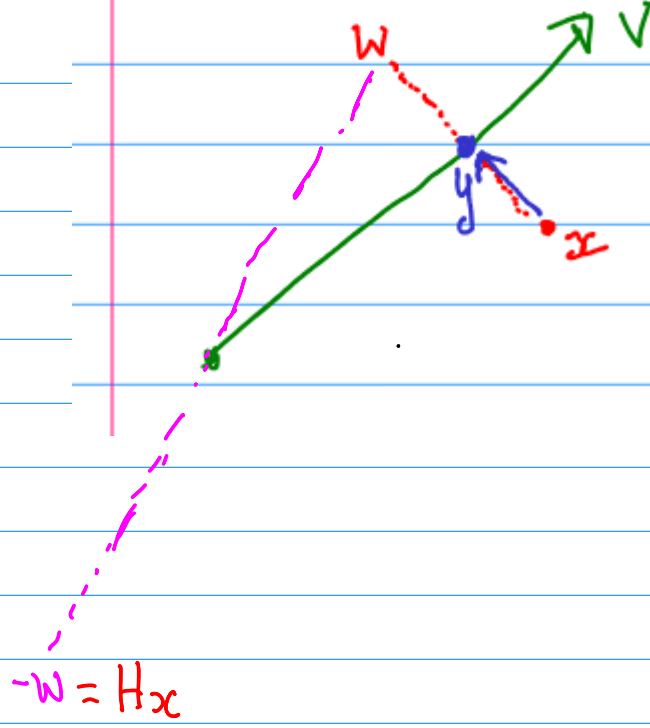

Define

$$H = I - 2vv^T$$

$$Hx = x - 2vv^Tx = x - 2v(v \cdot x)$$

We want to use this orthogonal matrix to factor $A = QR$

$$Q^TA = R$$

Look for an H such that

$$HA = \begin{bmatrix} c & \\ 0 & \\ \vdots & ? \\ 0 & \end{bmatrix}$$

$$A = \left[a_1 \mid a_2 \mid \cdots \mid a_n \right]$$

$H = I - 2vv^T$ where v is a unit vector

(assume a_1 is not already of the form ce_1).

$$HA = \left[Ha_1 \mid Ha_2 \mid \cdots \mid Ha_n \right] = \left[\begin{matrix} c \\ 0 \\ \vdots \\ 0 \end{matrix} \;\middle|\; Ha_2 \;\middle|\; \cdots \;\middle|\; Ha_n \right]$$

Solve for the unit vector v so $Ha_1 = ce_1$ Here $e_1 = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$

$$(I - 2vv^T)a_1 = ce_1$$

$$a_1 - 2v(v \cdot a_1) = ce_1$$

try this

$$2v\underbrace{(v \cdot a_1)}_{\text{scalar}} = a_1 - ce_1$$

$$v = \frac{a_1 - ce_1}{2v \cdot a_1}$$

Note that v is a unit vector and the denominator here is actually a normalization factor to make a unit vector

$$v = \frac{a_1 - ce_1}{\|a_1 - ce_1\|}$$

Now plug in v to find c..

$$a_1 - 2v(v \cdot a_1) = ce_1$$

$$a_1 - 2\frac{a_1 - ce_1}{\|a_1 - ce_1\|}\left(\frac{a_1 - ce_1}{\|a_1 - ce_1\|}\cdot a_1\right) = ce_1$$

$$a_1 - ce_1 - 2\frac{a_1 - ce_1}{\|a_1 - ce_1\|}\left(\frac{a_1 - ce_1}{\|a_1 - ce_1\|}\cdot a_1\right) = 0$$

$$\frac{(a_1 - ce_1)\|a_1 - ce_1\|^2}{\|a_1 - ce_1\|^2} - 2\frac{a_1 - ce_1}{\|a_1 - ce_1\|^2}\left(a_1\cdot a_1 - ce_1\cdot a_1\right) = 0$$

$$\frac{(a_1 - ce_1)\|a_1 - ce_1\|^2 - 2(a_1 - ce_1)\left(\|a_1\|^2 - ce_1\cdot a_1\right)}{\|a_1 - ce_1\|^2} = 0$$

$$(a_1 - ce_1)\underbrace{\|a_1 - ce_1\|^2}_{\text{expand}} - 2(a_1 - ce_1)\left(\|a_1\|^2 - ce_1\cdot a_1\right) = 0$$

$$\begin{aligned}\|a_1 - ce_1\|^2 &= (a_1 - ce_1)\cdot(a_1 - ce_1)\\ &= a_1\cdot a_1 - 2ce_1\cdot a_1 + ce_1\cdot ce_1\\ &= \|a_1\|^2 - 2ce_1\cdot a_1 + c^2\end{aligned}$$

$$(a_1 - ce_1)\left(\cancel{\|a_1\|^2} - \cancel{2ce_1\cdot a_1} + c^2\right) - (a_1 - ce_1)\left(\cancel{2}\|a_1\|^2 - \cancel{2ce_1\cdot a_1}\right) = 0$$

Therefore

$$\underbrace{(a_1 - ce_1)}_{\text{vector not zero}}\underbrace{(c^2 - \|a_1\|^2)}_{\text{must be zero}} = 0$$

Therefore $c = \pm\|a_1\|$

recall $$v = \frac{a_1 - ce_1}{\underbrace{\|a_1 - ce_1\|}_{\text{appears in the denominator}}}$$

To control rounding error want the divisor in the denominator as large as possible so

If $\|a_1 - \|a_1\|e_1\| < \|a_1 + \|a_1\|e_1\|$ then

choose $c = -\|a_1\|$ otherwise choose $c = \|a_1\|$.

