Homework6(SVD) Report for Algorithms for Big-Data Analysis

Haiwen Huang 1500010657

May 14, 2018

1 Algorithm

In this homework[1]. I implemented the prototype algorithm for Randomized SVD on page 227 of N. Halko, P. G. Martinsson, and J. A. Tropp, Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions,SIAM Rev., 53(2), 217288. The algorithm can be summarized in Figure 1.

PROTOTYPE FOR RANDOMIZED SVD

Given an $m \times n$ matrix $\boldsymbol{A}$, a target number k of singular vectors, and an exponent q (say, $q = 1$ or $q = 2$), this procedure computes an approximate rank-$2k$ factorization $\boldsymbol{U\Sigma V}^*$, where $\boldsymbol{U}$ and $\boldsymbol{V}$ are orthonormal, and $\boldsymbol{\Sigma}$ is nonnegative and diagonal.

Stage A:

1 Generate an $n \times 2k$ Gaussian test matrix $\boldsymbol{\Omega}$.

2 Form $\boldsymbol{Y} = (\boldsymbol{AA}^*)^q \boldsymbol{A\Omega}$ by multiplying alternately with $\boldsymbol{A}$ and $\boldsymbol{A}^*$.

3 Construct a matrix $\boldsymbol{Q}$ whose columns form an orthonormal basis for the range of $\boldsymbol{Y}$.

Stage B:

4 Form $\boldsymbol{B} = \boldsymbol{Q}^*\boldsymbol{A}$.

5 Compute an SVD of the small matrix: $\boldsymbol{B} = \widetilde{\boldsymbol{U}}\boldsymbol{\Sigma V}^*$.

6 Set $\boldsymbol{U} = \boldsymbol{Q}\widetilde{\boldsymbol{U}}$.

Note: The computation of $\boldsymbol{Y}$ in step 2 is vulnerable to round-off errors. When high accuracy is required, we must incorporate an orthonormalization step between each application of $\boldsymbol{A}$ and $\boldsymbol{A}^*$; see Algorithm 4.4.

Figure 1: Prototype Algorithm for Randomized SVD

The implementation can be seen in file prototype.m.

[1]FYI: I didn't do extra-credit part this time.

2 Random matrix test and results

We first test our algorithm on a random matrix A generated as specified in homework assignment. We compute $r \in \{5, 10, 15, 20\}$ largest singular values of A. And get their corresponding singular vectors. See random$_t$est.m.

We plot the the svd result of singular values and our approximate singular values below. We can see that the error is large only when $r = 5$. When $r \geq 10$, the approximate singular values are almost the same as svd singular values. Our error[2] when $r = \{5, 10, 15, 20\}$ is 1.1074e+03, 2.9887e-11, 3.2312e-12, 1.6449e-12.

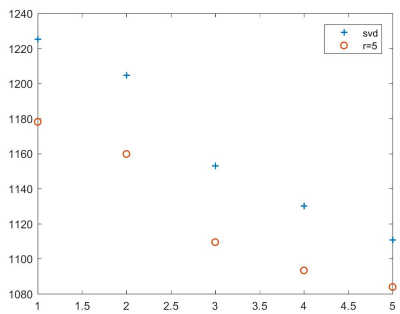

Figure 2: Random matrix; r=5

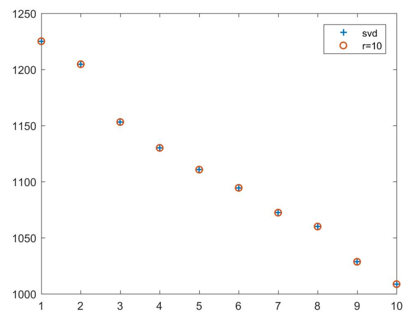

Figure 3: Random matrix; r=10

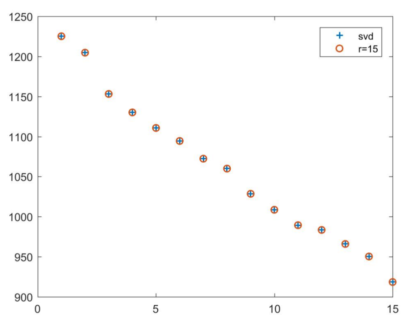

Figure 4: Random matrix; r=15

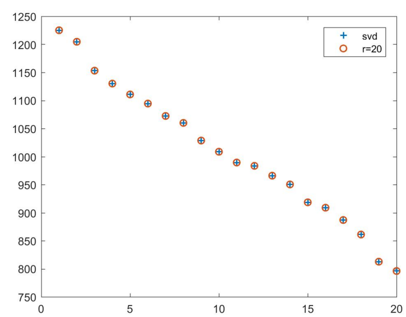

Figure 5: Random matrix; r=20

3 Practical data set test and results

We now use the generated PCA matrix data set in "Wenjian Yu, Yu Gu, Jian Li, Shenghua Liu, and Yaohang Li, Single-Pass PCA of Large High-Dimensional

[2] Error is defined as $||A - QQ^*A||$, see the referenced paper

Data" (Section4.1 Accuracy Validation). There are 5 types of matrices in the paper. We only show the results for type1. The results for the other types are pretty similar. And for this A, we also let it be 2048×512. We can also generate the exact singular values of A, which will be used in our comparison plot. The detail of the test can be found in test2.m.

Same as the previous section. We plot the exact singular values versus our approximate singular values. The error when when $r = \{5, 10, 15, 20\}$ is 0.0099, 1.4368e-04, 9.2612e-05, 8.6833e-05.

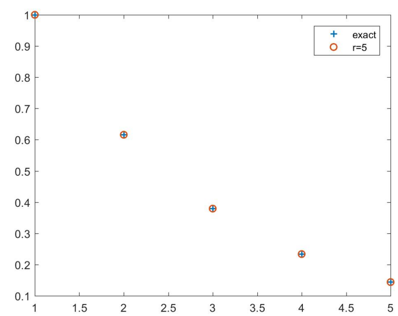

Figure 6: Generated matrix; r=5

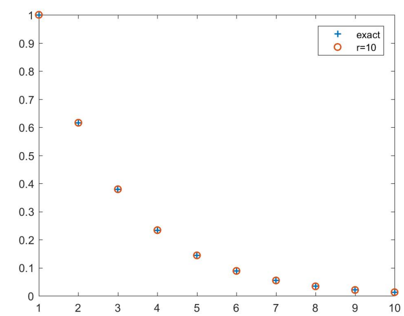

Figure 7: Generated matrix; r=10

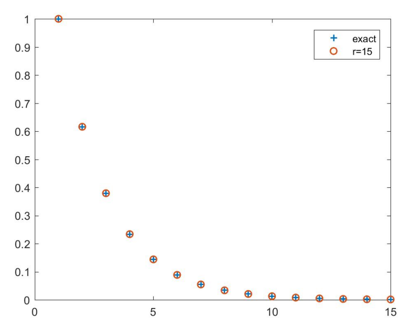

Figure 8: Generated matrix; r=15

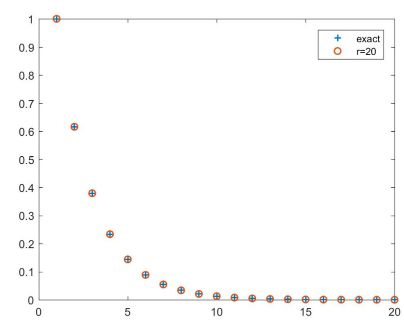

Figure 9: Generated matrix; r=20