Householder Transformation-based Algorithm

(See Stewart 1973) **Definition:** An elementary reflector is a matrix

\[ H = I - 2uu^T \]

where \( u^T u = 1 \). It is easily shown that

- \( H \) is symmetric (\( H^T = H \))
- \( H \) is orthogonal (\( H^T H = I \))
- \( H \) is involutory (\( H^2 = I \))
We can introduce 0's in a fashion similar to that used for \( LU \) factorization via Gauss transforms.

Let \( x \in \mathbb{R}^n, \gamma = \pm \|x\|_2 \), and assume \( x \neq -\gamma e_1 \). Define

\[
\begin{align*}
  u &= x + \gamma e_1 \\
  \pi &= \frac{1}{2} \|u\|_2^2 \\
  H &= I - \pi^{-1} uu^T
\end{align*}
\]

We then have

\[ Hx = -\gamma e_1 \]

This introduces 0s into all but the first element of the matrix and respects the properties of orthogonal transformations, in particular, it does not change the 2-norm of the vector.
We must specify how to choose the sign to finalize the algorithm.

\[
\begin{align*}
\gamma &= \text{sign}(\xi_1)||x||_2 \\
\xi_1 &\leftarrow \nu_1 = \xi_1 + \gamma \\
\pi &= \gamma \nu_1
\end{align*}
\]

Here we assume \( x^T = (\xi_1, \ldots, \xi_n) \), \( u^T = (\nu_1, \ldots, \nu_n) \) and that we overwrite the elements of \( x \) that become 0 with the elements of \( u \). This implies that we need an extra scalar per Householder transform to store \( \pi \). So if we apply a series of \( k \) transformations we need an extra vector of size \( k \). This is different from Gauss transformations.
This can be used to introduce 0s in positions $i+1$ to $n$ while altering position $i$ and leaving positions 1 to $i-1$ untouched by simply constraining $u$ to have 0s in positions 1 to $i-1$. (Exactly as we did for Gauss transforms.

Applying $H$ to a vector is simple as well. We assume $v \neq x$ where $x$ defined $u$ (we already know what $UX$ is and therefore need not compute it.)

\[
\begin{align*}
a &= Hv \\
   &= (I - \pi^{-1}uu^T)v \\
   &= v - \pi^{-1}uu^Tv \\
   &= v - \pi^{-1}(u^Tv)u \\
   &= v - (\pi^{-1}(u^Tv))u
\end{align*}
\]

So we see that it is an inner product (with the result scaled with $\pi^{-1}$), followed by a vector triad.

If this is applied to a matrix we have a matrix vector product followed by a rank-1 update.
So our transformation approach becomes clear:

Produce $H$ such that $HA = B$ where $B \in \mathbb{R}^{n \times k}$ with structure

$$B = \begin{pmatrix} R \\ 0 \end{pmatrix}$$

and $R \in \mathbb{R}^{k \times k}$ is an upper triangular matrix with positive diagonal elements.

$H$ is formed by the product $H_k H_{k-1} \cdots H_1$ (although it is never computed explicitly, the individual $u_i$ vectors and $\pi_i$ scalars are stored).

Initially

$$A^{(0)} = A$$

$$H_1 A^{(0)} = \begin{pmatrix} \rho_{1,1} & r_i^T \\ 0 & A_1 \end{pmatrix} = A^{(1)}$$

In general

$$H_i A^{(i-1)} = H_i \begin{pmatrix} R_{i-1,i-1} & R_{i-1}^T \\ 0 & A_{i-1} \end{pmatrix}$$

$$= \begin{pmatrix} R_{i,i} & R_i^T \\ 0 & A_i \end{pmatrix} = A^{(i)}$$

Where $R_{i,i} \in \mathbb{R}^{i \times i}$ is upper triangular with positive diagonal elements and $R_i^T \in \mathbb{R}^{i \times k-i}$. 
How do we use this to solve least squares?

\[ Ax = b \]

where \( A \in \mathbb{R}^{m \times k} \), \( b \in \mathbb{R}^m \).

