where $U$ and $V$ have orthonormal columns, and $\Sigma$ is a diagonal mat	as $A=U\Sigma V^*$ at the columns in $U$ and $V$ are organized to correspond to a list of singular values $\sigma_U$ so that $\sigma_{U} \geq \sigma_{\Omega} \geq \dots \geq \sigma_{N}$	$\tau > 0$ We also
	atrix. By convention, the columns in $U$ and $V$ are organized to correspond to a list of singular values $\sigma_{ii}$ so that $\sigma_{11} \geq \sigma_{22} \geq \cdots \geq \sigma_{n}$ be first $j$ columns of $A$ span the same space as the first $j$ columns of $U$ .  In set of orthonormal vectors $\{\mathbf{q}_1,\mathbf{q}_2,\ldots\mathbf{q}_n\}$ such that	$\sigma_{rr}>0$ . We also
where $\langle \ldots  angle$ denotes the space spanned of the enclosed vectors. V	$\langle {f a}_1,{f a}_2,\dots,{f a}_j angle = \langle {f q}_1,{f q}_2,\dots,{f q}_j angle, \qquad j=1,2,\dots,n$ We can then write ${f a}_1=r_{11}{f q}_1$	
	$egin{align} \mathbf{a}_2 &= r_{12}\mathbf{q}_1 + r_{22}\mathbf{q}_2 \ & \dots \ \mathbf{a}_n &= r_{1n}\mathbf{q}_1 + r_{2n}\mathbf{q}_2 + \dots + r_{nn}\mathbf{q}_n \ \end{pmatrix}$	
Question  How can we write this in matrix form?		
The QR decomposition  The matrix form suggested by the above set of equations is the QR.	R decompostion, given by	
The matrix form suggested by the above set of equations is the $QR$ where $Q=[{f q}_1,{f q}_2,\dots{f q}_n]$ is $m imes n$ , and $R$ is an upper triangular	A=QR	
As with the SVD, we also have a $full\ QR$ decomposition in which $Q$ Gram-Schmidt orthogonalization	Q is a square $m imes m$ matrix, and $R$ is $m imes n$ . For this purposes here, however, the $QR$ decomposition will refer to a <i>reduced</i> decom	nposition.
How can we compute such vectors $\mathbf{q}_j$ and the entries of $R$ ? In our	r first approach, we will use a classical algorithm called the Gram-Schmidt algorithm.	
Step 1: $j=1$ We know we need $\langle {f a}_1  angle = \langle {f q}_1  angle$ . Using our above formulation, we h	have	
Question	$\mathbf{a}_1 = r_{11}\mathbf{q}_1$	
What is $r_{11}$ ? What is $\mathbf{q}_1$ ?	$r_{11} = \ \mathbf{a}_1\  \qquad \mathbf{q}_1 = rac{\mathbf{a}_1}{r_{11}}$	
Note : $\mathbf{q}_1^*\mathbf{a}_1=r_{11}$ .		
$Step 2 \colon j = 2$	$\mathbf{a}_2=r_{12}\mathbf{q}_1+r_{22}\mathbf{q}_2$	
Question How do we find $r_{12}$ , $r_{22}$ and $\mathbf{q}_2$ ? From the above, we can write		
where $r_{12}$ is the projection of ${f a}_2$ onto ${f q}_1.$ We then subtract out this	${f q}_1^*{f a}_2=r_{12}$ is component of ${f a}_2$ in the direction of ${f q}_1$ to define an intermediate vector	
or	$\mathbf{v}=\mathbf{a}_2-r_{12}\mathbf{q}_1\equiv r_{22}\mathbf{q}_2$ $\mathbf{q}_2=rac{\mathbf{v}}{r_{22}}$	
where $r_{22} = \ \mathbf{a}_2 - r_{12}\mathbf{q}_1\ .$	$q_2-r_{22}$	
${\color{red} \textbf{Step 2}: j = k} \\$		
	$\mathbf{v} = \mathbf{a}_k - \sum_{j=1}^{k-1} r_{jk} \mathbf{q}_j$	
Then	$\mathbf{q}_k = rac{\mathbf{v}}{r_{kk}}$	
where $r_{jk} = \mathbf{q}_j^* \mathbf{a}_k$ for $j < k$ , and	$r_{kk} = \ \mathbf{v}\  = \left\ \mathbf{a}_k - \sum_{j=1}^{k-1} r_{jk} \mathbf{q}_j  ight\ $	
<pre>def display_mat(msg,A):     print(msg)     display(A)     print("")</pre>		
Gram-Schmidt algorithm		
Here is an outline of the classical Gram-Schmidt algorithm : $\bullet \ \ For \ j=1,2,\dots n$		
Set ${f v}={f a}_j$ , the $j^{th}$ column of $A$ Orthogonalize ${f v}$ against previous ${f q}_i, i=0,1,\ldots,j-1$ .		
Set $\mathbf{q}_j$ equal to normalized vector $\mathbf{v}$ . The code for the Gram-Schmidt algorithm is below.		
<pre># Classical Gram-Schmidt algorithm for orthogonalizing from numpy.linalg import norm  def gram_schmidt_classic(A):</pre>	a set of column vectors.	
<pre>m,n = A.shape assert n &lt;= m, 'We must have n &lt;= m' R = np.zeros((n,n)) Q = np.zeros((m,n)) tol = 1e-12 for j in range(n):</pre>		
<pre># Loop over columns of A; aj = A[:,j:j+1] v = aj # Orthogonalize against previous qi vectors, i for i in range(j):</pre>	$= 0, 1, 2, 3, \ldots, j-1$	
<pre>qi = Q[:,i:i+1]     R[i,j] = qi.T@aj # m ops     v = v - R[i,j]*qi  R[j,j] = norm(v,2) assert R[j,j] &gt; tol, "Columns are not linearly</pre>	<pre>independent"</pre>	
<pre>Q[:,j:j+1] = v/R[j,j] return Q,R  Before continuing, we set up a few matrices that we can use for example.</pre>	xamples. We'll put them in a function so that we can easily access different choices.	
<pre>def matrix_example(id):     # 3 x 1 example     A1 = np.array(np.mat('1; 3; 5'),dtype=float)</pre>		
<pre># A 3 x 2 example A2 = np.array(np.mat('1,2; 3,4; 5,-1'),dtype=float) # A 3x3 example.</pre>		
A3 = np.array(np.mat('1,2,-1; 3,4,4; 5,6,5'),dtype= A_mat = (A1,A2,A3)  assert id > 0 and id < 4, "Assert id must be 1,2,3.		
