CSE 383C: Numerical Linear Algebra

Fall 2016

1 Algorithm Complexity

Let $A \in \mathbb{C}^{m \times n}$, $x, y, z \in \mathbb{C}^n$, $H \in \mathbb{C}^{m \times m}$, $u, v \in \mathbb{C}^m$.

i. **Inner Product** $x^t y$ - $O(2n)$ time, $O(1)$ space.

ii. **Outer Product** xy^t - $O(n^2)$ time, $O(n^2)$ space.

iii. **Outer Product Vector** $(xy^t)z = x(y^t z)$ - $O(3n)$ time, $O(n)$ space.

iv. **Dense Matrix Vector** Ax - $O(2mn)$ time, $O(m)$ space.

v. **Sparse Matrix Vector** Ax - $O(m+n)$ time, $O(m+n)$ space.

vi. **Gram Schmidt** - $O(\frac{3}{2}mn^2)$ time.

vii. **Householder Vector** $Hv = (I - 2uu^*)v = v - 2uu^*v$ - $O(4m)$ time.

viii. **Householder** - $O(2mn^2 - \frac{2}{3}n^3)$ time.

ix. **Givens** - $O(mn^2)$ time.

x. **SVD** - $O(2mn^2 + 11n^3)$ time.

xi. **Backsubstitution** - $O(n^2)$ time.

xii. **LU (With or without PP)** - $O(\frac{2}{3}m^3)$ time.

xiii. **Cholesky** - $O(\frac{1}{3}m^3)$ time.

2 QR Factorizations

Given a matrix $A \in \mathbb{R}^{m\times n}$, we can write $A = QR$, where Q is orthonormal and R is upper triangular. This matrix factorization exists **for all** matrices.

2.1 Reduced QR

If $A \in \mathbb{R}^{m\times n}$, then $A = QR$ produces $Q \in \mathbb{R}^{m\times n}$ and $R \in \mathbb{R}^{n\times n}$.

The typical GS orthogonalization produces this.

2.2 Full QR

If $A \in \mathbb{R}^{m\times n}$, then $A = QR$ produces $Q \in \mathbb{R}^{m\times m}$ and $R \in \mathbb{R}^{m\times n}$.

A typical GS factorization loops over the columns of A and orthogonalizes that column with respect to the previous columns of A. But if $m > n$, there are only n columns of A and so there are $m - n$ more orthogonal vectors that we need to form a basis for $\mathbb{R}^{m\times m}$.

This means we need $m - n$ more linearly independent vectors. Well we can just pick random vectors to orthogonalize because the probability of picking a vector that aligns exactly with a previous one (linearly dependent vector) is 0.

Another option is just to use Householder QR, or Givens QR.

2.3 Gram Schmidt

Given a matrix $A \in \mathbb{R}^{m\times n}$, A full rank (why? explained later), we want to form an orthogonal basis for the range of A.

Pick the first column of A, called a_1. We want an orthogonal basis for span of a_1, well we can just pick $v_1 = a_1$.

Now we are on the second column of A, called a_2. We want to now find an orthogonal vector to v_1. Well we can just find the projection onto v_1, defined as $v_1 v_1^T a_2$, and then subtract this bit off off v_1. So $v_2 = (I - v_1 v_1^T)a_2$.

We are now on the third column of A, called a_3. We want to find an orthogonal vector to $v_1,\ v_2$. Well we can find this by finding the component of a_3 that lives in $span\{v_1,\ v_2\}$, then subtracting that component from a_3. So $v_3 = (I - v_1 v_1^T - v_2 v_2^T)a_3$.

We continue until we have gone through every column, now we have an orthogonal basis for $Range(A)$, but this is not orthonormal. We can simply normalize each column $q_i = \frac{v_i}{|v_i|}$.

Now we have formed our matrix Q, and the R follows. A column of R, say r_j, tells us the linear combination of Q that we need to form the corresponding column a_j. By construction, R is upper triangular.

Why does A have to be full rank? If the columns of A are not linearly independent, then when we try to find a orthogonal vector, we will get a $v_i = 0$, and get NaNs in our answer.

2.4 Modified Gram Schmidt

In CGS, we use $v_i = a_i - \sum_{j=1}^{i-1} q_j q_j^* a_i$, but if the columns of A are almost linearly dependent, the inner product and subtraction operations will cause large numerical instabilities, and cause $q_i \cdot q_j \neq 0$.

Instead, we will initialize $v_i = a_i$, but then for every iteration, we do $v_i = v_i - q_j q_j^* v_i \; \forall j < i$. This makes it so that even though we have some instabilities in R, we focus on the orthogonality of Q, and we can bound $|Q^*Q - I| = O(\kappa(A)\epsilon_m)$.

3 LU Decomposition

Let $A \in \mathbb{C}^{m \times m}$ and A nonsingular. Then, A admits a LU Decomposition of the form $A = LU$, where L lower triangular and U upper triangular, and both matrices have nonzeros along the diagonal.

Then, the solution to $LUx = b$ will be $Ux = L^{-1}b$, which has a forward substitution and a backwards substitution that take $O(2m^2)$.

If A is diagonally dominant, or symmetric positive definite, then the unpivoted LU decomposition exists, and the growth factor $p = O(1)$.

3.1 Pivoted LU Decomposition

Even if A is well conditioned, a naive LU decomposition will fail. We have to introduce pivoting at each step.

Theorem 1. Pivoted LU is backwards stable.

