**MATH 5330: Computational Methods of Linear Algebra**

# Lecture Note 10: Householder Transformation

Xianyi Zeng
Department of Mathematical Sciences, UTEP

## 1 The QR Decomposition

Using Givens rotations allows us to write $A = QE$ where $Q$ is orthogonal and $E$ is of the row echelon form. Note that the lower-triangular part of $E$ is always zero, i.e. $e_{ij} = 0$ if $i > j$; thus this is decomposition is also known as the QR decomposition, where "R" stands for right-triangular or upper triangular. Henceforth we will write $A = QR$ instead of $A = QE$. In the special case when $A$ is square and non-singular, we immediately see that $R = Q^{-1}A$ is also non-singular; hence all its diagonal elements are non-zero. Furthermore, we can compute the inverse of $A$ easily by:

$$A^{-1} = R^{-1}Q^t \,.$$

The advantage of using the QR decomposition over the LU decomposition to invert a non-singular matrix is that all its components have an *a priori* estimate. However, the QR decomposition requires much more computational cost ($\sim 2n^3$) than the Gaussian elimination ($\sim 2n^3/3$). Robustness does not come free!

Givens rotations belong to one of three widely used methods to compute $A = QR$:

- *Gram Schmidt*: Since every column of $A$ is a linear combination of the columns of $Q$, we have $\mathrm{col}(A) \subseteq \mathrm{col}(Q)$; thus in the end the QR decomposition can be reduced to orthogonalization of the column vectors of $A$. We have already seen in the Arnoldi's method that, this can be achieved by the Gram Schmidt process. The major problem of the Gram Schmidt process is similar to that of Gaussian elimination: The algorithm involves dividing by a number that cannot be estimated *a priori*, hence it can breakdown unexpected.

- *Givens rotations*: This method is more robust than the Gram Schmidt, and in each rotation only two adjacent rows are involved so it is more bandwidth efficient and reduce cache misses. A major objection for using the Givens rotation is its complexity in implementation; particularly people found out that the ordering of the rotations actually matter in practice [1], and determining the optimal order is a non-trivial problem.

- *Householder transformation*: This method is robust like the one using Givens rotations, easier to implement, and requires less computation. However, it is less bandwidth efficient and more difficult to parallelize than the latter.

At the end of this section, we prove that if $A$ is square and non-singular, the QR decomposition is unique, if we require in addition that all diagonal entries of $R$ are positive. In fact, if $A = Q_1R_1 = Q_2R_2$ where both $Q_1$ and $Q_2$ are orthogonal and both $R_1$ and $R_2$ are upper-triangular, we then have:

$$Q_2^tQ_1 = R_2R_1^{-1} \,.$$

The left hand side is orthogonal whereas the right hand side is upper triangular. It is not difficult to see that the only upper triangular matrix that is also orthogonal is diagonal matrices with diagonal elements being $\pm 1$. By the assumption that the diagonal elements of both $R_1$ and $R_2$ are positive, the only possibility is $R_2 R_1^{-1} = I$, or equivalently $Q_1 = Q_2$ and $R_1 = R_2$.

## 2 The Householder Transformation

A major motivation for using Givens transform to construct the QR decomposition is that rotations preserve the $L^2$-norm of vectors. These are, however, not the only operations that have this property. For example, the reflection about any plane also preserve the $L^2$-norm of vectors in $\mathbb{R}^n$. The method utilizing this latter property is built on the Householder transformation.

There are at least two ways to describe a Householder matrix. Algebraically, a Householder matrix differs from the identity matrix by a rank one matrix as follows:

$$H_{\boldsymbol{v}} = I - 2\boldsymbol{v}\boldsymbol{v}^t\,, \tag{2.1}$$

where $\boldsymbol{v}$ is a unit vector. Furthermore, $H_{\boldsymbol{v}}$ is symmetric and orthogonal; and we'll check the latter:

$$\begin{aligned} H_{\boldsymbol{v}}^t H_{\boldsymbol{v}} &= (I - 2\boldsymbol{v}\boldsymbol{v}^t)(I - 2\boldsymbol{v}\boldsymbol{v}^t) = I - 4\boldsymbol{v}\boldsymbol{v}^t + 4\boldsymbol{v}(\boldsymbol{v}^t\boldsymbol{v})\boldsymbol{v}^t \\ &= I - 4\boldsymbol{v}\boldsymbol{v}^t + 4\boldsymbol{v}\boldsymbol{v}^t = I\,. \end{aligned}$$