<pre>return A_mat[id-1]  A = matrix_example(3)  diamler mat('A = ' A)</pre>		
<pre>display_mat('A = ',A)  Q,R = gram_schmidt_classic(A)  display_mat('Q = ', Q) display_mat('P = ', P)</pre>		
<pre>display_mat('R = ',R) display_mat('QR = ',Q@R) display_mat('Q^*Q',Q.T@Q) A =</pre>		
array([[ 1., 2., -1.],		
<pre>array([[ 1., 2., -1.],</pre>		
<pre>array([[ 1., 2., -1.],</pre>		
<pre>array([[ 1., 2., -1.],</pre>	-16],	
$\begin{array}{llllllllllllllllllllllllllllllllllll$	-16], +00]])	
$\begin{array}{l} \operatorname{array}([[\ 1.,\ 2.,\ -1.],\ [\ 3.,\ 4.,\ 4.],\ [\ 5.,\ 6.,\ 5.]]) \\ Q = \\ \operatorname{array}([[\ 0.16903085,\ 0.89708523,\ -0.40824829],\ [\ 0.50709255,\ 0.27602622,\ 0.81649658],\ [\ 0.84515425,\ -0.34503278,\ -0.40824829]]) \\ R = \\ \operatorname{array}([[\ 5.91607978,\ 7.43735744,\ 6.08511063],\ [\ 0.\ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ $	-16], +00]])	
$\begin{array}{llllllllllllllllllllllllllllllllllll$	-16], +00]])	
$\begin{array}{lll} \operatorname{array}([[\ 1.,\ 2.,\ -1.], & [\ 3.,\ 4.,\ 4.], & [\ 5.,\ 6.,\ 5.]]) \\ Q = & \operatorname{array}([[\ 0.16903085,\ 0.89708523,\ -0.40824829], & [\ 0.50709255,\ 0.27602622,\ 0.81649658], & [\ 0.84515425,\ -0.34503278,\ -0.40824829]]) \\ R = & \operatorname{array}([[\ 5.91607978,\ 7.43735744,\ 6.08511063], & [\ 0. & ,\ 0.82807867,\ -1.51814423], & [\ 0. & ,\ 0. & ,\ 1.63299316]]) \\ QR = & \operatorname{array}([[\ 1.,\ 2.,\ -1.], & [\ 3.,\ 4.,\ 4.], & [\ 5.,\ 6.,\ 5.]]) \\ Q^* \times Q \\ & \operatorname{array}([[\ 1.000000000e+00,\ 2.77555756e-16,\ 0.00000000e+0,\ -9.15933995e-[\ 0.00000000e+00,\ -9.15933995e-16,\ 1.00000000e+0 \\ & [\ 0.00000000e+00,\ -9.15933995e-16,\ 1.00000000e+1 \\ & \text{Using QR to solve } A\mathbf{x} = \mathbf{b}. \\ & \text{Suppose we have a } QR \text{ factorization of a non-singular matrix } A. \text{ H} \\ & \text{Answer} \\ & 1.\ A = QR \\ & 2.\ QR\mathbf{x} = \mathbf{b} \\ & 3.\ R\mathbf{x} = Q^*\mathbf{b} \\ & 4. \text{ Use back substitution to solve for } \mathbf{x}. \\ \end{array}$	How can we use this factorization to solve $A\mathbf{x}=\mathbf{b}$ ?	
array([[ 1., 2., -1.],	How can we use this factorization to solve $A\mathbf{x}=\mathbf{b}$ ?	rithm.
$\begin{array}{llllllllllllllllllllllllllllllllllll$	How can we use this factorization to solve $A\mathbf{x}=\mathbf{b}$ ? $\mathbf{e}$ stable)  The very sensitive to small changes in the input vectors, e.g. columns of $A$ . For this reason, we used a "modified Gram-Schmidt" algored of vectors $[\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n]$ by successivly subtracting out projections on the previously found orthonormal set $[\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_j]$ .	rithm.
$\begin{array}{llllllllllllllllllllllllllllllllllll$	How can we use this factorization to solve $A\mathbf{x} = \mathbf{b}$ ?  How can we use this factorization to solve $A\mathbf{x} = \mathbf{b}$ ?  The very sensitive to small changes in the input vectors, e.g. columns of $A$ . For this reason, we used a "modified Gram-Schmidt" algored of vectors $[\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n]$ by successively subtracting out projections on the previously found orthonormal set $[\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_j]$ . $\mathbf{q}_1 = \frac{\mathbf{a}_1}{r_{11}}$ $\mathbf{q}_2 = \frac{\mathbf{a}_2 - (\mathbf{q}_1^* \mathbf{a}_2)\mathbf{q}_1}{r_{22}}$ $\mathbf{q}_3 = \frac{\mathbf{a}_3 - (\mathbf{q}_1^* \mathbf{a}_3)\mathbf{q}_1 - (\mathbf{q}_2^* \mathbf{a}_3)\mathbf{q}_2}{\mathbf{q}_3 - (\mathbf{q}_1^* \mathbf{a}_3)\mathbf{q}_1 - (\mathbf{q}_2^* \mathbf{a}_3)\mathbf{q}_2}$	rithm.
$\begin{array}{llllllllllllllllllllllllllllllllllll$	How can we use this factorization to solve $A\mathbf{x}=\mathbf{b}$ ?  Be stable  The very sensitive to small changes in the input vectors, e.g. columns of $A$ . For this reason, we used a "modified Gram-Schmidt" algor of vectors $[\mathbf{a}_1,\mathbf{a}_2,\ldots,\mathbf{a}_n]$ by successivly subtracting out projections on the previously found orthonormal set $[\mathbf{q}_1,\mathbf{q}_2,\ldots,\mathbf{q}_j]$ . $\mathbf{q}_1 = \frac{\mathbf{a}_1}{r_{11}}$ $\mathbf{q}_2 = \frac{\mathbf{a}_2}{r_{22}} \frac{\mathbf{q}_1}{r_{22}}$ $\mathbf{q}_3 = \frac{\mathbf{a}_3 - (\mathbf{q}_1^*\mathbf{a}_3)\mathbf{q}_1 - (\mathbf{q}_2^*\mathbf{a}_3)\mathbf{q}_2}{r_{33}}$ $\vdots$	rithm.