Let $PAQ = LU$ be the exact pivoted factorization of a non-singular matrix A. Let $\tilde{L}, \tilde{U},\ \tilde{P},\ \tilde{Q}$ be the computed factorization on an IEEE-754 machine.

Then,

$\tilde{L}\tilde{U} = \tilde{P}A\tilde{Q} + \delta A;\ \frac{\|\delta A\|}{\|A\|} = O(p\ \epsilon_m)$

Where p is called the growth factor of A and depends on the pivoting method.

There are a couple forms of pivoting - partial, full, and rook. In practice, partial pivoting is used, and the growth factor $p \leq 2^m$. So even though this algorithm is backwards stable, this can be potentially highly erroneous.

3.2 Cholesky Decomposition

If A is symmetric, positive-definite, then $A = LU = R^T R$, where R is upper triangular. This takes half the amount of time and space as a typical LU decomposition.

4 Least Squares

4.1 Underdetermined Systems

Let $A \in \mathbb{C}^{m \times n}$, where $m < n$, and A full rank. Now we an infinite number of solutions x to $Ax = b$. There are a couple methods to pick the "best" x.

Theorem 2. Underdetermined System Error Analysis

If $\frac{\|\Delta A\|}{\|A\|} < \sigma_{min}$, $\frac{\|\Delta b\|}{\|b\|} < \sigma_{min}$.

Then $\frac{\|\Delta x\|}{\|x\|} \leq \kappa(A)\{\frac{\|\Delta A\|}{\|A\|} + \frac{\|\Delta b\|}{\|b\|}\}$.

4.1.1 Regularized SVD

This method penalizes the norm of the solution x by a factor of β, known as the regularization term. We have some squared terms and $\frac{1}{2}$ terms to make differentiation easier, but it is the same minimization problem.

$$\arg\min_x \frac{1}{2}\|Ax - b\|_2^2 + \frac{1}{2}\beta\|x\|_2^2 \tag{1}$$

Using the reduced SVD, we have $A = U\Sigma V^*$, and $U \in \mathbb{C}^{m \times m}$, $\Sigma \in \mathbb{C}^{m \times m}$, and $V \in \mathbb{C}^{n \times m}$.

We substitute $x = Vy$, since the solution should live in the rowspace of A, and $q = U^* b$ and now we have the following.

$$\arg\min_y \frac{1}{2}\|\Sigma y - q\|_2^2 + \frac{1}{2}\beta\|y\|_2^2 \tag{2}$$

Now, each y_i term is independent, and taking the partial derivatives of y and setting it to 0, we can solve for the optimal y.

$$\arg\min_{y_i} J(y_i) = \frac{1}{2}(\sigma_i y_i - q_i)^2 + \frac{1}{2}\beta y_i^2 \tag{3}$$

$$\frac{\partial}{\partial y_i} J(y_i) = (\sigma_i^2 y - \sigma q_i) + \beta y_i \tag{4}$$

$$y_i = \frac{\sigma_i q_i}{\sigma_i^2 + \beta} \tag{5}$$

4.1.2 Truncated SVD

Truncated SVD is just a specific case of regularized SVD, with $\beta = 0$.

A full SVD decomposition of A gives us Σ with have $n-m$ zero columns, which correspond with the $n-m$ rightmost columns of V, where $span\{v_{n-m}, \dots, v_m\} = Null(A)$.

Truncated SVD says let's forget about the vectors in $Null(A)$, and take Σ_t to be the first m columns of Σ, and take V_t to be the first m columns of V. We now have $\Sigma_t \in \mathbb{C}^{m\times m}$, and $V_t \in \mathbb{C}^{n\times m}$. This is the reduced SVD of A.

The solution to $Ax = b$ is now clearly $x = V_t\Sigma_t^{-1}U^*b$.

4.2 Rank Deficient Systems

Let $A \in \mathbb{C}^{m\times n}$, where $m > n$, and $rank(A) < n$.

The SVD decomposition of A shows us that we have some singular values that are 0. We can ignore the bottom $n-r$ rows of Σ and the corresponding U and V vectors, and solve this with the techniques described in underdetermined systems.

4.3 Nearly Rank Deficient Systems

If we have $\kappa_2(A) = \frac{\sigma_{max}}{\sigma_{min}}$ very large, this tells us the spread of singular values is very large, and numerical methods will have high relative error. Using the SVD of A, we can easily tell A is ill conditioned, and set the corresponding singular values under some threshold τ to 0, and solve an underdetermined system.

However, SVD is not viable for large systems, and we have to use Pivoted QR, which will reveal the rank of A.

Theorem 3. The singular values of the block matrix produced by Column Pivoted QR $R_{k,k}$ are related to the singular values of A.

$\sigma_k(R_{k,k}) = O(\sigma_k(A))$

The R matrix produced by Column Pivoted QR will have small values along the diagonal, which tells us the corresponding vectors of Q that are not spanned. We can ignore these values and truncate our Q to $Q_t \in \mathbb{C}^{m\times r}$, and R to $R_t \in \mathbb{C}^{r\times r}$.

Now we can solve a generic $Q_tR_tx' = b$, for $x' \in \mathbb{C}^r$. We fill in $n-r$ values of x' to 0 to get $\bar{x} \in \mathbb{C}^n$. Finally, we have to permute the rows since $AP = QR$, and acquire our final solution $x = P\bar{x}$.

5 Eigenvalue Problems