_k = \vec{v}_k^0 \ \forall k \in V_0 \end{aligned}$$

Hint: Your answer should yield two linear systems $A\vec{x} = \vec{b}_x$ and $A\vec{y} = \vec{b}_y$.

(b) *Complete `problem1.m` to implement and solve this system in Matlab:*

- (i) *Compute the (symmetric positive definite) matrix A as a variable `A` and put both right hand sides into the $(n - |V_0|) \times 2$ matrix `rhs`. Make sure that you use a sparse matrix for `A`.*
- (ii) *Implement gradient descent on the x and y components simultaneously; the starter code will help you visualize the output during each iteration. For this and the next part, you will need to formulate and implement reasonable convergence conditions.*
- (iii) *Implement conjugate gradients for x and y simultaneously. You may wish to use the pseudocode in Appendix B2 of Shewchuk's document as a guide.*
- (iv) *Compare the number of iterations needed to reach a reasonable solution using both strategies.*
- EC. *Implement preconditioned conjugate gradients using a preconditioner of your choice. How much does convergence improve?*

Note: Two test cases, a small graph and a larger graph, are provided. You may wish to use the smaller example to debug, but make sure your code gives proper output on the larger example.

Problem 2 (25 points). *In this problem we will derive an iterative method of solving $A\vec{x} = \vec{b}$ via* splitting.

(a) *Suppose we decompose $A = M - N$, where M is invertible. Show that the iterative scheme $\vec{x}_k = M^{-1}(N\vec{x}_{k-1} + \vec{b})$ converges to $A^{-1}\vec{b}$ when $\max\{|\lambda| : \lambda \text{ is an eigenvalue of } M^{-1}N\} < 1$.*
Hint: Define $\vec{x}^* = A^{-1}\vec{b}$ and take $\vec{e}_k = \vec{x}_k - \vec{x}^*$. Show that $\vec{e}_k = G^k\vec{e}_0$, where $G = M^{-1}N$. For this problem, you can assume that the eigenvectors of G span $\mathbb{R}^n$ (in fact, it is possible to prove this statement without the assumption but doing so requires more analysis than we have covered in CS 205A).

(b) *Suppose A is diagonally dominant, that is, for each i it satisfies*

$$\sum_{j \neq i} |a_{ij}| < |a_{ii}|.$$

Suppose we define M to be the diagonal part of A and $N = M - A$. Show that the iterative scheme from part (a) converges in this case.
Note: You can assume the statement from (a) holds regardless of the eigenspace of G.

Problem 3 (25 points; CS 205A spring 2013). *Suppose $\vec{x}_1, \ldots, \vec{x}_k$ are A-orthogonal and nonzero. Show that if A is symmetric and positive definite, then $\vec{x}_1, \ldots, \vec{x}_k$ are linearly independent. Does this hold when A is symmetric but not positive definite?*