$\begin{array}{lll} \operatorname{array}([\{\ 1.,\ 2.,\ -1.\}, \\ [\ 3.,\ 4.,\ 4.], \\ [\ 5.,\ 6.,\ 5.]]) \end{array}) \\ \mathbb{Q} = \\ \operatorname{array}([\{\ 0.16903085,\ 0.89708523,\ -0.40824829], \\ [\ 0.50709255,\ 0.27602622,\ 0.81649658], \\ [\ 0.84515425,\ -0.34503278,\ -0.40824829]]) \\ \mathbb{R} = \\ \operatorname{array}([\{\ 5.91607978,\ 7.43735744,\ 6.08511063], \\ [\ 0.\ \ ,\ 0.82807867,\ -1.51814423], \\ [\ 0.\ \ ,\ 0.\ \ ,\ 1.63299316]]) \\ \mathbb{Q} = \\ \operatorname{array}([\{\ 1.,\ 2.,\ -1.], \\ [\ 3.,\ 4.,\ 4.], \\ [\ 5.,\ 6.,\ 5.]]) \\ \mathbb{Q}^*\mathbb{Q} \\ \operatorname{array}([\{\ 1.000000000e+00,\ 2.77555756e-16,\ 0.00000000e+ \\ [\ 2.77555756e-16,\ 1.00000000e+00,\ -9.15933995e-1 \\ [\ 0.000000000e+00,\ -9.15933995e-16,\ 1.00000000e+ \\ \mathbb{Q} = \\ \operatorname{array}([\{\ 1.00000000e+00,\ -9.15933995e-16,\ 1.00000000e+ \\ \mathbb{Q} = \\ \operatorname{array}([\ 1.000000000e+00,\ -9.15933995e-16,\ 1.00000000e+ \\ \mathbb{Q} = \\ \operatorname{array}([\ 1.00000000e+00,\ -9.15933995e-16,\ 1.00000000e+ \\ \mathbb{Q} = \\ \operatorname{array}([\ 1.000000000e+00,\ -9.15933995e-16,\ 1.00000000e+ \\ \mathbb{Q} = \\ \operatorname{array}([\ 1.00000000e+00,\ -9.15933995e-16,\ 1.00000000e+ \\ \mathbb{Q} = \\ \operatorname{array}([\ 1.000000000e+00,\ -9.15933995e-16,\ 1.00000000e+ \\ \mathbb{Q} = \\ \operatorname{array}([\ 1.000000000e+00,\ -9.15933995e-16,\ 1.00000000e+ \\ \mathbb{Q} = \\ \operatorname{array}([\ 1.00000000000e+00,\ -9.15933995e-16,\ 1.000000000e+ \\ \mathbb{Q} = \\ \operatorname{array}([\ 1.000000000000e+00,\ -9.15933995e-16,\ 1.00000000e+ \\ \mathbb{Q} = \\ \operatorname{array}([\ 1.000000000000e+00$	How can we use this factorization to solve $A\mathbf{x}=\mathbf{b}$ ?  Be stable)  The very sensitive to small changes in the input vectors, e.g. columns of $A$ . For this reason, we used a "modified Gram-Schmidt" algor of vectors $[\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n]$ by successivity subtracting out projections on the previously found orthonormal set $[\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_j]$ . $\mathbf{q}_1 = \frac{\mathbf{a}_1}{r_{11}}$ $\mathbf{q}_2 = \frac{\mathbf{a}_2 - (\mathbf{q}_1^* \mathbf{a}_2) \mathbf{q}_1}{r_{22}}$ $\mathbf{q}_3 = \frac{\mathbf{a}_3 - (\mathbf{q}_1^* \mathbf{a}_3) \mathbf{q}_1 - (\mathbf{q}_2^* \mathbf{a}_3) \mathbf{q}_2}{r_{33}}$ $\vdots$	rithm.
$\begin{array}{lll} \operatorname{array}([[\ 1.,\ 2.,\ -1.], & [\ 3.,\ 4.,\ 4.], & [\ 5.,\ 6.,\ 5.]]) \\ \operatorname{Q} = & \operatorname{array}([[\ 0.16903085,\ 0.89708523,\ -0.40824829], & [\ 0.50709255,\ 0.27602622,\ 0.81649658], & [\ 0.84515425,\ -0.34503278,\ -0.40824829]]) \\ \operatorname{R} = & \operatorname{array}([[\ 5.91607978,\ 7.43735744,\ 6.08511063], & [\ 0. &\ ,\ 0.82807867,\ -1.51814423], & [\ 0. &\ ,\ 0. &\ ,\ 0.63299316]]) \\ \operatorname{QR} = & \operatorname{array}([[\ 1.,\ 2.,\ -1.], & [\ 3.,\ 4.,\ 4.], & [\ 5.,\ 6.,\ 5.]]) \\ \operatorname{Q^**Q} = \operatorname{array}([[\ 1.000000000e+00,\ 2.77555756e-16,\ 0.00000000e+ & [\ 2.77555756e-16,\ 1.00000000e+ & [\ 2.77555756e-16,\ 1.000000000e+ & [\ 2.7755756e-16,\ 1.00000000e+ & [\ 2.77555756e-16,\ 1.000000000e+ & [\ 2.77555756e-16,\ 0.00000000e+ & [\ $	How can we use this factorization to solve $A\mathbf{x}=\mathbf{b}$ ?  Be stable)  The very sensitive to small changes in the input vectors, e.g. columns of $A$ . For this reason, we used a "modified Gram-Schmidt" algor of vectors $[\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n]$ by successivity subtracting out projections on the previously found orthonormal set $[\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_j]$ . $\mathbf{q}_1 = \frac{\mathbf{a}_1}{r_{11}}$ $\mathbf{q}_2 = \frac{\mathbf{a}_2 - (\mathbf{q}_1^* \mathbf{a}_2) \mathbf{q}_1}{r_{22}}$ $\mathbf{q}_3 = \frac{\mathbf{a}_3 - (\mathbf{q}_1^* \mathbf{a}_3) \mathbf{q}_1 - (\mathbf{q}_2^* \mathbf{a}_3) \mathbf{q}_2}{r_{33}}$ $\vdots$	rithm.
$\begin{array}{lll} \operatorname{array}([[\ 1.,\ 2.,\ -1.], & [\ 3.,\ 4.,\ 4.], & [\ 5.,\ 6.,\ 5.]]) \\ \operatorname{Q} = & \operatorname{array}([[\ 0.16903085,\ 0.89708523,\ -0.40824829], & [\ 0.50709255,\ 0.27602622,\ 0.81649658], & [\ 0.84515425,\ -0.34503278,\ -0.40824829]]) \\ \operatorname{R} = & \operatorname{array}([[\ 5.91607978,\ 7.43735744,\ 6.08511063], & [\ 0. &\ ,\ 0.82807867,\ -1.51814423], & [\ 0. &\ ,\ 0. &\ ,\ 0.63299316]]) \\ \operatorname{QR} = & \operatorname{array}([[\ 1.,\ 2.,\ -1.], & [\ 3.,\ 4.,\ 4.], & [\ 5.,\ 6.,\ 5.]]) \\ \operatorname{Q^**Q} = \operatorname{array}([[\ 1.000000000e+00,\ 2.77555756e-16,\ 0.00000000e+ & [\ 2.77555756e-16,\ 1.00000000e+ & [\ 2.77555756e-16,\ 1.000000000e+ & [\ 2.7755756e-16,\ 1.00000000e+ & [\ 2.77555756e-16,\ 1.000000000e+ & [\ 2.77555756e-16,\ 0.00000000e+ & [\ $	How can we use this factorization to solve $A\mathbf{x}=\mathbf{b}$ ?  The stable by the very sensitive to small changes in the input vectors, e.g. columns of $A$ . For this reason, we used a "modified Gram-Schmidt" algor of vectors $[\mathbf{a}_1,\mathbf{a}_2,\ldots,\mathbf{a}_n]$ by successivity subtracting out projections on the previously found orthonormal set $[\mathbf{q}_1,\mathbf{q}_2,\ldots,\mathbf{q}_n]$ . $\mathbf{q}_1 = \frac{\mathbf{a}_1}{r_{11}}$ $\mathbf{q}_2 = \frac{\mathbf{a}_2 - (\mathbf{q}_1^*\mathbf{a}_2)\mathbf{q}_1}{r_{22}}$ $\mathbf{q}_3 - \frac{\mathbf{a}_3 - (\mathbf{q}_1^*\mathbf{a}_3)\mathbf{q}_1 - (\mathbf{q}_2^*\mathbf{a}_3)\mathbf{q}_2}{r_{33}}$ $\vdots$ $\mathbf{q}_1  -1$ $\mathbf{a}_1 - r_{11}\mathbf{q}_1$	rithm.
