1. Review: Single-Pass Algorithm for Hermitian Matrices

This section is a review from class on 02/01/2016. Let $\mathbf{A}$ be an $n \times n$ Hermitian matrix. Further, define k to be our target rank and p the oversampling parameter. For notational convenience, let $l = k + p$.

1.1. Single-Pass Hermitian - Stage A.

1) Draw Gaussian matrix $\mathbf{G}$ of size $n \times l$
2) Compute $\mathbf{Y} = \mathbf{AG}$, our sampling matrix.
3) Find $\mathbf{Q} = orth(\mathbf{Y})$ via QR

Recall:
$\mathbf{A} \approx \mathbf{QQ}^*\mathbf{AQQ}^*$. Let $\mathbf{C} = \mathbf{Q}^*A\mathbf{Q}$. Calculate the eigendecomposition of $\mathbf{C}$ to find: $\mathbf{C} = \hat{\mathbf{U}}\mathbf{D}\hat{\mathbf{U}}^*$. Then $\mathbf{A} \approx \mathbf{QCQ}^* = \mathbf{Q}(\hat{\mathbf{U}}\mathbf{D}\hat{\mathbf{U}}^*)\mathbf{Q}^*$. Set $\mathbf{U} = \mathbf{Q}\hat{\mathbf{U}}$, $\mathbf{U}^* = \hat{\mathbf{U}}^*\mathbf{Q}^*$ and $\mathbf{A} \approx \mathbf{UDU}^*$. We find $\mathbf{C}$ by solving $\mathbf{C}(\mathbf{Q}^*\mathbf{G}) = \mathbf{Q}^*\mathbf{Y}$ in the least squares sense making sure to enforce $\mathbf{C}^* = \mathbf{C}$. Now consider replacing step (3) with the calculation of an 'econ' SVD on $\mathbf{Y}$. Let $\mathbf{Q}$ contain the first k left singular vectors of our factorization and proceed as normal. This will yield a substantially overdetermined system. The above procedure relies on $\mathbf{A}$ being symmetric, how do we proceed if this is not the case?

2. Single-Pass for General Matrix

Let $\mathbf{A}$ be a real or complex valued $m \times n$ matrix. Further, define k to be our target rank and p the oversampling parameter. For notational convenience, let $l = k + p$. We aim to retrieve an approximate SVD:

$$\underset{m \times n}{\mathbf{A}} \approx \underset{m \times k}{\mathbf{U}} \; \underset{k \times k}{\mathbf{D}} \; \underset{k \times n}{\mathbf{V}^*}$$

To begin, we will modify "Stage A" from Section 1.1 to output orthonormal matrices $\mathbf{Q_c}$, $\mathbf{Q_r}$ such that:

$$\underset{m \times n}{\mathbf{A}} \approx \underset{m \times k}{\mathbf{Q_c}} \; \underset{k \times m}{\mathbf{Q_c}^*} \; \underset{m \times n}{\mathbf{A}} \; \underset{n \times k}{\mathbf{Q_r}} \; \underset{k \times n}{\mathbf{Q_r}^*}$$

With $\mathbf{Q_c}$ an approximate basis for the column space of $\mathbf{A}$ and $\mathbf{Q_r}$ an approximate basis for the row space of $\mathbf{A}$. We will then set $\mathbf{C} = \mathbf{Q_c^*AQ_r}$ and proceed in the usual fashion. First let's justify the means in which we aim to find C. First, right multiply $\mathbf{C}$ by $\mathbf{Q_r^*G_c}$ to find: $\mathbf{CQ_r^*G_c} = \mathbf{Q_c^*AQ_rQ_r^*G_c} = \mathbf{Q_c^*AG_c} = \mathbf{Q_c^*Y_c}$. Similarly, left multiply $\mathbf{C}$ by $\mathbf{G_r^*Q_c}$ to find: $\mathbf{G_r^*Q_cC} = \mathbf{G_r^*Q_cQ_c^*AQ_r} = \mathbf{G_r^*AQ_r} = \mathbf{Y_r^*Q_r}$. Keep this in mind when we move to "Stage B".