Check how this works...

```
julia> A=[1 2 3; 4 5 6; 7 8 9]
3×3 Matrix{Int64}:
 1  2  3
 4  5  6
 7  8  9
```

```
julia> a1=A[:,1]
3-element Vector{Int64}:
 1
 4
 7
```

$a_1 = \begin{bmatrix} 1 \\ 4 \\ 7 \end{bmatrix}$

```
julia> e1=[1,0,0]
3-element Vector{Int64}:
 1
 0
 0

julia> using LinearAlgebra
```

If $\| a_1 - \|a_1\| e_1 \| < \| a_1 + \|a_1\| e_1 \|$ then choose $\boxed{c = -\|a_1\|}$ otherwise choose $c = \|a_1\|$.

```
julia> norm(a1-norm(a1)*e1)
10.758806773556634

julia> norm(a1+norm(a1)*e1)
12.175716685652304
```

$$v = \frac{a_1 - c e_1}{\|a_1 - c e_1\|}$$

```
julia> v=(a1-c*e1)/norm(a1-c*e1)
3-element Vector{Float64}:
 0.7493635602894408
 0.3285227558484129
 0.5749148227347227
```

$H = I - 2vv^T$

```
julia> H=I-2*v*v'
3×3 Matrix{Float64}:
 -0.123091  -0.492366   -0.86164
 -0.492366   0.784146   -0.377745
 -0.86164   -0.377745    0.338946
```

```
julia> H*A
3×3 Matrix{Float64}:
 -8.12404    -9.60114     -11.0782
  8.88178e-16  -0.0859656   -0.171931
  8.88178e-16  -0.90044     -1.80088
```

$= [c e_1 | Ha_2 | Ha_3]$

These are zero ...

work on this submatrix A_2

```
julia> A2=(H*A)[2:3,2:3]
2×2 Matrix{Float64}:
 -0.0859656  -0.171931
 -0.90044    -1.80088
```

```
julia> e1=[1,0]
2-element Vector{Int64}:
 1
 0
```

```
julia> a1=A2[:,1]
2-element Vector{Float64}:
 -0.0859655700236277
 -0.9004397475413497

julia> norm(a1-norm(a1)*e1)
1.3386116703491497

julia> norm(a1+norm(a1)*e1)
1.216900188483972
```

← bigger

If $\|a_1 - \|a_1\| e_1\| < \|a_1 + \|a_1\| e_1\|$ then choose $c = -\|a_1\|$ otherwise choose $c = \|a_1\|$.

$$v = \frac{a_1 - c e_1}{\|a_1 - c e_1\|}$$

denominator big.

```
julia> c=norm(a1)
0.9045340337332901

julia> v2=(a1-c*e1)/norm(a1-c*e1)
2-element Vector{Float64}:
 -0.7399454417565073
 -0.6726668887523506
```

```
julia> H2=I-2*v2*v2'
2×2 Matrix{Float64}:
 -0.0950385  -0.995474
 -0.995474    0.0950385

julia> v2x=[0,v2...]
3-element Vector{Float64}:
  0.0
 -0.7399454417565073
 -0.6726668887523506
```

want a 3×3 matrix so if can mult the original A.

```
julia> H2=I-2*v2x*v2x'
3×3 Matrix{Float64}:
 1.0   0.0         0.0
 0.0  -0.0950385  -0.995474
 0.0  -0.995474    0.0950385
```

```
julia> H2*H*A
3×3 Matrix{Float64}:
 -8.12404       -9.60114       -11.0782
 -8.88178e-16    0.904534        1.80907
 -8.88178e-16   -4.44089e-16     8.88178e-16
```

numerically zero

by luck was zero as well

because A wasn't invertible to start with.

$$H_2 H_1 A = R$$

$$\cancel{H_2^T} \cancel{H_2} H_1 A = H_2^T R$$

$$\cancel{H_1^T} \cancel{H_1} A = H_1^T H_2^T R$$

$$A = H_1^T H_2^T R = H_1 H_2 R = \underbrace{(H_1 H_2)}_{Q} R$$

Since the reflectors symmetric.

```
julia> Q=H*H2
3×3 Matrix{Float64}:
 -0.123091   0.904534   0.408248
 -0.492366   0.301511  -0.816497
 -0.86164   -0.301511   0.408248
```

$A = QR$

```
julia> R=H2*H*A
3×3 Matrix{Float64}:
 -8.12404        -9.60114       -11.0782
 -8.88178e-16     0.904534        1.80907
 -8.88178e-16  -4.44089e-16      8.88178e-16
```

```
julia> Q*R
3×3 Matrix{Float64}:
 1.0  2.0  3.0
 4.0  5.0  6.0
 7.0  8.0  9.0
```

it worked ...

The advantage of this method is one can choose $c = \pm \|a_1\|$ so the denominator in

$$v = \frac{a_1 - c e_1}{\|a_1 - c e_1\|}$$

is large to controll rounding errors...