Now let $\boldsymbol{u}$ be any vector, and we compute $H_{\boldsymbol{v}}\boldsymbol{u}$:

$$H_{\boldsymbol{v}}\boldsymbol{u} = (I - 2\boldsymbol{v}\boldsymbol{v}^t)\boldsymbol{u} = \boldsymbol{u} - 2(\boldsymbol{u}\cdot\boldsymbol{v})\boldsymbol{v}\,. \tag{2.2}$$

The second term looks familiar: $(\boldsymbol{u}\cdot\boldsymbol{v})\boldsymbol{v}$ is the projection of $\boldsymbol{u}$ onto the straight line spanned by the vector $\boldsymbol{v}$. Indeed, we see that:

$$\boldsymbol{u} = (\boldsymbol{u}\cdot\boldsymbol{v})\boldsymbol{v} + [\boldsymbol{u} - (\boldsymbol{u}\cdot\boldsymbol{v})\boldsymbol{v}]$$

is the orthogonal decomposition of $\boldsymbol{u}$ w.r.t. $\mathcal{V} = \mathrm{span}(\boldsymbol{v})$:

$$\boldsymbol{v}\cdot[\boldsymbol{u} - (\boldsymbol{u}\cdot\boldsymbol{v})\boldsymbol{v}] = \boldsymbol{v}\cdot\boldsymbol{u} - (\boldsymbol{u}\cdot\boldsymbol{v})||\boldsymbol{v}||^2 = 0\,.$$

Thus (2.2) gives the reflection of $\boldsymbol{u}$ w.r.t. $\mathcal{V}^\perp$, in the sense that the midpoint of the line connecting $\boldsymbol{u}$ and $H_{\boldsymbol{v}}\boldsymbol{u}$ is the orthogonal projection of $\boldsymbol{u}$ onto $\mathcal{V}^\perp$. This is the geometrical description of a Householder transform, and a 2D example is depicted in Figure 1.

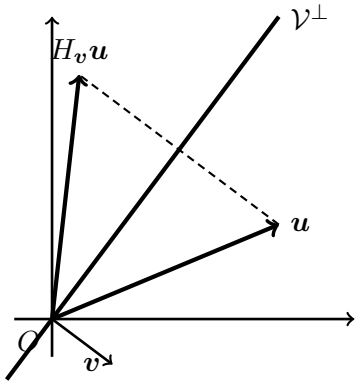


Figure 1: Reflection of $\boldsymbol{u}$ by the line $\mathcal{V}^\perp$ in $\mathbb{R}^2$.

In 2D, $\mathcal{V}^\perp$ itself is a straight line; in $\mathbb{R}^n$, $\mathcal{V}^\perp$ has dimension $n-1$ and it has normal vector $\boldsymbol{v}$. We call $\mathcal{V}^\perp$ a hyperplane with unit normal vector $\boldsymbol{v}$.

As an immediate application of the Householder transform, we can essentially map $\boldsymbol{u}$ to any other direction by choosing $\boldsymbol{v}$ carefully. In particular, we would like to find $\boldsymbol{v}$ such that $H_{\boldsymbol{v}}\boldsymbol{u}=||\boldsymbol{u}||\boldsymbol{e}_1$, i.e., all the components except the first one are zero. Let $\alpha=\boldsymbol{u}\cdot\boldsymbol{v}$ for now, then we need:

$$\begin{aligned} &u_1-2\alpha v_1=||\boldsymbol{u}||\,;\\ &u_i-2\alpha v_i=0\,,\ i=2,\cdots,n\,. \end{aligned}$$

Hence $\boldsymbol{v}$ is given by:

$$v_1=\frac{u_1-||\boldsymbol{u}||}{2\alpha}\,,\quad v_i=\frac{u_i}{2\alpha}\,,\ i=2,\cdots,n\,.$$

To proceed, note that $\boldsymbol{v}$ needs to be a unit vector, we thusly have:

$$4\alpha^2=(u_1-||\boldsymbol{u}||)^2+\sum_{i=2}^{n}u_i^2=2||\boldsymbol{u}||^2-2u_1||\boldsymbol{u}||\,.$$

Choosing $2\alpha=\sqrt{2||\boldsymbol{u}||^2-2u_1||\boldsymbol{u}||}$ we obtain the desired unit vector $\boldsymbol{v}$:

$$\boldsymbol{v}=\begin{bmatrix} \frac{u_1-||\boldsymbol{u}||}{\sqrt{2||\boldsymbol{u}||^2-2u_1||\boldsymbol{u}||}}\\ \frac{u_2}{\sqrt{2||\boldsymbol{u}||^2-2u_1||\boldsymbol{u}||}}\\ \vdots\\ \frac{u_n}{\sqrt{2||\boldsymbol{u}||^2-2u_1||\boldsymbol{u}||}} \end{bmatrix}. \tag{2.3}$$