2.1. Single-Pass General - Stage A.

1) Draw Gaussian matrices $\mathbf{G_c}$, $\mathbf{G_r}$ of size $n \times l$
2) Compute $\mathbf{Y_c} = \mathbf{AG_c}$, $\mathbf{Y_r} = \mathbf{A}^*\mathbf{G_r}$
3) Find [$\mathbf{Q_c}$, ,] = svd($\mathbf{Y_c}$,'econ'), [$\mathbf{Q_r}$, ,] = svd($\mathbf{Y_r}$,'econ')
4) $\mathbf{Q_c} = \mathbf{Q_c}(:, 1:k)$, $\mathbf{Q_r} = \mathbf{Q_r}(:, 1:k)$

2.2. Single-Pass General - Stage B.

5) Determine a $k \times k$ matrix $\mathbf{C}$ by solving:

$$\underset{k \times k}{\mathbf{C}} \; \underset{k \times l}{(\mathbf{Q_r^*G_c})} = \underset{k \times l}{\mathbf{Q_c^*Y_c}} \quad \textbf{and} \quad \underset{l \times k}{(\mathbf{G_r^*Q_r})} \; \underset{k \times k}{\mathbf{C}} = \underset{l \times k}{\mathbf{Y_r^*Q_r}}$$

in the least squares sense. (Note: There are $2k^2$ equations for k^2 unknowns which represents a system that is very overdetermined.)
6) Compute SVD: $\mathbf{C} = \hat{\mathbf{U}}\mathbf{D}\hat{\mathbf{V}}^*$
7) Set $\mathbf{U} = \mathbf{Q_c}\hat{\mathbf{U}}$, $\mathbf{V} = \mathbf{Q_r}\hat{\mathbf{V}}$

It should be noted that the General case reduces to the Hermitian case given a suitable matrix $\mathbf{A}$. A natural follow up questions targets the reduction of asymptotic complexity. Can we reduce the FLOP count, say from $O(mnk)$, to $O(mnlog(k))$?

3. Reduction of Asymptotic Complexity

3.1. **Review of RSVD.** Let $\mathbf{A}$ be a dense $m \times n$ matrix that fits in RAM, designate k, p, l in the usual fashion. When computing the RSVD of $\mathbf{A}$, there are two FLOP intensive steps that require $O(mnk)$ operations (Please see course notes from 1/29/2016 for more detail). We will first concentrate on accelerating the computation of $\mathbf{Y} = \mathbf{AG}$ with $\mathbf{G}$ an $n \times l$ Gaussian matrix. To do so, consider replacing $\mathbf{G}$ by a new random matrix, $\mathbf{\Omega}$ with a few (seemingly contradictory) properties. These are:

- $\mathbf{\Omega}$ has enough structure to ensure that $\mathbf{A\Omega}$ can be evaluated in $(mnlog(k))$ flops.
- $\mathbf{\Omega}$ is random enough to be reasonably certain that the columns of $\mathbf{Y} = \mathbf{A\Omega}$ approximately span the column space of $\mathbf{A}$.

How can such an $\mathbf{\Omega}$ be found? Are there any examples of one?

3.2. **Example of $\mathbf{\Omega}$:** Let $\mathbf{F}$ be the $n \times n$ DFT and note $\mathbf{F}^*\mathbf{F} = \mathbf{I}$ ($\mathbf{F}$ is called a ”rotation”). Define $\mathbf{D}$ to be diagonal with random entries and $\mathbf{S}$ a subsampling matrix. Let $\mathbf{\Omega}$ be:

$$\underset{n \times l}{\mathbf{\Omega}} = \underset{n \times n}{\mathbf{D}} \; \underset{n \times n}{\mathbf{F}} \; \underset{n \times l}{\mathbf{S}^*}$$

We are one step closer to the mythical $\mathbf{\Omega}$. Further details in subsequent lectures.