**Householder QR**

- **Householder reflectors** are matrices of the form
  \[ P = I - 2ww^T, \]
  where \( w \) is a unit vector (a vector of 2-norm unity)

Geometrically, \( Px \) represents a mirror image of \( x \) with respect to the hyperplane \( \text{span}\{w\}^\perp \).

**Problem 1.** Given a vector \( x \neq 0 \), find \( w \) such that
\[ (I - 2ww^T)x = \alpha e_1, \]
where \( \alpha \) is a (free) scalar.

Writing \( (I - \beta vv^T)x = \alpha e_1 \) yields
\[ \beta(v^T x) v = x - \alpha e_1. \]

- Desired \( w \) is a multiple of \( x - \alpha e_1 \), i.e., we can take:
  \[ v = x - \alpha e_1 \]
- To determine \( \alpha \) recall that
  \[ \| (I - 2ww^T)x \|_2 = \| x \|_2 \]
- As a result: \( |\alpha| = \| x \|_2 \), or \( \alpha = \pm \| x \|_2 \)
- Should verify that both signs work, i.e., that in both cases we indeed get \( Px = \alpha e_1 \) [exercise]

**A few simple properties:**

- For real \( w \): \( P \) is symmetric – It is also orthogonal \( (P^T P = I) \).
- In the complex case \( P = I - 2ww^H \) is Hermitian and unitary.
- \( P \) can be written as \( P = I - \beta vv^T \) with \( \beta = 2/\|v\|^2_2 \), where \( v \) is a multiple of \( w \). [storage: \( v \) and \( \beta \)]
- \( Px \) can be evaluated \( x - \beta(x^T v) v \) (op count?)
- Similarly: \( PA = A - vz^T \) where \( z^T = \beta \ast v^T \ast A \)

- **NOTE:** we work in \( \mathbb{R}^m \), so all vectors are of length \( m \), \( P \) is of size \( m \times m \), etc.

- Next: we will solve a problem that will provide the basic ingredient of the Householder QR factorization.

\[ \ldots \text{Show that} \ (I - \beta vv^T)x = \alpha e_1 \text{ when } v = x - \alpha e_1 \text{ and } \alpha = \pm \| x \|_2. \]

**Solution:** Equivalent to showing that
\[ x - (\beta x^T v)v = \alpha e_1 \text{ i.e., } x - \alpha e_1 = (\beta x^T v)v \]
but recall that \( v = x - \alpha e_1 \) so we need to show that
\[ \beta x^T v = 1 \text{ i.e., that } \frac{2}{\|x - \alpha e_1\|_2^2}(x^Tv) = 1 \]

- Denominator = \( \|x\|_2^2 + \alpha^2 - 2\alpha e_1^T x = 2(\|x\|_2^2 - \alpha e_1^T x) \)
- Numerator = \( 2x^Tv = 2x^T(x - \alpha e_1) = 2(\|x\|_2^2 - \alpha x^Te_1) \)

Numerator/ Denominator = 1. \( \blacksquare \)
Which sign is best? To reduce cancellation, the resulting $x - \alpha e_1$ should not be small. So, $\alpha = -\text{sign}(\xi_1)\|x\|_2$, where $\xi_1 = e_1^T x$

\[ v = x + \text{sign}(\xi_1)\|x\|_2 e_1 \] and $\beta = 2/\|v\|_2^2$

\[ v = \begin{pmatrix} \hat{\xi}_1 \\ \hat{\xi}_2 \\ \vdots \\ \hat{\xi}_{m-1} \\ \hat{\xi}_m \end{pmatrix} \quad \text{with} \quad \hat{\xi}_1 = \begin{cases} \xi_1 + \|x\|_2 & \text{if } \xi_1 > 0 \\ \xi_1 - \|x\|_2 & \text{if } \xi_1 \leq 0 \end{cases} \]

OK, but will yield a negative multiple of $e_1$ if $\xi_1 > 0$.

Alternative:

Define $\sigma = \sum_{i=2}^m \xi_i^2$.

Always set $\hat{\xi}_1 = \xi_1 - \|x\|_2$. Update OK when $\xi_1 \leq 0$

When $\xi_1 > 0$ compute $\hat{x}_1$ as

\[ \hat{x}_1 = \xi_1 - \|x\|_2 = \frac{\xi_1^2 - \|x\|_2^2}{\xi_1 + \|x\|_2} = \frac{-\sigma}{\xi_1 + \|x\|_2} \]

So:

\[ \hat{\xi}_1 = \begin{cases} -\sigma/\xi_1 + \|x\|_2 & \text{if } \xi_1 > 0 \\ \xi_1 - \|x\|_2 & \text{if } \xi_1 \leq 0 \end{cases} \]

It is customary to compute a vector $v$ such that $v_1 = 1$. So $v$ is scaled by its first component.

If $\sigma = 0$, we get $v = [1; x(2:m)]$ and $\beta = 0$.