Finally, it is not difficult to check that:

$$\boldsymbol{v}\cdot\boldsymbol{u}=\frac{u_1^2-u_1||\boldsymbol{u}||}{\sqrt{2||\boldsymbol{u}||^2-2u_1||\boldsymbol{u}||}}+\sum_{i=2}^{n}\frac{u_i^2}{\sqrt{2||\boldsymbol{u}||^2-2u_1||\boldsymbol{u}||}}=\frac{||\boldsymbol{u}||^2-u_1||\boldsymbol{u}||}{\sqrt{2||\boldsymbol{u}||^2-2u_1||\boldsymbol{u}||}}=\frac{2\alpha^2}{2\alpha}=\alpha\,.$$

For convenience, we shall denote this Householder matrix by $H(\boldsymbol{u})$, i.e., $I-2\boldsymbol{v}\boldsymbol{v}^t$ where $\boldsymbol{v}$ is given by (2.3).

# 3 The QR Decomposition using Householder Transformations

The idea is to use Householder transformation to construct a sequence of matrices $H_1$, $H_2$, $\cdots$, $H_n$ such that $H_nH_{n-1}\cdots H_1A$ is of the row echelon form. Again, the purpose of each $H_i$ is to put the $i$-th column of $A$ in the desired form without touching the previous $i-1$ columns.

The purpose of $H_1$ is to have the first column of $H_1A$ possess at most one non-zero, which is the first component. This is clearly achieved by defining $H_1$ as the Householder matrix that corresponds to (2.3) where $\boldsymbol{u}$ is replaced by the first column of $A$. Let $\boldsymbol{a}_1^{(0)}$ be the first column of

$A^{(0)} \stackrel{\text{def}}{=\!=} A$, following our previous notation we have $H_1 = H(\boldsymbol{a}_1^{(0)})$:

$$[A]_1 \mapsto [H_1 A]_1 \quad \text{is given by} \quad \begin{bmatrix} a_{11} \\ a_{21} \\ a_{31} \\ \vdots \\ a_{m1} \end{bmatrix} \mapsto \begin{bmatrix} \sqrt{a_{11}^2 + a_{21}^2 + \cdots + a_{m1}^2} \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Let $A^{(1)} = H_1 A$; now we look at the second column. Again, there are two situations: if $a_{11}^{(1)} = \sqrt{a_{11}^2 + \cdots + a_{m1}^2} = 0$, we define $H_2 = H(\boldsymbol{a}_2^{(1)})$, where $\boldsymbol{a}_2^{(1)}$ is the entire second column of $A^{(1)}$:

$$[A^{(1)}]_2 \mapsto [H_2 A^{(1)}]_2 \quad \text{is given by} \quad \begin{bmatrix} a_{12}^{(1)} \\ a_{22}^{(1)} \\ a_{32}^{(1)} \\ \vdots \\ a_{m2}^{(1)} \end{bmatrix} \mapsto \begin{bmatrix} \sqrt{(a_{12}^{(1)})^2 + (a_{21}^{(1)})^2 + \cdots + (a_{m1}^{(1)})^2} \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

If, however, $a_{11}^{(1)}$ is not zero, we do not want to touch the first row of $A^{(1)}$ and we only need to put the last $(m-2)$ elements of the second column of $A^{(1)}$ into zero. To this end, we abuse the notation and denote:

$$\boldsymbol{a}_2^{(1)} = \begin{bmatrix} a_{22}^{(1)} \\ a_{32}^{(1)} \\ \vdots \\ a_{m2}^{(1)} \end{bmatrix}$$

and define:

$$H_2 = \begin{bmatrix} 1 & \\ & H(\boldsymbol{a}_2^{(1)}) \end{bmatrix}.$$

It is not difficult to check that $H_2 A^{(1)}$ keeps the first column of $A^{(1)}$ unchanged; furthermore, the first two columns of $A$ transform as:

$$[A]_{1:2} \mapsto [H_2 H_1 A]_{1:2} \quad \text{is given by} \quad \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \\ \vdots & \vdots \\ a_{m1} & a_{m2} \end{bmatrix} \mapsto \begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} \\ 0 & \sqrt{(a_{22}^{(1)})^2 + (a_{32}^{(1)})^2 + \cdots + (a_{m2}^{(1)})^2} \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \end{bmatrix}.$$

Now suppose we have already computed $H_1$, $\cdots$, $H_k$. In order to construct$H_{k+1}$, we first identify the vector $\boldsymbol{a}_{k+1}^{(k)}$, which is composed of the last $m-p$ elements of the $(k+1)$-th column of $A^{(k)}$ that we need to transform, and then define:

$$H_{k+1} = \begin{bmatrix} I_p & \\ & H(\boldsymbol{a}_{k+1}^{(k)}) \end{bmatrix}.$$

Here $I_p$ is the identity matrix in $\mathbb{R}^{p\times p}$.

