Optimization Theory and Algorithms

Gauss-Newton Method for Truncated SVD

Optimization Model

We consider the nonlinear least squares problem:

$$\min_{X \in \mathbb{R}^{n \times k}} \; f(x) \equiv \frac{1}{4} \left\| XX^T - A \right\|_F^2, \tag{1}$$

where $X \in \mathbb{R}^{n \times k}$ with $k < n$, and $A \in \mathbb{R}^{n \times n}$ is given and symmetric positive semi-definite. The problem is to find the best rank-k approximation of A, and solvable by eigen-decomposition. When $A = B^T B$, it is solvable by computing the SVD of B. However, we will try a pure Gauss-Newton method proposed and studied in the paper: *An Efficient Gauss-Newton Algorithm for Symmetric Low-Rank Product Matrix Approximations* (by X. Liu, Z. Wen and Y. Zhang, **SIAM Journal on Optimization**. 25-3 (2015), pp. 1571-1608. Link: `http://dx.doi.org/10.1137/140971464`). Glance the paper to get a rough idea on what the paper is about, but do try to understand the mechanics of Algorithm 2.

Assignment

Implement Algorithm 2 in the above paper (a pure Gauss-Newton method) by writing a Matlab function

```
[x, iter] = myGN(A,X0,tol,maxit);
```

where A is the given matrix, `X0` is an initial guess, `tol` is an tolerance value for termination, and `maxit` the the maximum number of iterations allowed. The termination criterion is

$$\frac{\|X^k - X^{k-1}\|_F}{\|X^{k-1}\|_F} \leq tol.$$

Requirements

Download and the zip file `handout-GN.zip` which contains two images and a code by the instructor, and the test script `test_panda.m` which can be run without your code. You can use the smaller Image #1 for debugging and experimenting.

Once ready, run the test script `test_panda` on Image #2 and submit the following items:

- A copy of your code, Matlab screen printout, and the figure generated.
- A short summary (typed, half a page or less) about your experiments and observations that you consider to be the most relevant and most important.