Matlab function:

```matlab
function [v,bet] = house (x)
%% computes the householder vector for x
m = length(x);
v = [1 ; x(2:m)];
sigma = v(2:m)' * v(2:m);
if (sigma == 0)
    bet = 0;
else
    xnorm = sqrt(x(1)^2 + sigma) ;
    if (x(1) <= 0)
        v(1) = x(1) - xnorm;
    else
        v(1) = -sigma / (x(1) + xnorm) ;
    end
    bet = 2 / (1+sigma/v(1)^2);
    v = v/v(1) ;
end
```

Problem 2: Generalization.

Given an $m \times n$ matrix $X$, find $w_1, w_2, \ldots, w_n$ such that

\[ (I - 2w_nw_n^T) \cdots (I - 2w_2w_2^T)(I - 2w_1w_1^T)X = R \]

where $r_{ij} = 0$ for $i > j$

First step is easy: select $w_1$ so that the first column of $X$ becomes $\alpha e_1$

Second step: select $w_2$ so that $x_2$ has zeros below 2nd component.

etc.. After $k - 1$ steps: $X_k \equiv P_{k-1} \cdots P_1 X$ has the following shape:
To do: transform this matrix into one which is upper triangular up to the \( k \)-th column...

... while leaving the previous columns untouched.

\[
X_k = \begin{pmatrix}
  x_{11} & x_{12} & x_{13} & \cdots & \cdots & x_{1n} \\
  x_{21} & x_{22} & x_{23} & \cdots & \cdots & x_{2n} \\
  x_{31} & x_{32} & x_{33} & \cdots & \cdots & x_{3n} \\
  \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\
  x_{k1} & x_{k2} & x_{k3} & \cdots & \cdots & x_k^{n+1} \\
  x_{m1} & x_{m2} & x_{m3} & \cdots & \cdots & x_{mn}
\end{pmatrix}
\]

To leave the first \( k-1 \) columns unchanged \( w \) must have zeros in positions 1 through \( k-1 \).

\[
P_k = I - 2w_kw_k^T, \quad w_k = \frac{v}{\|v\|_2},
\]

where the vector \( v \) can be expressed as a Householder vector for a shorter vector using the matlab function house,

\[
v = \begin{pmatrix}
  0 \\
  \text{house}(X(k : m, k))
\end{pmatrix}
\]

The result is that work is done on the \( (k : m, k : n) \) submatrix.

**ALGORITHM : 1. Householder QR**

1. For \( k = 1 : n \) do
2. \([v, \beta] = \text{house}(X(k : m, k))\)
3. \(X(k : m, k : n) = (I - \beta vv^T)X(k : m, k : n)\)
4. If \((k < m)\)
5. \(X(k + 1 : m, k) = v(2 : m - k + 1)\)
6. end
7. end

In the end:

\[
X_n = P_nP_{n-1} \ldots P_1X = \text{upper triangular}
\]

Yields the factorization:

\[
X = QR
\]

where:

\[
Q = P_1P_2 \ldots P_n \text{ and } R = X_n
\]

**Example:**

Apply to system of vectors:

\[
X = [x_1, x_2, x_3] = \begin{pmatrix}
  1 & 1 & 1 \\
  1 & 1 & 0 \\
  1 & 0 & -1 \\
  1 & 0 & 4
\end{pmatrix}
\]

Answer:

\[
x_1 = \begin{pmatrix}
  1 \\
  1 \\
  1
\end{pmatrix}, \quad \|x_1\|_2 = 2, \quad v_1 = \begin{pmatrix}
  1 + 2 \\
  1 \\
  1
\end{pmatrix}, \quad w_1 = \frac{1}{2\sqrt{3}} \begin{pmatrix}
  1 + 2 \\
  1 \\
  1
\end{pmatrix}
\]
$P_1 = I - 2w_1w_1^T = \frac{1}{6} \begin{pmatrix} -3 & -3 & -3 & -3 \\ -3 & 5 & -1 & -1 \\ -3 & -1 & 5 & -1 \\ -3 & -1 & -1 & 5 \end{pmatrix},$

$P_1X = \begin{pmatrix} -2 & -1 & -2 \\ 0 & 1/3 & -1 \\ 0 & -2/3 & -2 \\ 0 & -2/3 & 3 \end{pmatrix},$ $\|\tilde{x}_2\|_2 = 1,$

$\tilde{x}_2 = \begin{pmatrix} 0 \\ 1/3 \\ -2/3 \\ -2/3 \end{pmatrix},$

$v_2 = \begin{pmatrix} 0 \\ 1/3 + 1 \\ -2/3 \\ -2/3 \end{pmatrix},$

$P_2 = I - \frac{2}{v_2v_2}v_2v_2^T = \frac{1}{3} \begin{pmatrix} 3 & 0 & 0 & 0 \\ 0 & -1 & 2 & 2 \\ 0 & 2 & 2 & -1 \\ 0 & 2 & -1 & 2 \end{pmatrix},$

$P_2P_1X = \begin{pmatrix} -2 & -1 & -2 \\ 0 & -1 & 1 \\ 0 & 0 & -3 \\ 0 & 0 & 2 \end{pmatrix}$ $\bar{x}_3 = \begin{pmatrix} 0 \\ 0 \\ -2 \\ 3 \end{pmatrix},$ $\|\bar{x}_3\|_2 = \sqrt{13},$

$v_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix},$

$X = P_1P_2P_3R,R_1.$

**Question:** How to obtain $X = Q_1R_1$ where $Q_1$ = same size as $X$ and $R_1$ is $n \times n$ (as in MGS)?

**MAJOR** difference with Gram-Schmidt: $Q$ is $m \times m$ and $R$ is $m \times n$ (same as $X$). The matrix $R$ has zeros below the $n$-th row. Note also: this factorization always exists.