To summarize, the Householder version of QR decomposition is given by Algorithm 3.1.

**Algorithm 3.1** Orthogonal Reduction by Householder Transformation

1: Set $p=1$ and $Q=I_m$
2: **for** $i=1,2,\cdots,n$ **do**
3: Compute $s=a_{p+1,i}^2+\cdots+a_{m,i}^2$
4: **if** $s\neq 0$ **then**
5: Compute $\beta=\sqrt{s+a_{pi}^2}$
6: Compute $\gamma=\sqrt{2(\beta^2-a_{pi}\beta)}$
7: Compute $\boldsymbol{v}=[(a_{pi}-\beta)/\gamma\,,\,a_{p+1,i}/\gamma\,,\,\cdots\,,\,a_{mi}/\gamma]^t$
8: $A[p:m,i:n]\leftarrow H_{\boldsymbol{v}}A[p:m,i:n]$
9: $Q[p:m,p:m]\leftarrow Q[p:m,p:m]H_{\boldsymbol{v}}$
10: **end if**
11: **if** $a_{pi}!=0$ **then**
12: Set $p\leftarrow p+1$
13: **end if**
14: **end for**

Here we use a notation $M[\mathcal{I},\mathcal{J}]$ to denote a sub-matrix of $M$: $\mathcal{I}$ is a subset of all the row indices of $M$ and $\mathcal{J}$ is a subset of all column indices of $\mathcal{J}$; and $M[\mathcal{I},\mathcal{J}]$ is the $|\mathcal{I}|\times|\mathcal{J}|$ matrix composed of elements at the intersections of rows denoted by $\mathcal{I}$ and columns designated by $\mathcal{J}$.

# 4 Analysis and Other Aspects

First of all, we note that updating the matrices $A[p:m,i:n]$ and $Q[p:m,p:m]$ in Algorithm 3.1 does not require constructing $H_{\boldsymbol{v}}$ explicitly. Let us assume for simplicity $H_{\boldsymbol{v}}\in\mathbb{R}^{m\times m}$ and we want to compute $H_{\boldsymbol{v}}A$. Note that:

$$H_{\boldsymbol{v}}A=(I-2\boldsymbol{v}\boldsymbol{v}^t)A=A-2\boldsymbol{v}(\boldsymbol{v}^tA)\,,$$

can be accomplished column by column. That is, for each column $\boldsymbol{a}_i$ of $A$, we replace $\boldsymbol{a}_i$ by:

$$\boldsymbol{a}_i\leftarrow\boldsymbol{a}_i-2(\boldsymbol{a}_i\cdot\boldsymbol{v})\boldsymbol{v}\,,$$

which requires $4m$ flops and essentially no additional storage. Similarly, computing $QH_{\boldsymbol{v}}$ leads to:

$$QH_{\boldsymbol{v}}=Q(I-2\boldsymbol{v}\boldsymbol{v}^t)=Q-2(Q\boldsymbol{v})\boldsymbol{v}^t\,,$$

and this can be done row by row, with $4m$ flops for updating each row of $Q$.

Finally, we look at the complexity of Algorithm 3.1. Note that the line 3 will be executed every outer loop, unless we smartly terminate the program when $p$ reaches $r=\min(m,n)$. But in any case, the computational cost associated with line 3 is bounded by $(2m-1)n\sim 2mn$ flops. The execution of the block from line 4 to line 10, however, will leads to an increase in the value of $p$; hence this

block can be run at most $r$ times, with the complexity bounded by $2r$ square root operations and the number of flops:

$$\sum_{p=1}^{r}[2+3+m-p+2+4(m-p+1)(n-i_p+1)]$$

where $i_p$ is the value $i$ such that the `if` block with $p$ is executed. Clearly $i_p \geq p$, and we obtain the estimate:

$$\begin{aligned}\sum_{p=1}^{r}[2+3+m-p+2+4(m-p+1)(n-i_p+1)] &\leq \sum_{p=1}^{r}[m-p+7+4(m-p+1)(n-p+1)] \\ &\sim 2mr(n-r)+2nr(m-r)+\frac{2}{3}r^3\,.\end{aligned}$$

Comparing Householder transformation and Givens rotation, the former requires only nearly two thirds of the computational cost of the latter; however, because each Householder transformation work on (almost) the entire column of $A$ simultaneously, it is less friendly to parallelization.

## References

[1] M. I. Gillespie and D. O. Olesky. Ordering givens rotations for sparse QR factorization. SIAM J. Matrix Anal. A., 16(3):1024–1041, 1995.