$\begin{array}{lll} \operatorname{array}([[\ 1.,\ 2.,\ -1.], & [\ 3.,\ 4.,\ 4.], & [\ 5.,\ 6.,\ 5.]]) \\ \operatorname{Q} = & \operatorname{array}([[\ 0.16903085,\ 0.89708523,\ -0.40824829], & [\ 0.50709255,\ 0.27602622,\ 0.81649658], & [\ 0.84515425,\ -0.34503278,\ -0.40824829]]) \\ \operatorname{R} = & \operatorname{array}([[\ 5.91607978,\ 7.43735744,\ 6.08511063], & [\ 0. &\ ,\ 0.82807867,\ -1.51814423], & [\ 0. &\ ,\ 0. &\ ,\ 0.63299316]]) \\ \operatorname{QR} = & \operatorname{array}([[\ 1.,\ 2.,\ -1.], & [\ 3.,\ 4.,\ 4.], & [\ 5.,\ 6.,\ 5.]]) \\ \operatorname{Q^**Q} = \operatorname{array}([[\ 1.000000000e+00,\ 2.77555756e-16,\ 0.00000000e+ & [\ 2.77555756e-16,\ 1.00000000e+ & [\ 2.77555756e-16,\ 1.000000000e+ & [\ 2.7755756e-16,\ 1.00000000e+ & [\ 2.77555756e-16,\ 1.000000000e+ & [\ 2.77555756e-16,\ 0.00000000e+ & [\ $	How can we use this factorization to solve $A\mathbf{x}=\mathbf{b}$ ?  Be stable)  The extra sequence of vectors $[\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n]$ by successivity subtracting out projections on the previously found orthonormal set $[\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_p]$ . $\mathbf{q}_1 = \frac{\mathbf{a}_1}{r_{11}}$ $\mathbf{q}_2 = \frac{\mathbf{a}_2 - (\mathbf{q}_1^* \mathbf{a}_2) \mathbf{q}_1}{r_{22}}$ $\mathbf{q}_3 - \frac{\mathbf{a}_3 - (\mathbf{q}_1^* \mathbf{a}_3) \mathbf{q}_1 - (\mathbf{q}_2^* \mathbf{a}_3) \mathbf{q}_2}{r_{33}}$ $\vdots$ $\mathbf{q}_1 = 1$ $\mathbf{q}_1 = 1$ $\mathbf{q}_2 = 1$ $\mathbf{q}_3 = 1$	rithm.
$\begin{array}{lll} \operatorname{array}([[\ 1.,\ 2.,\ -1.], & [\ 3.,\ 4.,\ 4.], & [\ 5.,\ 6.,\ 5.]]) \\ \operatorname{Q} = & \operatorname{array}([[\ 0.16903085,\ 0.89708523,\ -0.40824829], & [\ 0.50709255,\ 0.27602622,\ 0.81649658], & [\ 0.84515425,\ -0.34503278,\ -0.40824829]]) \\ \operatorname{R} = & \operatorname{array}([[\ 5.91607978,\ 7.43735744,\ 6.08511063], & [\ 0. &\ ,\ 0.82807867,\ -1.51814423], & [\ 0. &\ ,\ 0. &\ ,\ 0.63299316]]) \\ \operatorname{QR} = & \operatorname{array}([[\ 1.,\ 2.,\ -1.], & [\ 3.,\ 4.,\ 4.], & [\ 5.,\ 6.,\ 5.]]) \\ \operatorname{Q^**Q} = \operatorname{array}([[\ 1.000000000e+00,\ 2.77555756e-16,\ 0.00000000e+ & [\ 2.77555756e-16,\ 1.00000000e+ & [\ 2.77555756e-16,\ 1.000000000e+ & [\ 2.7755756e-16,\ 1.00000000e+ & [\ 2.77555756e-16,\ 1.000000000e+ & [\ 2.77555756e-16,\ 0.00000000e+ & [\ $	How can we use this factorization to solve $A\mathbf{x} = \mathbf{b}$ ?  How can we use this factorization to solve $A\mathbf{x} = \mathbf{b}$ ?  The stable is expressed as the input vectors, e.g. columns of $A$ . For this reason, we used a "modified Gram Schmidt" algor of vectors $[\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n]$ by successivity subtracting out projections on the previously found orthonormal set $[\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_n]$ . $\mathbf{q}_1 = \frac{\mathbf{a}_1}{r_{11}}$ $\mathbf{q}_2 = \frac{\mathbf{a}_2 - (\mathbf{q}_1^* \mathbf{a}_2)\mathbf{q}_1}{r_{22}}$ $\mathbf{q}_3 = \frac{\mathbf{a}_3 - (\mathbf{q}_1^* \mathbf{a}_3)\mathbf{q}_1 - (\mathbf{q}_2^* \mathbf{a}_3)\mathbf{q}_2}{r_{33}}$ $\vdots$ $\mathbf{a}_1 = r_{11}\mathbf{q}_1$ $\mathbf{a}_1 = r_{11}\mathbf{q}_1$ $\mathbf{a}_1 = r_{11}\mathbf{q}_1$	rithm.
$ \begin{aligned} & \text{array}([\ 1\ ,\ ,\ 2\ ,\ -1\ ,\ ]\ (3\ ,\ 4\ ,\ 4\ ,\ 4\ )\ (5\ ,\ 6\ ,\ 5\ ,\ )]) \\ & 0 = \\ & \text{array}([\ 0\ .16903085,\ 0\ .89708523,\ -0\ .40824829]), \\ & (0\ .828151825,\ -0\ .34503278,\ -0\ .40824829])) \\ & R = \\ & \text{array}([\ 5\ .91607978,\ 7\ .43735744,\ 6\ .08511063], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814423], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814423], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814423], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814423], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814423], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814423], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814423], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814423], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814423], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814223], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814223], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814223], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814223], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814223], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814223], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814223], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814223], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814223], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814223], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814223], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814223], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814223], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814223], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .51814223], \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .1828239316]) \\ & [\ 0\ ,\ 0\ .82807867,\ -1\ .18282393956-16, \ 0\ .00000000000000000000000000000000$	thow can we use this factorization to solve $A\mathbf{x} = \mathbf{b}$ ?  The stable is expressive to small changes in the input vectors, e.g. columns of $A$ . For this reason, we used a "modified Gran-Schmidt" algor of vectors $[\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n]$ by successivity subtracting out projections on the previously found orthonormal set $[\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_n]$ . $\mathbf{q}_1 = \frac{\mathbf{a}_1}{r_{11}}$ $\mathbf{q}_2 = \frac{\mathbf{a}_2 - (\mathbf{q}_1^* \mathbf{a}_2)\mathbf{q}_1}{r_{22}}$ $\mathbf{q}_3 = \frac{\mathbf{a}_2 - (\mathbf{q}_1^* \mathbf{a}_3)\mathbf{q}_1 - (\mathbf{q}_2^* \mathbf{a}_3)\mathbf{q}_2}{r_{33}}$ $\vdots$ $\mathbf{q}_1 = \mathbf{r}_{11} \mathbf{q}_1$ $\mathbf{q}_1 = \mathbf{r}_{11} \mathbf{q}_1$ $\mathbf{q}_2 - (\mathbf{q}_1^* \mathbf{a}_2)\mathbf{q}_1 + r_{22}\mathbf{q}_2$	rithm.