Cost of Householder QR? Compare with Gram-Schmidt.
Answer: simply use the partitioning

\[ X = (Q_1 \ Q_2) \begin{pmatrix} R_1 \\ 0 \end{pmatrix} \rightarrow X = Q_1 R_1 \]

- Referred to as the “thin” QR factorization (or “economy-size QR” factorization in matlab)
- How to solve a least-squares problem \( Ax = b \) using the Householder factorization?
  - Answer: no need to compute \( Q_1 \). Just apply \( Q^T \) to \( b \).
  - This entails applying the successive Householder reflections to \( b \)

The rank-deficient case

- Result of Householder QR: \( Q_1 \) and \( R_1 \) such that \( Q_1 R_1 = X \). In the rank-deficient case, can have \( \text{span}\{Q_1\} \neq \text{span}\{X\} \) because \( R_1 \) may be singular.
- Remedy: Householder QR with column pivoting. Result will be:

\[
A \Pi = Q \begin{pmatrix} R_{11} & R_{12} \\ 0 & 0 \end{pmatrix}
\]

- \( R_{11} \) is nonsingular. So \( \text{rank}(X) = \text{size of } R_{11} = \text{rank}(Q_1) \) and \( Q_1 \) and \( X \) span the same subspace.
- \( \Pi \) permutes columns of \( X \).

Algorithm: At step \( k \), active matrix is \( X(k : m,k : n) \). Swap \( k \)-th column with column of largest 2-norm in \( X(k : m,k : n) \). If all the columns have zero norm, stop.

Practical Question: How to implement this ???

Suppose you know the norms of each column of \( X \) at the start. What happens to each of the norms of \( X(2 : m,j) \) for \( j = 2, \ldots, n \)? Generalize this to step \( k \) and obtain a procedure to inexpensively compute the desired norms at each step.
Properties of the QR factorization

Consider the 'thin' factorization $A = QR$, (size$(Q) = [m,n] = $ size $(A)$). Assume $r_{ii} > 0$, $i = 1,\ldots,n$

1. When $A$ is of full column rank this factorization exists and is unique.

2. It satisfies:
   $$\text{span}\{a_1, \cdots, a_k\} = \text{span}\{q_1, \cdots, q_k\}, \quad k = 1,\ldots,n$$

3. $R$ is identical with the Cholesky factor $G^T$ of $A^T A$.
   - When $A$ is rank-deficient and Householder with pivoting is used, then $\text{Ran}\{Q_1\} = \text{Ran}\{A\}$

Cost of Householder QR

Look at the algorithm: each step works in rectangle $X(k : m, k : n)$. Step $k$: twice $2(m - k + 1)(n - k + 1)$

$$T(n) = \sum_{k=1}^{n} 4(m - k + 1)(n - k + 1)$$

$$= 4 \sum_{k=1}^{n} [(m - n) + (n - k + 1)](n - k + 1)$$

$$\approx (m - n) \cdot 2n^2 + 4n^3/3$$

$$= 2mn^2 - 2/3 n^3$$

Givens Rotations

Matrices of the form

$$G(i, k, \theta) = \begin{pmatrix}
1 & \cdots & 0 & \cdots & 0 & 0 \\
\vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\
0 & \cdots & c & \cdots & s & \cdots & 0 \\
\vdots & \ddots & \vdots & \ddots & \ddots & \ddots & \ddots \\
0 & \cdots & -s & \cdots & c & \cdots & 0 \\
\vdots & \ddots & \vdots & \ddots & \ddots & \ddots & \ddots \\
0 & \cdots & 0 & \cdots & \cdots & \cdots & 1
\end{pmatrix}$$

with $c = \cos \theta$ and $s = \sin \theta$

represents a rotation in the span of $e_i$ and $e_k$.

Main idea of Givens rotations consider $y = Gx$ then

$$y_i = c \cdot x_i + s \cdot x_k$$
$$y_k = -s \cdot x_i + c \cdot x_k$$
$$y_j = x_j \quad \text{for} \quad j \neq i, k$$

Can make $y_k = 0$ by selecting

$$s = x_k/t; \quad c = x_i/t; \quad t = \sqrt{x_i^2 + x_k^2}$$

This is used to introduce zeros in the first column of a matrix $A$ (for example $G(m - 1, m), G(m - 2, m - 1)$ etc..$G(1,2)$).

See text for details.