Assume we have \( H \in \mathbb{R}^{m \times m} \) such that

\[ HA = \begin{pmatrix} R \\ 0 \end{pmatrix} \]

and \( R \in \mathbb{R}^{k \times k} \) is an upper triangular matrix with positive diagonal elements. Let

\[ Hb = \begin{pmatrix} c \\ d \end{pmatrix} \]

It follows that

\[ H(b - Ax) = \begin{pmatrix} c \\ d \end{pmatrix} - \begin{pmatrix} R \\ 0 \end{pmatrix} x \]
But since $H$ is orthogonal we have

$$\|r\|_2^2 = \|Hr\|_2^2 = \left\| \begin{pmatrix} c \\ d \end{pmatrix} - \begin{pmatrix} R \\ 0 \end{pmatrix} \right\|_2^2 = \left\| \begin{pmatrix} c - Rx \\ d \end{pmatrix} \right\|_2^2 = \|c - Rx\|_2^2 + \|d\|_2^2$$

This is minimized when

$$Rx = c$$

and we have at the minimizer $x = x_{min}$,

$$\|r\|_2^2 = \|d\|_2^2$$

$$r = H^T \begin{pmatrix} 0 \\ d \end{pmatrix}$$

$H$ is computed via $k$ Householder reflectors and store in factored form in the positions of the elements of $A$ that we zeroed out and an extra vector of $k$ elements for the $\pi_i$. Therefore additional righthand side vectors can be handled.

The algorithms requires $2mk^2 - (2/3)k^3$ operations.
Perturbation Theorem for Least Squares

**THEOREM:** (Stewart 1973) Let $A \in \mathbb{R}^{m \times k}$ have linearly independent columns. Let $E \in \mathbb{R}^{m \times k}$ and $b \in \mathbb{R}^m$. Let $E_1$ and $b_1$ be projections of $E$ and $b$ onto $\mathcal{R}(A)$ and let $E_2$ and $b_2$ be projections of $E$ and $b$ onto $\mathcal{R}^\perp(A)$. If

$$\|A^\dagger\|_2\|E_1\|_2 < \frac{1}{2}$$

then the columns of $A + E$ are linearly independent and if

$$x = A^\dagger b$$
$$\bar{x} = (A + E)^\dagger b$$

then we have

$$\frac{\|x - \bar{x}\|_2}{\|x\|_2} \leq 2\kappa \frac{\|E_1\|_2}{\|A\|_2} + 4\kappa^2 \frac{\|E_2\|_2\|b_2\|_2}{\|A\|_2 \|b_1\|_2} + 8\kappa^3 \frac{\|E_2\|_2}{\|A\|_2}$$

where $kappa = \|A\|_2\|A^\dagger\|_2$ and it can be shown that $kappa^2 = \kappa(A^TA)$. 
• The third term can usually be ignored.

• The first term is like our bound on solving systems and says that the part of the relative error that is in \( \mathcal{R}(A) \) is expanded by the condition number.

• The second term has a larger expansion factor, \( \kappa^2 \), but is weighted by the size of the projection of \( b \) that is perpendicular to \( \mathcal{R}(A) \).
• The Householder method is very stable when computing the transformed problem (and when used to compute projections or an orthogonal basis).

• CGS and MGS are unreliable as producers of an orthogonal basis except for well conditioned problems unless reorthogonalization is used which doubles the operation count.

• MGS is as stable as Householder for the least squares problem even though the computed $Q$ may not be orthogonal.

• The normal equations can have difficulties from the backward error point of view (the perturbation that one must use is proportional to the condition number). However, for moderately well-conditioned problems in practice it is often accurate enough.

• The smallest operation count for the least squares problem is for the normal equations (by a factor of 2).

• In general, for black box library routines the Householder (or other orthogonal transformation-based triangularization) tends to be used.

• For less well-conditioned problems, column pivoting can be added to Householder, MGS and CGS.

• Column pivoting is also needed when you are attempting to perform a rank-revealing $QR$ factorization.