$ \begin{aligned} & \operatorname{array}([[\ 1.,\ 2.,\ -1.],\ [\ 5.,\ 6.,\ 5.]]) \\ & = \\ & \operatorname{array}([[\ 0.16903085,\ 0.89708523,\ -0.40824829],\ [\ 0.84515425,\ 0.27602622,\ 0.81649658],\ [\ 0.84515425,\ 0.34503278,\ -0.40824829]]) \\ & R = \\ & \operatorname{array}([[\ 5.91607978,\ 7.43735744,\ 6.08511063],\ [\ 0.,\ 0.82807867,\ -1.51814423],\ [\ 0.,\ 0.,\ 1.63299316]]) \\ & \operatorname{array}([[\ 1.,\ 2.,\ -1.],\ [\ 5.,\ 6.,\ 5.]]) \\ & \operatorname{array}([[\ 1.,\ 2.,\ -1.],\ [\ 5.,\ 6.,\ 5.]]) \\ & \operatorname{array}([[\ 1.000000006+00,\ 2.77555756e-16,\ 0.00000000e+0,\ -9.15933995e-16,\ 1.00000000e+0,\ -9.15939$	thow can we use this factorization to solve $A\mathbf{x} = \mathbf{b}$ ?  The stable is expressive to small changes in the input vectors, e.g. columns of $A$ . For this reason, we used a "modified Gran-Schmidt" algor of vectors $[\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n]$ by successivity subtracting out projections on the previously found orthonormal set $[\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_n]$ . $\mathbf{q}_1 = \frac{\mathbf{a}_1}{r_{11}}$ $\mathbf{q}_2 = \frac{\mathbf{a}_2 - (\mathbf{q}_1^* \mathbf{a}_2)\mathbf{q}_1}{r_{22}}$ $\mathbf{q}_3 = \frac{\mathbf{a}_2 - (\mathbf{q}_1^* \mathbf{a}_3)\mathbf{q}_1 - (\mathbf{q}_2^* \mathbf{a}_3)\mathbf{q}_2}{r_{33}}$ $\vdots$ $\mathbf{q}_1 = \mathbf{r}_{11} \mathbf{q}_1$ $\mathbf{q}_1 = \mathbf{r}_{11} \mathbf{q}_1$ $\mathbf{q}_2 - (\mathbf{q}_1^* \mathbf{a}_2)\mathbf{q}_1 + r_{22}\mathbf{q}_2$	rithm.
$ \begin{aligned} & \operatorname{array}([[\ 1.,\ 2.,\ -1.],\ [\ 5.,\ 6.,\ 5.]]) \\ & = \\ & \operatorname{array}([[\ 0.16903085,\ 0.89708523,\ -0.40824829],\ [\ 0.84515425,\ 0.27602622,\ 0.81649658],\ [\ 0.84515425,\ 0.34503278,\ -0.40824829]]) \\ & R = \\ & \operatorname{array}([[\ 5.91607978,\ 7.43735744,\ 6.08511063],\ [\ 0.,\ 0.82807867,\ -1.51814423],\ [\ 0.,\ 0.,\ 1.63299316]]) \\ & \operatorname{array}([[\ 1.,\ 2.,\ -1.],\ [\ 5.,\ 6.,\ 5.]]) \\ & \operatorname{array}([[\ 1.,\ 2.,\ -1.],\ [\ 5.,\ 6.,\ 5.]]) \\ & \operatorname{array}([[\ 1.000000006+00,\ 2.77555756e-16,\ 0.00000000e+0,\ -9.15933995e-16,\ 1.00000000e+0,\ -9.15939$	How can we use this factorization to solve $A\mathbf{x} = \mathbf{b}$ ?  Be stable)  The very sensitive to small changes in the input vectors, e.g. columns of $A$ . For this reason, we used a "modified Gram-Schmidt' algor of vectors $[\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n]$ by successivity subtracting out projections on the previously found orthonormal set $[\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_j]$ . $\mathbf{q}_1 = \frac{\mathbf{a}_1}{r_{11}}$ $\mathbf{q}_2 = \frac{\mathbf{a}_3}{r_{22}} = \frac{(\mathbf{q}_1^* \mathbf{a}_3) \mathbf{q}_1}{r_{23}}$ $\mathbf{q}_3 = \frac{\mathbf{q}_1^* \mathbf{a}_3}{r_{33}} = \frac{\mathbf{q}_1^* \mathbf{a}_3}{r_{33}}$ $\vdots$ $\mathbf{a}_1 = r_{11} \mathbf{q}_1$ $\mathbf{a}_2 = (\mathbf{q}_1^* \mathbf{a}_2) \mathbf{q}_1 + r_{22} \mathbf{q}_2$ The coplanar.	rithm.
$ \begin{aligned} & \operatorname{array}([[\ 1.,\ 2.,\ -1.],\ [\ 5.,\ 6.,\ 5.]]) \\ & = \\ & \operatorname{array}([[\ 0.16903085,\ 0.89708523,\ -0.40824829],\ [\ 0.84515425,\ 0.27602622,\ 0.81649658],\ [\ 0.84515425,\ 0.34503278,\ -0.40824829]]) \\ & R = \\ & \operatorname{array}([[\ 5.91607978,\ 7.43735744,\ 6.08511063],\ [\ 0.,\ 0.82807867,\ -1.51814423],\ [\ 0.,\ 0.,\ 1.63299316]]) \\ & \operatorname{array}([[\ 1.,\ 2.,\ -1.],\ [\ 5.,\ 6.,\ 5.]]) \\ & \operatorname{array}([[\ 1.,\ 2.,\ -1.],\ [\ 5.,\ 6.,\ 5.]]) \\ & \operatorname{array}([[\ 1.000000006+00,\ 2.77555756e-16,\ 0.00000000e+0,\ -9.15933995e-16,\ 1.00000000e+0,\ -9.15939$	How can we use this factorization to solve $A\mathbf{x}=\mathbf{b}$ ?  Be stable)  The very sensitive to small changes in the input vectors, e.g. columns of $A$ . For this reason, we used a "modified Gram-Schmidt" algor of vectors $[\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n]$ by successivly subtracting out projections on the previously found orthonormal set $[\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_n]$ . $\mathbf{q}_1 = \frac{\mathbf{q}_1}{\mathbf{q}_2} = \frac{\mathbf{q}_1 - (\mathbf{q}_1^2 \mathbf{q}_2) \mathbf{q}_1}{\mathbf{q}_3} = \frac{\mathbf{q}_1 - (\mathbf{q}_1^2 \mathbf{q}_3) \mathbf{q}_1}{\mathbf{q}_3} = \frac{\mathbf{q}_1 - (\mathbf{q}_1^2 \mathbf{q}_3) \mathbf{q}_2}{\mathbf{q}_3}$ $\mathbf{q}_1 = \mathbf{r}_{11} \cdot \mathbf{q}_1$ $\mathbf{q}_1 = \mathbf{r}_{11} \cdot \mathbf{q}_1$ $\mathbf{q}_2 = \mathbf{q}_1^2 \mathbf{a}_2 \cdot (\mathbf{q}_1^2 \mathbf{a}_2) \mathbf{q}_2$ as follows. $\mathbf{q}_1 = \mathbf{r}_{11} \cdot \mathbf{q}_2$ as coplanar.	rithm.
$ \begin{aligned} & \operatorname{array}([[\ 1.,\ 2.,\ -1.],\ [\ 5.,\ 6.,\ 5.]]) \\ & = \\ & \operatorname{array}([[\ 0.16903085,\ 0.89708523,\ -0.40824829],\ [\ 0.84515425,\ 0.27602622,\ 0.81649658],\ [\ 0.84515425,\ 0.34503278,\ -0.40824829]]) \\ & R = \\ & \operatorname{array}([[\ 5.91607978,\ 7.43735744,\ 6.08511063],\ [\ 0.,\ 0.82807867,\ -1.51814423],\ [\ 0.,\ 0.,\ 1.63299316]]) \\ & \operatorname{array}([[\ 1.,\ 2.,\ -1.],\ [\ 5.,\ 6.,\ 5.]]) \\ & \operatorname{array}([[\ 1.,\ 2.,\ -1.],\ [\ 5.,\ 6.,\ 5.]]) \\ & \operatorname{array}([[\ 1.000000006+00,\ 2.77555756e-16,\ 0.00000000e+0,\ -9.15933995e-16,\ 1.00000000e+0,\ -9.15939$	How can we use this factorization to solve $A\mathbf{x}=\mathbf{b}$ ?  Be stable)  The very sensitive to small changes in the input vectors, e.g. columns of $A$ . For this reason, we used a "modified Gram-Schmidt" algor of vectors $[\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n]$ by successivly subtracting out projections on the previously found orthonormal set $[\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_n]$ . $\mathbf{q}_1 = \frac{\mathbf{q}_1}{\mathbf{q}_2} = \frac{\mathbf{q}_1 - (\mathbf{q}_1^2 \mathbf{q}_2) \mathbf{q}_1}{\mathbf{q}_3} = \frac{\mathbf{q}_1 - (\mathbf{q}_1^2 \mathbf{q}_3) \mathbf{q}_1}{\mathbf{q}_3} = \frac{\mathbf{q}_1 - (\mathbf{q}_1^2 \mathbf{q}_3) \mathbf{q}_2}{\mathbf{q}_3}$ $\mathbf{q}_1 = \mathbf{r}_{11} \cdot \mathbf{q}_1$ $\mathbf{q}_1 = \mathbf{r}_{11} \cdot \mathbf{q}_1$ $\mathbf{q}_2 = \mathbf{q}_1^2 \mathbf{a}_2 \cdot (\mathbf{q}_1^2 \mathbf{a}_2) \mathbf{q}_2$ as follows. $\mathbf{q}_1 = \mathbf{r}_{11} \cdot \mathbf{q}_2$ as coplanar.	rithm.
$ \begin{aligned} & \operatorname{array}([[\ 1.,\ 2.,\ -1.],\ [\ 5.,\ 6.,\ 5.]]) \\ & = \\ & \operatorname{array}([[\ 0.16903085,\ 0.89708523,\ -0.40824829],\ [\ 0.84515425,\ 0.27602622,\ 0.81649658],\ [\ 0.84515425,\ 0.34503278,\ -0.40824829]]) \\ & R = \\ & \operatorname{array}([[\ 5.91607978,\ 7.43735744,\ 6.08511063],\ [\ 0.,\ 0.82807867,\ -1.51814423],\ [\ 0.,\ 0.,\ 1.63299316]]) \\ & \operatorname{array}([[\ 1.,\ 2.,\ -1.],\ [\ 5.,\ 6.,\ 5.]]) \\ & \operatorname{array}([[\ 1.,\ 2.,\ -1.],\ [\ 5.,\ 6.,\ 5.]]) \\ & \operatorname{array}([[\ 1.000000006+00,\ 2.77555756e-16,\ 0.00000000e+0,\ -9.15933995e-16,\ 1.00000000e+0,\ -9.15939$	How can we use this factorization to solve $Ax = b^2$ e stable)  be very sensitive to small changes in the input vectors, e.g., columns of $A$ . For this reason, we used a 'modified Gram-Schmidt' alpor of vectors $[a_1, a_2, \dots, a_n]$ by execossively subtracting out projections on the proviously found orthonormal set $[a_1, a_2, \dots, a_n]$ . $ \begin{array}{c} a_1 = \frac{a_1}{r_{11}} \\ a_2 = \frac{a_2 - (a_1^2 a_2) a_1}{a_2} \\ a_3 = (a_1^2 a_2) a_1 - (a_2^2 a_3) a_2 \end{array} $ $ \begin{array}{c} a_1 = r_{11} \\ a_2 = r_{12} \\ a_3 = coplanar. $ a $a_1 = r_{12} $ $ \begin{array}{c} a_1 = r_{12} \\ a_2 = r_{13} \\ a_3 = r_{14} \\ a_4 = r_{14} \\ a_5 = r_{15} \\ a_6 = (a_1^2 a_2) a_1 + r_{25} a_2 \end{array} $	rithm.
$ \begin{aligned} & \operatorname{array}([[\ 1.,\ 2.,\ -1.],\ [\ 5.,\ 6.,\ 5.]]) \\ & = \\ & \operatorname{array}([[\ 0.16903085,\ 0.89708523,\ -0.40824829],\ [\ 0.84515425,\ 0.27602622,\ 0.81649658],\ [\ 0.84515425,\ 0.34503278,\ -0.40824829]]) \\ & R = \\ & \operatorname{array}([[\ 5.91607978,\ 7.43735744,\ 6.08511063],\ [\ 0.,\ 0.82807867,\ -1.51814423],\ [\ 0.,\ 0.,\ 1.63299316]]) \\ & \operatorname{array}([[\ 1.,\ 2.,\ -1.],\ [\ 5.,\ 6.,\ 5.]]) \\ & \operatorname{array}([[\ 1.,\ 2.,\ -1.],\ [\ 5.,\ 6.,\ 5.]]) \\ & \operatorname{array}([[\ 1.000000006+00,\ 2.77555756e-16,\ 0.00000000e+0,\ -9.15933995e-16,\ 1.00000000e+0,\ -9.15939$	Lies can we use this factorization to solve $A\mathbf{x} = \mathbf{h}$ ?  Be stable)  The very sensitive to annull changes in the input vectors, e.g. columns of $A$ . For this reason, we used a "modified Gram-Schmidt" algor of vectors $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n$ by successivity subtracting out projections on the proviously found orthonormal set $[\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_i]$ and $\mathbf{q}_1 = \frac{\mathbf{a}_1}{r_{21}}$ and $\mathbf{q}_2 = \frac{\mathbf{a}_2 - (\mathbf{q}_1^2\mathbf{a}_2)\mathbf{q}_1}{r_{22}}$ is $\mathbf{q}_1 = \mathbf{q}_1 = \mathbf{q}_1 = \mathbf{q}_2 = \mathbf{q}_1^2\mathbf{a}_2$ .  If $\mathbf{q}_1 = \mathbf{q}_1 = \mathbf{q}_1$ and $\mathbf{q}_2 = \mathbf{q}_1^2\mathbf{q}_2$ and $\mathbf{q}_3 = \mathbf{q}_4 = \mathbf{q}_4^2\mathbf{q}_2$ by $\mathbf{q}_4 = \mathbf{q}_4 = \mathbf{q}_4^2\mathbf{q}_4$ .  The proviously found orthonormal set $[\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_k]$ is $\mathbf{q}_4 = \mathbf{q}_4 = \mathbf{q}_4^2\mathbf{q}_4$ .  The proviously found orthonormal set $[\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_k]$ is $\mathbf{q}_4 = \mathbf{q}_4 = \mathbf{q}_4$ .  The proviously found orthonormal set $[\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_k]$ is $\mathbf{q}_4 = \mathbf{q}_4 = \mathbf{q}_4$ .  The proviously found orthonormal set $[\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_k]$ is $\mathbf{q}_4 = \mathbf{q}_4 = \mathbf{q}_4$ .  The proviously found orthonormal set $[\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_k]$ is $\mathbf{q}_4 = \mathbf{q}_4 = \mathbf{q}_4$ .  The proviously found orthonormal set $[\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_k]$ is $\mathbf{q}_4 = \mathbf{q}_4 = \mathbf{q}_4$ .  The proviously found orthonormal set $[\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_k]$ is $\mathbf{q}_4 = \mathbf{q}_4 = \mathbf{q}_4$ .	rithm.
$ \begin{aligned} & \operatorname{array}([[\ 1.,\ 2.,\ -1.],\ [\ 5.,\ 6.,\ 5.]]) \\ & = \\ & \operatorname{array}([[\ 0.16903085,\ 0.89708523,\ -0.40824829],\ [\ 0.84515425,\ 0.27602622,\ 0.81649658],\ [\ 0.84515425,\ 0.34503278,\ -0.40824829]]) \\ & R = \\ & \operatorname{array}([[\ 5.91607978,\ 7.43735744,\ 6.08511063],\ [\ 0.,\ 0.82807867,\ -1.51814423],\ [\ 0.,\ 0.,\ 1.63299316]]) \\ & \operatorname{array}([[\ 1.,\ 2.,\ -1.],\ [\ 5.,\ 6.,\ 5.]]) \\ & \operatorname{array}([[\ 1.,\ 2.,\ -1.],\ [\ 5.,\ 6.,\ 5.]]) \\ & \operatorname{array}([[\ 1.000000006+00,\ 2.77555756e-16,\ 0.00000000e+0,\ -9.15933995e-16,\ 1.00000000e+0,\ -9.15939$	How can we use this factorization to solve $Ax = b7$ e stable)  be very sensitive to small changes in the input vectors, e.g. columns of $A$ for this reason, we used a "modified Grant-Schmidt" algorization of vectors $[a_1, a_2, \ldots, a_n]$ by sectosibly subtracting our projections on the previously found orthonormal set $[a_1, a_2, \ldots, a_n]$ . $a_1 = \frac{a_1}{r_2} = \frac{a_1}{(a_1^2 a_1)a_1} = \frac{a_2}{r_2} = \frac{a_1 - (a_1^2 a_2)a_2}{r_2}$ $a_2 = \frac{a_1 - (a_1^2 a_2)a_2}{r_2} = \frac{a_2 - (a_1^2 a_2)a_2}{r_3}$ $a_1 = r_{11}q_1$ $a_1 = r_{11}q_1$ Plant Containing $a_1 = r_{12}q_2$ resplants.	ithm.
acray([1], 2, 2, -2-1); [3.4, 4.1] [3.4, 4.4]; [3.4, 4.1] [5.6, 6.5, 5.1]) 0.5 [5.6, 6.5, 5.1]) 0.5 [5.6, 6.5, 5.1]) 0.5 [6.8503053, -0.40024829]; [0.50090255, 0.27602622, -0.20449589]; [0.50090255, 0.27602622, 0.21649589]; [0.50090250, 0.27602622, 0.21649589]; [0.50090250, 0.27602622, 0.21649589]; [0.50090250, 0.27602622, 0.21649589]; [0.50090250, 0.27602622], 0.2009020000000000000000000000000000000	e stable)  e stable)  be very servalve to another function to solve $A_{A} = \mathbf{b}$ ?  e stable)  be very servalve to another function in the input vectors, $\mathbf{a}_{0}$ , endures of $A$ . For this reason, we used a "recidited Gram-Schmidt" algorithm of vectors $[a_{1}, a_{2}, \ldots, a_{n}]$ by successed $[a_{1}, a_{2}, \ldots, a_{n}]$ by successed $[a_{1}, a_{2}, \ldots, a_{n}]$ and $[a_{1}^{*}(a_{1}^{*}a_{2})]a_{1}$ and $[a_{1}^{*}(a_{1}^{*}a_{2})]a_{2}$ and	rithm.
array([] 1, 2, -1, -1), [3, 4, 4, 4], [3, 4, 4, 4], [3, 4, 4, 4], [5, 6, 5, 5], [5], [6, 5, 5], [7], [6, 5, 5], [7], [9], [9], [9], [9], [9], [9], [9], [9	From can we use this footonication to color $Ax = 1/2$ as stable)  as very sensitive to small changes in the input vectors, e.g. columns of $A$ . For this mose, we used a freedified Gram Schmidt algorishment of vectors $[a_1,a_2,\ldots,a_n]$ by successivity submouthing out projections on the previously found anthonormal set $[a_1,a_2,\ldots,a_k]$ . $a_1 = \frac{a_1 - (a_1)a_1}{a_2}$ $a_2 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_3 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_4 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_4 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_4 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_1 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_2 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_3 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_4 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_1 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_2 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_3 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_4 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_1 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_2 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_3 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_4 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_1 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_2 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_3 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_4 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_1 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_2 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_3 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_4 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_1 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_2 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_3 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_4 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_4 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_1 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_2 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_3 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_4 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_1 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_2 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_3 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_4 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_1 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_1 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_2 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_3 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_4 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_1 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_2 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_3 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_4 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_1 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_2 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_3 = \frac{a_1 - (a_1)a_2}{a_2}$ $a_4 = \frac{a_1 - (a_1)a_2}{a_2}$	rithm.
actray([[ 1., 2., -1.]), [ 3.+, 4., 4.], [ 3.+, 4.], 3.7, 4.+, 4.], 3.7, 4.+, 4.], 3.7, 4.+, 4.], 3.7, 4.+, 4.], 3.7, 4.1, 4.1, 4.1, 4.1, 4.1, 4.1, 4.1, 4.1	e stable)  e estable)  e overy semilibre to annual changes in the insut vectors, e.g., columns of A. For this reserve, we used a "modified Gram-Gehricht" agove overy semilibre as annual changes in the insut vectors, e.g., columns of A. For this reserve, we used a "modified Gram-Gehricht" agove over years and ago, ago, ago, ago, ago, ago, ago, ago,	rithm.
stroy([1], 2., -1.], [3., 4., 4.], 3.], 3.4., 4.], 3.], 3.1., 4., 4.], 3.], 3.1., 3.],	e stable)  e estable)  e overy semilibre to annual changes in the insut vectors, e.g., columns of A. For this reserve, we used a "modified Gram-Gehricht" agove overy semilibre as annual changes in the insut vectors, e.g., columns of A. For this reserve, we used a "modified Gram-Gehricht" agove over years and ago, ago, ago, ago, ago, ago, ago, ago,	rithm.
array(1, 1, 2, 2, -1, 1, 3, 4, 4, -1), 3, 4, 4, -1), 3, 4, 4, -1), 3, 4, 4, -1), 3, 4, 4, -1), 3, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1	e stable)  e estable)  e overy semilibre to annual changes in the insut vectors, e.g., columns of A. For this reserve, we used a "modified Gram-Gehricht" agove overy semilibre as annual changes in the insut vectors, e.g., columns of A. For this reserve, we used a "modified Gram-Gehricht" agove over years and ago, ago, ago, ago, ago, ago, ago, ago,	rithm.
A key observation in the above is that $\mathbf{q}_2$ lies in a plane or thogonal result the classical Gram-Schmidt algorithm (more than a cuttle of the Cassical Gram-Schmidt (CGS) or thogonalizes a set of the classical Gram-Schmidt (CGS) or the clast (CGS) or the classical Gram-Schmidt (CGS) or the classical Gr	its open we can the this factorization to valve day — 1/2  e stable)  extra we can the this factorization to valve day — 1/2  extra value to small charges in the man vectors, e.p. columns of A. For the reason, we used a finalised size in State of Tage of Process of State of Tage of Tag	rithm.
$ \begin{array}{ll} a_{ij}, a_{ij$	time can be used the factorization to sover $dx = 0.7$ e stable)  e stable)  e vary sensitive to small charges in the injust vectors, e.g. volume of $A$ . For this resour, we ever a "modified Corrected midd" export on each $a_1, a_2, \dots, a_n$ , by secondary participation in the processor year and an angular transformation and $[a_1, a_2, \dots, a_n]$ , and $a_1 = \frac{a_1}{a_2 a_1} \frac{(a_1 a_1)a_2}{a_1 a_2} \frac{a_2}{a_2 a_2} \frac{(a_1 a_2)a_2}{a_2 a_2} \frac{a_2}{a_2 a_2} \frac{a_2}{a_2} \frac{a_2}$	rithm.
accept [1] 1, 2, 2, -1, 1, 3, 4, 2, 1, 2, -1, 1, 3, 4, 4, 2, 1, 2, -1, 1, 3, 4, 4, 2, 1, 2, -1, 1, 2, -1,	time can be used the factorization to sover $dx = 0.7$ e stable)  e stable)  e vary sensitive to small charges in the injust vectors, e.g. volume of $A$ . For this resour, we ever a "modified Corrected midd" export on each $a_1, a_2, \dots, a_n$ , by secondary participation in the processor year and an angular transformation and $[a_1, a_2, \dots, a_n]$ , and $a_1 = \frac{a_1}{a_2 a_1} \frac{(a_1 a_1)a_2}{a_1 a_2} \frac{a_2}{a_2 a_2} \frac{(a_1 a_2)a_2}{a_2 a_2} \frac{a_2}{a_2 a_2} \frac{a_2}{a_2} \frac{a_2}$	rithm.
active (1 1, $r_2$ , $r_1$ , $r_1$ ), $r_2$ , $r_3$ , $r_4$ , $r_4$ ), $r_5$ ,	time can be used the factorization to sover $dx = 0.7$ e stable)  e stable)  e vary sensitive to small charges in the injust vectors, e.g. volume of $A$ . For this resour, we ever a "modified Corrected midd" export on each $a_1, a_2, \dots, a_n$ , by secondary participation in the processor year and an angular transformation and $[a_1, a_2, \dots, a_n]$ , and $a_1 = \frac{a_1}{a_2 a_1} \frac{(a_1 a_1)a_2}{a_1 a_2} \frac{a_2}{a_2 a_2} \frac{(a_1 a_2)a_2}{a_2 a_2} \frac{a_2}{a_2 a_2} \frac{a_2}{a_2} \frac{a_2}$	rithm.
array((f. 1, 2, -1, 1), [2, 4, 4, 4, 1], [2, 4, 4, 1], [2, 4, 4, 4, 1], [2, 5, 6, 5, 5, 1]) [2, 1], [2, 6, 5, 5, 1]] [2, 1], [2, 6, 5, 5, 1]] [2, 6, 5, 5, 1]] [2, 6, 5, 5, 1]] [2, 6, 5, 7, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,	time can be used the factorization to sover $dx = 0.7$ e stable)  e stable)  e vary sensitive to small charges in the injust vectors, e.g. volume of $A$ . For this resour, we ever a "modified Corrected midd" export on each $a_1, a_2, \dots, a_n$ , by secondary participation in the processor year and an angular transformation and $[a_1, a_2, \dots, a_n]$ , and $a_1 = \frac{a_1}{a_2 a_1} \frac{(a_1 a_1)a_2}{a_1 a_2} \frac{a_2}{a_2 a_2} \frac{(a_1 a_2)a_2}{a_2 a_2} \frac{a_2}{a_2 a_2} \frac{a_2}{a_2} \frac{a_2}$	rithm.
array([1], 2, 2, -1.], [3, 4, 4], [3], [3, 4, 4], [3], [4], 4], [4], [4], [4], [4], [4], [4]	time can be used the factorization to sover $dx = 0.7$ e stable)  e stable)  e vary sensitive to small charges in the injust vectors, e.g. volume of $A$ . For this resour, we ever a "modified Corrected midd" export on each $a_1, a_2, \dots, a_n$ , by secondary participation in the processor year and an angular transformation and $[a_1, a_2, \dots, a_n]$ , and $a_1 = \frac{a_1}{a_2 a_1} \frac{(a_1 a_1)a_2}{a_1 a_2} \frac{a_2}{a_2 a_2} \frac{(a_1 a_2)a_2}{a_2 a_2} \frac{a_2}{a_2 a_2} \frac{a_2}{a_2} \frac{a_2}$	rithm.
exercy([1], 2, 2, -1.], [3, 4, -2], [3, 4, -2], [3, 4, -2], [3, 4, -2], [3, 4, -2], [3, 4, -2], [3, -	time can be used the factorization to sover $dx = 0.7$ e stable)  e stable)  e vary sensitive to small charges in the injust vectors, e.g. volume of $A$ . For this resour, we ever a "modified Corrected midd" export on each $a_1, a_2, \dots, a_n$ , by secondary participation in the processor year and an angular transformation and $[a_1, a_2, \dots, a_n]$ , and $a_1 = \frac{a_1}{a_2 a_1} \frac{(a_1 a_1)a_2}{a_1 a_2} \frac{a_2}{a_2 a_2} \frac{(a_1 a_2)a_2}{a_2 a_2} \frac{a_2}{a_2 a_2} \frac{a_2}{a_2} \frac{a_2}$	rithm.
energy ([1,1, 2, -1]), { 3, 4, -1}, { 3, 4, -1}, { 3, 4, -1}, { 3, 4, -1}, { 3, 4, -1}, { 3, 4, -1}, { 3, 4, -1}, { 3, 4, -1}, { 3, 4, -1}, { 3, 4, -1}, { 3, 4, -1}, { 3, 4, -1}, { 3, 4, -1}, { 3, 4, -1}, { 3, 4, -1}, { 3, 4, -1}, { 4, -1}, { 5, -1}, { 6,	time can be used the factorization to sover $dx = 0.7$ e stable)  e stable)  e vary sensitive to small charges in the injust vectors, e.g. volume of $A$ . For this resour, we ever a "modified Corrected midd" export on each $a_1, a_2, \dots, a_n$ , by secondary participation in the processor year and an angular transformation and $[a_1, a_2, \dots, a_n]$ , and $a_1 = \frac{a_1}{a_2 a_1} \frac{(a_1 a_1)a_2}{a_1 a_2} \frac{a_2}{a_2 a_2} \frac{(a_1 a_2)a_2}{a_2 a_2} \frac{a_2}{a_2 a_2} \frac{a_2}{a_2} \frac{a_2}$	rithm.